

PhD thesis

**Analytic provenance for sensemaking behavioural markers in
visual analytics**

Islam, M.

Full bibliographic citation: Islam, M. 2021. Analytic provenance for sensemaking behavioural markers in visual analytics. PhD thesis Middlesex University

Year: 2021

Publisher: Middlesex University Research Repository

Available online: <https://repository.mdx.ac.uk/item/148z1z>

Middlesex University Research Repository makes the University's research available electronically.

Copyright and moral rights to this work are retained by the author and/or other copyright owners unless otherwise stated. The work is supplied on the understanding that any use for commercial gain is strictly forbidden. A copy may be downloaded for personal, non-commercial, research or study without prior permission and without charge.

Works, including theses and research projects, may not be reproduced in any format or medium, or extensive quotations taken from them, or their content changed in any way, without first obtaining permission in writing from the copyright holder(s). They may not be sold or exploited commercially in any format or medium without the prior written permission of the copyright holder(s).

Full bibliographic details must be given when referring to, or quoting from full items including the author's name, the title of the work, publication details where relevant

(place, publisher, date), pagination, and for theses or dissertations the awarding institution, the degree type awarded, and the date of the award.

If you believe that any material held in the repository infringes copyright law, please contact the Repository Team at Middlesex University via the following email address: repository@mdx.ac.uk

The item will be removed from the repository while any claim is being investigated.

See also repository copyright: re-use policy: <https://libguides.mdx.ac.uk/repository>

Analytic Provenance for Sensemaking Behavioural Markers in Visual Analytics



Md. Junayed Islam

Faculty of Science & Technology
Middlesex University
London

A thesis submitted to Middlesex University
in partial fulfilment of the requirements for the degree of
Doctor of Philosophy

September 2021

*For my parents,
Metu
&
Sahiba*

Acknowledgements

First and foremost I am extremely grateful to my esteemed supervisors, Professor B. L. William Wong and Associate Professor Kai Xu for their continuous support and patience during my PhD study. I was selected by them for this funded PhD study as well as work for a large EU funded research project named as below, without which it was not possible for me to reach at this stage. I would like to express my sincere gratitude to Professor William for giving me opportunity to join in different project consortium meetings, conferences, workshops and gather knowledge from international research communities. It was the unique chance in my life to learn and experience how a large research project is being carried out with the aim of contributing to the enhanced security to UK/EU nations. I am deeply grateful to Middlesex University to arrange everything I needed to gain such experiences and continue my study.

I would like to offer my special thanks and gratitude to Associate Professor Kai Xu to stand beside me during the crucial moment of my life when my wife's situation deteriorated after giving birth to my daughter. I still can remember his emails by asking about my condition at that time. He wrote statement in support of mine to grant my required study interruptions to support my wife and new born. Without such co-operation it wouldn't be possible for me to come back and finish this PhD. I shall remember this support with due to respect throughout rest of my life. Thanks to Professor Balbir Barn, the academic dean of the faculty for considering and granting my request.

While working for the EU project as part of my research, I met with many talented individuals. Professor Henrik Eriksson, Robin Keskisarkka were among many whom I found extremely knowledgeable and efficient. It was interesting for me to present my work to Belgium and West Midlands police analysts to get feedback for further development. I still can remember discussion sessions with Rani Pinchuck, Professor Ifan Shepherd, Dr. Simon Attfield, Dr. Lishi Zhang, Dr. Peter Passmore and many other consortium members. My gratitude also extends to Dr. Nadeem Qazi and Dr. Craig Anslow due to their support and encouragement during the ups and downs of my study.

It was a long tough journey to finish this PhD. Thanks to all the staff members of computer science faculty and research degrees offices who were extremely helpful and processed my raised requests very efficiently. My heartfelt thanks to Pragya Paudyal, Celeste Groenewald, Phong Hai Nguyen, Unai and others who were part of discussions, brainstorming and providing feedbacks to my research. I would also like to express my gratitude to Dr. Neesha Kodagoda, Dr. Chris Rooney, Patrick Seidler for their insightful comments and suggestions on the implementation results of my PhD.

Lastly, I would like to convey my very special thanks to the people who surrounded me at all times, gave me courage and psychological support. They are none but my father Md. Shafiqul Islam, my mother Jahan-Ara Begum, my wife Afrin Sultana and my little princess Sahiba Islam. I have finally become able to finish this thesis for their patience and continuous support. Hence, I am dedicating this thesis to them. Overall, thanks to almighty Allah who is the most merciful and beneficent.

Abstract

Studying how analysts use interactions in visualization systems is an important part of evaluating how well these interactions support analysis needs, making sense of data or generating insights. As sensemaking is inherently a fluid activity involving transitions between mental and interaction states, lack of accuracy or precision into adopted visualization techniques can create a gap between cognitive constructs and manipulations or interactions humans apply to think or reason about the data. To tackle the problem, this thesis proposes 'Behavioural Markers (BMs)' which are representatives of the action choices that analysts make during their analytical processes as the bridge between that gap. Appropriate tools, techniques are required to log individual processing activities and utilize those to complement the information entailed with transparent processing operations.

As a first step to achieve the goal of bridging between human cognition and analytic computation through interactions at micro-analytic level, this thesis contributes to an extensive research with groups of real police intelligence analysts for designing and developing a visual judgemental system named as 'PROV' according to W3C standard. Secondly to explain how human cognition leads to interactions and vice versa, it contributes to development of an exhaustive list of behavioural constructs and detection of those ingrained cognitive constituents through interaction network graph analysis and translate those by theories of psychology for externalizing human thought processes. Recovering cognitive reflection on analytic reasoning processes from extended log data or only by observing is a difficult task. Due to cognitive and perceptual variances, conventional clustering or pattern mining techniques for user behaviour modelling, task identification, clickstream modelling don't fit very well with this purpose. To overcome these limitations as third step, this research proposes 'BreakPoints (BPs)' as the way to pinpoint internal transitions in perception and cognition which are nipped into analytic interactions. This research has contributed to development of machine learning models to contextualize those streams of actions, infer cognitive transition points into both known and unknown task scenarios. Proposed approaches have significantly improved results compared to existing techniques. Finally for transparent validation of all computational outcomes in terms of reliability, accuracy, relevance and to build human trust on those results, this research has presented visual explanations of machine produced results by unfolding blackbox calculations.

The major research results reported into this thesis have contributed to the project VALCRI (Visual Analytics for Sensemaking in Criminal Intelligence), Analysis which has received funding from the European Union Seventh Framework Programme FP7/2007-2013 through Project VALCRI, European Commission Grant Agreement No- FP7-IP-608142, awarded to Middlesex University and partners.

Contents

List of figures	x
List of tables	xiv
The Project VALCRI	xv
Chapter 1 Introduction	16
1.1 Background	17
1.2 Research Problems	19
1.3 Research Approach.....	22
1.3.1 Data Used	22
1.3.2 Summary of Contributions.....	23
1.4 Publications	25
1.5 Outline.....	26
Chapter 2 Literature Review	28
2.1 Analytic Provenance.....	29
2.1.2 Process Provenance	29
2.1.2.1 Process Models.....	29
2.1.2.2 Process Scenarios and Strategies	32
2.1.2.3 Investigative Scenarios	32
2.1.2.4 Investigative Strategies	32
2.1.2.5 Visual Analytics for Process Provenance.....	34
2.1.2.6 Flexible Mechanisms for Source Management.....	35
2.1.2.7 Responsive User Interface.....	35
2.1.2.8 Direct Attention to Critical Information	35
2.1.3 Reasoning Provenance.....	36
2.1.3.1 Process and Reasoning: Interrelated Concepts	36
2.1.3.2 Reasoning Scenarios and Strategies	37
2.1.3.3 Reasoning Scenarios and Challenges	37
2.1.3.4 Reasoning Strategies	38
2.1.3.5 Visual Analytics for Reasoning Provenance	42
2.1.4 Examining Analytic Provenance in Visual Analytic Systems ...	43
2.1.4.1 Challenges to Examining Analytic Provenance.....	43
2.1.4.2 Capture, Visualization and Utilization.....	45
2.1.4.3 Capture Methods	45
2.1.4.4 Manual Capture.....	45
2.1.4.5 Automatic Capture.....	48
2.1.4.6 Automatic Capture with Visual Analytic Systems.....	49

- 2.1.4.7 Automatic Capture with Specific Software..... 52
- 2.1.5 Visualization..... 55
- 2.1.6 Utilization..... 56
- 2.1.7 Summary..... 57

- 2.2 Behavioural Markers (BMs) 58
 - 2.2.1 Overview..... 58
 - 2.2.2 BM System Development and Evaluation 60
 - 2.2.3 Correlation: Technical Skills & Non-Technical Skills 61
 - 2.2.4 Coupling Technical Skills & Non-Technical Skills..... 62
 - 2.2.5 Summary..... 63

- 2.3 Inferring Sensemaking Tasks 64
 - 2.3.1 Clickstream Modelling 64
 - 2.3.1.1 Click Sequence Model..... 64
 - 2.3.1.2 Time-based Model 65
 - 2.3.1.3 Hybrid Model 65
 - 2.3.2 Computing Sequence Similarity..... 65
 - 2.3.2.1 Common Sub-sequences 65
 - 2.3.2.2 Common Sub-sequences with Counts..... 66
 - 2.3.2.3 Distribution-based Method..... 66
 - 2.3.3 Task Identification..... 66
 - 2.3.4 User Behaviour Modelling..... 69
 - 2.3.5 Summary..... 72

- 2.4 Machine Learning for Inference Making 72
 - 2.4.1 Data Simulation..... 73
 - 2.4.2 Classification Algorithms 74
 - 2.4.2.1 Single-Class Classification..... 74
 - 2.4.2.2 Multi-Class Classification 75
 - 2.4.2.3 Reasoning Task Classification..... 75
 - 2.4.2.4 Non-Contextual Classification..... 76
 - 2.4.2.5 Machine Learning for Visual Analytic Systems..... 77
 - 2.4.3 Contextual Classification - Attention Model..... 79
 - 2.4.3.1 Global Attention..... 81
 - 2.4.3.2 Local Attention 82
 - 2.4.3.3 Self Attention..... 82
 - 2.4.3.4 Encoder-Decoder..... 83
 - 2.4.3.5 The Transformer 84
 - 2.4.3.6 Bidirectional Encoder Representation from Transformer84
 - 2.4.3.7 BERT Pre - Training..... 85
 - 2.4.3.8 BERT Fine - Tuning..... 86

2.4.4 Summary.....	86
2.5 Explainable AI (XAI)	88
2.5.1 Explainability and Interpretability	88
2.5.1.1 Interpretability.....	88
2.5.1.2 Explainability	89
2.5.2 Taxonomy of XAI Methods	90
2.5.2.1 Intrinsic or Post Hoc?	90
2.5.2.2 Model-Specific or Model Agnostic?.....	90
2.5.2.3 Local or Global?.....	91
2.5.3 Popular XAI Techniques.....	91
2.5.3.1 Local Interpretable Model-Agnostic Explanations.....	91
2.5.3.2 Shapley-Additive Explanation.....	93
2.5.3.3 ELI5 and Permutation Importance	95
2.5.4 Summary.....	96
Chapter 3 Analytic Provenance for Sensemaking	97
3.1 Chapter Overview.....	98
3.2 Introduction.....	99
3.3 Approach.....	102
3.3.1 Analytic Task Model.....	104
3.3.2 Perceiving Data	106
3.3.2.1 Uncertainty in Visualization	106
3.3.2.2 Crime Analysis Under Uncertainty.....	107
3.3.2.3 Case Study on a Criminal Situation	109
3.3.2.4 Findings.....	112
3.3.3 Capturing Data	117
3.3.3.1 Requirements Analysis	118
3.3.3.2 System Design.....	122
3.3.3.3 Analytic Provenance Visualization.....	126
3.3.4 Recovering Data.....	128
3.3.4.1 Analytic Path	128
3.3.4.2 Schematization.....	130
3.3.5 Reusing Data	131
3.3.5.1 Repetitive Replicating Playback (RRP)	131
3.4 Evaluation.....	134
3.5 Discussion.....	138
Chapter 4 Sensemaking Behavioural Markers	140
4.1 Chapter Overview.....	141

4.2 Introduction.....	142
4.3 The problem.....	143
4.4 Development Approach	144
4.5 Behavioural Markers (BMs) Detection	152
4.5.1 Quantitative Approach.....	152
4.5.1.1 Action Sequence Computation.....	153
4.5.2 Qualitative Approach.....	158
4.5.2.1 Methodology.....	158
4.5.2.2 Participants.....	159
4.5.2.3 Procedure	159
4.5.2.4 Data Collection	162
4.5.2.5 Study Setup	162
4.5.2.6 Assessment Method	162
4.5.2.7 Results.....	164
4.6 Discussion	169

Chapter 5 Sensemaking Task Inference 172

5.1 Chapter Overview.....	173
5.2 Introduction.....	174
5.3 Approach and Experiments.....	176
5.3.1 Experiment 1.....	176
5.3.1.1 Dataset.....	176
5.3.1.2 Pre-processing.....	177
5.3.1.3 Evaluation.....	177
5.3.1.4 Discussion.....	184
5.3.2 Contextual Attention.....	185
5.3.2.1 The Context.....	185
5.3.2.2 The Attention	186
5.3.3 Breakpoint for Action Chunking	187
5.3.3.1 Definition	187
5.3.3.2 Detecting Breakpoints.....	188
5.3.4 Experiment 2.....	191
5.3.4.1 Supervised Learning for Breakpoint Detection.....	191
5.3.4.2 Dataset.....	191
5.3.4.3 Pre-processing.....	192
5.3.4.4 Implementation	193
5.3.4.5 Accuracy Scores	203
5.3.4.6 Discussion.....	203
5.3.4.7 Unsupervised Learning for breakpoint Detection.....	205
5.3.4.8 Data Transformation	207
5.3.4.9 Implementation	208

5. 3.4.10 Discussion.....	213
Chapter 6 Validations	216
6.1 Chapter Overview.....	217
6.2 XAI with Shapley Additive Explanations.....	218
6.2.1 Prediction Explainer	218
6.2.1.1 Local Interpretability.....	218
6.2.2 Model Explainer.....	219
6.2.2.1 Global Interpretability.....	219
6.2.2.2 Elapsed Time Effects.....	219
6.2.2.3 Dependence Plot.....	220
6.2.2.4 Decision Plot.....	221
6.2.2.5 Summary Plot	221
6.3 XAI with Local Interpretable Model-Agnostic Explanations.....	222
6.4 Explaining Model’s Decision Making.....	224
6.4.1 Visualizations of Leaves Impurities.....	225
6.4.2 Decision Tree Regressor.....	229
6.5 Feature Importance	232
6.6 Discussion.....	237
Chapter 7 Conclusion	241
7.1 Research Contributions.....	244
7.1.1 Hypothesis 1.....	244
7.1.2 Hypothesis 2.....	246
7.1.3 Hypothesis 3.....	249
7.2 Additional Work and Scope of Further Development	254
7.2.1 Ontological Approach for Data Provenance.....	256
7.2.2 Ontological Approach for Analytic Provenance.....	258
References.....	260

List of Figures

1.2	Bridging the gap between computation (TS) and cognition (NTS) for human in the loop visual analytics.....	19
2.1	Sensemaking loop for intelligence analysis.....	30
2.2	Hierarchical analytic provenance model.....	36
2.3	The Jigsaw list view showing connections between concepts,	46
2.4	The Aruvi information visualization framework.....	49
2.5	Scalable Reasoning System (SRS) web client.....	51
2.6	GeoTime and HTVA.....	53
2.7	Temporal relationships and links across different sections of LifeLines.....	54
2.8	SensePath and Vistories.....	55
2.9	TS and NTS scores.....	61
2.10	A generalizable model for coupling cognition and computation.....	62
2.11	Discretizing two clickstreams into event sequences.....	65
2.12	Illustration of SC, GC and SCM.....	67
2.13	Task forming time.....	68
2.14	Progger log showing user scenarios.....	69
2.15	Hierarchy of the behavioural clusters.....	70
2.16	Whisper behavioural clusters.....	71
2.17	Accuracy of classifiers – machine learning for inference making....	73
2.18	Reasoning task classification.....	75
2.19	Finding Waldo – machine learning for visual analytic systems.....	77
2.20	The attention model.....	79
2.21	Global attention model.....	80
2.22	Local attention model.....	81
2.23	Scaled Dot-Protect and Multi-Head Attentions models.....	82
2.24	The Transformer – model architecture.....	83
2.25	Differences among BERT, OpenAI GPT and ELMo in pre-training model architecture.....	84
2.26	Overall pre-training and fine-tuning procedures for BERT.....	85

2.27	Taxonomy mind-map of Machine Learning Interpretability Techniques.....	89
2.28	LIME - Explaining individual predictions.....	90
2.29	LIME equation and proximity calculations.....	92
2.30	SHAP local explanation based on assigning a numeric measure of credit to each input feature.....	93
3.1	Typical analytic task model.....	101
3.2	PROV - Four linked views for analytical provenance capture and representation system.....	103
3.3	Proposed analytic task model and RRP system.....	105
3.4	The park map, visualization paradigm, temporal view, spatial view.....	110
3.5	Park visitor's check-in visualizations.....	111
3.6	Spatial determinacy problem.....	113
3.7	Temporal determinacy problem.....	114
3.8	PROV – System architecture, internal system function calls.....	121
3.9	PROV - Simplified state event sequence diagram upon interactions on visualizations into Analyst's user Interface (AUI)...	123
3.10	PROV - Provenance data flow diagram of Analyst's User Interface (AUI) back-end.....	124
3.11	Manually captured states panel with annotation add/edit and automatically captured log panel for Analyst's User Interface (AUI).....	126
3.12	Analytic path showing annotations set by analysts with captured states & their relationships.....	127
3.13	Schematization of analytic path in a visuo-spatially manner.....	129
3.14	Repetitive Replicating Playback (RRP) System shows results with source state id information after running batch of saved group of states.....	132
3.15	Visual representation of saved RRP batches of captured states and tracing those back by time gliding or by selecting colour coded users (analysts) or by keyword searching and selecting from RRP list.....	132
4.1	Means-Ends abstraction hierarchy to illustrate the decomposition approach to identify Behavioural Markers (BMs).....	147
4.2	An analytic path showing annotations set by analysts with captured states & their relationships.....	153
4.3	Indegrees [$\text{deg}^- (V)$] of action sequence graph indicative of restoring previous analytic states.....	155
4.4	Outdegrees [$\text{deg}^+ (V)$] of action sequence graph indicative of generating more alternative approaches.....	156
4.5	Calculating centrality or approximate importance of an action sequence graph.....	157

List of Figures

4.6	VALCRI’s phase-1 system evaluation: User feedback approach based on open-ended questionnaire to identify how it’s AUI system encourages or hinders insight, creativity and imagination.....	166
4.7	VALCRI’s phase-1 system evaluation: NASA-TLX Mental Workload Rating Scale.....	167
5.1	Evaluation results of Popularity Model on first 10 out of 1139 users.....	178
5.2	Popular Topics.....	179
5.3	Relevance measure of top 10 tokens for user profile.....	180
5.4	Evaluation results of Content-Based Filtering Model on first 10 out of 1139 users.....	181
5.5	Evaluation results of Collaborative Filtering Model on first 10 out of 1139 users.....	182
5.6	Evaluation results of Hybrid Filtering Model on first 10 out of 1139 users.....	183
5.7	Comparison of Top-N accuracy values calculated as Recall@N from 100 random test data by using data filtering models.....	184
5.8	Attention to next word at layer 2, head 0. Left: attention weights for all tokens. Right: attention weights for selected token (“my”)...	194
5.9	Attention to previous word at layer 6, head 11. Left: attention weights for all tokens. Right: attention weights for selected token (“dishes”).....	195
5.10	Attention to identical/related tokens at layer 2, head 6. Left: attention weights for all tokens. Right: attention weights for selected token (“youtube”).....	196
5.11	Attention to identical/related words in other sentence at layer 10, head 10. Left: attention weights for all tokens. Right: attention weights for selected token (“chain”).....	197
5.12	Attention to other words predictive of word at layer 2, head 1. Left: attention weights for all tokens. Right: attention weights for selected token (“re”).....	198
5.13	BERT base model visualizations for 12 layers and 12 heads resulting in a total of $12 \times 12 = 144$ distinct attentions for Text A and Text B.....	200
5.14	Text A and Text B focused next word attention pattern at layer 2, head 0 of the BERT-base pre-trained model.....	201
5.15	Elementwise and dot products of query (q) and key (k) vectors for next word attentions at layer 2, head 0.....	202
5.16	BERT classification reports.....	203
5.17	Detecting change points.....	205
5.18	Topic predictions visualization by using LDA model while relevance metric $\lambda=1$	206
5.19	t-distributed stochastic neighbour embedding (t-SNE) visualization of inferred chunks in 2D.....	207
5.20	Autoencoder evaluation results for inferring breakpoints.....	209

6.1	SHAP local interpretation for <code>X_train.iloc[0,:]</code> & <code>X_train.iloc[421,:]</code>	219
6.2	Shapley value - Global interpretability and Elapsed Time Effects....	220
6.3	Dependence plot and decision plot.....	221
6.4	Summary plot and LightGBM feature importance.....	222
6.5	LIME - Tabular representation and text highlights for <code>test.loc[310]</code> .	223
6.6	Scikit-learn visualization of decision trees for <code>max_depth=4</code> , <code>random state = 310</code> (<code>test.loc[310]</code>).....	226
6.7	<code>dtreeviz</code> visualization of decision tree classifier for <code>max_depth=4</code> , <code>random state = 310</code> (<code>test.loc[310]</code>).....	227
6.8	Gini purity for each leaf, leaves purities distribution, number of leaves grouped by target class and leaves sample distribution.....	228
6.9	Prediction path visualization for <code>random state = 310</code> (<code>test.loc[310]</code>).....	229
6.10	Feature importance based on prediction path nodes, leaf target distribution for regression decision trees, number of samples from each leaf, Mean Absolute Error (MAE) for each leaf.....	230
6.11	Decision tree regressor for <code>max_depth=4</code> , <code>random state = 310</code> (<code>test.loc[310]</code>).....	231
6.12	ELI5 - Global feature importance, local TextExplainer for <code>train.values[1000]</code> with Ridge classifier.....	234
6.13	ELI5 - Global feature importance, TextExplainer with Random Forest classifier.....	235
6.14	ELI5 - Explanation as decision tree (partial view), feature importance of <code>X_test.iloc[310]</code> , <code>X_test.iloc[1]</code> for decision tree classifier.....	236
7.1	Thesis contributions.....	243
7.2	ProvViz - An analytic state suggestion system (GST View) and it's preliminary ontology development.....	255
7.3	Preliminary version of analytic provenance ontology for 'PROV' – WebProtege class view, WebVowl visualization.....	257

List of Tables

2.1	Evaluation hypotheses, data sources and analysis techniques.....	59
4.1	Behavioural Attributes.....	144
4.2	Observable behavioural markers and their constructs for criminal intelligence analysis.....	149
4.3	Description of behavioural constructs.....	160
4.4	Observed analysis techniques followed by crime analysts.....	163
5.1	Chrome browsing history information.....	192
5.2	Un-supervised model performances of inferring breakpoints.....	210
5.3	Classification reports after hyperparameter optimization.....	214

VALCRI

VISUAL ANALYTICS FOR SENSE-MAKING
IN CRIMINAL INTELLIGENCE ANALYSIS

The VALCRI project was funded by the European Commission to undertake R&D with a view to developing an integrated software support system for police forces across partner countries. This software system, known as VALCRI, will be used by police analysts to investigate crimes and crime-related behaviour, complementing and enhancing current police capabilities. The consortium includes partners and activities aimed at designing the technology from cognitive, legal, ethical and privacy perspectives so that the rights of the individual to security and liberty will be respected while ensuring the good of society. It will also enable law enforcement agencies to make their processes more transparent, so that the processes by which their conclusions are reached are made easier to inspect.

The purpose of Project VALCRI was to create a Visual Analytics-based sense-making capability for criminal intelligence analysis by developing and integrating a number of technologies into a coherent working environment for the analyst named as the Reasoning Workspace. Conceptually, the Reasoning Workspace comprises three areas: (i) a Data Space which will enable an analyst to see what data and themes exist, (ii) an Analysis Space to which data can be brought into to carry out various computational analyses including statistical and text analysis, and (iii) a Hypothesis Space that will enable the analysts to assemble their evidence into coherent arguments that lead to meaningful and valid conclusions.

At the cutting edge of intelligence-led policing, VALCRI is a semi-automated analysis system that helps find connections humans often miss. When pre-empting crime or investigating a case, it can be deployed by analysts to reconstruct situations, generate insights and discover leads.

Through autonomous work or collaboration with a human team, VALCRI creatively analyses data from a wide range of mixed-format sources. It displays its findings with easy-to-digest visualisations, comes up with possible explanations of crimes, and paves the way for rigorous arguments. Protecting against human error and bias, VALCRI works with objective intelligence, speed and precision.

Project Links -

* Website: <https://www.euprojectvalcri.org/>

φ EC: <https://cordis.europa.eu/project/id/608142>

"The research leading to the results described in this report has received funding from the European Union Seventh Framework Programme (FP7/2007-2013) to Project VALCRI under the EC Grant Agreement N° FP7-IP-608142 awarded to Middlesex University and partners."



VALCRI Context and Objectives

The purpose of VALCRI was to develop the next generation criminal intelligence analysis system for European LEAs. Working closely with three European police forces, the project researched and developed at TRL-5, an integrated system of over 75 software components of advanced data processing, analytic and sense-making tools. It includes multiple applications spanning strategic intelligence analysis to tactical intelligence and individual case management.

The VALCRI system was routinely evaluated with project end-users. In the final nine months, it has been evaluated with 214 LEA officers in 50 agencies in 16 countries and 2 international LEAs (Europol and NATO Intelligence Fusion Centre). VALCRI used a cognitive engineering approach to create a human-technology team that combined advanced concepts of human reasoning and analytic discourse with machine learning and database technologies. The result has been a semi-automated human-mediated semantic knowledge extraction capability that can facilitate and improve investigative sense making and problem solving in crime analysis and criminal investigation in a high ambiguity and constantly evolving environment.

Key Distinguishing Features

1. Support How Analysts Think, Rather Than What Analysts Do

If VALCRI were designed to mainly support what analysts do, then the system would primarily automate current tasks and workflows. Instead, by designing for how analysts think, the VALCRI system is better able to respond to the variety of sense making, reasoning and inference making and problem solving strategies presented by human analysts.

2. Facilitate Expert Intuition To Scientific Method

In many investigations, analysts are often only presented with fragments of data from which to create an understanding of the situation and to anticipate what might happen. Expert intuition is very useful in generating “hunches”, or early, plausible and tentative hypotheses. However, hunches can be error prone and subject to cognitive biases. VALCRI has designed quick ways for analysts to use the scientific method to test their hunches so that they may easily discard it if proven wrong.

3. Humans Decide, Machines Do The Heavy Lifting

VALCRI has been designed so that humans and machines do what each is good at: Humans make decisions under ambiguity; machines are fast at tedious and repetitive task. So, when an analyst instructs VALCRI to “find me more reports like this ...”, the machine learning-based automation will trawl through large volumes of structured and un-structured data (e.g. free text) to retrieve, triage, collate, thematically analyse the data, and then combines and presents the reports in context of the crime problem being investigated, e.g. Comparative Case Analysis.

4. Ethics, Legal and Privacy By Design

In many LEA data analytics systems, once a person’s data is enmeshed in the system-data networks, that person will continue to be linked to those criminal profiles. Such profiles will be used by the system to predict membership characteristics and to set up alerts for “persons of interest”. This can lead to further stops and searches of the person, even though he may be innocent. This interferes with his private life. VALCRI advocates the need for ‘computational transparency’ as a mitigating approach: make visible the inner workings of ‘black box’ automated algorithms. A lower TRL prototype has been implemented in VALCRI to investigate how fine grain data access controls may be combined with computational transparency so that analysts and investigators are aware of the provenance of algorithm’s computed results and protect the rights of individuals.

5. Up-skilling of Analytic Abilities

VALCRI has also identified and addressed varying deficiencies in the abilities of the intelligence analysis community. Some of this have been formalised in a new Master degree level analytics training course at Aston University in Birmingham, in partnership with the West Midlands Police; and some have been formalised into commercial intelligence analysis training packages focusing on analytic reasoning.

6. Research Data

Partner AES worked with West Midlands Police to make anonymous three years of actual police data: over 1 million crime reports including structured and un-structured data, and over 6 million ANPR records. This data includes spelling errors, duplicates, similar but different data, and so forth. This dataset has been a crucial enabler.

Project Objectives

OBJECTIVE 1: Human Issues Framework

(a) Ethics, Privacy, Law. Comparative analysis of law in Germany, Belgium and UK, led to specification of legal requirements in VALCRI; Evaluated impact of removing ethically sensitive data from data analysis; Developed understanding of Ethics by Design in VALCRI; Set up *Ethics Working Group* in West Midlands Police to assess ethical issues in criminal intelligence analysis; (b) Cognitive bias and sense making. Operationalized insight, imagination, fluidity and rigour, transparency for experimental evaluation; Evaluated visualisation designs for insight, sense-making, cognitive bias, structuring of arguments.

OBJECTIVE 2: Analyst User Interface

A suite of AUI tools based around the reasoning workspace developed to orchestrate ML and database capabilities with interaction and visualisation functions to facilitate analytic reasoning and investigative sensemaking. The AUI tools include maps and timelines, network evolution, dispersion diagrams, and statistical process charts, with touch-enabled, multiple-coordinated views. It is designed to encourage analysts to ask questions – an important part of sense making and coping with ambiguity.

OBJECTIVE 3: Semantic search and retrieval

Semantic search capabilities include an interactive dimension-reduction tool for data exploration and sense making with the Knowledge Generation Model. ML algorithms applied to read and select appropriate texts from crime reports, show feature set and create a first draft Comparative Case Analysis table. Associative Search identifies new associations or links between criminal entities by exploiting information, criminal behaviour, modus operandi, geographical and temporal proximity, and associations between unsolved crimes and offenders to generate suspects lists.

OBJECTIVE 4: Crime situation re-construction

Developed a method for visual storytelling using argumentation theory to assist with the re-construction of crime situations. Explanations comprising fragments of data can then be formulated into defensible assessments. It enables analysts to record their evolving reasoning during investigations based on inferences from data, visualisations, and can be linked to conclusions through inferential networks.

OBJECTIVE 5: Secure, scalable and distributed architecture

The security architecture is implemented through OpenPMF with a Domain Specific Language DLS to configure Attribute and Proximity Based Access Controls (ABAC, PBAC) that translates human readable security policies into machine enforceable code; PET (Privacy Enhancing Technologies) to rapidly anonymise or pseudonymise data so it can be used without

compromising privacy; HALA security test-bed set up for High Assurance Logging and Audit method based on a 'Vault' to provide hardware separation.

OBJECTIVE 6: Anonymised dataset

Partner WMP supplied three years of actual fine-grain police data comprising over 6 million crime reports and others, and over 58 million ANPR records. Led by AES, the data was anonymised at a deep level. This dataset was used in the development of the VALCRI system. However, internal tests showed that it was possible to de-anonymise the data. For confidentiality reasons, the data will not be released to the research community.

Other Results

Harvester - A stand-alone application where police users can search and mark up interesting text in PDF documents, harvest and store in a knowledge base.

Analysts Training Courses - The VALCRI Analytic Reasoning Training Curriculum (TN 13.4) has been developed into commercial courses: i-Intel's 3-day CPD courses in intelligence analysis have been evaluated with 123 LEA officers in 40 agencies in 13 countries; A Master-level Advanced Analyst qualification had been developed by AES, WMP, and Aston University, Birmingham.

Provenance - Recording, playback and state saving features integrated at TRL-5, with advanced analytic provenance being researched (TRL 2-3).

Project Results and Impacts

1. The main outcome is an integrated multi-application criminal intelligence analysis system at TRL-5. Using a cognitive engineering approach, we implemented the concept of a joint cognitive system, demonstrating how mixed-initiative systems can be developed to enable proactive and reactive system behaviours to create a human-machine team. This creates a test-bed for further research: (i) study the impact on operational use of criminal intelligence analysis systems of how the laws and privacy regulations are implemented, (ii) advancement of the semantic search algorithms, (iii) inclusion of formal concept analysis techniques to associative search, (iv) application of hybrid AI techniques to semantic knowledge extraction, (v) investigate alternative methods for storytelling and argumentation to support work with uncertainty, ambiguity and deception, (vi) It will also create opportunities to re-factor the integration platform code to enable plug and play capability, (vii) provide an environment for police to experiment with new methods based on the new VALCRI capabilities, (viii) use behavioural markers for automatic classification of analytic reasoning activities from user interactions with the system.
2. The VALCRI system is not one single application, but a complex multi-application industrial scale system using the following technology stack: Java, Javascript, GWT and ERRAI, Docker containers, RESTful interfaces, Jena/Fuseki RDF triple store, MongoDB, SQL Postgres DB with Elasticsearch, OpenPMF and a Central Authority Service, Graylog, NLP pipeline for concept extraction, ML-based semantic search functions.
3. Training courses have been developed around the analytic reasoning research in VALCRI. These courses are in high demand. New insights about analytic reasoning and new VALRI technologies have created opportunities for new techniques to be developed. By embedding the knowledge into CPD and Master-level courses, opportunities are being created for propagating the knowledge beyond the police intelligence communities.
4. Research into legal, ethical and privacy requirements in Europe has identified key issues and translated them into system design specifications and implementation trade-offs e.g. how to show data or node in a network visualisation graph that may be confidential for security, privacy or ethical reasons?

5. Cognitive engineering research has helped to understand how analysts think. This has enabled us to design how software might facilitate the reasoning in uncertain, ambiguous and deceptive environments through designs that encourage the asking of questions.
6. Partners have implemented different methods for semantic knowledge extraction and associative search. This opens opportunities for new research e.g. computational transparency – how we make the results of black box automated analyses understandable and verifiable by users; computational steering of algorithms such as the use of sub-space clustering methods to discover low frequency but operationally significant events; use VALCRI as a test-bed for investigating hybrid intelligent technologies in a joint cognitive system approach; navigating uncertainty when using the products of such methods given ambiguous and deceptive situations.
7. WMP provided real data that was large and complex enough for developing real systems. The data was anonymized and used to develop the VALCRI prototype system. However, internal evaluations determined that the data could be de-anonymized due to the richness of the data contained in the un-structured text. Therefore the anonymized data cannot be released to the research community as originally planned.
8. Exploitation. A variety of IP has been produced with plans for commercial exploitation and further research. Instead of tying partners down to the usual single exploitation plan, an exploitation agreement was reached for VALCRI that freed partners to exploit the IP they owned as they wish. The 9-point agreement is based on three ideas (a) freedom to commercially exploit IP that is individually owned, (b) freedom to join another partner to create products or services that create commercial value, and (c) profits to be shared only by those who generated the profit.
9. Impact. Most significant is the independent decisions by the Metropolitan Police Service London and the Pasco County Sheriff's Department in Florida to adopt the VALCRI system for trials with actual data. The VALCRI system was installed at both sites. They are in the process of ingesting actual data to solve actual cases. They are not members of the project consortium and are not obliged to adopt nor trial the VALCRI system.

Summary of VALCRI Achievements

- i. **The VALCRI System Prototype** - VALCRI has been designed around a knowledge extraction engine which uses machine learning techniques for semantic similarity analysis undertaken in both reactive and proactive modes with the analyst. Crime-related data are stored in two databases: an unstructured database (UDB) for free text fields and video data, and a structured database (SDB) for structured text and data extracted by parsing free-text. A combination of Open Source technologies is being adapted and integrated to undertake varied forms of data analysis across different crime categories and multiple data sets. All components have already been built for semantic data mining, associative search, and Comparative Case Analysis (CCA).
- ii. **VALCRI Technology Readiness Level (TRL)** - The majority of the VALCRI prototype will be functionally integrated into a single TRL-5 platform by the extended project end date. The problems addressed by software components developers have proved to be more difficult than anticipated and so the entire system is now being developed primarily at TRL-5, with those components at lower TRLs being made available on separate branches.
- iii. **VALCRI User Interface** - The VALCRI user interface (UI) design is based on the concept of tactile interaction, driven by a visual analytic perception-action cycle, guided by the fluidity and rigour model. The design has been further informed by principles and requirements from user practice, human factors and psychology principles, and our own studies of analytic reasoning and sensemaking. The design has been implemented in the GWT (Google Web Toolkit) environment, within which we have developed the Analyst User Interface (AUI). This

manages the windowless AUI environment where data records fluidly transition into abstract visual representations on dual screens which can be manipulated to carry out numerous analytical operations. The AUI is further integrated with dynamic visual querying techniques for fast response times across multiple-coordinated and faceted views involving maps and timelines, statistical process charts, crime hotspot analyses, and dispersion diagrams. These tools help the analyst to generate and test the logic of explanations that connect assemblies of propositions, data and assumptions, structured and presented in ways to facilitate inference making, storytelling, the creation of explanations, and the formulation and testing of hypotheses.

- iv. **Ethical, privacy and legal issues** - Studies have been undertaken to compare applicable laws in Germany, Belgium and the UK to determine how legal principles may be implemented within the prototype. These include: purpose limitation, data minimization, the treatment of data subjects, handling of ethically sensitive data, and data storage and deletion.
- v. **Security Test-bed for High Assurance Logging and Audit** - A security test-bed has been set up in a Berlin location by partner Object Security to develop and test secure logging method, referred to as the High Assurance and Logging Auditing to create secure crime analysis logs that cannot be tampered with. Object Security has designed the 'Vault' which provides hardware separation through trusted key storage, high performance, trusted crypto operations, trusted mass storage, trusted user I/O, and trusted processing. This permits all system log data to be sent in real time to the Vault from application/middleware, and optionally from kernel modules.
- vi. **Patent** - The project partner Object Security has registered a patent with the US Patent and Trademark Office based on research undertaken as part of the VALCRI project. It described a system and method for managing the implementation of policies in an IT system by automatically or semi-automatically generating machine-enforceable rules and/or configurations. This is being adapted to translate European laws and regulations into rules that can guide access to crime-related data in VALCRI.
- vii. **Anonymised Datasets** - Three years of police data, comprising over 6 million crime reports, stop and search, stolen property reports, intelligence reports, nominal, and custody reports, and over 58 million ANPR records, have been anonymised at a deep level by VALCRI partner AES from raw data supplied by West Midlands Police. Unlike most publicly available crime data, these are fine-grained, and are being used by VALCRI partners to undertake research, and develop and test the prototype in readiness for operational use by LEAs. Tests are currently under way to determine whether the procedures used to anonymise these data can resist de-anonymisation. At the end of the project, the VALCRI data set will be made available to the broader research community.
- viii. **Development Environment and User Access** - The VALCRI software development environment is hosted at three partner locations: London, Linköping, and Brussels. The primary project source code is stored at Middlesex University, and managed through GitLab. Developers with sufficient machine resources can pull the code from Middlesex and images from Space and run the full stack locally. Resource-limited partners can get some of the images to run locally, and connect to running versions of the other images hosted at Linköping. All users, whether analysts or non-technical partners, can accessing VALCRI in two ways: use a web browser to access a release version (TRL-5) on a server hosted at SPACE (via VPN access) who is a project partner; or download and run it locally on their own machines.
- ix. **VALCRI Deployed at Police locations and Consortium Partners' locations** - The VALCRI system prototype, comprising the Analyst Workstation has been deployed in all three police end-user environments so they can learn to use the software in their own time. They will initially use the VALCRI-developed crime dataset, and will migrate to using larger samples of old but real data when appropriate security procedures have been established. This will help them determine what ways VALCRI assists or hinders the criminal intelligence analysis process. For security reasons, the VALCRI prototype will not be connected to any live police

systems. The VALCRI system prototype has also been deployed to all other VALCRI partners to enable local familiarisation, and to enable partners to use it for carrying out experiments and studies.

- x. **Analyst Training Courses** - Partners involved in commercial training for intelligence analysis have developed multiple courses. Eight workshops have been run for police analysts. Additionally, a Masters-level (Level-7) Advanced Analyst training qualification has been developed in conjunction with VALCRI police partner, West Midlands Police, and Aston University, Birmingham. The course will include subjects in criminal behaviour, criminal networks, crime linking, crime and criminal profiling, from a critical thinking perspective in the context of data science.

Introduction

1

chapter

1.1 Background

Visual Analytics is playing a major role in providing insights into the relationships of complex data sets across a number of domains. It helps making sense of complex systems interactions and interrelations, by utilizing systems thinking [67] during analytic processes. A central concept of visual analytics is that the development of human insight aided by interactions with visual interface, and the steps that a user takes to discover insights, are often as important as the final product itself [68]. The analytic processes not only provide relevant information on individual insights but also how the users arrive at these insights. The area of research that focuses on understanding of user's reasoning process through the study of their interactions with a visualization is called '*Analytic Provenance*' and has demonstrated great potential in becoming a foundation of the science of visual analytics. It is a broad topic and has many meanings in different contexts. On the otherhand, visual analytics is the science of analytical reasoning facilitated by interactive visual interfaces [14] which are powerful means of making sense of data. But it is an obvious challenge to track and utilize those analytical data due to complexity of the event-driven systems around us. Because those systems are computationally extensive where data flows from one process to another as it is transformed, filtered, fused and used in complex models in which computations are triggered in response to events. Without appropriate support and technique for capturing the deluge of event-driven data, it becomes difficult to render an opaque reasoning process transparent such that analysts can view, trace and probe how conclusions came about [69]. Alongside the problem of maintaining transparency, it also becomes crucial to detect and mitigate pitfalls such as human biases, that can lead to errors in data assessment and making judgements [70]. This can occur during an analytical scenario where solutions are found through serendipity instead of a rules. These incorporate uncertainty into visual representation of data that may lead to erroneous insights. So, accuracy and precision of adopted visualization techniques have got a greater role in trustworthiness of the outcome [71]. Visual analytics in such

cases can be improved via better understanding of behaviour during the analytic process in support of sensemaking. Provenance can be used for self-reflection, exploration guidance which can also support collaboration and help to understand what can be trusted from possibly uncertain data.

This research has considered above problems and leveraged analytic provenance as the means for providing insight into data processing operation in question with the aim to contribute it's results to the project *VACLRI.

In criminal intelligence analysis provenance is one of the best means to provide necessary support to explain in a clear way how decisions or choices were made, what they were based on, how steps in a selection process were made, provide information grounds to justify and answer claims of bias or discrimination, and show compliance. All these are enablers of fairness and lawfulness of the data processing activities from the legal framework. Transparency in criminal intelligence analysis is an important requirement for maintaining respective LEP (*Legal, Ethical, and Privacy*) guidelines. This is the property that all operations on data including legal, technical, organizational setting and the correlating decisions based on the results can be understood and reconstructed at any time. So, '*Transparency*' can be regarded as the underlying foundation of the analytical provenance. Analytical activities performed by analysts should be recorded for supporting '*Accountability*' for particular action of analysis process. Analytical provenance data has got greater influence in this regard [72].

Capturing analytical provenance has also got a significant role in criminal intelligence analysis, because the legal directive foresees an obligation to provide competent legal authorities with information about the processing operation upon request. Competent authorities are any public authority or any other entrusted body by national law to exercise public authority and public powers for the purposes of the prevention, investigation, detection or prosecution of criminal offences or the execution of criminal penalties, including the safeguarding against and the prevention of threats to public security. Analytical provenance data can help to validate the processing operation in such case [72].

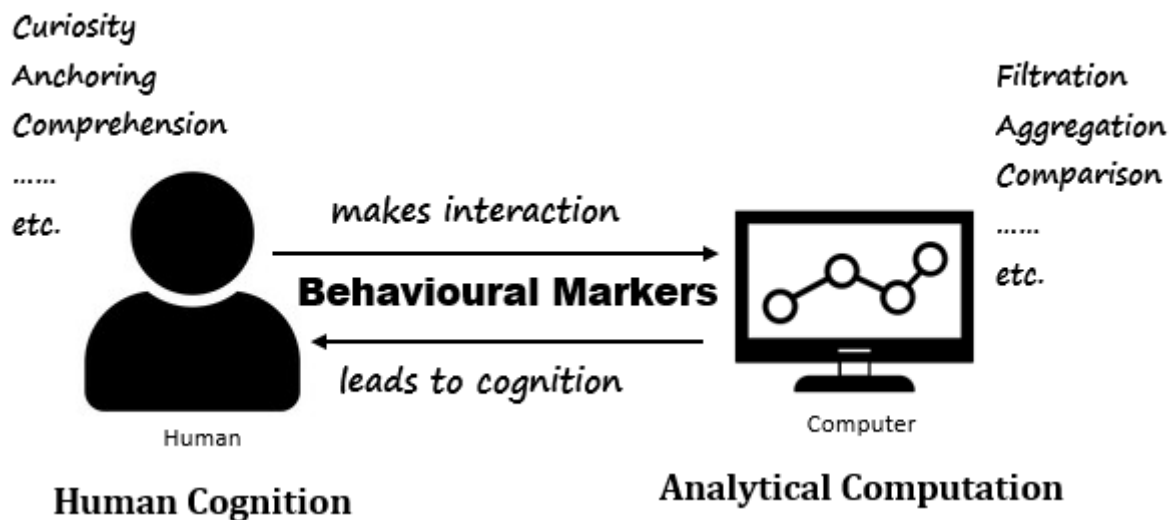


Figure 1.1: Bridging the gap between computation and cognition for human in the loop visual analytics.

1.2 Research Problems

The central research question (RQ) of this thesis is –

How can analytic provenance be leveraged for detecting sensemaking behavioural markers in visual analytic systems?

This research addresses the challenges of working on uncertain data visualizations within ‘*Legal, Ethical & Privacy (LEP)*’ framework and maintaining process ‘*transparency*’ of outcomes to be plausible by using visual analytic systems. To find out solutions of these challenges this research aims to investigate – how to capture and utilize large flow of analytic provenance data? how can different cognitive constructs be translated in terms of computational interactions? how to pinpoint transitions among those cognitive constructs? and how can those concepts be utilized to model user profile and understand their analytic behaviour? The aim is to bridge the gap between human cognition and analytic computation as shown in figure 1.1. We have aimed to leverage analytic provenance data for this purpose as it can bridge between higher-level logical constructs and the lowest-level user

interaction events as claimed by Gotz & Zhou et. Al [10]. The approach is to detect sensemaking '*Behavioural Markers* (BMs)' which are ingrained with different cognitive reflections through interactions. We have formulated couple of hypotheses (H) to test our approach and several research questions (RQs) to carry out this research. Brief descriptions of those outlining all research problems are as followed:

H1: Capturing user's interactions with a visual interface can retrieve some aspects of the transparency of user's reasoning processes in intelligence analysis.

H1 demands logs of the individual processing activities within an automated processing system to complement the information entailed and to provide enhanced transparency of the operations.

- **RQ1:** How to develop a system that tackles large flow of heterogeneous analytical data and supports W3C PROV-AQ^T?
RQ1 outlines the requirements and development challenges of front-end techniques, back-end modelling for generically capturing different complex visual analytical states, automatically processing, storing as well as recalling those as per query to maintain traceability.
- **RQ2:** How to utilize captured analytic provenance data for sensemaking?
RQ2 seeks to find out techniques of utilizing captured analytic data to support transparent sensemaking, mitigation of uncertainties in visualizations and build trust on visual analytic systems.

H2: Behavioural Markers (BMs)' can act as attributes for bridging between human cognition and analytic computation through interactions during fluid transitions between mental and analytic processes at the micro-analytic level.

H2 addresses the gap between cognitive constructs and manipulations or interactions humans employ to think and reason about data as identified by many researchers and proposes that constructs of '*Behavioural Markers* (BMs)' can bridge such gap alongside performance measurement.

^T <https://www.w3.org/TR/prov-aq/>

- **RQ3: What are the constructs of Behavioural Makers (BMs)?**
RQ3 aims to identify and form an exhaustive list of behavioural constructs for criminal intelligence by considering human factors and cognitive engineering principles.
- **RQ4: How to translate reasoning processes to Behavioural Markers (BMs)?**
RQ4 seeks to find out the way of sequencing captured analytic actions for meaningful representation of BMs.
- **RQ5: How to externalize thinking processes from the constructs of Behavioural Markers (BMs)?**
RQ5 focuses on representation of interaction network to visualize analytic steps as well as extract BMs by using translation theories of RQ4.

H3: Inferring chains of low-level analytic actions can be of assistance for understanding multi-tasking behaviour.

H3 leads us to understand what the user is trying to do by classifying system interactions at different granular levels. It will pinpoint user's cognitive transitions as a way of chunking action streams at lower level and then classifying those.

- **RQ6: How can meaningful units of task execution be produced from captured interaction logs?**
RQ6 addresses the fundamental problem of finding out the way of breaking down a search session into meaningful chunks to detect user's task switch points.
- **RQ7: How precisely multi-task switches be inferred during execution of interactive tasks?**
RQ7 targets to prove/disprove the above hypothesis (H3) and evaluate the results by developing machine learning models for both known and unknown task scenarios.
- **RQ8: How to validate inference making results for building trust on machine learning models and maintain transparency?**
RQ8 seeks to find out techniques of explaining machine learning model's decision making processes, unfolding blackbox calculations of probabilities towards predictions, computing feature importance, understanding their local and global implications to show algorithmic transparency of machine learning outcomes for inferring task switch points from uncertain log dataset.

Thus it also helps to explain evaluation results found from RQ6 & RQ7 and prove their validities through human judgemental process.

1.3 Research Approach

This research endeavours to come up with visual analytic and machine learning solutions of above problems described into research questions by adopting hybrid (qualitative and quantitative) research approach and leveraging analytic provenance for understanding higher level complementary analytical behaviours unleashed from lower level [10] sensemaking interactions. Qualitative research approach includes conducting requirement analysis, creating an issue specific knowledgebase by carrying out focus group discussions and receiving feedback through structured interviews. Quantitative research approach aims to develop mathematical and machine learning models on theories found from the qualitative approach and compute those. We aim to evaluate all experimental findings both qualitatively and quantitatively where those apply and validate for the purpose of explanation and human judgement.

1.3.1 Data Used

Besides recording data during qualitative research sessions and analyzing afterwards, we also have used or captured following other sets of data from different sources that fits with the experimental settings of quantitative research approaches.

1.3.1.1 Geospatial-Temporal Crime Datasets

To understand various kinds of uncertainties that exist into dataset from geospatial and temporal dimensions, their effects on intelligence analysts while carrying out sensemaking activities, finding out tools and techniques to visualize data and leverage analytical activities on those for improved intelligence we have used datasets from following sources:

- Vast Challenge[‡] dataset on a fictitious crime incident to answer the questions of who, where, when, what and how etc.

- Anonymized police dataset found from the project VALCRI* to develop systems for real intelligence analysis.

1.3.1.2 User's Log Dataset

To understand visualization challenges of log dataset (known as analytical provenance) for supporting sensemaking, we have used/captured user's analytical activities from several systems under different scenarios as follows:

- VALCRI's* *Analyst's User Interface (AUI)* to capture and visualize complex form of log dataset from heterogeneous modules under large platform that supports real sensemaking tasks of police intelligence analysts.
- Deskdrop² log dataset³ which provides contextual data of multi-users to use for understanding how machine can perceive user's intention.
- Google Chrome bulk dataset captured by History View⁴ software to utilize user's internet browser based cumbersome sensemaking data and apply machine learning techniques for understanding user's cognitive transitions.

1.3.2 Summary of Contributions

This thesis contributes by proposing following tools, techniques, ideas and theories to address the central research question (RQ) on detecting '*Behavioural Markers (BMs)*'.

1.3.2.1 PROV for Analyst's User Interface (AUI)

PROV is a prototype analytic visual judgmental system for '*Analyst's User Interface (AUI)*' of the project VALCRI*. This module includes tools for capturing a large dataset of heterogeneous flows of analytical data. It is built on a proposed complex dataflow model, a temporal visualization of all captured reasoning states that supports W3C PROV-AQ[‡],

‡ <http://vacommunity.org/VAST+Challenge+2015>

² <https://deskdrop.co/>

³ <https://github.com/yunshuipiao/sw-kaggle/tree/master/recommend-system/datasets>

⁴ https://www.nirsoft.net/utils/browsing_history_view.html

a tool for replaying work-flows known as RRP (*Repetitive Replicating Playback*) built on proposed task model, a visualization of analytic path that can be schematized in a visuo-spatial manner to enable tactile reasoning. PROV has been developed through a step-by-step research approach and evaluated with the real police intelligence analysts to support aspects of transparency i.e, source, process, accountability, series of events in criminal intelligence analysis. This part of research addresses RQ1, RQ2. More details on development, contribution can be found in chapter 3 and section 7.1.1 .

1.3.2.2 Behavioural Markers (BMs) for Bridging Cognition and Analytic Computation

To capture tacit information that resides in a human analyst as they perform their analysis role, this research proposes '*Behavioural Markers (BMs)*' as attributes for bridging the gap between human cognition and analytic computation through interactions. BMs are commonly known as observable Non-Technical Skills that contribute to performance within an work environment. As part of the detection approach of BMs this research has contributed to the development of an exhaustive list of behavioural constructs by arranging a workshop, proposed a network graph based computational approach to detect cognitive constituents and translated those by using theories of psychology. As the computational approach is automated, but lacks expert judgement, so a CTA (*Cognitive Task Analysis*) based experiment has also been used (as part of VALCRI's* AUI system evaluation) to detect those observable behaviours manually. This part of research addresses RQ3, RQ4, RQ5. Further details on experimental results are available in chapter 4 and section 7.1.2.

1.3.2.3 BreakPoints (BPs) of Multi-tasking Behaviour

To gain a deeper understanding of human multi-tasking behavior, this research proposes to use the concept of *BreakPoints (BPs)* as semantic boundaries among chains of actions. BPs pinpoint internal transitions in perception and cognition which are hidden into captured analytic dataset as found from previous CTA. Machine Learning (ML) models have been developed to infer where those pinpoints are,

completely in data-driven ways. Both known and unknown task scenarios have been evaluated for these experiments. This part of research addresses RQ6, RQ7. More descriptions on BPs, ML experimental settings and detailed evaluation results are available in chapter 5 and section 7.1.3.

1.3.2.4 Computational Transparency and Human Trust

Building

The more explainable a model, the deeper the understanding that humans achieve in terms of the internal algorithmic procedures that take place while the model is making decisions. This is important for transparent validation of outcomes in terms of reliability, accuracy and relevance. To unfold all black-box calculations, 'eXplainable AI (XAI)' techniques have been used at the final stage of this thesis to explain model's decision making process, computing importance of different features or their influences on model predictions both locally and globally. Thus black-box calculations have opened up opportunities for making human judgements on evaluation results and building trust on machine produced results. This part of research addresses RQ8. More details can be found in chapter 6.

1.4 Publications

JOURNAL

-
- **Islam, Junayed**, Xu, Kai and Wong, B. L. William (2018) *Analytic provenance for criminal intelligence analysis*. Chinese Journal of Network and Information Security, 4 (2) . pp. 18-33. ISSN 2096-109X [Article] (doi:10.11959/j.issn.2096-109x.2018016).

BOOK SECTION

-
- **Islam, Junayed**, Wong, B. L. William and Xu, Kai (2018) *Analytic provenance as constructs of behavioural markers for externalizing thinking processes in criminal intelligence analysis*. In: Community-Oriented Policing and Technological Innovations. Leventakis, Georgios and Haberfeld, M. R., eds. SpringerBriefs in Criminology . Springer, pp. 95-105. ISBN 9783319892931. [Book Section] (doi:10.1007/978-3-319-89294-8_10).

CONFERENCE OR WORKSHOP ITEMS

- **Islam, Junayed**, Xu, Kai and Wong, B. L. William (2018) *Uncertainty of visualizations for SenseMaking in criminal intelligence analysis*. EuroVis Workshop on Reproducibility, Verification, and Validation in Visualization (EuroRV3). In: EuroRV3: EuroVis Workshop on Reproducibility, Verification, and Validation in Visualization (2018), 04-08 June 2018, Brno, Czech Republic. ISBN 9783038680666. [Conference or Workshop Item] (doi:10.2312/eurorv3.20181145).
- **Islam, Junayed** and Wong, B. L. William (2017) *Behavioural markers: bridging the gap between art of analysis and science of analytics in criminal intelligence*. European Intelligence and Security Informatics Conference (EISIC). In: 2017 European Intelligence and Security Informatics Conference, 11-13 Sept 2017, Dekelia Air Base, Attica, Greece. ISBN 9781538623855. [Conference or Workshop Item] (doi:10.1109/EISIC.2017.30).
- **Islam, Junayed**, Anslow, Craig, Xu, Kai, Wong, B. L. William and Zhang, Leishi (2016) *Towards analytical provenance visualization for criminal intelligence analysis*. Computer Graphics and Visual Computing (CGVC). In: Computer Graphics & Visual Computing (CGVC) 2016, 15-16 Sept 2016, Bournemouth University, United Kingdom. ISBN 9783038680222. [Conference or Workshop Item] (doi:10.2312/cgvc.20161290).
- Groenewald, Celeste, Anslow, Craig, **Islam, Junayed**, Rooney, Chris, Passmore, Peter J. and Wong, B. L. William (2016) *Understanding 3D mid-air hand gestures with interactive surfaces and displays: a systematic literature review*. HCI 2016 - Fusion! Proceedings of the 30th International BCS Human Computer Interaction Conference (HCI 2016). In: HCI 2016 - Fusion! 30th International BCS Human Computer Interaction Conference (HCI 2016), 11-15 July 2016, Bournemouth University, Poole, United Kingdom. . ISSN 1477-9358 [Conference or Workshop Item] (doi:10.14236/ewic/HCI2016.43).

1.5 Thesis Outline

This thesis is divided into following seven chapters:

- | | |
|------------------|---|
| Chapter 1 | Describes research background, problems, summary of intended contributions and publications. |
| Chapter 2 | Presents literature reviews on some of existing relevant research on analytic provenance visualizations, sensemaking, behavioural markers, machine learning models, inference making and eXplainable AI techniques. |

- Chapter 3** Includes research and development approaches of analytic provenance visualizations for sensemaking in criminal intelligence analysis, proposed underlying data-flow architecture, front-end analytic task model and system evaluation results.
- Chapter 4** Contributes to development approaches of sensemaking behavioural marker system including exhaustive list of behavioural constructs, their detection approaches both computationally and through qualitative experimental observations to externalize human thinking processes.
- Chapter 5** Describes experiments to understand human multi-tasking behavior by using machine learning techniques.
- Chapter 6** This chapter is an extension of previous chapter, which presents some machine learning model explanation methods to build trust on machine produced results.
- Chapter 7** Summarizes research contributions of all chapters, possible future works and concludes this thesis.



Literature Review



chapter

2.1 Analytic Provenance

In this section we define analytic provenance. We categorize it into two main types: (i) process provenance and (ii) reasoning provenance. We also review what types of analytic provenance are important to capture for intelligence analysis through literature survey.

2.1.1 Definition of Analytic Provenance

Analytic provenance captures the interactive data exploration process and human reasoning process involved in sensemaking [1]. As explained by Chang et. al. [2], analytic provenance describes methods for extracting user intent and reasoning from user behaviours and interaction logs. In the following sections we will differentiate between '*process provenances*', referring to the tracking of the data exploration process and the methodologies used by analysts, and '*reasoning provenance*' relating to the capturing of the human reasoning process.

2.1.2 Process Provenance

Görg et al. [3] explain that in order to better understand intelligence analysis, it is important to explore the methodologies of analysts as well as the fundamental processes they conduct. Process provenance refers to the procedural steps followed by analysts to achieve a given end.

2.1.2.1 Process Models

Several process models depict the intelligence analysis process in an abstract way. As Görg et al. [3] explain, most process models involve some form of iterative cycle of exploration, including steps such as planning and direction, data collection, processing, analysis and production, and dissemination. Pirolli et. al.'s [4] '*Sensemaking Loop Model*' for intelligence analysis as shown in Figure 2.1 has been widely cited and adopted by researchers within the visual analytics community. It consists of a linear set of states characterizing both data and process flow in an investigation. Analysts iterate through this process over the course of an investigation.

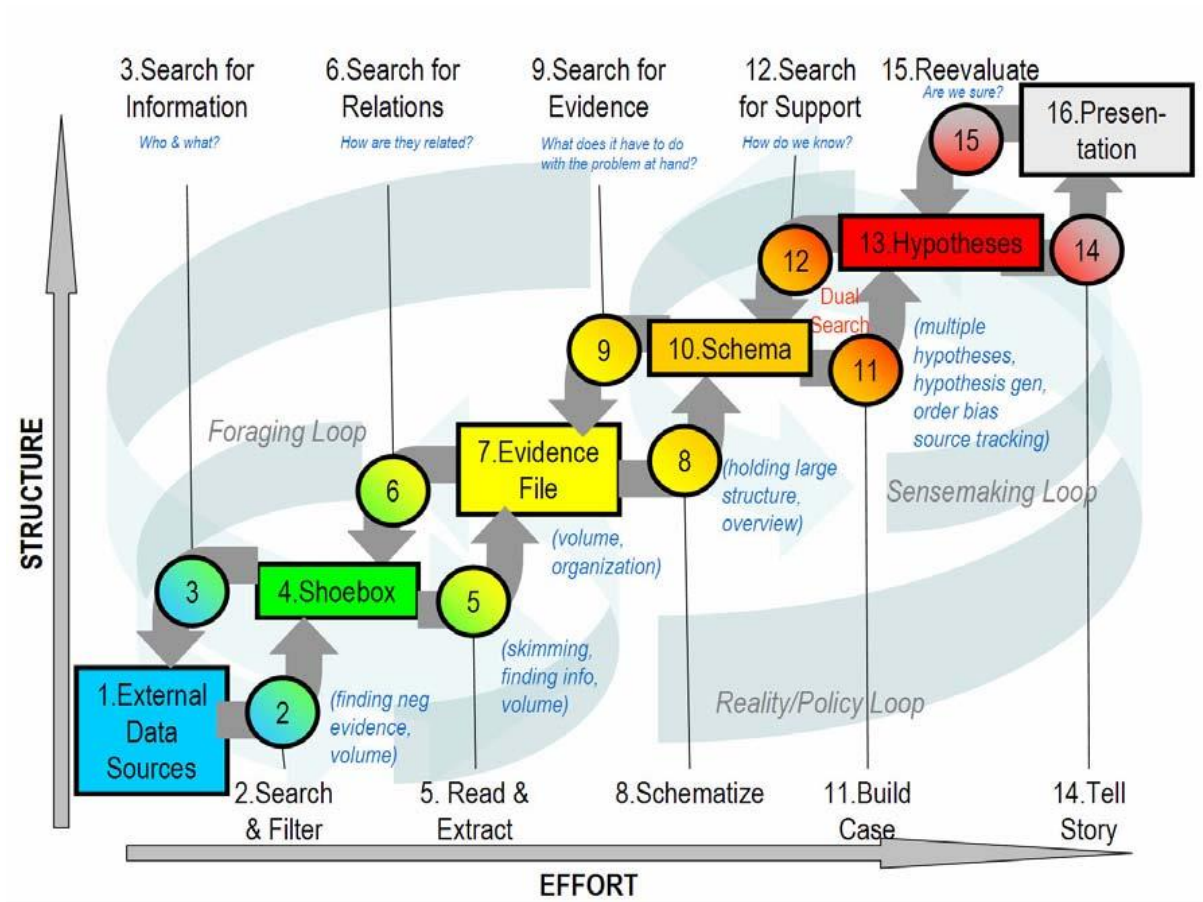


Figure 2.1: Sensemaking loop for intelligence analysis [4].

At a high level, the model contains two primary loops: a foraging loop in which analysts collect data and evidence, and a sensemaking loop in which analysts reflect on the data in order to generate schema and hypotheses about the situation and ultimately construct a presentation of the findings. Each loop contains three stages that further refine the process and both loops are connected through an overarching reality/policy loop.

The Sensemaking Loop Model is however subject to criticism and limitations. While useful and frequently cited, it describes the data transaction and information transformation processes in an abstract way, instead of describing how analysts work and how they transition from one step to another [6].

Görg et al. [8] mention alternative process models that could contribute to the analysis of process provenance. They mention the model of parallel tasks envisaged by Dr. Kristan Wheaton, Professor at the Department of Intelligence Studies at Mercyhurst College. Wheaton proposed an intelligence process model in which several tasks such as collection, analysis, and production stages take place in parallel, with different emphases over the course of an investigation.

Another alternative process model described by Görg et al. [8] is the one developed by Kang et. al. [7] from empirical observations. To better understand the analytical process and its requirements in the intelligence domain Kang and Stasko conducted their own qualitative user study.

They found that four processes dominated the overall workflow of the analyst:

- 1) Construction of a conceptual model
- 2) Collection
- 3) Analysis, and
- 4) Production

The main findings of Kang et. al. [7] regarding the process followed by intelligence analysts, referenced by Görg et. al. [8], were the following:

- The process is organic and parallel. Analysis is typically not about finding an answer to a specific problem and it does not evolve in a sequential manner. Instead, analysis is often about determining how to answer a question, what to research, what to collect, and what criteria to use to achieve a goal.
- The process is collaborative. Intelligence analysts do not operate as lone investigators, researching some problem in an isolated space. Kang et. al. [6] found that during the intelligence process collaboration is ‘commonplace and crucial, frequently being asynchronous’ [8].
- The process requires the use of diverse and flexible applications. The student analysts observed by Kang et. al. [7] did not seek “*grand, monolithic computational analysis tools*” (Görg et. al. [8]). Instead, they used a variety of

small applications for specific purposes. Kang et. al. [7] observed that student analysts sought ways to integrate “*existing tools and easy-to-use new tools that leveraged existing analysis methods*” (Görg et al. [8]).

2.1.2.2 Process Scenarios and Strategies

Intelligence analysis requires investigators to gather as much available data as possible in order to better understand a situation and then make judgments about the appropriate next steps to take. Research has shown that during this process the analysts encounter two basic investigative scenarios and employ different investigative strategies.

2.1.2.3 Investigative Scenarios

Görg et al. [8] describe two fundamental types of investigative scenarios within the intelligence domain:

- 1) targeted analysis scenarios, in which analysts are tasked with examining specific ‘people, organizations, or incidents, as well as locations and dates, in order to either investigate past events or uncover an imminent threat’ (Görg et. al. [8]), and
- 2) open ended, strategic analysis scenarios, in which analysts are tasked with “learning as much as possible about a person, organization, country, or situation in order to gain a deeper understanding, conduct an accurate assessment, and possibly make a prediction on the likely chain of events that will occur at a later point in time” (Görg et. al. [8]).

2.1.2.4 Investigative Strategies

Kang et al. [5] conducted an empirical study to find out what strategies investigators use throughout their analysis process. They identify four common investigative strategies, whose effectiveness they judge depending on the analysis results. The four identified strategies are:

- Overview-Filter-Detail (OFD),
- Build-from-Detail (BFD),

- Hit-the-Keyword (HTK) and
- Find-a-Clue-Follow-the-Trail (FCFT)

Overview, Filter, Detail (OFD)

Kang et. al. [5] found that the most commonly used investigative strategy was the one they called “*Overview, Filter and Detail*” (OFD). This strategy employed by analysts consists of three steps:

- Overview. Analysts first gain an overview of the available information by scanning documents, building rough ideas and jotting down important keywords with corresponding document numbers. They then draw circles and lines to indicate connections between keywords and documents to later use these notes as an index for finding relevant documents.
- Filter and Select. After scanning all documents analysts revisit relevant documents selectively, either by direct looking up the documents or by searching for a keyword that stands out.
- Elaborate on details. After filtering and selecting relevant documents, analysts read each document carefully and extract key information.

Kang et. al. [5] concluded that OFD is an appropriate strategy for small data sets only, since analysts need to make decisions about the importance of each document or keyword based on subjective judgements. The strategy presents the danger of missing important details.

Build from Detail (BFD)

The strategy “*Build from Detail*” (BFD) contrasts the previous one. The experiments conducted by Kang et. al. [5] have shown that, when employing this strategy, investigators start the analysis from specific details from each document. They use the search function when important phrases or words arise and write down important keywords for every document. This strategy turned out to be the most time-consuming because of the analysts paying attention to every detail. The strategy impeded analyst to see the “*big picture*” of the plot and turned out to be least effective of the different strategies employed.

Hit the Keyword (HTK)

This strategy consists in an intensive keyword-based exploration. When employing HTK analysts do not begin the analysis by reading a specific document, but directly look for a few specific keywords such as, for example, “terrorist”. They read only the related documents and then search for other terms that emerge during that time. The danger of employing this strategy is that it does not cover all of the documents and that it leads to a situation in which participants ignore the rest of the documents. The effectiveness of this strategy depends highly on the appropriateness of the terms chosen in the initial stage.

Find a Clue, Follow the Trail (FCFT)

The ‘*Find a clue, follow a trail*’ (FCFT) strategy is a hybrid approach of the previous strategies. It starts by reading some first few documents to understand context and find a clue. The second step consists in following the trail rigorously using search or other functionalities. Kang et. al. [5] argue that this strategy was the most effective of all four employed. It allows the analyst to focus her/his attention on relevant documents only. Also, it is considerably less time consuming than BFD. The initial time investment of reading a few documents pays off because it increases the possibility of finding the right clue. Kang et. al. [5] conclude that this strategy leads to satisfactory results and that it may be a fruitful strategy when there are large numbers of documents. There still exists a possibility; however, of a dead-end if the analyst follows a wrong trail. In that case, the ability to quickly turn to another trail is crucial.

2.1.2.5 Visual Analytics System for Process Provenance

The empirical study conducted by Kang et al. [5] and their findings about the common strategies investigators use throughout their analysis process leads us to conclusions about how visual analytics systems can support process provenance representation. Kang et al.’s study refers to strategies employed during document analysis. However, they point out to the fact that their findings can be applied to source analysis in general.

2.1.2.6 Flexible Mechanisms for Source Management

The use of strategies such as OFD or BFD has shown that intelligence analysts require more flexible mechanisms for source management activities such as import, storing, filtering, and maintaining in order to be able to save time. Visual Analytics Systems need to enable the analyst to manage both pushed and pulled information and organize sources meaningfully. As explained by Cybenko et. al. [9], pulled information refers to an analyst's specific information requests. Information sent in anticipation of the analyst's need and information not directly solicited is characterized as pushed information. Also, flexible mechanisms for source management need to support analysis with constantly changing information as well as in integrating collection and analysis into a single system. This supports the analyst in using structured methods during information collection.

2.1.2.7 Responsive User Interface

The study of Kang et. al. [5] has shown that during the initial overview process analysts need to be able to identify important keywords and in this process they then draw circles and lines to indicate connections between keywords and documents. In terms of the user interface this means that investigators want to be able to annotate the system views, highlight particular items, and add notes and comments on top of the visual representations.

2.1.2.8 Direct attention to critical information

Since Kang et. al. [5] concluded that FCFT was the most effective strategy, we learn that investigative analysis tools need to support analysts in finding appropriate starting points or clues and then following the trail of these clues efficiently. If analysts are able to focus from the beginning on relevant documents, they are likely to perform very well. Therefore, investigative analysis tools need to direct the analyst's attention to the most critical information Görg et al. [8].

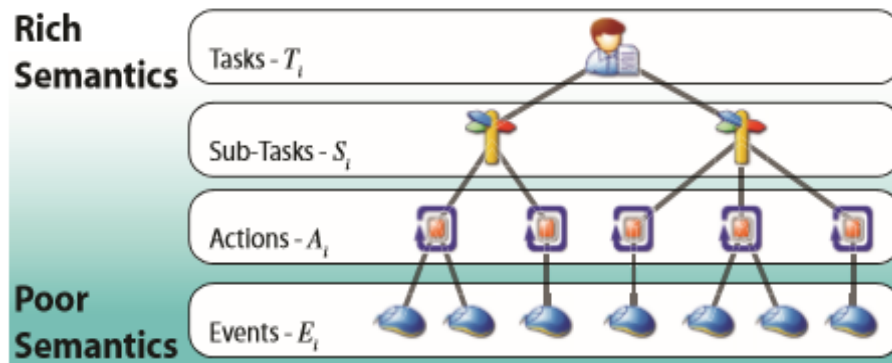


Figure 2.2: Hierarchical analytic provenance model. This model is an example of analyzing the stock market. The semantic richness increases from bottom to top. The bottom layer includes events such as key presses and mouse clicks, which have little semantics. The next level up are actions such as database querying and visualization zooming. Further up are the subtasks, which usually are the analyses performed during sensemaking. The top-level tasks are the overall sensemaking undertaking [10].

2.1.3 Reasoning Provenance

We explained in the previous section that analytic provenance captures, on the one hand, the concrete steps of the data exploration process and, on the other hand, the human reasoning process that supports sensemaking (Kai Xu et al. [1]). We can thus differentiate between ‘*process provenances*’, referring to the workflow during the data exploration process and the methodologies used by analysts, and ‘*reasoning provenance*’ relating to the capturing of the human reasoning process. Accordingly, process provenance answers the question about what steps the analyst engages in during the data analysis. Reasoning provenance refers to a meta-cognitive dimension: to the how and why the analyst’s ideas evolve over time. Analytical provenance refers to all three questions regarding the what, why and how of the data exploration process: the mechanical and more tangible steps together with the more intangible, meta-cognitive phenomena the analyst is involved in.

2.1.3.1 Reasoning Provenance and Process Provenance: Two Interrelated Concepts

At this point we would like to point out that the differentiation between the process followed by the analyst and the reasoning s/he involves in is an artificial one. As explained by Chang et. al. [2], analytic provenance records the analyst’s reasoning,

her/his behaviours as well as the interaction logs and steps taken during the process, and these topics are strongly interrelated and interdependent. If we take a closer look at the levels of visual analytic activity we understand how process provenance and reasoning provenance are two sides of the same coin. Mechanically, the process can be divided into tasks, sub-tasks, actions and events. But each of these mechanical steps has its own semantic meaning and belongs to the realm of meta-cognition reasoning. Tasks and sub-tasks represent logical structures of a user's reasoning process, such as the user's cognitive goals and sub-goals (Gotz et. al. [10]) as shown in figure 2.2. Tasks capture a user's highest-level cognitive goals and sub-Tasks correspond to more objective, concrete cognitive goals. These cognitive goals, which are often open ended or ambiguous, are what drive a user's overall analysis process [10]. Actions represent the individual executable semantic steps, such as making a data inquiry, taken by a user while working toward their analytic goal. *'The action tier uniquely bridges the gap between higher-level logical constructs and the lowest-level user interaction events'* [10]. Events correspond to the lowest-level of user interaction events such as a mouse click or a menu item selection, which carry very little semantic meaning, as explained by Gotz et. al. [10].

2.1.3.2 Reasoning Scenarios and Strategies

As explained in previous section, reasoning provenance refers to the meta-cognitive questions of how we think and reach insights and of why we reach particular insights. Reasoning provenance is therefore often referred to as *'insight provenance'*.

2.1.3.3 Reasoning Scenarios and Challenges

Heuer [11] examined the psychology of intelligence analysis and identified scenarios and challenges analysts must confront in the analytical reasoning process. According to Heuer intelligence analysis faces three main cognitive scenarios and challenges (1999, p. xx):

- i. **Uncertainty:** The analyst's faces both inherent uncertainty (surrounding complex intelligence issues) and induced uncertainty (the "man-made" internal uncertainty) [11].

- ii. **Cognitive Biases:** The analyst is not prepared to deal effectively with uncertainty because of the danger to fall prey to cognitive biases, '*such as the tendency to see information confirming an already-held judgment more vividly than one sees 'disconfirming information'*' [11].
- iii. **Lack of Appropriate Tools and Techniques:** In order to be able to confront the scenario of uncertainty and cognitive biases, the analyst needs tools and techniques for applying higher levels of critical thinking. If in lack of specific techniques for structuring information, challenging assumptions, and exploring alternative interpretations, the analyst is unable to improve analysis on complex issues on which information is incomplete [11].

Taking into consideration the scenarios presented to analysts, we define that reasoning provenance is about tracking strategies of problem-solving and decision-making under the conditions of uncertainty, cognitive biases and lack of appropriate tools and techniques.

2.1.3.4 Reasoning Strategies

Conventional Intuitive Analysis and Cognitive Heuristics

When analysts encounter a situation of uncertainty, they are trained to develop multiple hypotheses and have them compete against each other. However, if analysts are not trained to use a structured methodology such as the Analysis of Competing Hypotheses (ACH), they employ the conventional intuitive analysis, characterized by the following circumstances (Heuer [11], p. 108):

- Analysis starts with a most likely alternative for which the analyst seeks confirmation, rather than with a full set of alternative possibilities. This does not ensure that alternative hypotheses receive equal treatment.
- The fact that key evidence may also be consistent with alternative hypotheses is rarely considered explicitly and often ignored.
- Conventional analysis generally entails looking for evidence to confirm a favoured hypothesis.

Tversky et. al. [12] also discuss how the analyst can enter a process of rapid diagnosis employing the intuitive system. They explain that in this intuitive system s/he involves past experience as well as current knowledge. This process is however inaccessible to conscious control. In order to reach conclusions analysts often employ rapid mental comparisons of current cases with abstract prototypical pictures. These mental comparisons and shortcuts based on intuitive judgment are called "*heuristics*". The heuristics used in judgments under uncertainty identified by Tversky et. al. [12] belong to the most commonly encountered cognitive biases during the analysis and production phase:

- i. **The representative heuristic:** the probability of a problem is judged by how closely a case presentation matches a prototypical case.
- ii. **The availability heuristic:** the probability of a problem is judged on the basis of how easily that problem is recalled, which is often skewed by recent and memorable cases.
- iii. **The anchoring heuristic:** involves clinging to initial diagnostic hypotheses even as contradictory evidence accumulates.
- iv. **Premature closure:** settling on a diagnosis without sufficient evidence or without seeking or carefully considering contradictory information.
- v. **Confirmation bias:** is the tendency to look for evidence to support a working hypothesis, ignore contradictory evidence, and misinterpret ambiguous evidence.

Less-than-Optimal Strategies for Making Decisions

Alexander George [138] identified a number of less-than-optimal strategies for making decisions in the face of uncertainty and incomplete information. These strategies are: satisficing, incrementalism, consensus, reasoning by analogy and relying on a set of principles that distinguish '*good*' from '*bad*' alternatives (Heuer [11], p. 43).

- i. **Satisficing** - consists in selecting the first identified alternative that appears "*good enough*" rather than examining all alternatives to determine which is "*best*".

- ii. **Incrementalism** - refers to the strategy of focusing on a narrow range of alternatives representing marginal change, without considering the need for dramatic change from an existing position.
- iii. **Consensus** - consists in opting for the alternative that will elicit the greatest agreement and support. Simply reporting to a superior what s/he wants to hear is one example of this.
- iv. **Reasoning by Analogy** - refers to choosing the alternative that appears most likely to avoid some previous error or to duplicate a previous success.
- v. **Relying on a Set of Principles or maxims that distinguish a 'good' from a 'bad' alternative** - is the last less-than-optimal strategy mentioned by George (1980) for making decisions in the face of uncertainty and incomplete information.

The 1980 work of Alexander George [138] applied to strategies used by decision makers in order to choose among alternative policies. Heuer argues however that the same strategies apply to decision making in the realm of intelligence analysis. He identifies three specific weaknesses of the satisficing strategy when selecting a hypothesis (Heuer [11], p. 45):

- i. **Selective Perception:** Analysts, Heuer argues, like people in general, tend to see what they are looking for and to overlook that which is not specifically included in their search strategy. They tend to limit the processed information to that which is relevant to the current hypothesis.
- ii. **Failure to Generate Appropriate Hypotheses:** If tentative hypotheses determine the criteria for searching for information and judging its relevance, it follows that the analyst may “overlook the proper answer if it is not encompassed within the several hypotheses being considered” (Heuer [11], p. 45).
- iii. **Failure to Consider Diagnosticity of Evidence:** In the absence of a complete set of alternative hypotheses, the analyst is not able to evaluate the ‘*diagnosticity*’ of evidence.

Analysis of Competing Hypotheses (ACH)

When analysts encounter a situation of uncertainty, they are usually trained to develop multiple hypotheses and seek out information that can discredit many of the hypotheses. The analysis of competing hypotheses (ACH) is a tool to aid judgment on important issues requiring careful weighing of alternative explanations or conclusions. “It helps an analyst to overcome, or at least minimize, some of the cognitive limitations that make prescient intelligence analysis so difficult to achieve” (Heuer [11], p. 95). ACH requires an analyst to have alternatives compete against each other, rather than evaluating their plausibility one at a time. Therefore, ACH is used when there are multiple competing hypotheses to analyze.

ACH comprises the following steps, as described by Heuer ([11], p. 97):

- Identification of possible hypotheses to be considered.
- Compilation of a list of significant evidence and arguments for and against each hypothesis.
- Preparation of a matrix with hypotheses across the top and evidence down the side.
- Analysis of the ‘*diagnosticity*’ of the evidence and arguments.
- Identification of the items those are most helpful in judging the relative likelihood of the hypotheses.
- Redefinition of the matrix by reconsidering the hypotheses and deleting evidence and arguments that have no diagnostic value.
- Formulation of tentative conclusions about the relative likelihood of each hypothesis by disproving the hypotheses rather than proving them.
- Verification of how sensitive a conclusion is to a few critical items of evidence.
- Reporting of conclusions by discussing the relative likelihood of all the hypotheses, not just the most likely one.
- Identification of milestones for future observation that may indicate events are taking a different course than expected.

Heuer’s concept of Analysis of Competing Hypotheses (ACH) is among the most important contributions to the development of an intelligence analysis methodology.

It represents a very thorough and effective strategy. At its core lies the notion of competition among a series of plausible hypotheses. The surviving hypotheses are subjected to further testing. ACH, Heuer concedes, will not always yield the right answer. But it can help analysts overcome the cognitive limitations (Heuer [11], p. xxxiii).

2.1.3.5 Visual Analytics for Reasoning Provenance

Visual analytics systems are often evaluated based on how effectively they allow the generation of insights but also on how easily reasoning and insight creation can be tracked (Görg et al. [8]). Lessons learned from reasoning scenarios and strategies dictate that visual analytics systems need to help the analyst externalize the thinking process and create convincing production by supporting insight provenance and sanity checks of analytical products. Inspired by Sørmo [13], we argue that the effectiveness of visual analytics systems in terms of capturing reasoning provenance can be judged according to the following criteria and by answering the following questions:

- i. **Transparency:** Is it clear how the analyst reached insights and answers with the help of the system? The transparency principle focuses on the main condition necessary for examining insight provenance, the historical record of the process and rationale by which an analyst derives insights. Transparent systems allow the analyst to visualize and understand the entire reasoning path.
- ii. **Justification:** Is it clear why the insights represent “good” insights? The justification principle is about judging the quality of insights and about verifiability. Systems that permit justification will allow the analyst to record the justification of each step during the process and also formulate a posteriori explanations.
- iii. **Relevance:** Is it clear why questions asked were relevant? This principle refers to the mental models used by the analyst and her/his reasoning strategy. The analyst needs to be able to record why a question asked was relevant to the

task at hand. An explanation of this type justifies the strategy pursued. A system complying with the relevance principle comprehensibly display the analyst's reasoning strategy by recording the relevance behind each question asked.

- iv. **Conceptualization:** Is it clear that terms and definitions mean? Analysts do not always understand, or have a common understanding, about all the terms encountered in a query. This may be because the analyst is a novice in a particular field, but also because different analysts can use terms differently or organize the knowledge in different ways.
- v. **Learning:** Is it clear what lessons can be learned during and/or after solving a case? The use of a visual analytics system for solving a particular case should increase the analyst's understanding about the different domains encountered. A system complying with the learning principle is able to train the analyst in solving problems by, for example, pointing out to similar cases from the past or helping her/him reapply insights to a new data or domain.

2.1.4 Examining Analytic Provenance in Visual Analytic Systems

In this section we survey how to track and visualize analytic provenance. We also review design alternatives to keep track of analytic provenance.

2.1.4.1 Challenges to Examining Analytic Provenance

The key to the research on analytic provenance is the belief that by capturing a user's interactions with a visual interface, some aspects of the user's reasoning processes can be retrieved [14]. North et al. [14] argue that the research of analytic provenance can be examined in five interrelated stages: perceive, capture, encode, recover and reuse. Each of these stages presents its own challenges.

"Perceive": How does the analyst perceive the visualization of data?

To correlate a user's interactions with visualization to her reasoning process, the research must begin with understanding how the data is presented to the user. Since the user's interaction can only begin after perceiving the visualization of data, the

analytic provenance research also needs to start with the understanding of how information is perceived by the user [14].

“Capture”: How can analytic provenance be captured?

As the user interacts with visualization, the series of interactions can be considered as a linear sequence of actions. Researchers have shown that additional semantic information is necessary to adequately represent a user’s analysis process. Semantic information can be directly annotated by the user, modeled based on task analysis, or correlated with the visualization elements, but identifying the most appropriate representation remains an open challenge [14].

“Encode”: How can captured analytic provenance be described?

Encoding refers to the process of describing the captured provenance in predefined formats. While many systems implicitly have their own encoding schema for capturing analytic provenance for specific tasks and domains, few generalizable schemas exist. North et al. [14] explain that researchers have attempted using XML, declarative pattern language, logic-programming, and dynamic scripts, but in most cases these schemas only record the “*how*”, but not always the “*why*”. By using these schemas, the user can reapply interaction, but the semantic meanings behind these steps are often unclear.

“Recover”: How can we make sense of provenance?

Once the user’s provenance has been captured and encoded, the challenge becomes making sense of the provenance. As noted by Jankun-Kelly et. al. [15], history alone is not sufficient for analyzing the analytical process with visualization tools. Often, there are relationships between the results and other elements of the analysis process which are vital to understanding analytic provenance. North et. al. [14] argue that while some of the relationships have been shown to be recoverable through manual inspection, whether the same can be done using automated techniques is still an open question.

“Reuse”: How can a user’s insight be reapplied to a new data or domain?

The research goal in analytic provenance is to be able to automatically reapply a user’s insights to a new data or domain. Most systems that are successful at encoding a user’s interactions have mechanisms that allow for the reapplication of the interactions within the same system. However, North et. al. [14] argue that in most analytical environments, analysts often utilize multiple tools simultaneously which renders the use of existing methods inadequate. They conclude that a more comprehensive and cohesive encoding, recovering, and reusing process is therefore necessary to support the analysts in their natural working environments.

2.1.4.2 Capture, Visualization and Utilization for Analytic Provenance

Analytic provenance consists of three stages: capturing the provenance of the analysis process, visualizing the captured information, and utilizing the visualized provenance. Significant amount of research have been carried out for developing a usable and manageable provenance tracker along with the user interface for representation, access to provenance information.

2.1.4.3 Capture Methods (Manual vs Automatic)

2.1.4.4 Manual Capture

Manual approaches of analytic provenance include ‘*user-created notes, manually authored diagrams illustrating a user’s analytic steps and user-built structured argumentation graphs*’ (Gotz et. al. [10], p. 124).

As Gotz et. al. [10] argue, the manual capture approach can be very effective for capturing the high-level rationale by which analysts “*connect individual insights into an overall conclusion*”. During a visual analytic task, users typically perform a very large number of activities at a very fast pace. Each of these activities (queries, filtering and sorting processes) is motivated by a logical rationale. As Gotz et. al. [10] argue, it is however often too laborious and impractical for an analyst to manually record each individual activity and it’s rationale due to the overwhelming amount of

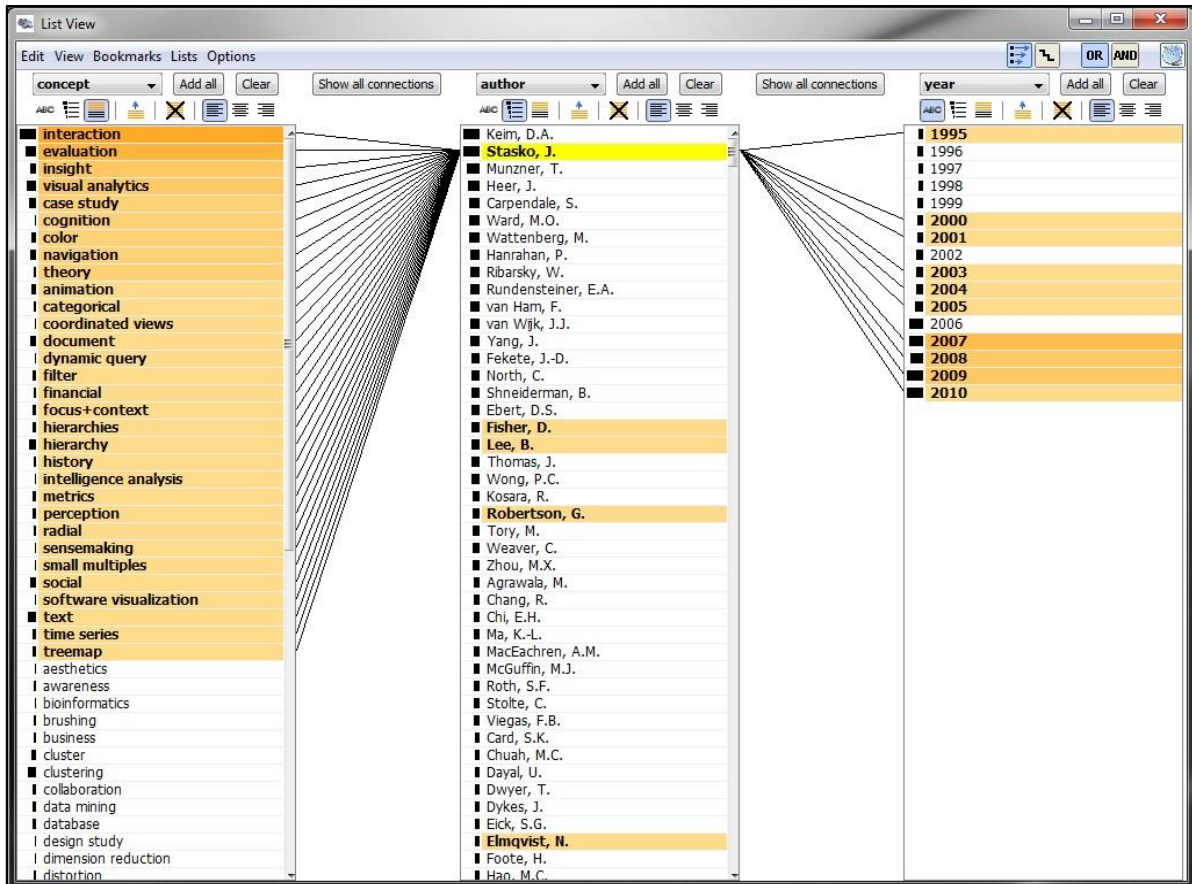


Figure 2.3: The Jigsaw list view showing connections between concepts, names and dates. Retrieved from <http://www.cc.gatech.edu/gvu/ii/jigsaw/views.html>.

information involved. For this reason, analysts often record just the final state of visualization and tag it with a high-level description. Analysts *'typically omit from their documentation the intermediate steps that led to the insight'*. Moreover, they often omit seemingly unimportant visualizations from their notes even if they directly motivated additional lines of inquiry. As a result, *'critical information may be lost and the manual approach fails to capture a user's insight provenance comprehensively'* [10].

A Case Study

We present a case study of manual capture of analytic provenance of analysts using the Jigsaw system. In this case we will refer to *'manual'* as opposed to the term *'automatic'* or *'computational'* capture. Automatic or computational capture which refers to systems that integrate analytic provenance capture. In the case presented below we refer to manual capture as a *'by-hand'* documentation process, including the self-documentation process of analyst as well as the a posteriori process of examining

provenance. We first illustrate the Jigsaw system and its features to then describe what the provenance capturing process was about.

Jigsaw is a visual analytics tool for exploring and understanding document collections. It was developed by researchers Stasko, Görg, Liu, Sainath and Stolper at Georgia Tech. Jigsaw combines automated text analysis with interactive visualizations for exploring and analyzing collections of unstructured and semi-structured text documents. It automatically identifies entities of interest in the documents, such as concepts, people, places, dates and organizations, and then shows connections between those entities across the entire collection, as well as connections between documents and entities. Figure 2.3 above, shows how Jigsaw's list view establishes connections between particular concepts, authors and years.

Connections are defined by co-occurrence: if two entities co-occur in the same document, they are connected to each other as well as to that document. If entities co-occur in many documents they have a stronger connection. Görg et al. [8] argue that even though this untyped connection model based on co-occurrence is very simple, it has turned out to be a powerful tool for investigative analyses. It works best if the documents are not too large, as it is often the case for news articles or case reports that usually span a few paragraphs. The list view presented in figure 2.3 is just one of the possible views offered by Jigsaw. Jigsaw was not specifically designed to automatically capture analytic provenance. In order to examine analytic provenance when using Jigsaw's, Kang et al. [7] conducted an evaluation of the visual analytic system Jigsaw and compared its use to three other more traditional methods of analysis. The study of Kang et al [7] consisted in recruiting sixteen students, dividing them into four groups, and asking them to conduct a document and identify a hidden threat.

- Participants in the first group only worked with pencil and paper. They received a printout of all the reports and some blank sheets for note taking.
- Participants in the second group received an electronic copy of the reports and could use basic text editing software for reading and searching the documents.

- Participants in the third group used only the Document View of the Jigsaw system to read and analyze the document collection. This setup was similar to the previous one, providing functionality for reading and searching; however, the Document View also highlighted identified entities within the documents.
- Participants in the fourth group used the entire suite of visualizations in Jigsaw to conduct the analysis.

In order to ‘*manually*’ capture analytic provenance for all four groups Kang et al. [7] used a posteriori semi-structured interviews, where analysts made use of their own notes. Kang et al. also video-taped and analyzed a posteriori all sessions. They mainly based their capture of analytic provenance on observations, interviews, videos, and log analyses. They not only identified several investigative strategies employed by analysts but also the benefits and limitations of Jigsaw.

Kang et al. [7] found that overall the participants using the full Jigsaw system outperformed all other groups on average. The benefits of Jigsaw were that the system supported different investigative strategies, that it showed connections between entities, that it helped users find a right clue and that it also helped them focus on essential information. The limitations of Jigsaw resulted from the analysts’ wishlist, who asked for better ways to work with only subsets of their document collections and to be able to dynamically filter out documents in an investigation, but also maintain the ability to reintroduce filtered documents as desired.

2.1.4.5 Automatic Capture

Automatic approaches for capturing analytic provenance attempt to systematically capture the full history of a user’s analytic process. There are visual analytic systems that record histories of user visual operations and the parameters of these operations. But although these tools ‘*comprehensively and faithfully record user analytic activity, they cannot abstract the high-level semantic constructs obtained in the manual approach*’ [10]. As explained by Gotz et. al. [10], most existing

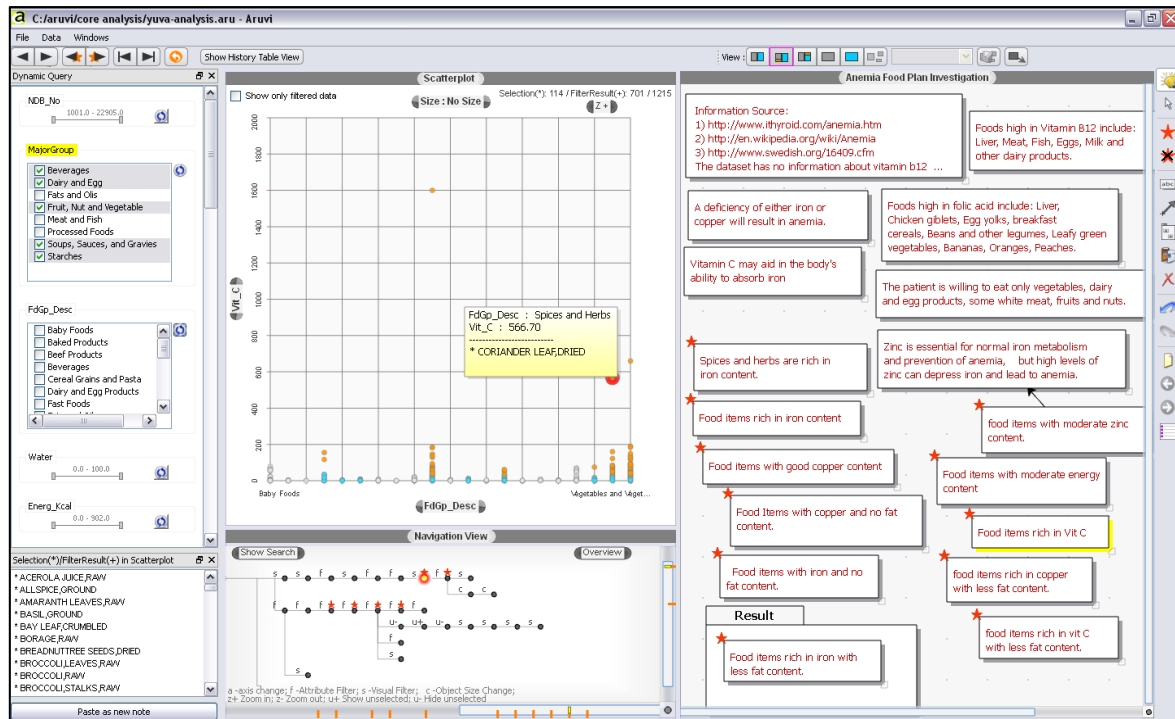


Figure 2.4: The Aruvi Information Visualization Framework: The data view, knowledge view and navigation view. Retrieved from <http://www.win.tue.nl/vis1/home/yedendra/imgs/aruvi.png>.

visual analytic systems are event-based systems that are designed to recognize and process specific, often low-level user interaction events like mouse clicks and drags, but can rarely understand and capture the semantics of such events (e.g., the analytic purpose of a user's mouse drag). In addition, during visual analysis the high rate of user activity often creates a large number of low-level user interaction events that grows enormously as the analysis unfolds. Gotz et. al. [10] conclude that it is extremely difficult for systems to organize the large linear list of user interaction events into semantically meaningful segments of activity. *'It is even more challenging for such system to infer the high level semantic constructs that can capture the complex, non-linear nature of a user's visual analysis process'* [10].

2.1.4.6 Automatic Capture with Visual Analytic Systems

Visual analytics is a relatively new research field that integrates the interactive visualization and exploration of data with computational data analyses [16]. It represents *'the science of analytical reasoning facilitated by interactive visual interfaces'* (Thomas et. al. [17], p. 28). Intelligence analysis challenges investigators to

examine large collections of data and documents and come to a deeper understanding of the information and events contained within them. Visual analytics technologies thus hold great promise as potential aids for intelligence analysis professionals [8].

Aruvi

Aruvi is an information visualization framework that supports the analytical reasoning process presented by Shrinivasan et. al. [18]. It contains three main views: the data view, the navigation view and the knowledge view, as represented in the figure 2.4 below. The data view is the visual analytical tool itself, the navigation view is a panel for visually tracking the user's history, and lastly the knowledge view allows the user to interactively record his reasoning process through the creation of a node-link diagram.

Shrinivasan et. al. [18] explain that when using Aruvi analysts can also organize the analysis artifacts in the knowledge view to build a case to support or contradict an argument. They can establish a link between an analysis artifact in the knowledge view and a visualization state in the navigation view. Hence, they can revisit a visualization state from both navigation and knowledge views to review the analysis and to validate the findings. After revisiting the visualization state, the user can reuse it to look for alternate views. Aruvi supports two ways of preserving a user's insight provenance:

- First, it automatically records a user's navigational steps.
- Second, it provides an interface component that allows users to manually add notes.

Gotz et. al. [10] argue however that the granularities of a user's navigational steps are determined by '*application-specific heuristics*', like for example when the mouse pointer exits from a particular UI (User Interface) panel.

Scalable Reasoning System (SRS)

The Scalable Reasoning System(SRS) is a web service-based analytic toolkit (figure 2.5) that allows data clustering, temporal trend identification and geographic analysis

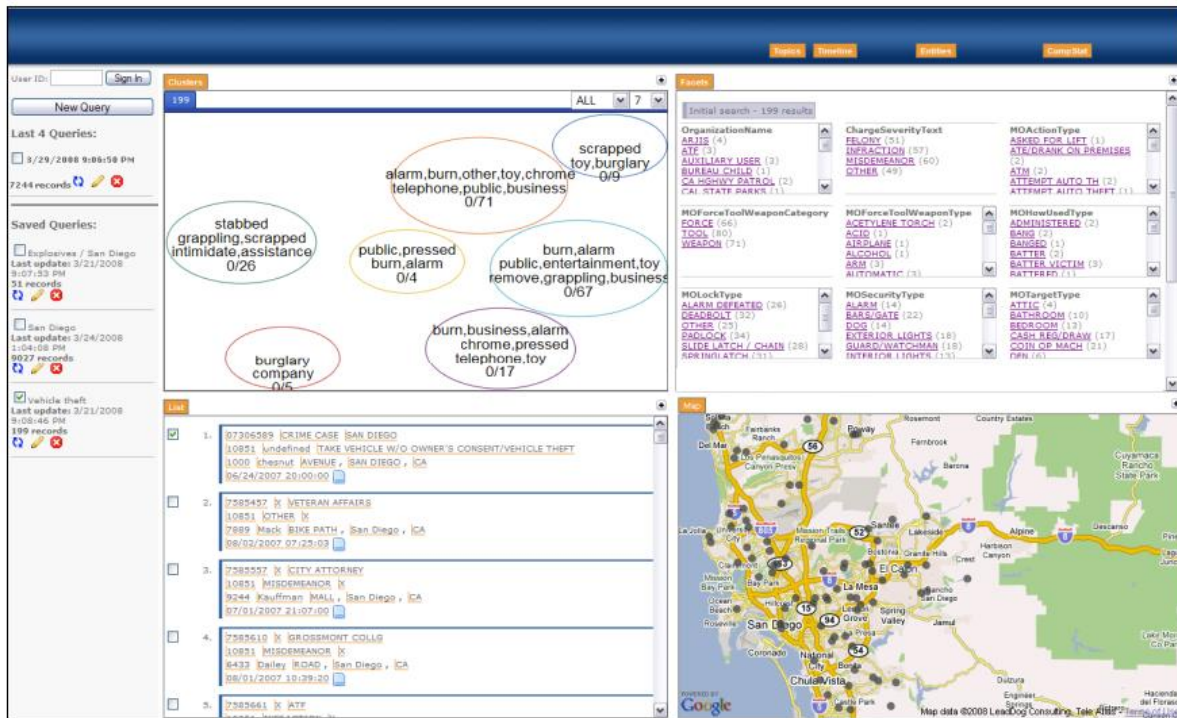


Figure 2.5: SRS web client showing, from top left, cluster, faceted, list, and map views of vehicle theft incidents [19].

[19]. It was initially developed by the U.S. Department of Homeland Security to support rapid application deployment to users that needed access from a variety of locations to a rapidly broadening and changing set of analytical capabilities. In SRS, analysts can graphically describe their knowledge construction workflows, linking hypothesis creation and testing to the visualizations that were used to derive these hypotheses. The SRS hypothesis construction workspace allows any feature of a view, or an entire view itself, to be saved as a *'reasoning artifact'*. Pike et al. [19] explain that reasoning artifacts are *'essentially pieces of information that the analyst has tagged with a role; these roles are defined through a customizable taxonomy of knowledge structures'*. At any point during an analysis, a user can open a reasoning whiteboard on the SRS web site and begin to create artifacts that link features in SRS views to reasoning roles. Features such as incident clusters can be dragged out of the view and onto the whiteboard to capture them as a reasoning artifact, where they are converted to a small sticky note. Annotation artifacts can also be created to record information such as assumptions. SRS also provides explicit support for the analytic reasoning process. Too often the insight generated through visual discovery is left tacit in the

analyst's mind or recorded in forms disconnected from exploratory tools. SRS clients can embed a graphical '*reasoning whiteboard*' on which users can link features discovered through exploratory visualization with reasoning structures such as emerging hypotheses.

2.1.4.7 Automatic Capture with Specific Provenance Software

Jankun-Kelly et. al. [15] argues that the lower the barrier to capture process and reasoning information and annotate it, the more data will be generated. Jankun-Kelly offers two examples of software that can be integrated into visual analytics toolkits: *VisTrails* – an open source software based on tree representations and the *PSet* – a software which is available on request and based on graphical representations.

VisTrails

VisTrails is an open-source scientific workflow and provenance management system, developed at the University of Utah, which provides support for simulations, data exploration and visualization. VisTrails enables interactive multiple-view visualizations by simplifying the creation and maintenance of visualization pipelines, and by optimizing their execution. VisTrails design goals included:

- creating an infrastructure that maintains the provenance of a large number of visualization data products.
- providing a general framework for efficiently executing a large number of potentially complex visualization pipelines.
- providing a user interface that simplifies multiple-view comparative visualization.

VisTrails uses an action-based provenance model that uniformly captures changes to both parameter values and pipeline definitions by unobtrusively tracking all changes that users make to pipelines in an exploration task. Silva et al. [20] refer to this detailed provenance of the pipeline evolution as a visualization trail, or *vistrail*.

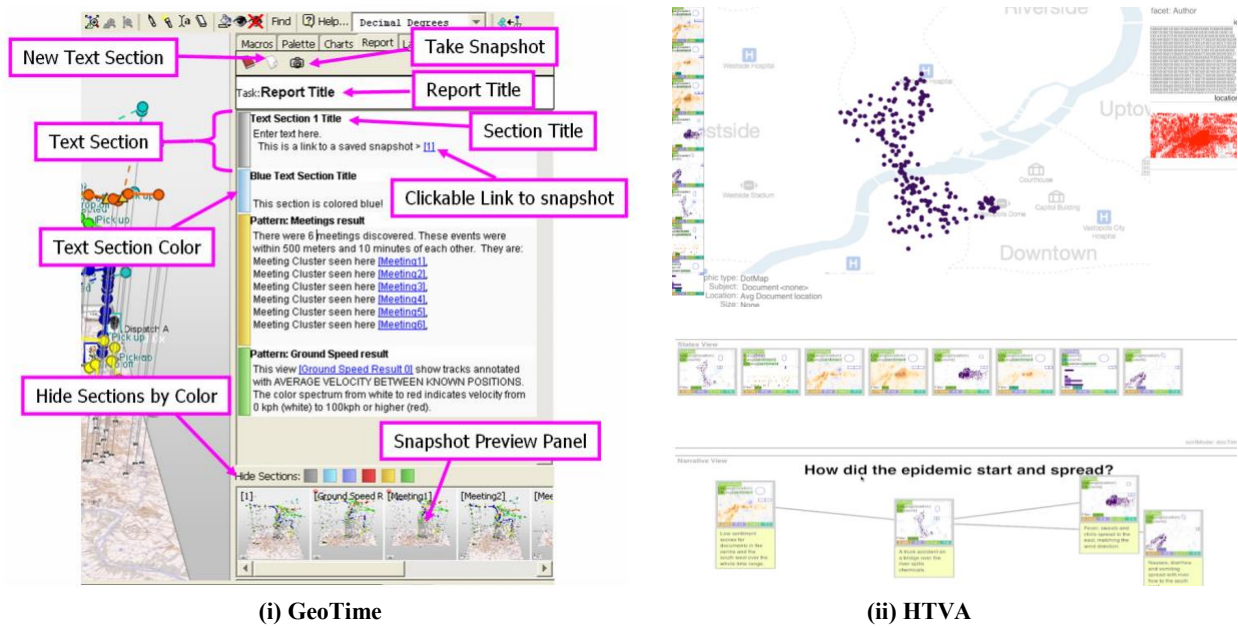


Figure 2.6: (i) The story window to the right of the *GeoTime* scene with Snapshot Preview Panel (Eccles et al. [21]). (ii) A narrative has been formed by dragging bookmarks into an ordered group and attaching narrative text. Each bookmark can be linked back to the live data (Walker et al. [22]).

PSet Software

Jankun-Kelly et al. [15] argues that there is a need for formal representations of visual explorations. The PSet software provides metrics to evaluate visualizations. The PSet software uses different graph measures to describe the visualization. It encapsulates *‘the interactions a user can have with a visualization system and how these interactions are part of the greater exploration session’* [15]. The P-Set Model of visualization exploration formalizes the iterative visualization cycle by describing a user’s interaction with a visualization system. During such interaction, a user manipulates parameter values to form a parameter set (a p-set). A p-set is a collection of parameter values of different types. Created p-sets are used to generate new results. For each result generated during visualization exploration, four items are stored: a timestamp, parameter derivation information, p-set derivation information, and result derivation information.

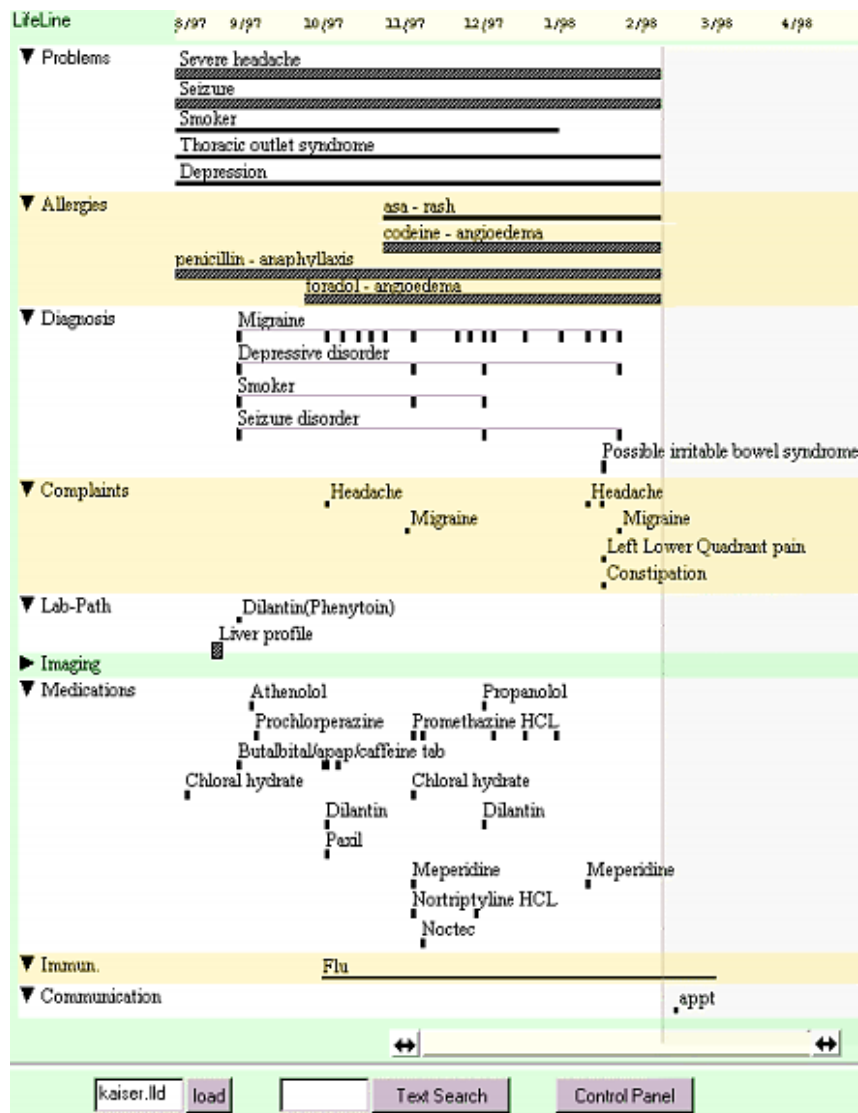


Figure 2.7: Temporal relationships and links across different sections of the record are made apparent (e.g., the temporal coincidence between the constipation and the closely preceding administration of meperidine is suspicious (Plaisant et. al. [23])).

- The timestamp indicates when the derivation was performed; it is possible for multiple results to be generated during the same timestamp as a consequence of a single user interaction.
- The parameter and p-set derivations describe which previous parameters and p-sets were used to create the new parameters and p-sets.
- Finally, the created results are identified by the p-sets. Each explored result is encoded by the model.

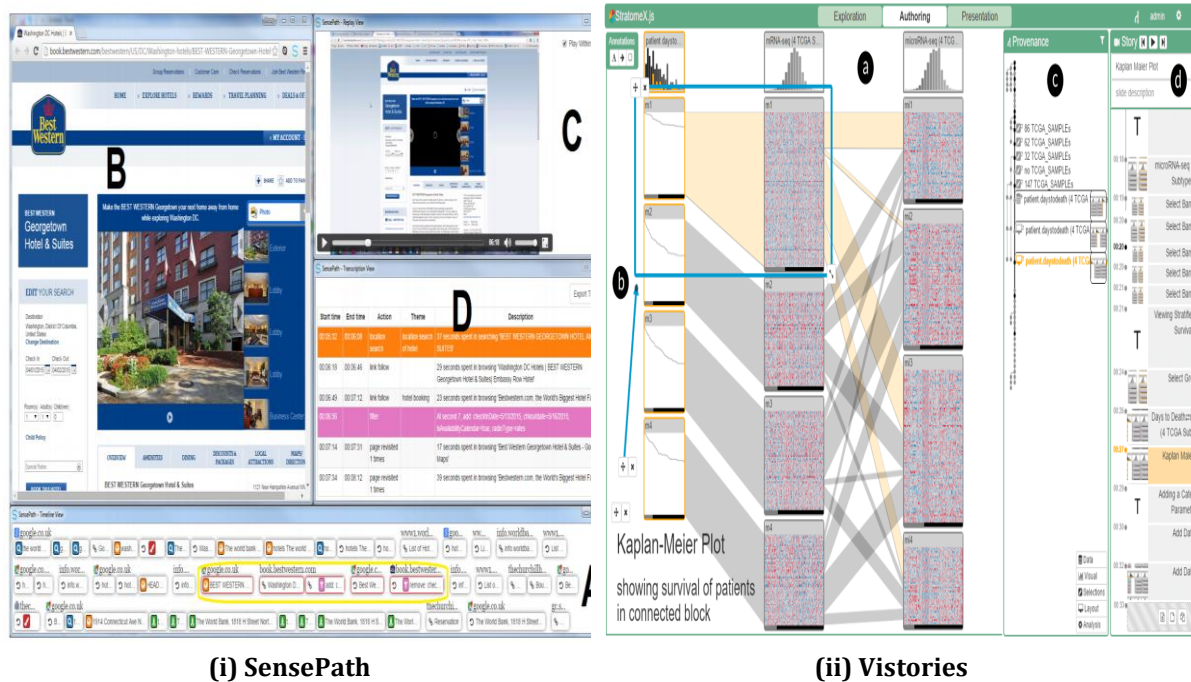


Figure 2.8: (i) Four linked views of *SensePath*. **A:** The timeline view shows all captured sensemaking actions in temporal order. **B:** The browser view displays the web page where an action was performed. **C:** The replay view shows the screen capture video and can automatically jump to the starting time of an action when it is selected in another view. **D:** The transcription view displays detailed information of selected actions (Nguyen et. al. [24]).
(ii) Screenshot of CLUE applied to the StratomeX technique **(a)** in authoring mode. An annotation **(b)** highlights relevant aspects. The provenance graph view **(c)** and story view **(d)** show the history of the analysis and a Vistory being created (Gratzl et. al. [25]).

2.1.5 Visualization

GeoTime

GeoTime (Eccles et. al. [21]) is a commercial geo-temporal event visualization tool as shown in figure 2.6(i) that can capture a screen shot of the tool and perform text or graphical annotation. It also allows users to construct a report of the analysis. Tableau Public offers a story telling feature, which consists of several pages or story points, each is a captured visualization with annotation.

HTVA

To reuse captured states, the human terrain visual analytics system (HTVA) proposed by Walker et al. [22] as shown in figure 2.6(ii) , allows the analyst to drag and drop

captured visualizations automatically onto an empty space and add narrative to each visualization to build the story.

LifeLines

To visualize captured information, LifeLines (Plaisant et. al. [23]) as shown in figure 2.7, is a visualization for personal histories, which uses icons to indicate discrete events and thick horizontal lines for continuous ones. Typically, the system begins with an initial state (node). When the user performs an action, a new node is created for the current state, and a new edge is added to connect the previous node with the current node.

2.1.6 Utilization

SensePath

SensePath (Nguyen et. al. [24]) as shown in Figure 2.8(i), is a tool for understanding sensemaking process through analytic provenance. SensePath provides four linked views of i.e, a timeline view that shows all captured sensemaking actions in temporal order, a browser view that displays the web page where an action was performed, a replay view that shows the captured screen video and can automatically jump to the starting time of an action when it is selected in another view, a transcription view that displays detailed information of selected actions.

Vistories

“Vistories” as shown in Figure 2.8(ii), is a visual story based history exploration system by following the CLUE (Capture, Label, Understand, Explain) model proposed by Gratzl et. al. [25]. This tool has an authoring mode, a provenance graph view, a story view for showing the history of the analysis and a Vistory being created.

2.1.7 Summary

As mentioned by Heuer [11] due to lack of appropriate tools and techniques, analysts may be unable to apply higher levels of critical thinking on critical issues in intelligence analysis. So, tools with fine-tuned computation led cognition technique and vice-versa having support to bridge the gap [10] between those is required for successful analytical activities. The above section 2.1 has provided a concrete definition of analytic provenance as proposed by Xu et al. [1] and described it's different potential stages of research (North et al. [14]). It has also mentioned how can analytic provenance be examined in existing visual analytic systems and utilized for sensemaking. This section has also described cognitive (reasoning provenance) and investigative strategies (process provenance) for their respective scenarios. Currently, no analytic task model is available for intelligence analysis which will amalgamate both computation and cognition for large visual analytic system compatible with it's complex system architecture. More research is required to understand the requirement, development challenges of both back-end data modelling and front-end visual interface design supporting transparency in decision making.

2.2 Behavioural Markers (BMs)

2.2.1 Overview

Behavioural Marker (BM) systems are now being developed for performance measurement in a range of organizational settings, especially in high reliability industries such as air aviation, nuclear power, maritime transport, and medicine. They are usually structured into a set of categories (e.g. co-operation, decision making, and situational awareness). Normally, these categories are then sub-divided into more specific nontechnical skills or elements. The seminal research on behavioral markers comes from studies of civilian pilots carried out by Helmreich and colleagues at the University of Texas. In the late 1980s they developed a data collection form called the LINE/LOS Checklist (LLC) to gather information on flight crews' crew resource management performance [26]. This checklist has been used as the basis of many airlines' behavioral marker systems [27]. Behavioral marker systems have also been developed for using by anesthesiologists [28], surgeons [29], scrub nurses [30], and nuclear power control room teams [31]. Flin et al. [31] identified significant limitations of such behavioral marker systems such as – not being capable of capturing every possible aspects of performance, absence of conflict management, bringing own biases and perceptions by the raters. Recently, the Behavioural Markers (BMs) concept is not only used to measure team performance in aviation or medical sectors but also their use for evaluating visualization are noticeable. C. North et. al. [32] claims that the purpose of visualization is insight and to determine to what degree visualizations achieve this purpose. He listed some of the characteristics of insight such as – complex, deep, qualitative, unexpected and relevant. P. Saraiya et. al. [33] defined insight as an individual observation about the data, a unit of discovery. They presented several characteristics of insight while running a pilot study on biological and microarray data such as – observation, time, domain value, hypotheses, directed versus unexpected, breadth and depth, category.

Table 2.1: Evaluation hypotheses, data sources and analysis techniques [35].

Evaluation Criterion	Hypothesis	Data Source and Analysis
<i>Validation</i>		
Completeness	The ANTS (<i>Anaesthetists Non-Technical Skills</i>) system provides a suitably comprehensive set of categories and elements to describe anaesthetists' non-technical skills.	<i>Questionnaire data:</i> basic frequency analysis and content review to identify any superfluous or missing elements
Observability	Anaesthetists' non-technical skills can be identified by observation of behaviour using the ANTS system.	<i>Ratings data:</i> basic descriptive statistics and χ^2 tests to establish the extent to which non-technical skills were observed vs not observed. <i>Questionnaire data:</i> frequency analysis, content review and <i>t</i> tests where appropriate.
<i>Reliability</i>		
Inter-rater agreement	Using the ANTS system to rate non-technical skills, participants will achieve inter-rater agreement at (a) category level and (b) element level consistent with recognised criteria for acceptance.	<i>Ratings data:</i> within-group inter-rater agreement statistic [55, 56] to show the level of rater consensus (i.e. whether they rate performances the same): $r_{wg} = 1 - (S\chi^2/\sigma_E^2)$, where $S\chi^2$ = variance of observed ratings and σ_E^2 = population variance for a discrete rectangular distribution of ratings (i.e. it represents a random response where each scale point would have an equal number ratings). This is calculated as $\sigma_E^2 = (A^2 - 1)/12$, where <i>A</i> is the number of points on the scale.
Accuracy/sensitivity	Category and element ratings given by participants will be consistent with reference ratings agreed by a panel of experts.	<i>Ratings data:</i> mean absolute deviation (MAD) from the reference ratings [57, 58] and basic difference from reference ratings to establish the level of accuracy or error for ratings.
Internal consistency	The ANTS system has an acceptable level of internal consistency between the categories and their elements.	<i>Ratings data:</i> Cronbach α coefficient for correlation between elements within a category and Pearson reliability coefficient for mapping of elements to categories.
<i>Usability</i>		
Acceptability	The ANTS system is an acceptable tool for (a) training and (b) assessing non-technical skills in anaesthesia.	<i>Questionnaire data:</i> basic descriptive statistics and content review to establish the level of acceptance for different uses of the system.
Usability	The ANTS system is straight forward for anaesthetists to use to rate non-technical skills.	<i>Questionnaire data:</i> basic descriptive statistics and content review. <i>Ratings data:</i> overall indication of effective use of the system

In a case study with the popular visual analytics application Jigsaw, Kang et al. [5] found that analysts' interaction histories showed evidence of the high-level sensemaking processes. Reda et al. [35] approached interaction and sensemaking by combining interaction logs and user-reported mental processes into an extended log and modeling the log using transition diagrams to better understand the transition between mental and interaction states.

2.2.2 BM System Development and Evaluation

The BM tool is designed in the form of a structured list of behaviours. The observers then use this form during a selected work situation to rate performance within a work environment. Lacher et. al. [36] proposed a BM system development for measurement of the non-technical skills of software professionals. They performed a systematic literature review as the *first step* by addressing the high-level question – '*What are the Non-Technical Skills (NTS) required of software professionals performing well in their field?*' The cognitive or other personal skills that complement human technical skills and contribute to overall task performance are termed as Non-Technical Skills (NTS). Technical Skills (TS) refer to techniques applied as part of ongoing computation by human interactions with the system. The output of this step was an initial list of 35 NT skills that were clustered into four major categories: communication, interpersonal, problem solving, and work ethic. During the *second step*, the initial list of NT skills had their quality assessed and were validated by focus group of experts in industry and academia. They evaluated the percentage of positive ratings, and developed a binary data set for statistical analyses. By inspecting the distributions of the raters when examining the skills, a critical value (specific to each NT skill) was chosen to separate the 0 or 1. Next, a *McNemar's test* was used to evaluate whether or not there are significant differences between the raters. A value of $p < 0.05$ would tell us that there is a significant difference between the raters and p value greater than 0.05 would signify inter-rater reliability [54].

Fletcher et. al. [34] have presented an experimental evaluation for '*Anaesthetists Non-Technical Skills (ANTS)*' by using human factors research techniques to establish its basic psychometric properties and usability. The design of the study required trained

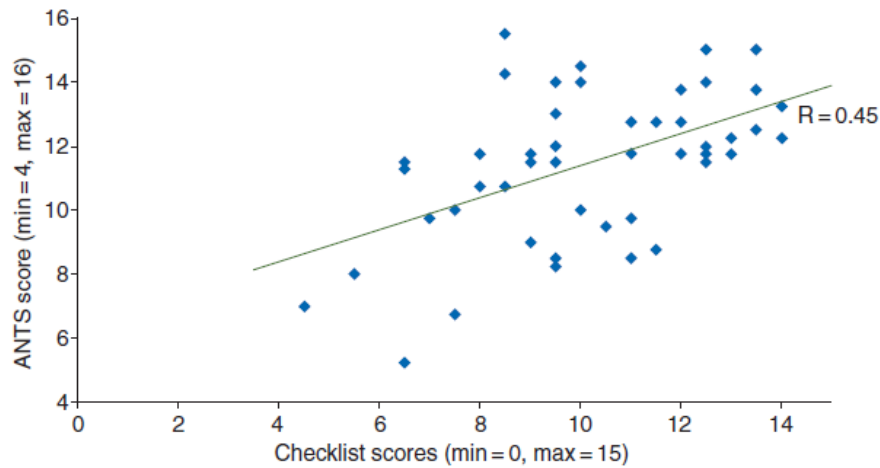


Figure 2.9: TS and NTS scores [59].

participants to watch videos of scripted anaesthetic situations and to rate the non-technical skills of the main anaesthetist in each scenario using the ANTS system. A number of specific experimental hypotheses were developed and used to drive the data collection and analysis process as shown in table 2.1.

2.2.3 Do technical skills (TS) correlate with non-technical skills (NTS)?

Riem et. al. [37] hypothesized that both TS and NTS are not independent of each other for ensuring patient safety in acute care practice and effective crisis management. They aimed to evaluate the relationship between TS and NTS during a simulated intraoperative crisis scenario. They conducted a study with 50 anaesthesiology residents who managed a simulated crisis scenario of an intraoperative cardiac arrest secondary to a malignant arrhythmia. They used a modified Delphi approach to design a TS checklist, specific for the management of a malignant arrhythmia requiring defibrillation. All scenarios were recorded. Each performance was analysed by four independent experts. For each performance, two experts independently rated the technical performance using the TS checklist, and two other experts independently rated NTS using the Anaesthetists' Non-Technical Skills score.

Their study showed that TS and NTS are associated and are not independent from each other during intraoperative crisis management. Technical performance, as

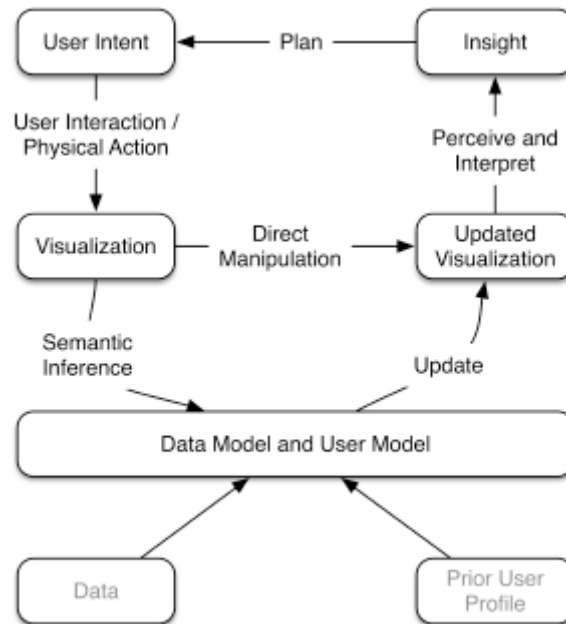


Figure 2.10: A generalizable model for coupling cognition and computation. Plans generate intents that are externalized by users via interactions and physical actions. Data and user models can be inferred from these actions, and used to update a visualization to continue the analytic process [60].

measured by their TS checklist score, and NTS, as measured by the total ANTS score, reached a correlation of 0.45 ($P < 0.05$). The relationship between ANTS categories and the TS checklist score had statistically significant correlations, with r ranging from 0.31 to 0.45 as shown in Figure 2.

2.2.4 Coupling TS and NTS

As we can see that TS and NTS have significant correlations, Endert et. al. [38] introduced the idea of coupling those for interactive analytics. The idea was – coupling those would support a true human-machine symbiotic relationship where users and machines work together collaboratively and adapt to each other to advance an interactive analytic process. They suggested ‘*semantic interaction*’ a solution concept to couple the cognitive (NTS) and computational components (TS) by binding the user interactions used for visual sensemaking. Their proposed generalized model (Figure 2.10) for coupling cognition (NTS) and computation (TS) takes an approach of directly binding model steering techniques to the interactive affordances created by the

visualization. The user interaction is directly applied in the visual metaphor, creating a bi-directional medium between the user and the analytic models. The bi-directionality afforded by semantic interaction comes via binding the parameter controls traditionally afforded by the GUI directly within the visual metaphor. Data and user models can be inferred from these actions to continue the coupled interaction process [38].

2.2.5 Summary

Whereas the previous section 2.1 addresses the gap between cognition and computation, the idea of '*Behavioural Marker (BM)*' can bridge this as Riem et. al. [37] found a correlation of 0.45 ($P < 0.05$) between TS and NTS through their study (section 2.2). We have aimed to test the hypothesis through this thesis work by using the concept '*semantic interaction*' as introduced by Endert et. al. [38]. Although the research on BMs came from the studies of civilian pilot and their non-technical skills (NTS), however BM systems are now being widely used for performance measurement in a range of organizational settings including evaluation of different visual analytic tools. Endert et. al. [38] also proposed a generalized model for coupling cognition and computation to infer data and user models, however more work is needed to test their model. How different cognitive constructs contribute to pinpoint their transitions, how such constructs can be translated in terms of computational interactions and how all of these concepts can be utilized to model user profile and understand their analytical behaviour. Section 2.2 has presented relevant literature where authors provided the idea of developing BM system and couple TS-NTS concepts in terms of performance analysis. We have aimed to utilize these ideas for our seminal research of finding out techniques to detect those BMs from sensemaking activities and automatically infer those for enhanced decision support systems.

2.3 Inferring Sensemaking Tasks

2.3.1 Clickstream Modelling

Wang et. Al. [39] proposed clickstream models to characterize user behaviour in large online services. By analyzing clickstream traces (i.e., sequences of click events from users), they sought to achieve two goals: (1) *detection*: to capture distinct user groups for the detection of malicious accounts, and (2) *understanding*: to extract semantic information from user groups to understand the captured behaviour. To run experiments they used ‘Renren’ (one of the largest online social networks in Chinese) dataset having goal to prevent attackers from creating large numbers of fake identities (*Sybils*) to disseminate unwanted contents. The ‘Renren’ dataset contained 5,998 normal users and their clickstream traces over 2 months in 2011 including 9,994 *Sybil* accounts randomly sampled from all previously banned accounts by *Renren*. Alongside they also used another 135 million click events from 100K users on ‘Whisper’, a popular anonymous social network app. To provide semantic interpretations on captured behaviour, authors proposed an *iterative feature pruning* algorithm to partition the clickstream similarity graph. The result is a hierarchy of clusters, where higher-level clusters represent more general user behaviour patterns and lower-level clusters further identify smaller groups that differ in key behavioural patterns.

As part of clickstream behaviour detection and interpretation, authors [39] built models of user activity patterns that can effectively distinguish *Sybils* from normal users. Their goal was to cluster similar clickstreams together to form general user “profiles” that capture specific activity patterns. To begin with this, they defined following three models to represent a user’s clickstream and for each model they described similarity metrics that allow to cluster similar clickstreams together.

2.3.1.1 Click Sequence Model: *Sybils* and normal users exhibit different click transition patterns and focus their energy on different activities. The Click Sequence

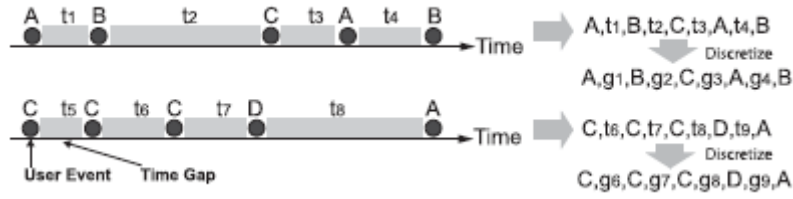


Figure 2.11: Discretizing two clickstreams into event sequences [39].

(CS) model treats each user’s clickstream as a sequence of click events, sorted by order of arrival.

2.3.1.2 Time-Based Model: The Time-based model focuses on the distribution of gaps between events: each user’s clickstream is represented by a list of interarrival times $[t_1, t_2, t_3, \dots, t_n]$ as shown in Figure 2.11, where n is the number of clicks in a user’s clickstream.

2.3.1.3 Hybrid Model: The Hybrid model combines click types and click interarrival times. Each user’s clickstream is represented as an in-order sequence of clicks along with interevent gaps between clicks. An example is shown in Figure 2.11. $[A, t_1, B, t_2, C, t_3, A, t_4, B]$, where A, B, C are click types, and t_i is the time interval between the i^{th} and $(i + 1)^{\text{th}}$ event.

2.3.2 Computing Sequence Similarity

Having defined three models of clickstream sequences, Wang et. al. [39] then investigated methods to quantify the similarity between clickstreams. In other words computing the distance between pairs of clickstreams. Authors defined three distance functions as follows:

2.3.2.1 Common Sub-sequences: It involves locating the common subsequences of varying lengths between two clickstreams. Authors [39] formalized a clickstream as a sequence $S = (s_1 s_2 \dots s_i \dots s_n)$, where s_i is the i^{th} element in the sequence. They then defined T_N as the set of all possible k -grams (k consecutive elements) in sequence S where $k \leq N$: $T_N(S) = \{k\text{-gram} | k\text{-gram} = (s_i s_{i+1} \dots s_{i+k-1}), i \in [1, n+1 - k], k \in [1, N]\}$. Simply put, each k -gram in $T_N(S)$ is a subsequence of S with a length of k . Finally, the distance between two sequences can then be computed based on the number of

common subsequences shared by the two sequences. Inspired by the *Jaccard Coefficient* [13], they defined the distance between sequences S_1 and S_2 as -

$$D_N(S_1, S_2) = 1 - \frac{|T_N(S_1) \cap T_N(S_2)|}{|T_N(S_1) \cup T_N(S_2)|}$$

2.3.2.2 Common Sub-sequences with Counts: The common subsequence metric defined above only measures distinct common subsequences; that is, it does not consider the frequency of common subsequences. Wang et. Al. [39] proposed a second distance metric that rectifies this by taking the count of common subsequences into consideration. For sequences S_1 , S_2 and a chosen N , we first compute the set of all possible subsequences from both sequences as $T = T_N(S_1) \cup T_N(S_2)$. Next, authors counted the frequency of each subsequence within each sequence i ($i = 1, 2$) as array $[c_{i1}, c_{i2}, \dots, c_{in}]$, where $n = |T|$. Finally, *Euclidean Distance* distance between S_1 and S_2 is -

$$D(S_1, S_2) = \frac{1}{\sqrt{2}} \sqrt{\sum_{j=1}^n (c_{1j} - c_{2j})^2}$$

2.3.2.3 Distribution-Based Method: The prior metrics cannot be applied to sequences of continuous values (i.e., the Time-based model). Instead, for continuous value sequences S_1 and S_2 , authors [39] computed the distance by comparing their value distribution using a two-sample *Kolmogorov-Smirnov* test (*K-S* test). A two-sample *K-S* test is a general nonparametric method for comparing two empirical samples. It is sensitive to differences in location and shape of the empirical *Cumulative Distribution Functions* (CDFs) of the two samples. They defined the distance function using *K-S* statistics:

$$D(S_1, S_2) = \sup_t |F_{n,1}(t) - F_{n,2}(t)|,$$

Where $F_{n,i}(t)$ is the CDF of values in sequence S_i .

2.3.3 Task Identification

Hua et. al. [40] built a conceptualization mechanism based on an external knowledgebase known as Probase [41] to infer the underlying conceptual meanings

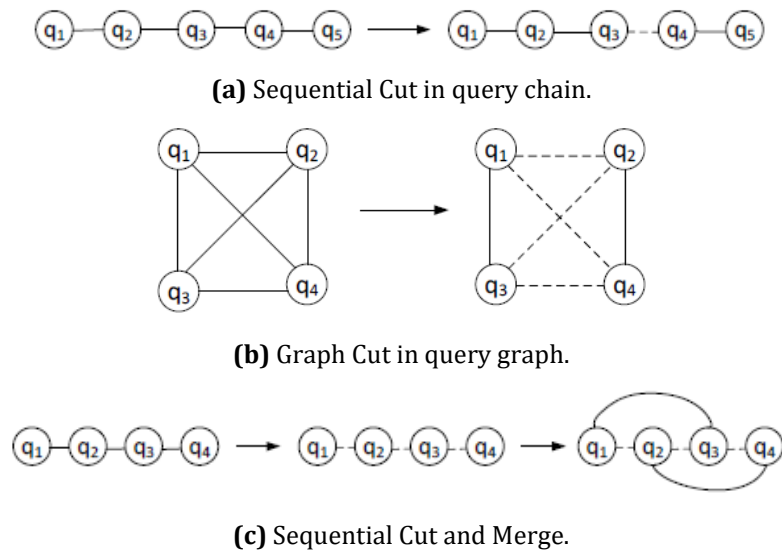


Figure 2.12: Illustration of SC (a), GC (b) and SCM (c) [40].

of queries and reduce query ambiguity. They employed lexical, conceptual, template and temporal features to measure query similarities, which are then used to estimate whether two queries should be clustered into one task and found that it can increase the task identification accuracy by 13.8% on average.

Most previous works assume that tasks are never interleaved with each other, so they simply try to detect task boundaries between consecutive queries. More specifically, they model each session as a query chain $G_1 = (V, E_1)$, in which $V = \{q_i\}$ is the set of queries, and $E_1 = \{edge(q_i, q_{i+1})\}$ is the set of undirected edges between consecutive queries. They then examine each edge in the query chain and remove it when the similarity between the two consecutive queries connected by that edge is smaller than the similarity threshold θ . As shown in Figure 2.12 (a), edge $edge(q_3, q_4)$ is removed from the query chain, or in other words, a task boundary between queries q_3 and q_4 is detected after edge examination. Finally, for each pair of consecutive queries, if they are still connected in the query chain after edge examination, they will be clustered together into one task. Authors denoted this process of task identification as *Sequential Cut (SC)*. To detect interleaved tasks, Jones et. al. [42] employed a *Graph Cut (GC)* algorithm. Particularly, they modelled each session as a query graph $G_2 = (V, E_2)$, in which $V = \{q_i\}$ is the set of queries, and $E_2 = \{edge(q_i, q_j) | i \neq j\}$ is the set of undirected edges between each pair of queries. They then examined each edge in the

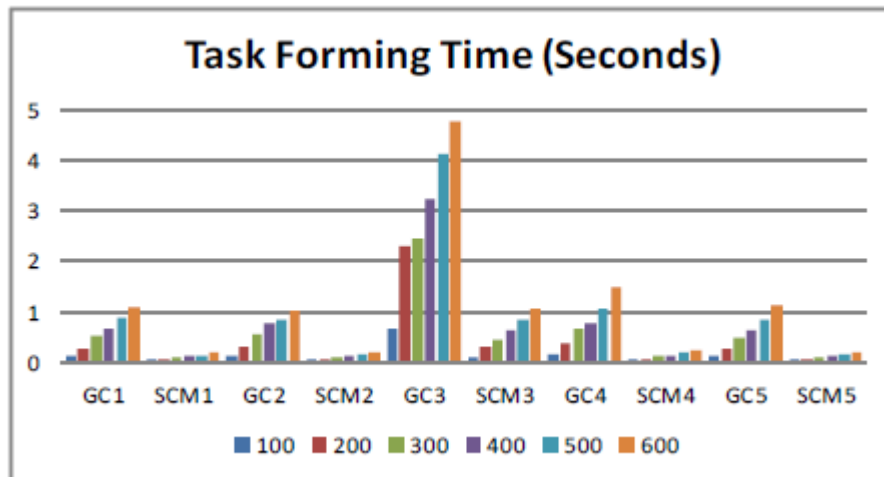


Figure 2.13: Task forming time [40].

query graph and removed it when the similarity between the two queries connected by that edge was insignificant. As shown in Figure 2.12 (b), edges $edge(q_1, q_2)$, $edge(q_1, q_4)$, $edge(q_2, q_3)$ and $edge(q_3, q_4)$ are removed from the query graph after edge examination.

In order to detect interleaved tasks more quickly and avoid over-merging at the same time, Hua et. al. [40] proposed a new algorithm for task identification, which they call *Sequential Cut and Merge (SCM)*, which can be considered as a combination of *SC* and *GC*. Or more specifically, authors first applied *SC* on the target session and referred to the tasks derived from *SC* as subtasks. They merged together the *Bag-of-Words (BoW)* interpretations of queries contained in a subtask to form a new query, which is used to represent that subtask. They then applied *GC* to the set of subtasks. In other words, authors [40] built a subtask graph $G_3 = (V', E_3)$ on the derived subtasks similar to the query graph. Here, $V' = \{Q_1, Q_2, \dots, Q_m\}$ is the set of subtasks, and E_3 is the set of edges connecting each pair of subtasks. They examined each edge in the subtask graph and removed it when the similarity between the two subtasks (represented by the new queries) connected by that edge was smaller than the similarity threshold θ . Finally, they merged together queries contained in subtasks that were still connected in the subtask graph after edge examination. As shown into Figure 2.12(c), in the *SC* process, edges $edge(q_1, q_2)$, $edge(q_2, q_3)$ and $edge(q_3, q_4)$ are removed from the query chain after edge examination, resulting in four subtasks with each subtask consisting

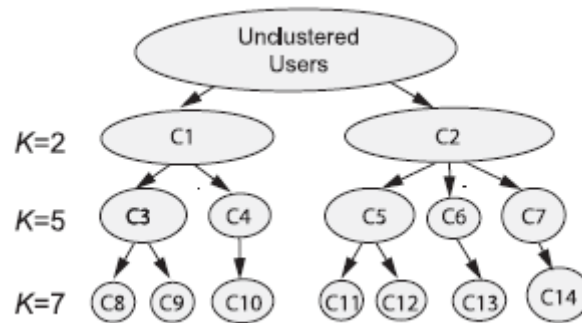


Figure 2.15: Hierarchy of the behavioural clusters [39].

To extend their work on clickstream analysis, authors [39] aimed to identify prevalent user behaviours as their next stage of research in a given service without any prior knowledge of labels (unsupervised). At the high level, they assumed that human behaviour naturally forms clusters.

Despite users' differences in personalities and habits, their behavioural patterns within a given service cannot be entirely disparate. The goal was to identify such natural clusters as behavioural models. In addition, user's behaviour is likely multidimensional. User clusters are likely to fall into a tree hierarchy instead of a one-dimensional structure as shown in Figure 2.15. In this hierarchy, most prominent features are used to place users into high-level categories, while less significant features characterize detailed substructures. Wang et. Al. [39] built a system based on their proposed algorithm to capture hierarchical clickstream clusters called *iterative feature pruning* which means of identifying fine grained behavioural clusters within existing clusters and recursively partitioning the similarity graph. As shown in Figure 2.15, by partitioning the similarity graph C_1 and C_2 are considered as the top-level clusters. Suppose C_1 is the current parent cluster. Then authors performed feature selection to determine the key features (i.e., k -grams) that classify users into C_1 . Then, to partition C_1 , they removed those top k -grams from the feature set and used the remaining k -grams to compute a new similarity graph for C_1 . In this way, secondary features can step out to partition C_1 into C_3 and C_4 . They ran the same process recursively to produce more fine-grained sub-clusters until the partition could not be split any further.

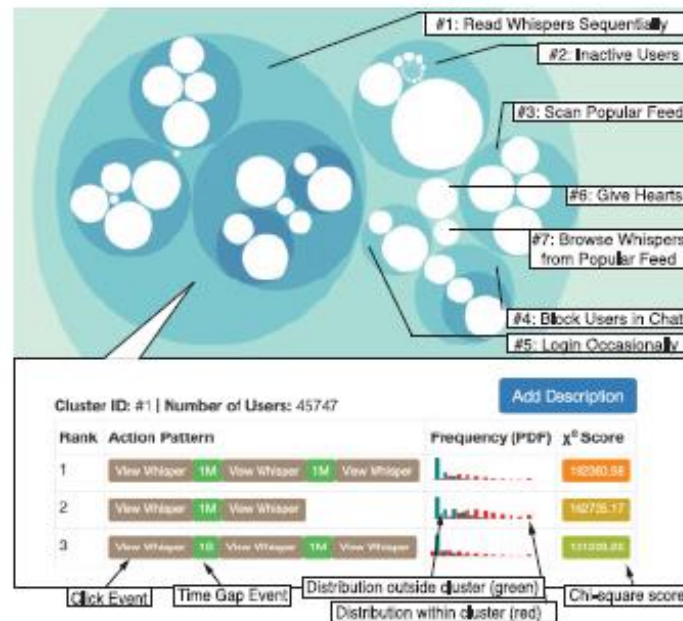


Figure 2.16: Whisper behavioural clusters. Cluster labels are manually input based on results of each cluster. The pop-up window shows users in Cluster #1 tend to sequentially read whispers [39].

As shown in Figure 2.16, authors [39] displayed the cluster hierarchy by using *Packed Circle*, where child clusters are nested within their parent cluster. This gives a clear view of the hierarchical relationships of different clusters. Circle sizes reflect the number of users in the cluster, which allows service providers to quickly identify the most prevalent user behaviours. They showed the basic cluster information (Figure 2.16) on top, including clusterID and the number of users. Below that a list of ‘Action Patterns’ (k-grams) selected by their ‘Feature Pruning’ algorithm to describe how users behave. The ‘Frequency (PDF)’ column shows how frequently each action pattern appears among users of this cluster. The red bar indicates the pattern frequency (*probability density function*) inside the cluster, and the green bar denotes frequency outside of this cluster. Intuitively, the more divergent the two distributions are, the more distinguishing power the pattern has. The aim of this pattern analysis was to model online user behaviour and detect malicious user accounts.

2.3.5 Summary

For automatically inferring user's intention while sensemaking, the first goal should be detecting clickstream behaviour detection and its modelling. By analyzing clickstream traces Wang et. al [12] sought to achieve two goals - (i) detecting distinct user groups and (ii) understanding semantic information from user groups. As part of clickstream behaviour detection and interpretation, they built models of user activity patterns and to provide semantic interpretations on captured behaviour they proposed an iterative feature pruning algorithm to partition the clickstream similarity graph. But it is not always possible to separate those entirely due to differences in personalities and habits; their behavioural patterns within a given service. Authors built a system based on their proposed algorithm to capture hierarchical clickstream clusters and visualized those by using packed circle as shown in Figure 2.16. Their system shows the basic cluster information, list of action patterns selected by their algorithm to describe how users behave. The frequency *PDF* into their system shows how frequently each action patterns appear among users of a cluster. The main aim of this pattern analysis was to model online user behaviour. However, due to cognitive variances and followed approaches pattern analysis can not be the only measurement of understanding user's tasks. We have addressed this issue in our research and aim to come up with a solution that will consider it while understanding user's task behaviour.

2.4 Machine Learning for Inference Making

It is generally difficult to infer meaningful actions quickly from the deluge of log data. Hence, inferring user actions from analytical log data is still a challenge. Li et. al. [43] addressed this issue and set out to infer user actions from a kernel-based cloud data provenance logger known as *Progger*. The key aspects of their approach were identifying the data pre-processing steps and attribute selection. They then used four standard classification models and identified the most accurate inference on user actions.

Classifier	Length	Window Length								
		1	2	3	4	5	6	7	8	Whole
Naive Bayes	Yes	70.83%	88.02%	90.10%	92.19%	92.71%	92.71%	93.23%	93.23%	86.46%
	No	41.67%	82.29%	89.06%	92.19%	92.71%	92.71%	93.23%	93.23%	80.21%
Multinomial Naive Bayes	Yes	47.40%	59.38%	53.65%	57.29%	63.54%	49.48%	47.40%	43.23%	7.29%
	No	37.50%	65.63%	73.44%	84.90%	86.46%	86.46%	89.06%	90.10%	79.69%
Nearest-neighbour (IB1)	Yes	78.65%	89.58%	92.19%	92.19%	92.71%	92.71%	92.71%	91.15%	84.90%
	No	33.33%	77.08%	86.46%	90.63%	93.23%	92.71%	92.71%	91.15%	78.13%
Decision Tree (J48)	Yes	78.65%	90.63%	91.67%	93.23%	92.19%	93.23%	92.71%	91.67%	83.85%
	No	40.63%	82.81%	88.02%	91.67%	92.19%	93.23%	92.71%	91.67%	71.35%

(i)

Classifier	Length	Window Length	Sixfold Cross-validation	Mono-scenario	Interleaving Scenario
Naive Bayes	Yes	8	93.23%	50.00%	66.67%
	No	8	93.23%	50.00%	83.33%
Multinomial Naive Bayes	Yes	5	63.54%	50.00%	66.67%
	No	8	90.10%	66.67%	83.33%
Nearest-neighbour (IB1)	Yes	6	92.71%	83.33%	83.33%
	Yes	7	92.71%	58.33%	55.56%
	No	5	93.23%	83.33%	83.33%
Decision Tree (J48)	Yes	4	93.23%	58.33%	83.33%
	Yes	6	93.23%	66.67%	83.33%
	No	6	93.23%	66.67%	83.33%

(ii)

Figure 2.17: Accuracy of classifiers on (i) Sixfold cross-validation, (ii) Supplied testing datasets [18].

2.4.1 Data Simulation

Li et. al. [43] used a training data set which is a log file generated by implementing each of the user scenario (Figure 2.14) several times. This includes the logs generated for different users as well as the logs generated for the same user at different instances. They also used two testing data sets: (a) *mono-scenario* testing data set [Figure 2.14(v)], (b) *interleaving scenario* testing data set [Figure 2.14(iv)]. The *mono-scenario* refers to a situation where several scenarios are implemented in a sequential fashion, i.e. one after the other. On the other hand, the *interleaving scenario* refers to a more realistic situation where several scenarios are run concurrently (i.e., log entries from different scenarios may interleave each other). For inferring user actions, authors also considered “*Window Length*” [Figure 2.14(ii), (iii)] as another important attribute which represents the number of log entries that are related to users scenarios.

2.4.2 Classification Algorithms

2.4.2.1 Single-Class Classification

Li et. al. [43] employed four commonly used classification methods, namely, *Naive Bayes Classifier*, *Multinomial Naive Bayes Classifier*, *Nearest-Neighbour Classifier (IB1)* and *Decision Tree (J48)* to find the best performing algorithm for classifying Progger logs. They found that the classification accuracy achieved in most cases was in the range of about 80% to 90% as shown in Figure 2.17(i), for all possible combinations of the two attributes “*Length*” and “*Window Length*”. The *Nearest-neighbour (IB1)*, the *Naive Bayes* and the *Decision Tree (J48)* seem to perform well but the accuracy achieved using the *Multinomial Naive Bayes* algorithm is significantly less. It is clear that including the length information does not necessarily improve the accuracy significantly. In general, the length feature seems to be only helpful when the window length is small (less than four) or equal to the length. For *Multinomial Naive Bayes*, the length lowers the classification accuracy except when the window length equals to one [43].

The four classification algorithms were then implemented using the optimal combinations of attributes on the two testing data sets, namely, the *mono-scenario testing data set* and *interleaving scenario testing data set*. The results are summarized in Figure 2.17(ii). *Nearest-neighbour (IB1)* with length attribute, window length = 6 and without length attribute, window length = 5 seems to outperform the others on the classification accuracy achieved on both the *mono-scenario* data set as well as the *interleaving scenario* data set. But in some cases, for e.g., *IB1* with window length = 7, the accuracy on the test data sets is significantly lower than the cross validation accuracy. This is possibly because the test datasets were small in size and did not incorporate a wide range of user scenarios [43].

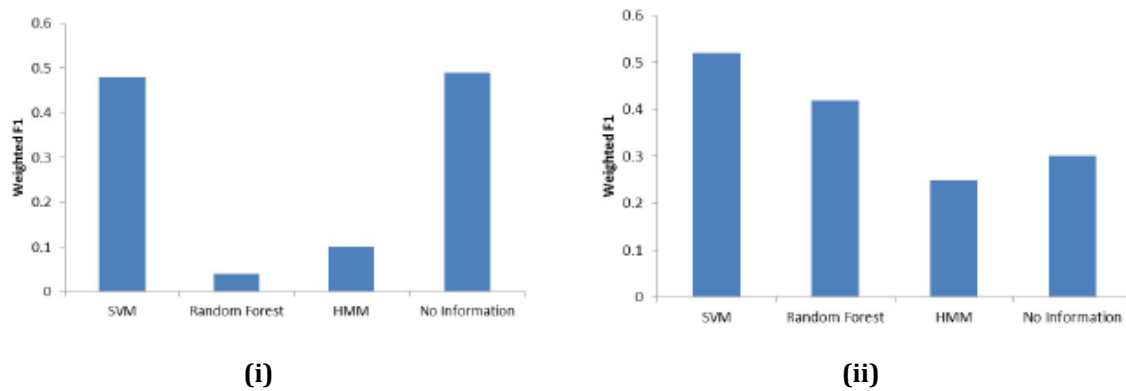


Figure 2.18: Reasoning task classification (i) Study A, weighted average F1 Measure model comparison, (ii) Study B, weighted average F1 measure model comparison [45].

2.4.2.2 Multi-Class Classification

Zhang et. al. [44] attempted to infer user’s online activities through traffic analysis in real time by using no more information than packet size, timing and direction. It was a challenging task to do this accurately with such limited information, especially among a wide range of network applications, such as web browsing, online chatting, online gaming, downloading, uploading, online video and BitTorrent. Specially, submerging applications and changeable features make the accurate identification of user’s online activities even more difficult. To overcome these challenges authors [44] explored an online *hierarchical classification system* based on machine learning (ML) techniques to map traffic features to the online activities and showed that their system can distinguish different online applications on the accuracy of about 80% in 5 seconds and over 90% accuracy if the eavesdropping lasts for 1 minute. Their classification system performed *multiclass classification* by taking advantage of both the efficient computation of *decision-tree* structure and the high classification accuracies of *Support Vector Machine (SVM)* and *Neural Network (NN)* algorithms.

2.4.2.3 Reasoning Task Classification

Kodagoda et al. [45] presented a novel method for reconstructing reasoning provenance from analysis provenance records to map actions to reasoning by using modified ‘*Data-Frame Model*’ version of Klein et al. [46]. The approach that they took for such reconstruction was to automatically infer reasoning from low-level user interaction logs. Novel machine learning methods were used for inference making and

two user studies were conducted to collect data. In *Study A* [Figure 2.18(i)], participants used web browser and for *Study B* [Figure 2.18(ii)], a visual search/query system called *INVISQUE* [7] was used for performing same intelligence analysis task. All interaction logs were manually encoded captured from think-aloud protocol and post-task interviews to map cognitive actions. They used SVM (*Support Vector Machine*), RF (*Random Forest*), HMM (*Hidden Markov Model*) classifiers to test the hypothesis that computer-based interactions can provide information to aid in recovering reasoning. Performances were evaluated by looking at the overall classification accuracy of the three models for both studies, mainly by considering weighted average F1 measure.

Through their studies, SVM was found as the best model in *Study A* whereas none of the models significantly outperformed the no information rate classifier ($p < 0.05$). The RF model performed particularly badly because of the high bias present in the data.

HMM also did not perform well in this study. In *Study B*, SVM was also found as the best model and it did significantly outperform the no information rate classifier ($p < 2.2e-16$). The RF model performed significantly better than the no-information rate ($p = 2.272e-12$) as there is less bias in the main feature set. The HMM model did not perform better than other models in this study.

2.4.2.4 Non-Contextual Classification

Gramazio et. al. [47] conducted studies to understand the degree to which anonymized interaction logs could be used to understand analytic intent given the complete omission of context. They used twelve automated visual analysis task classification models including k -nearest neighbours, linear support vector machines (SVMs), random forests (RFs) to hand-coded task inferences. Aims of their studies were to test how consistently tasks [10] can be inferred using only low level interaction logging data. For model's classification evaluation they considered three feature sets: 'dwell', 'region-of-interest (ROI) transition', a novel 'mouse tracking' approach and 'all' which combined features from aforementioned sets. Their final experimental design consisted of twelve classification models (3 classifiers \times 4 feature sets). They evaluated those 12 classifiers to test whether automated classification



Figure 2.19: The interface from Brown et. al.'s [48] study in which participants found Waldo while authors recorded their mouse interactions. Inset (a) shows Waldo himself, hidden among the trees near the top of the image. Distractors such as the ones shown in inset (b) and (c) help make the task difficult.

could predict visual analysis tasks with comparable accuracy to domain experts. match-any accuracy ties. The twelve models' match-any accuracies ranged from 38% (linear SVM, dwell) to 73% (random forest, mouse tracking) and the modal accuracies ranged from 18% (k-nearest neighbours, dwell) to 56% (random forest, mouse tracking). But their evaluation results only considered supervised learning approaches, which left the potential effect of unsupervised approaches an open problem. This open problem can be tested in the future by evaluating whether clustering based on geometric-temporal distances of interaction segments can accurately predict visual analysis tasks. However, one barrier to this approach, which must also be examined, is how to best segment interaction logs into discrete components that accurately represent stages of visual analysis.

2.4.2.5 Machine Learning for Visual Analytic Systems

Shen et. al. [49] proposed a *TaskTracer* system that helps multi-tasking users manage the resources that they create and access while carrying out their work activities. *TaskTracer* assumes that "activities" provide a useful abstraction for organizing and accessing resources. They developed *TaskTracer* system based on two main premises: (a) the behaviour of the user at the desktop is a mixture of different activities and (b) each activity is associated with a set of resources relevant to that activity. The first system is *TaskPredictor.WDS*, and it predicts the current task based on properties of

the window currently in focus. The second system is *TaskPredictor.email*, and it predicts the current task based on properties of incoming email messages (sender, recipients, subject, etc). The authors also proposed a *TaskPredictor* that attempts to predict the current task of users in case they forget to notify the system every time they change activities. The authors adapted machine learning techniques for predicting the current task of the user. They demonstrated that three machine learning techniques gave improved performance with these systems: 1) feature selection via mutual information, 2) a threshold for making classification decisions, and 3) a hybrid approach in which a generative model (Naive Bayes) is first applied to decide whether to make a prediction and then a discriminative model (linear support vector machines) is applied to make the prediction itself. The experiments show that the hybrid method gives slightly better performance than either Naive Bayes or SVMs alone. The overall results show that *TaskPredictor.WDS* can achieve more than 80% precision with 10–20% coverage (i.e., proportion of the time that a prediction is made). *TaskPredictor.email* can achieve more than 90% precision with 65% coverage.

Brown et. al. [48] demonstrated a small visual analytics subtask to show that it is indeed possible to automatically extract high-level semantic information about users and their analysis processes from mouse and keyboard interactions. They utilized those interaction data and applied machine learning techniques to predict user's (1) task performance and (2) infer personality traits. For the visual analytics task authors chose *Waldo* as shown in Figure 2.19, which is a famous children's game consisting of illustration spreads in which children are asked to locate the character *Waldo*. Participants were asked to navigate the image by clicking the interface's control bar. For the analysis, mouse click events on interface buttons were logged with a record of the specific button pressed and a time stamp. To establish labels for the machine learning analysis of performance outcomes and personality traits, authors recorded both completion time and personality survey scores for each participant. By using low-level interaction data they created three encodings: (1) *state-based*, which captures the total state of the software based on what data (portion of the image) is showing, (2) *event-based*, which captures the user's actions through statistics of the

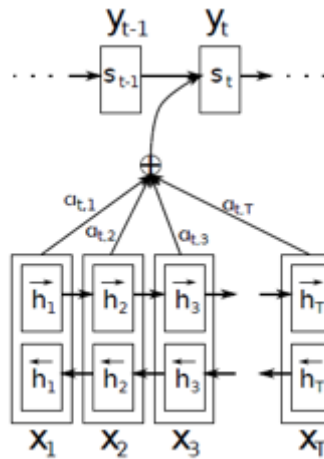


Figure 2.20: Attention model proposed by Bahdanau et. al. [50].

raw mouse activity, and (3) *sequence-based*, which encodes sequences of clicks on the interface’s buttons. The encoded information was then analyzed using well-known machine learning techniques such as support vector machines (SVM) and decision trees to classify groups of users with performance outcomes and individual differences. Authors attained 62% and 83% accuracy at differentiating participants who completed the task quickly versus slowly, with state-based yielding up to 83%, event-based up to 79% accuracy, and sequence-based 79%.

The ability to classify users is interesting on its own, but an adaptive system could test the feasibility of applying this type of results in real time. Different cognitive traits may prove more fruitful for adaptation. Also in sequence-based analysis, authors used pair n-grams with decision trees for readability, but there are plenty of existing treatments of sequence data that remain to be tried for this type of data classification on visual analytic tasks, including sequence alignment algorithms, and random process models, e.g., Markov models.

2.4.3 Contextual Classification – Attention Model

Conventional ‘*Topic modelling*’ technique can chunk semantically similar text from a large corpus and tag those with a topic name. These topic names are useful for clustering or organizing large blocks of textual data, information retrieval from unstructured text and feature selection. However, ‘tags’ as representatives of semantic texts may not express user’s actual intention and requires computation of

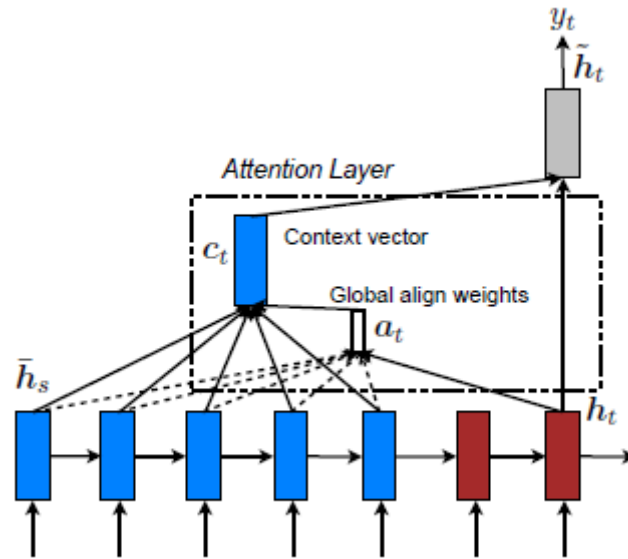


Figure 2.21: Global attention model as proposed by Luong et. al. [51] - at each time step t , the model infers a variable-length alignment weight vector α_t based on the current target state h_t and all source states \bar{h}_s . A global context vector c_t then computed as the weighted average, according to α_t , over all the source states.

intention from association of different contexts. Bahdanau et. al. [50] proposed the first ‘*Attention Model*’ as shown in Figure 2.20.

The ‘*Bidirectional Long Short Term Memory (LSTM)*’ [52] used here generates a sequence of annotations $(h_1, h_2, \dots, h_{T_x})$ for each input sentence. All the vectors h_1, h_2, \dots , etc., used in their work are basically the concatenation of forward and backward hidden states in the encoder [50].

$$h_j = \lceil \vec{h}_j^T; \overleftarrow{h}_j^T \rceil^T$$

To put it in simple terms, all the vectors $h_1, h_2, h_3, \dots, h_{T_x}$ are representations of T_x number of words in the input sentence. In the simple encoder and decoder model, only the last state of the encoder *LSTM* [52] was used (h_{T_x} in this case) as the *context vector* c_i for the output word y_i which is generated using the weighted sum of the annotations:

$$c_i = \sum_{j=1}^{T_x} \alpha_{ij} h_j$$

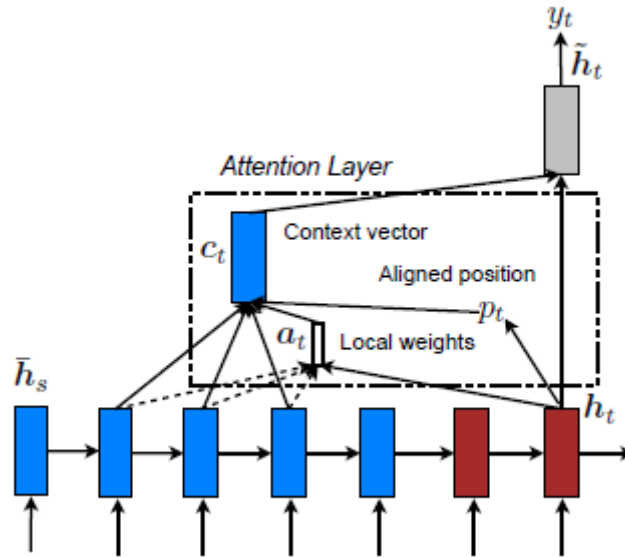


Figure 2.22: Local attention model as proposed by Luong et. al. [52] - the model first predicts a single aligned position p_t for the current target word. A window centered around the source position p_t is then used to compute a context vector c_t , a weighted average of the source hidden states in the window. The weights α_t are inferred from the current target state h_t and those source states h_s in the window.

The weights α_{ij} are computed by a *softmax function* given by the following equation:

$$\alpha_{ij} = \frac{\exp(e_{ij})}{\sum_{k=1}^{T_x} \exp(e_{ik})}$$

$$e_{ij} = a(s_{i-1}, h_j)$$

e_{ij} is the output score of a feedforward *neural network* described by the function a that attempts to capture the alignment between input at j and output at i [50].

2.4.3.1 Global Attention

The idea of a '*global attentional*' model (Figure 2.21) is to consider all the hidden states of the encoder when deriving the context vector c_t . In this model type, a variable-length alignment vector α_t , whose size equals the number of time steps on the source side, is derived by comparing the current target hidden state h_t with each source hidden state \bar{h}_s [51]:

$$\alpha_t(s) = \text{align}(h_t, \bar{h}_s)$$

$$= \frac{\exp(\text{score}(h_t, \bar{h}_s))}{\sum_{s'} \exp(\text{score}(h_t, \bar{h}_s))}$$

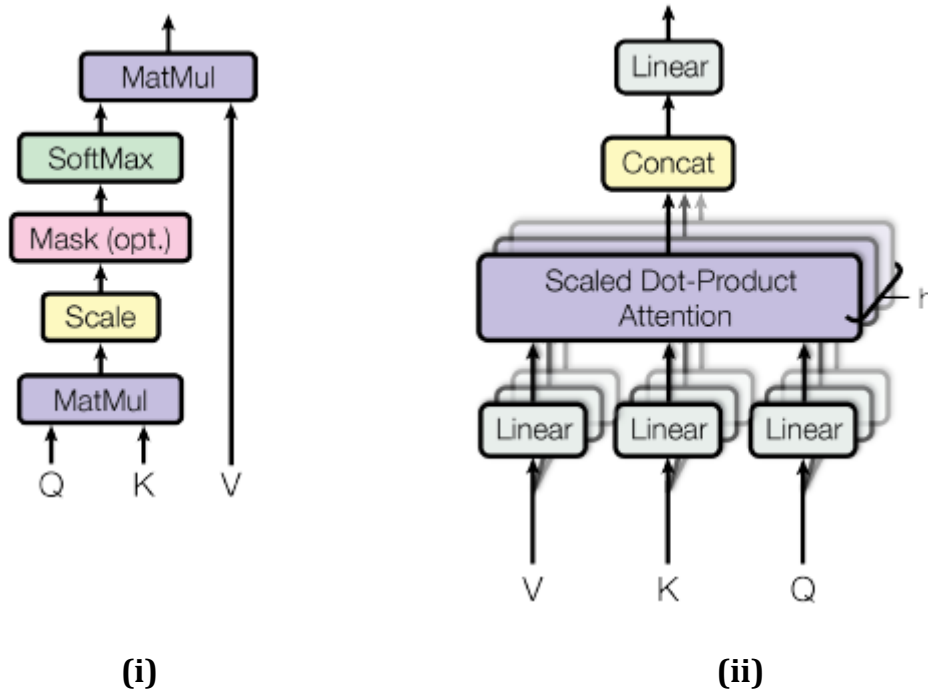


Figure 2.23: Attention models proposed by Vaswani et. al [53] – (i) Scaled Dot-Protect Attention, (ii) Multi-Head Attention consists of several attention layers running in parallel.

2.4.3.2 Local Attention

The ‘*global attention*’ has a drawback that it has to attend to all words on the source side for each target word, which is expensive and can potentially render it impractical to translate longer sequences, e.g., paragraphs or documents. To address this deficiency, Luong et. al. [51] proposed a ‘*local attentional*’ mechanism (Figure 2.22) that chose to focus only on a small subset of the source positions per target word.

2.4.3.3 Self-Attention

Self-attention, sometimes called ‘*intra-attention*’ [52] is an attention mechanism relating different positions of a single sequence in order to compute a representation of the sequence [53]. *Self-attention* allows the model to look at the other words in the input sequence to get a better understanding of a certain word in the sequence. Vaswani et. al [53] computed the attention function on a set of queries simultaneously, packed together into a matrix Q . And keys and values are packed into matrices K and V , where Q, K, V and *output* are all vectors. These vectors are trained and updated during the training process. They computed the matrix of outputs as:

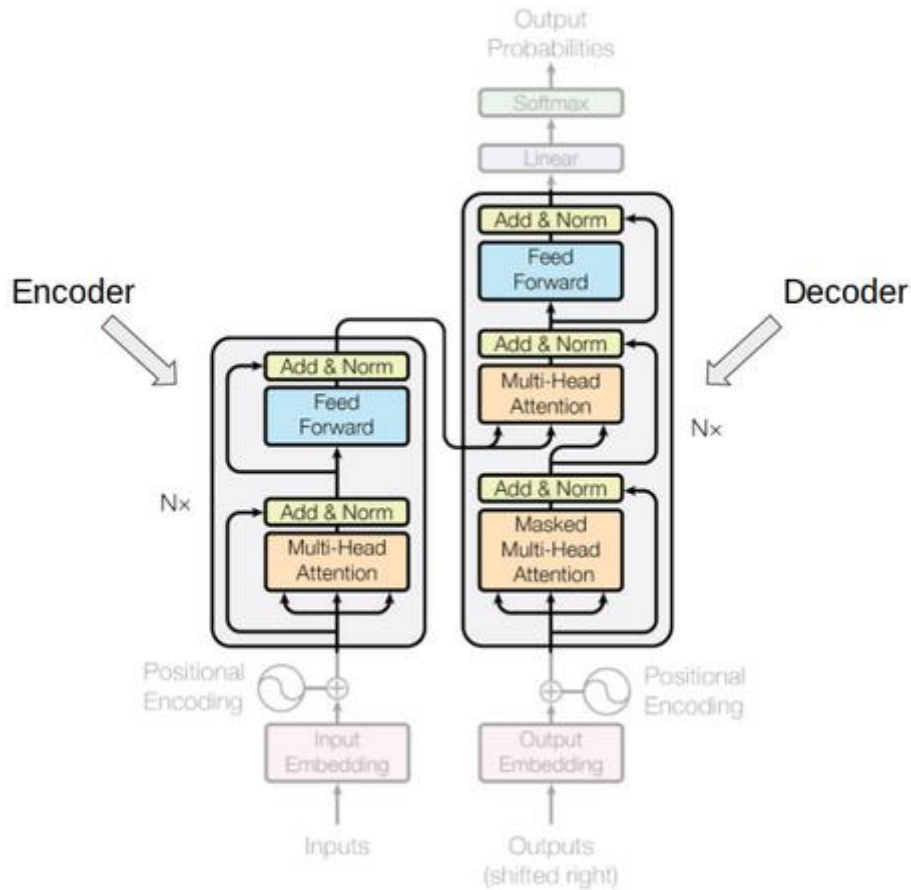


Figure 2.24: The Transformer – model architecture [53].

$$Attention(Q, K, V) = softmax\left(\frac{QK^T}{\sqrt{d_k}}\right)V$$

Self-attention is computed not once but multiple times in the ‘Transformer’ architecture [53, 54], in parallel and independently.

Multi-head attention allows the model to jointly attend to information from different representation subspaces at different positions. It is therefore referred to as ‘*Multi-head Attention*’ [53]. The outputs are concatenated and linearly transformed as shown in Figure 2.23(ii).

2.4.3.4 Encoder-Decoder

Most competitive neural sequence transduction models have an *encoder-decoder* structure. Here, the *encoder* maps an input sequence of symbol representations (x_1, \dots, x_n) to a sequence of continuous representations $z = (z_1, \dots, z_n)$. Given z , the

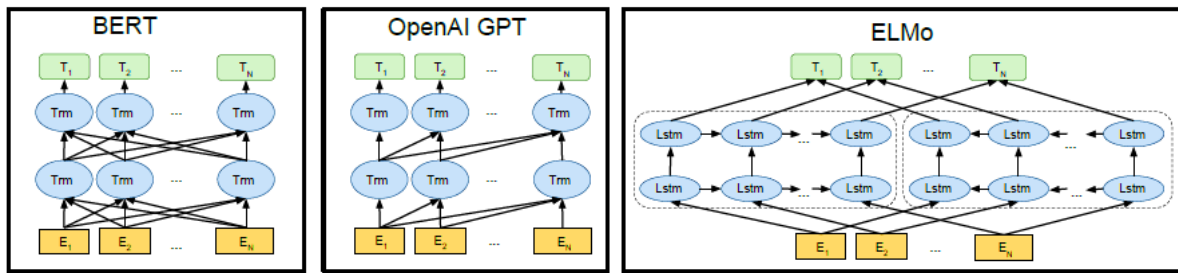


Figure 2.25: Differences among BERT [55], OpenAI GPT [56] and ELMo [57] in pre-training model architecture.

decoder then generates an output sequence (y_1, \dots, y_m) of symbols one element at a time. At each step the model is auto-regressive, consuming the previously generated symbols as additional input when generating the next [53].

2.4.3.5 The Transformer

The *Transformer* in NLP is a novel architecture that aims to solve ‘*sequence-to-sequence tasks*’ while handling long-range dependencies with ease. Vaswani et. al [53] described *Transformer* an overall architecture using stacked ‘*self-attention*’ and point-wise, fully connected layers for both the ‘*encoder*’ and ‘*decoder*’. As shown in Figure 2.24, the *encoder* block has 1 layer of a ‘*Multi-Head Attention*’ followed by another layer of ‘*Feed Forward*’ neural network. The decoder, on the other hand, has an extra ‘*Masked Multi-Head Attention*’. The encoder and decoder blocks are actually multiple identical encoders and decoders stacked on top of each other. Both the encoder stack and the decoder stack have the same number of units [54].

2.4.3.6 Bi-directional Encoder Representation from Transformers (BERT)

The BERT framework is a new language representation model developed by Google AI team. It uses *pre-training* and *fine-tuning* to create state-of-the-art models for a wide range of tasks [54]. As *pre-training* process, the model is trained on unlabelled data (un-supervised or semi-supervised). Then for *fine-tuning*, the BERT model is first initialized with the pre-trained parameters, and all of the parameters are fine-tuned using labelled data from the downstream tasks (supervised). BERT uses a multi-layer bidirectional Transformer encoder. Its self-attention layer performs self-attention in both directions. Google has released two variants of the model:

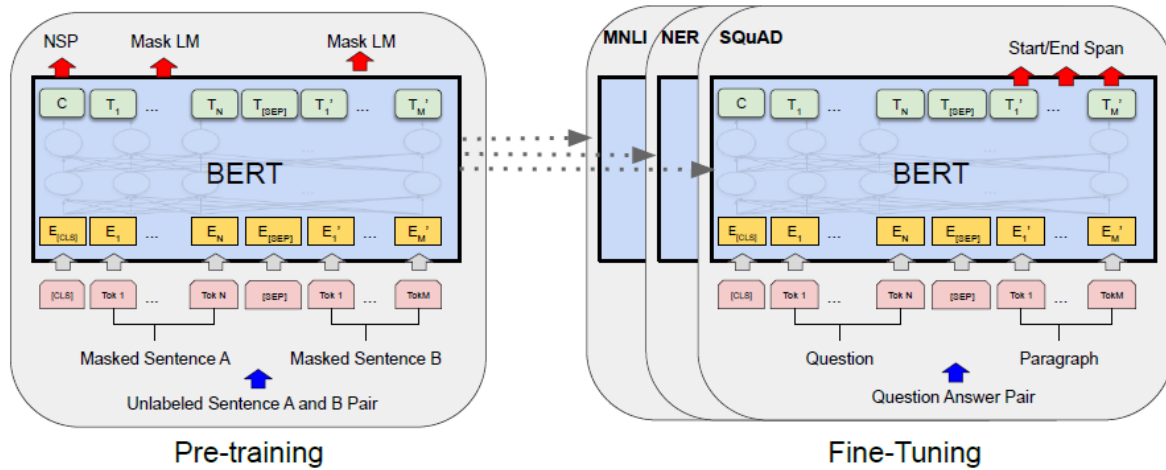


Figure 2.26: Overall pre-training and fine-tuning procedures for BERT. Apart from output layers, the same architectures are used in both pre-training and fine-tuning. The same pre-trained model parameters are used to initialize models for different down-stream tasks. During fine-tuning, all parameters are fine-tuned. [CLS] is a special symbol added in front of every input example, and [SEP] is a special separator token (e.g., separating questions/answers) [55].

BERT Base: Number of Transformers layers = 12, Total Parameters = 110M

BERT Large: Number of Transformers layers = 24, Total Parameters = 340M

2.4.3.7 BERT Pre-Training

BERT is pre-trained using the following two unsupervised prediction tasks.

i. Masked Language Modelling (MLM)

The masked language model randomly masks some of the tokens from the input, and the objective is to predict the original vocabulary id of the masked word based only on its context. Unlike the left-to-right language model pre-training, the MLM objective allows the representation to fuse the left and the right context, which allows to pre-train a deep bidirectional Transformer [55].

Authors used the below technique for pre-training:

- 80% of the time the words were replaced with the masked token [MASK].
- 10% of the time the words were replaced with random words.
- 10% of the time the words were left unchanged.

Differences (as shown in Figure 2.25) between BERT and other models (i.e, OpenAI GPT, ELMo) in pre-training are - BERT uses a bidirectional Transformer. OpenAI GPT uses a left-to-right Transformer. ELMo uses the concatenation of independently trained left-to-right and right-to-left LSTMs to generate features for downstream tasks. Among the three, only BERT representations are jointly conditioned on both left and right context in all layers. In addition to the architecture differences, BERT and OpenAI GPT are *fine-tuning* approaches, while ELMo is a *feature-based* approach [55].

ii. Next Sentence prediction (NSP)

In order to train a model that understands sentence relationships, authors [55] pre-trained for a binarized *next sentence prediction* task that can be trivially generated from any monolingual corpus. Specifically, when choosing the sentences *A* and *B* for each pre-training example, 50% of the time *B* is the actual next sentence that follows *A* (labelled as *IsNext*), and 50% of the time it is a random sentence from the corpus (labelled as *NotNext*). As shown in figure 2.26, *C* is used for *Next Sentence Prediction* (NSP). The vector *C* is not a meaningful sentence representation without *fine-tuning*, since it was trained with NSP.

2.4.3.8 BERT Fine-tuning

The pre-trained BERT which is trained on huge dataset as a starting point, can then be used further for training the smaller dataset. This process is known as *model fine tuning* [54]. Fine-tuning is straightforward since the *self-attention* mechanism in the *Transformer* allows BERT to model many downstream tasks—whether they involve single text or text pairs—by swapping out the appropriate inputs and outputs.

2.4.4 Summary

From section 2.3 we have found that authors have raised concerns of using data pattern or cluster analysis for user's behaviour modelling due to differences in personalities and habits. In section 2.4, we have presented some existing machine learning approaches to infer meaningful actions due to this problem. Li et. al.

[18] described two data simulation approaches for preparing training datasets. One of those is '*mono scenario*' referring to situation where scenarios are implemented in a sequential fashion and the other one is '*interleaving scenario*' referring to more realistic situation where several scenarios are run concurrently. They used '*Naive Bayes Classifier*', '*Multinomial Naive Bayes Classifier*', '*Nearest-Neighbour Classifier (IB1)*' and '*Decision Tree (J48)*' as classifiers to find out the best performing algorithm for both scenarios. Brown et. al. [48] created three encodings of low level interaction data and achieved accuracies up to (i) 83% for state-based, (ii) 79% for event-based, (iii) 79% for sequence-based by applying SVMs. Zhang et. al. [44] performed multiclass classification by taking advantage of both the efficient computation of decision-tree structure and the high classification accuracies of '*Support Vector Machine (SVM)*' and '*Neural Network (NN)*' algorithms and achieved accuracy of about 80%. We have presented some other aspects of task classification from deluge of log dataset. Kodagoda et. al. [45] used Klein et' al's [46] data frame model for inferring reasoning tasks from low-level user interaction logs and found '*SVM (Support Vector Machine)*' model outperforms other classifiers for this purpose.

So far we have discussed about all known scenarios and applying supervised algorithms for developing models. Gramazio et. al. [47] conducted studies to understand the degree to which anonymized interaction logs could be used to understand analytic intent. After applying combination of 12 classification models they found accuracies ranged from 18% - 73%. We have addressed this issue in our research and attempted to contextualize such data by using a bi-directional encoder-representation [55] originally proposed by Google to improve search results. To contextualize it uses some attention models [50] i.e, local [51], global [51] and self-attentions [52]. This approach is known as BERT [29] in NLP built upon the '*Transformer*' model which solves '*sequence-to-sequence*' tasks while handling long-range dependencies. We have described in section 2.4.3.6 how to implement this model by pre-training (trained on unlabelled data) and fine-tuning (by using labelled data).

2.5 eXplainable AI (XAI)

Although machine learning models can produce good results, it is often unclear how those AI techniques make decisions. Decision making needs to be transparent for building trust in machine learning models. *Explainable AI (XAI)* answers those questions to build trusts of users on AI systems. Such as -

- Why does the model predict that result?
- What are the reasons for a prediction?
- What is the prediction interval?
- How does the model work?

Basically, most of the machine learning models are referred to as *black-boxes* in terms of interpretability. So, *model explainability* in terms of human understanding has high priority challenge in today's machine learning community.

2.5.1 Explainability and Interpretability

The terms '*interpretability*' and '*explainability*' are usually used by researchers interchangeably [58]. There is not a concrete mathematical definition for interpretability or explainability, nor have they been measured by some metric.

2.5.1.1 Interpretability - One of the most popular definitions of interpretability is the one of Doshi-Velez and Kim, who, in their work [59], define it as “the ability to explain or to present in understandable terms to a human”. Another popular definition came from Miller in his work [61], where he defines interpretability as “the degree to which a human can understand the cause of a decision”. Although intuitive, these definitions lack mathematical formality and rigorousness [60]. Interpretability is mostly connected with the intuition behind the outputs of a model; with the idea being that the more interpretable a machine learning system is, the easier it is to identify cause-and-effect relationships within the system's inputs and outputs [58].

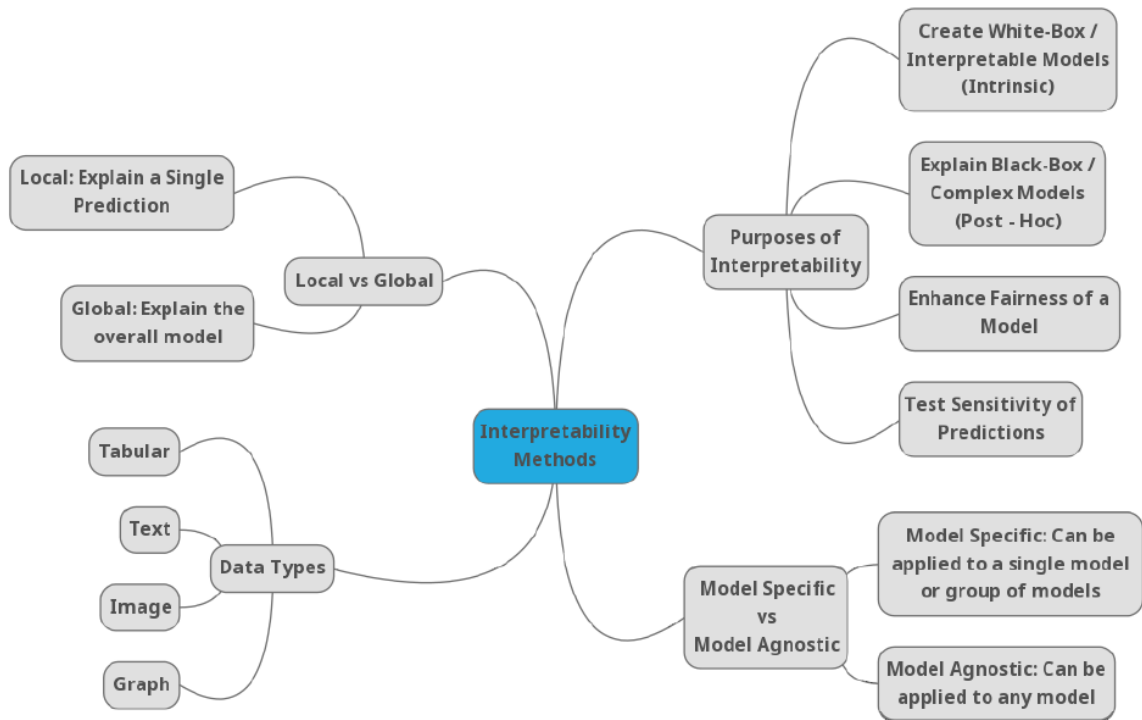


Figure 2.27: Taxonomy mind-map of Machine Learning Interpretability Techniques [58].

2.5.1.2 Explainability - Explainability, on the other hand, is associated with the internal logic and mechanics that are inside a machine learning system. The more explainable a model, the deeper the understanding that humans achieve in terms of the internal procedures that take place while the model is training or making decisions [58].

An interpretable model does not necessarily translate to one that humans are able to understand the internal logic of or its underlying processes. Therefore, regarding machine learning systems, interpretability does not axiomatically entail explainability, or vice versa. As a result, Gilpin et al. [62] supported that interpretability alone is insufficient and that the presence of explainability is also of fundamental importance. Mostly aligned with the work of Doshi-Velez and Kim [59], our research considers interpretability to be a broader term than explainability.

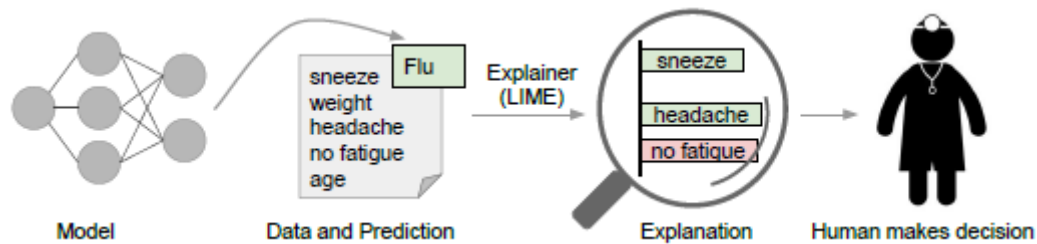


Figure 2.28: Explaining individual predictions. A model predicts that a patient has the flu, and LIME highlights which symptoms in the patient’s history led to the prediction. Sneezes and headaches are portrayed as contributing to the “flu” prediction, while “no fatigue” is evidence against it. With these, a doctor can make an informed decision about the model’s prediction. [63].

2.5.2 Taxonomy of XAI Methods

Different view-points exist when it comes to looking at the emerging landscape of interpretability methods, such as the type of data these methods deal with or whether they refer to ‘*global*’ or ‘*local*’ properties etc. As shown in Figure 2.27, Linardatos et al. [58] presents a summarized mindmap, which visualizes the different aspects by which an interpretability method could be classified. These aspects should always be taken into consideration by practitioners, in order for the ideal method with respect to their needs to be identified. Definitions of few of the criteria as shown in Figure 2.27 are as follows:

2.5.2.1 Intrinsic or Post Hoc?

Intrinsic explainability refers to machine learning models that are considered interpretable due to their simple structure, such as short decision trees or sparse linear models. *Post hoc* explainability refers to the application of interpretation methods after model training. Permutation feature importance is, for example, a post hoc explanation method.

2.5.2.2 Model-Specific or Model-Agnostic?

Model-specific methods are limited to specific model classes. The explanation of regression weights in a linear model is a model-specific explanation. *Model-Agnostic* methods can be used on any machine learning model and are applied after the model

has been trained (post hoc). These agnostic methods usually work by analyzing feature input and output pairs.

2.5.2.3 Local or Global?

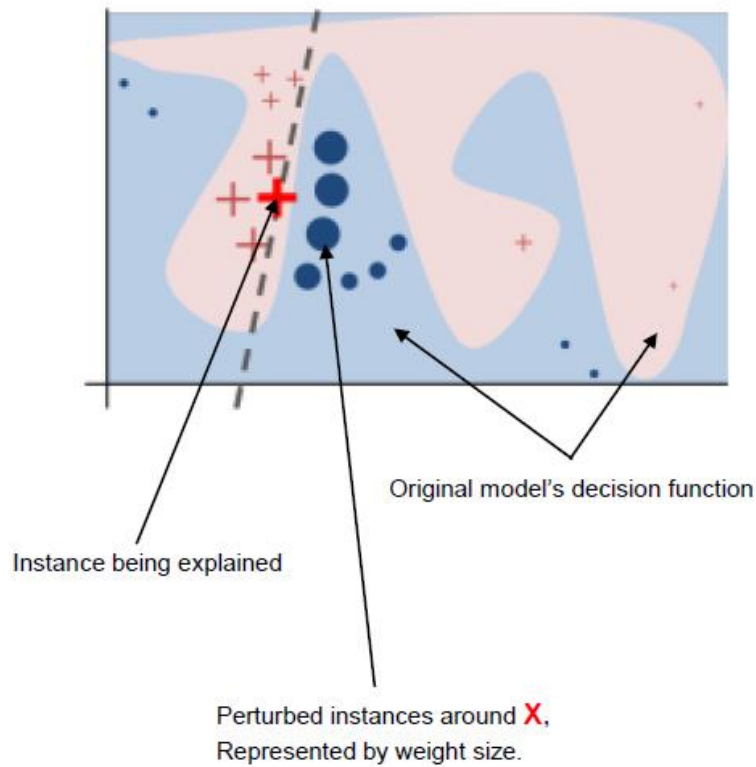
Does the method explain an individual prediction or the entire model behaviour? Or is the scope somewhere in between?

The above taxonomy (Figure 2.27), focuses on the purpose that these methods were created to serve and the ways through which they accomplish this purpose. As a result, according to the presented taxonomy, four major categories for interpretability methods are identified: methods for explaining complex black-box models, methods for creating white-box models, methods that promote fairness and restrict the existence of discrimination, and, lastly, methods for analysing the sensitivity of model predictions.

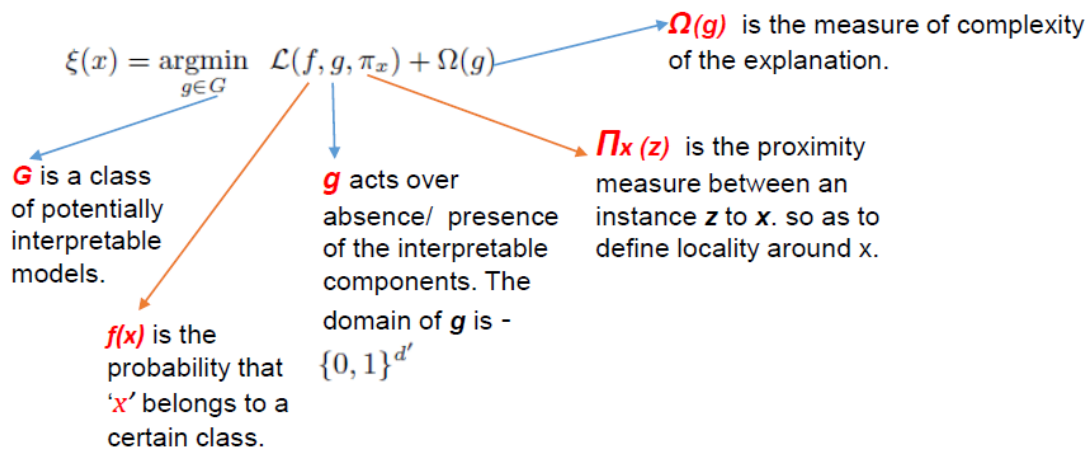
2.5.3 Popular XAI Techniques

2.5.3.1 Local Interpretable Model-Agnostic Explanations (LIME)

In 2016, Ribeiro et al. [63] introduced '*Local Interpretable Model-Agnostic Explanations (LIME)*' to derive a representation that is understandable by humans. LIME algorithm can explain the predictions of any classifier or regressor in a faithful way, by approximating it locally with an interpretable model. As shown in figure 2.28, LIME highlights the most and least contributing symptoms for the prediction 'flu'. LIME uses a local surrogate model trained on perturbations of the data point we are investigating for explanations. Figure 2.29(i) provides better understanding of how LIME represents perturbed instances around the explained instance. The original model's decision function is represented by the '*blue/pink*' background, and is clearly nonlinear. The '*bright red cross*' is the instance being explained (denoted as X). Instances are perturbed around X, and weighted according to their proximity to X (weight here is represented by size).



(i)



(ii)

Figure 2.29: (i) Representation of proximity calculation of LIME instances, (ii) The LIME equation [63].

Original model's prediction is obtained on these perturbed instances, and then a linear model (dashed line) is learnt that approximates the model well in the vicinity of X. The explanation in this case is not faithful globally, but it is faithful locally around X [63].



Figure 2.30: SHAP local explanation based on assigning a numeric measure of credit (marginal contribution) to each input feature [64].

2.5.3.2 Shapley Additive exPlanations (SHAP)

The Shapley value is the average contribution of a feature value to the prediction in different coalitions. For each of coalitions prediction is calculated with or without the feature value. The feature value is the numerical or categorical value of a feature and instance; the Shapley value is the feature contribution to the prediction. The effect of each feature is the weight of the feature times the feature value.

Computing feature contribution - Let's assume a linear model prediction for one data instance -

$$\hat{f} = \beta_0 + \beta_1 x_1 + \dots + \beta_p x_p$$

where 'X' is the instance for which we want to compute the contributions. Each x_j is a feature value, with $j = 1, \dots, \dots, p$. The β_j is the weight corresponding to feature j . So, the contribution ϕ_j of the j -th feature on the prediction $\hat{f}(x)$ can be denoted as -

$$\phi_j(\hat{f}) = \beta_j x_j - E(\beta_j X_j) = \beta_j x_j - \beta_j E(X_j)$$

Where $E(\beta_j X_j)$ is the mean effect estimate for feature j . The contribution is the difference between the feature effect minus the average effect. If we sum all the feature contributions for one instance, the result is the following:

$$\begin{aligned}
\sum_{i=1}^P \phi_j(\hat{f}) &= \sum_{j=1}^P (\beta_j x_j - E(\beta_j X_j)) \\
&= \left(\beta_0 + \sum_{j=1}^p \beta_j x_j \right) - \left(\beta_0 + \sum_{j=1}^p E(\beta_j X_j) \right) \\
&= \hat{f}(x) - E(\hat{f}(x))
\end{aligned}$$

This is the predicted value for the data point 'X' minus the average predicted value. Feature contributions can be negative.

Computing Shapley value - The shapley value [65] of a feature value is its contribution, weighted and summed over all possible feature value combinations as shown below:

$$\phi_j(val) = \sum_{S \subseteq \{x_1, \dots, x_p\} \setminus \{x_j\}} \frac{|S|!(p - |S| - 1)!}{p!} (val(S \cup \{x_j\}) - val(S))$$

where 'S' is a subset of the features used in the model, 'x' is the vector of feature values of the instance to be explained and 'p' the number of features. $val_x(S)$ is the prediction for feature values in set 'S' that are marginalized over features that are not included in set S:

$$val_x(S) = \int \hat{f}(x_1, \dots, x_p) dP_{x \notin S} - E_x(\hat{f}(X))$$

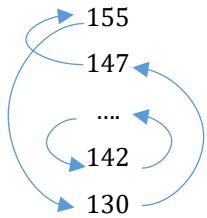
Prediction Explainer - Lundberg et. al. [64] showed how these features each contributing to push the model output from the base value (the average model output passed over the training dataset) to the model output. As shown in figure 2.30, features pushing the prediction higher are shown in 'red', those pushing the prediction lower are in 'blue'. Another way to visualize the same explanation is to use a force plot [66].

2.5.3.3 ELI5 and Permutation Importance

eli5 provides a way to compute feature importances for any black-box estimator by measuring how score decreases when a feature is not available; the method is also known as ‘*permutation importance*’ or ‘*Mean Decrease Accuracy (MDA)*’.

One of the most basic questions we might ask of a model is: What features have the biggest impact on predictions? To calculate this let’s consider the following table:

Height at age 20 (cm)	Height at age 10 (cm)	...	Socks owned at age 10
182	155	...	20
175	147	...	10
...
156	142	...	8
153	130	...	24



Let’s assume we want to predict a person’s height when they become 20 years old, using data that is available at age 10. The above example includes useful features (*height at age 10*), features with little predictive power (*socks owned*), as well as some other features we won’t focus on in this explanation. Now if we randomly shuffle a single column of the validation data, leaving the target and all other columns in place, how would that affect the accuracy of predictions in that now-shuffled data? Randomly re-ordering a single column should cause less accurate predictions, since the resulting data no longer corresponds to anything observed in the real world. Model accuracy especially suffers if we shuffle a column that the model relied on heavily for predictions. In this case, shuffling height at age 10 would cause terrible predictions. If we shuffled socks owned instead, the resulting predictions wouldn’t suffer nearly as much. We can use these predictions and the true target values to calculate how much the loss function suffered from shuffling. That performance deterioration measures the importance of the variable we just shuffled. Such permutation importance is calculated after a model has been fitted. We can then visualize those importance measures by using eli5 library.

2.5.4 Summary

Section 2.4 described different approaches of machine learning for classification, used by authors to achieve promising results and satisfy their research goals. All those are basically blackboxes which did not include any interpretation of how those results were produced. To solve this problem, we have explained eXplainable AI (XAI) into current section 2.5 as a technique of providing interpretation of those model operations and build trust in machine produced results. We have described their taxonomy and visualized how do those operate on underlying features and influence prediction results. For providing local interpretation, 'LIME (Local Interpretable Model-Agostic Explanations)' as introduced by Ribeiro et. al. [63] has been used in our research to explain how contributing features weights are calculated by perturbing data points and interpret local outcomes. On the otherhand, for providing global interpretations we have presented Lundberg et. al.'s [64] description on showing how average feature contributions known as '*shapley*' [65] values are calculated to express left/right pushes to the model from the base value prediction which are positive/negative impacts of features. We also have described another approach of computing feature importance for any blackbox estimator known as ELI5. It measures how do prediction scores get changed in absence/presence of features. This method is also known as 'permutation importance' or 'Mean Decrease Accuracy (MDA)'.



Analytic Provenance for Sensemaking



3 chapter



3.1 Chapter Overview

In intelligence analysis domain where solution discovery is often serendipitous, demands techniques to provide transparent evidences of top-down and bottom-up analytical processes of analysts while sifting through or transforming sourced data to provide plausible explanation of the fact. To complement the information entailed and to provide enhanced transparency of the operations, it demands logs of the individual processing activities within an automated processing system. We hypothesize that ***-capturing user's interactions with a visual interface can retrieve some aspects of the transparency of user's reasoning processes in intelligence analysis.*** Management and tracing of such security sensitive analytical information flow originated from tightly coupled visualizations into large visual analytic systems for intelligence analysis that triggers huge amount of analytical information on a single click, involves design and development challenges. The research in this chapter, contributes to solutions of these issues by considering following research questions:

RQ1: How to develop a system that tackles large flow of heterogeneous analytical data and supports W3C PROV-AQ: Provenance Access and Query standard factors i.e., Recording - represent, denote; Querying - identify, pingback; Accessibility - locate, retrieve into a multi-modular environment?

- RQ1 outlines the requirements, development challenges of front-end techniques and back-end modelling for generically capturing different complex visual analytical states, automatically processing and storing as well as recalling those as per query to maintain traceability. We have proposed a data flow model for tackling large, modular, heterogeneous platform's clickstream with internal system call architecture, visualization techniques to represent and query back those captured analytic data. We also have proposed an analytic task model to operate on this system and supporting real intelligence analyst's strategies of sensemaking.

RQ2: How to utilize captured analytic provenance data for sensemaking?

- RQ2 seeks to find out the techniques of utilizing captured analytic data to support transparent sensemaking, mitigation of uncertainties in visualizations and build trust on visual analytic systems. We have conducted a case study in this chapter to show impacts of such uncertainties on transparency principle and developed an analytic visual judgemental system by utilizing captured series of event sequences and their interrelationships to support human perception, cognition and understand the entire reasoning path which is obvious for intelligence analysis.

3.2 Introduction

Now-a-days the large and complex event-driven systems around us are computationally intense where data flows from one process to another as it is transformed, filtered, fused, and used in complex models in which computations are triggered in response to events. Capturing provenance and representing to support judgmental process specially in intelligence analysis by using such computation systems with hundreds of interconnected services that creates huge volume of data at a single run is a matter of obvious challenge. It has greater impact on understanding the process by which the decision has been made. Provenance is a broad topic that has many meanings in different contexts. According to W3C (World Wide Web Consortium) incubator group report, provenance normally relates to 'source', 'process', 'accountability', 'causality' or 'identity' of series of events. It implies provenance recording system should include – the collection, alteration, consultation, disclosure including transfers, combination or and erasure of personal data, whereby the logs of consultation and disclosure will make it possible to establish the reasons for, date and time of such operations and, as far as possible, the identification of the person who consulted or disclosed data, and the identity of the recipients of such data. Xu et al. [1] explain, it can be valuable to maintain such historical records detailing the evolution of data, proceeding of process, and changes to reasoning which take place during sensemaking. Sensemaking is a process of generating meaning from information. It involves activities such as information foraging and hypothesis generation.

Wong et al. [69] propose a three-layer provenance model which describes the relationship between the provenance and the intelligence process. During this process information is located, collected, analyzed, transformed, and communicated. Wong et al. focus on the traceability of this workflow '*as an essential part of individual and collaborative reflective dialogues with both evolving and completed analyses*'. In order to support this traceability, Wong et al. offer a conceptual framework which shows the analyst's problem space as divided into three complementary work areas. Together they form what Wong et al. refer to as the '*Intelligence Analysis Reasoning Workspace*'. The three levels are:

- i. **The data level.** This level includes raw data which is derived from different external intelligence sources such as documents, financial records, signals intelligence reports or photographs.
- ii. **The analysis products level.** This level includes the results of data manipulations. At this level the analyst creates abstract representations to draw out key facts, without going beyond sorting and structuring the available information.
- iii. **The reasoning products level.** As Wong et al. [69] explain, this level integrates the findings and high-level reasoning artifacts such as interpretations, assumptions and hypotheses.

As Wong et al. [69] explain, provenance operates at these three levels. At the data level provenance tracks data and information resources. We refer to provenance at this level as **data provenance**. At the analysis products level provenance tracks the process of '*data manipulations and analytic moves*' (Wong et al., [69]). This **process provenance** reveals, for example, how the analyst integrates and summarizes data. At the reasoning products level provenance follows the analyst's lines of reasoning and argument. This **reasoning provenance** thus reveals how knowledge is used to interpret, create hypotheses and draw conclusions. The latter two categories - process provenance and reasoning provenance are often referred to in the literature as **analytic provenance**.

The research of analytic provenance can be examined in five interrelated stages as proposed by North et al. [14]. They are - *perceive, capture, encode, recover* and *reuse*. These are mainly non-cognitive aspects of analytic provenance information. But for a successful analysis the importance of analyst's cognitive stages must be addressed as they are complement to each other. To make these visible we have presented a typical analytic task model (below) by combining cognitive and non-cognitive aspects. During an analytic task it usually starts from the low level interactions on data heading towards high level reasoning tasks in combination with human cognition. At

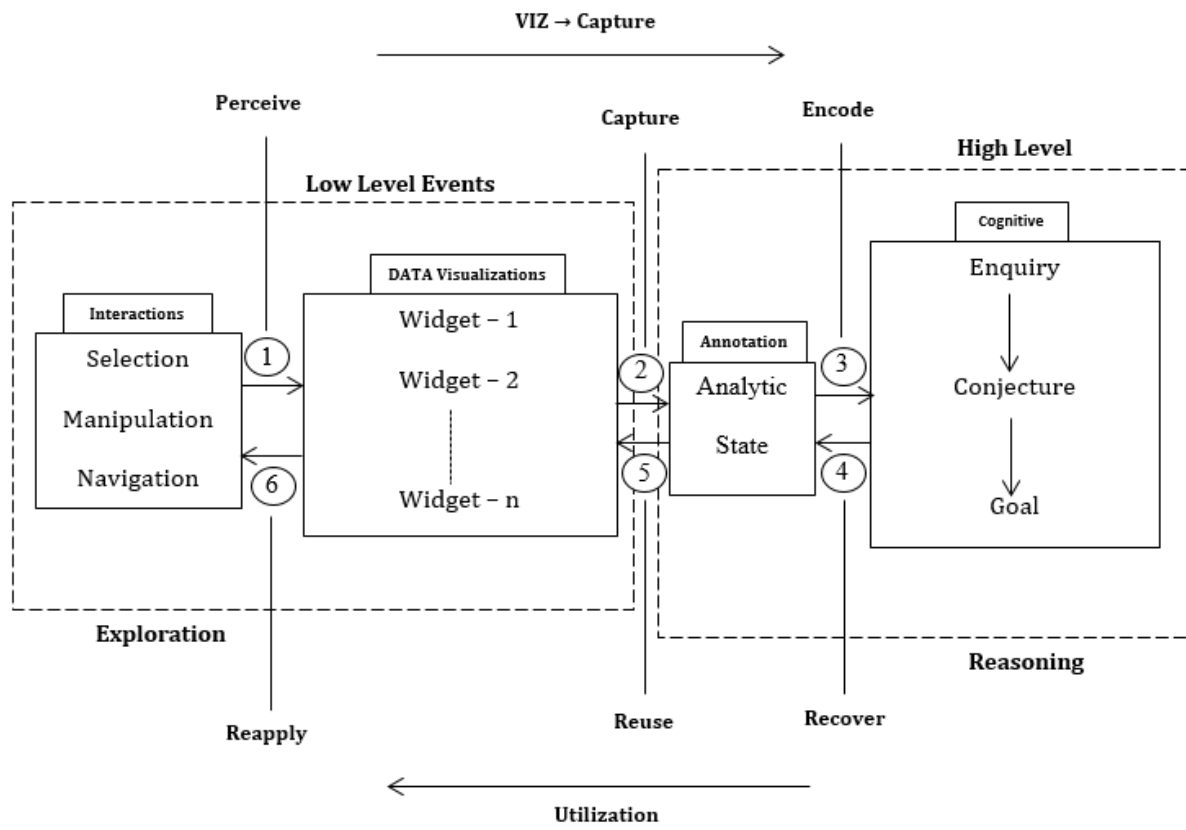


Figure 3.1: Typical analytic task model.

the later stage (Chapter 4) of this research, we have addressed the importance of capturing these cognitive aspects of analytical processes and described a computational approach to detect and utilize those.

However, suitable tools and techniques need to be in place to capture and interpret those data. Throughout this research work we have identified the problems of existing approaches, attempted to overcome by proposing our techniques and evaluate those. The current chapter of this research, only includes the challenges of developing appropriate tools and techniques for capturing analytic tasks in structured ways. To progress with this, we arranged several focus group studies to gather requirements of the analytic provenance system, identify cognitive constructs of intelligence analysts during their analytical process.

3.3 The Approach

Our research, design and development described into next few sections are based on the architecture of a prototype Analyst's User Interface (AUI) of a funded integrated project through European Commission's 7TH Framework Programme named as below (Visual Analytics for Sensemaking in Criminal Intelligence). The results of this research work aims to contribute to the project for sensemaking in criminal intelligence analysis.

The Analyst's User Interface (AUI) as shown in below is a prototype visual analytics system developed for criminal intelligence analysis based on the '*Thinking Landscape*' design concept as proposed by Wong et. al. [73] . The '*Thinking Landscape*' is a UI design concept that embodies the idea of externalizing the thinking and reasoning processes of the analyst in a way that gives abstract concepts a tangible expression within the computer user interface. The AUI architecture harmonizes interaction across different applications, based on conventional interaction techniques that fluidly support direction manipulation with information, and complies with Human Issues Framework. It's visualization and interaction methods supports representing and working with data, tracing of how decisions and conclusions were arrived, and how to make these conclusion pathways visible to the users and co-workers. We progressed developing a widget named as 'PROV' for provenance visualization as shown in below, to *capture, visualize and utilize analytical provenance information*. In the previous section we presented a literature review on how provenance is tracked and represented, and implemented in the few systems that operate in the different data, analysis and hypothesis spaces. The literature review also describes what is important to the analyst about preserving and tracking the provenance of the data, of the analytical process, and the ways that the analytical provenance data can be used in reviewing, re-playing, or assess the considerations made with what data or evidence.

*VALCRI - <http://valcri.org/>

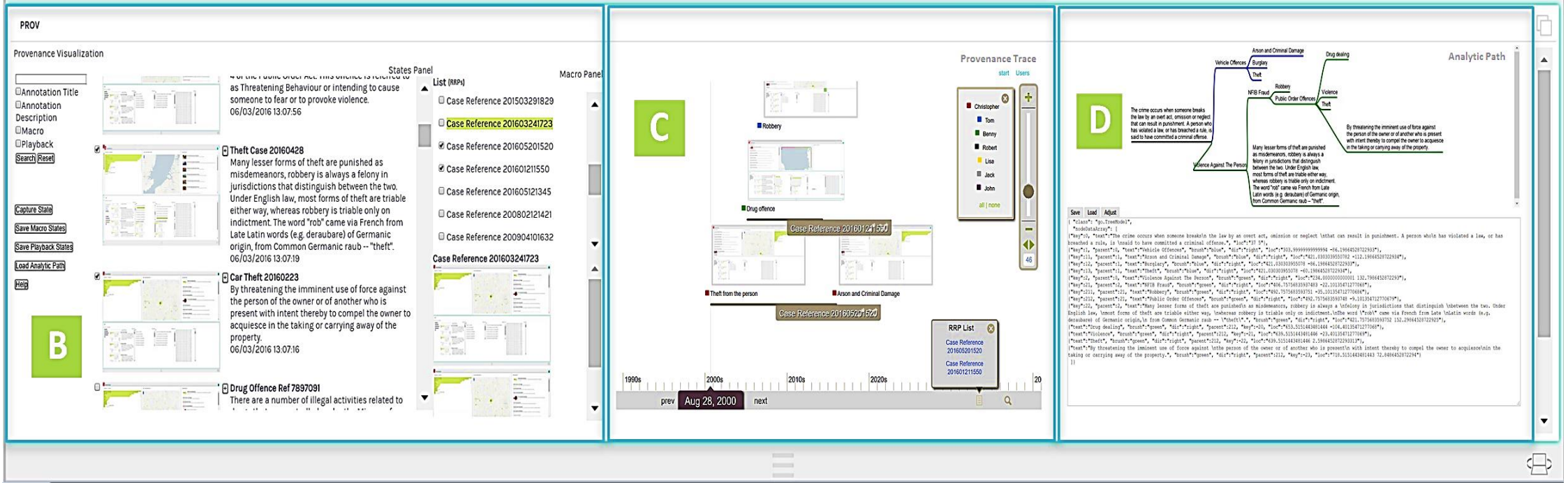
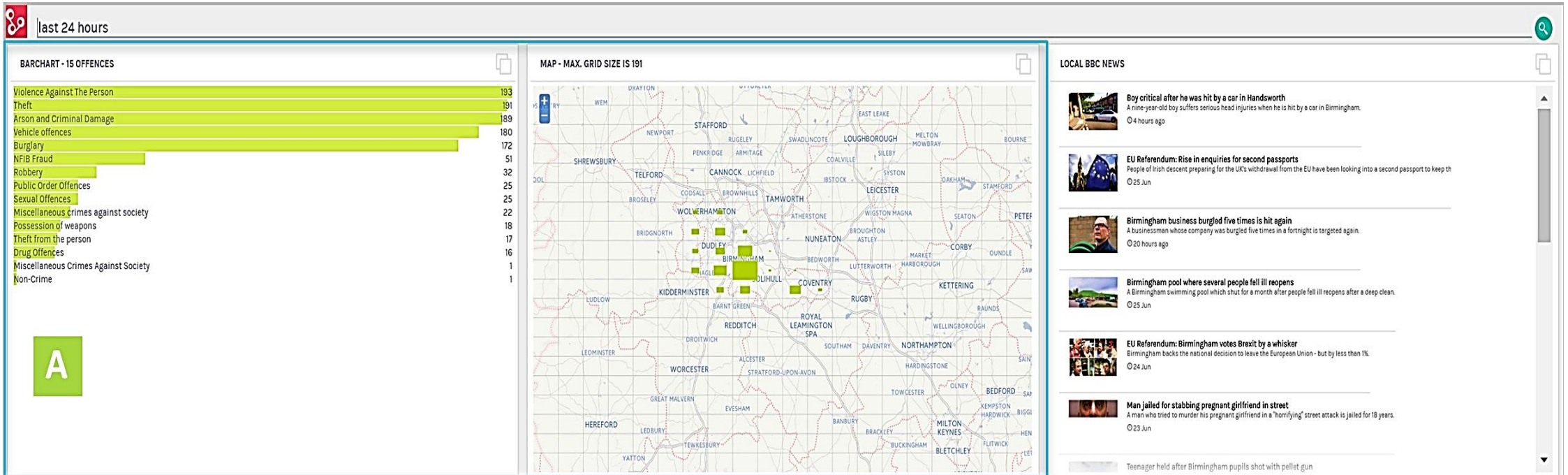


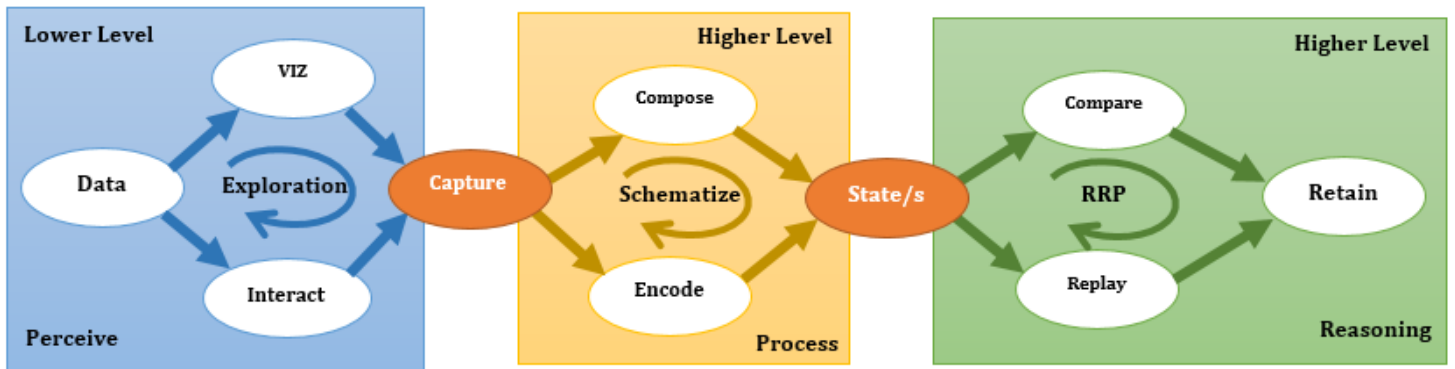
Figure 3.2: Four linked views for analytical provenance capture and representation system – **A.** Interactive partial views of AUI for total number of offences during last 24 hours on map and bar chart views, **B.** Captured states, macro playback panels, **C.** Provenance Trace panel by temporal & colour coded users (analysts) filtering, keyword search and multiple timelines selection for macro states, **D.** Analytic Path showing annotations set by analysts with captured states & their relationships based on interactions with colour coded users (analysts) information.

3.3.1 Proposed Analytic Task Model

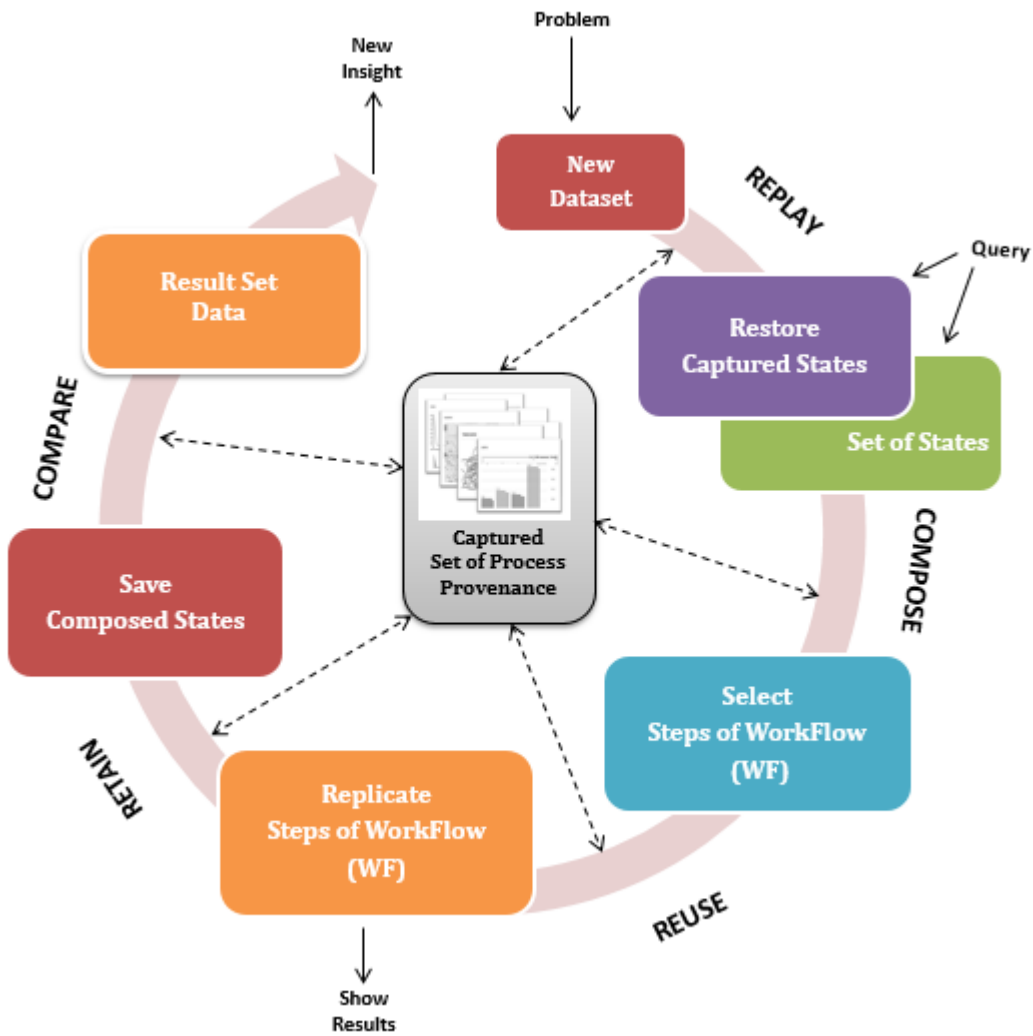
By considering the research issues i.e., how will provenance be captured, saved, tracked and implement those into AUI system that operates on different data, analysis, hypothesis spaces; it is important understand how analysts think and what they do to achieve their goals. We organized series of focus group discussion with crime analysts to understand their requirements from the perspective of system support to carry out a fruitful analysis. We have accumulated those concepts and presented those as correlating analytic activities into a proposed '*Analytic Task Model*' as shown in below. The proposed model is consisting of following analytical steps:

- **Perceive:** At this stage the analysts try to gain preliminary understanding of presented data as visualization or some other format. This is an exploration stage when analysts perform low level events [10] and capture those to use for the next level of analysis.
- **Process:** This is the stage of higher level sub-task [10] when analysts try to organize their findings by composing analytical states in more meaningful way or encode those and use for next level of analysis. These sub-tasks are also known as schematization.
- **RRP:** The final reasoning state at higher level allows analysts to apply their approaches on a different scenario for comparative analysis. It mainly follows *compose* → *replicate* → *retain* steps [below] to help generating a new insight. This is the top level sensemaking undertaking step which can be saved for future use and replayed. We have named this technique as RRP (*Repetitive Replicating Playback*).

The system design of 'PROV' system (above) considers above three vital steps of analytical processes usually applied by criminal intelligence analysts. We have only considered non-cognitive analytic activities into our proposed task model and also includes interrelated stages to be examined during analytic provenance research as argued by North et. al. [14]. We conducted an evaluation study as well with the end-users of the project to gather some feedback for further research and development.



(i)



(ii)

Figure 3.3: Proposed – (i) Analytic Task Model , (ii) Repetitive Replicating Playback (RRP) system.

3.3.2 Perceiving Data

How does the analyst perceive visualization of data?

It is important to understand how the analyst “Perceives” the data to start reasoning process and correlate interactions with visualization. It is also important to understand how the data is presented to the analyst. Crime analysis encompasses a range of data analysis activities. Many tasks, however, require analysts to study large collections of crime reports in order to identify aberrant or exceptional patterns of activity, identify new and emerging crime series, or sometimes suggest crime suspects that may be linked to a crime phenomenon. There are seldom concrete, single or certain approaches or techniques that can be taken at each of these stages and often solutions are found through serendipity instead of rules. These incorporate uncertainty into visual representation of data that may lead to erroneous insights. Our literature review has found that denoted this uncertainty as *Uncertainty of Visualization* [74], which considers how much inaccuracy occurs through the pipeline of data processing. Such uncertainty becomes more problematic in the crime solving domain as it may have negative consequences for individuals. Current state of the art *Uncertainty of Visualization* differs with the concept *Visualization of Uncertainty* - which considers how we depict uncertainty specified with the data [74] and on which lot of research work have been carried out to find techniques and develop tools. Current state of the art demands more work to find out causes and effects of uncertainty of visualization in criminal intelligence analysis.

3.3.2.1 Uncertainty in Visualization

During analysis numerous techniques are followed for allowing analysts to make observations and research claims with varying levels of authority. Failing to acknowledge uncertainties around such analysis task, dataset and analysis technique may lead to a cavalier and superficial data analysis: making faulty claims with confidence that may lead to poor decision making.

Criminal intelligence analysts commonly work with incomplete, ambiguous and often contradictory data. Such incomplete collection of data may cause flaws in logic, vague

or misapplied similarities to unrelated events, failures of imaginations to find a viable solution. A good analyst highlights such information gaps, the strengths/weaknesses of current dataset and pinpoints the way forward. The emergent growth of data capture technologies has made it possible to analyze those with the exponential growth of human activities now-a-days. The big problems of handling such data are – various sources, poorly structured, unreliable etc, which have made the analysis process more complex.

3.3.2.2 Crime Analysis Under Uncertainty

After being collected, information is processed or arranged in a way that enables the analytic effort. Processing can involve any number of activities including data collation, data mining, entity extraction, translation etc. Typically, however, processing involves the structuring of information so as to enable the search for relationships and meaning within one's data. The techniques followed for these processing activities may be new to analysts and introduce concerns such as - unawareness of the errors or uncertainties that occur as a result of the data transformations required by the algorithms, building trust in outputs and their analytical techniques etc. During this phase, an analyst needs to deal with the following occurred uncertainties during dataset analysis:

- **Visualization Biases** – analysts see patterns into data plots (e.g. on a scatterplot) when the data is in fact a random distribution. Two things are occurring here, (i) the user is unaware that a random sample does not generate an even distribution of points on a simple scatterplot or in coin tossing, a fairly balanced sequence of heads and tails; and (ii) humans are predisposed to finding patterns, even very insignificant ones such as three points in a row amongst hundreds of scattered points. This cognitive bias is one that has already been identified, and is in fact a visualization bias [75] rather than analytic.
- **Trust Building** – obviously, the chance of human error is highest when uncertainty is present in the system and the analyst is not aware of it, or mistakenly believes that there are no uncertainties. Uncertainty in visual

analytics originates and propagates from the system – that is the datasets, data model and visualizations and is then passed to the analyst as findings and insights are discovered, resulting in knowledge generation. Uncertainties affect human trust building processes using the knowledge generation model [76] for visual analytics.

- **Personal uncertainty** - processing obliges the analyst to test the assumptions and hypotheses they have hitherto been operating with. The analyst has to ensure that the way in which information is organized enables a sober and unbiased evaluation of its contents. Errors introduced here can seriously affect any subsequent analysis.
- **Task uncertainty** - the quality of outputs obtained during processing influences analysis. Accordingly, the outputs of processing have to be adjusted to a particular case, circumstance, or analytic need. This requires a solid understanding of the task and processes involved.
- **Outcome uncertainty** - the outputs of processing are inputs to analysis. Consequently, processing should be oriented to helping the analytic process. Similarly, if the steps to be taken during analysis are not clear, processing will be muddled.
- **Issue uncertainty** - as with other steps in the cycle, understanding the issue at hand enables processing by providing the analyst with one or more concept models that can be applied to the structuring of information. Such models can be tacit or explicit in nature, technology driven or merely pen-and-paper representations to help the analyst filter and organize the data collected.
- **Course-of-action uncertainty** - processing is greatly enabled by clarifying the analytical steps that will follow. Thus, knowing what analytic or data visualization tools will be employed can help the analyst orient the processing effort accordingly.
- **Decision uncertainty** - processing encompasses the broadest range of possible activities. Given the resource constraints, analysts are often required to weigh the options available and determine which are most likely to generate new insights or ideas.

- **Informational uncertainty** - processing offers another opportunity to critically assess the information collected in terms of its reliability, accuracy and relevance, as well as to verify the quality of sources from which the data originated.

Techniques that provide accurate estimates of uncertainties are therefore vital. By understanding the uncertainties, analysts better trust their acquired knowledge and can report findings with greater rigour and authority.

3.3.2.3 Case Study on a Criminal Situation

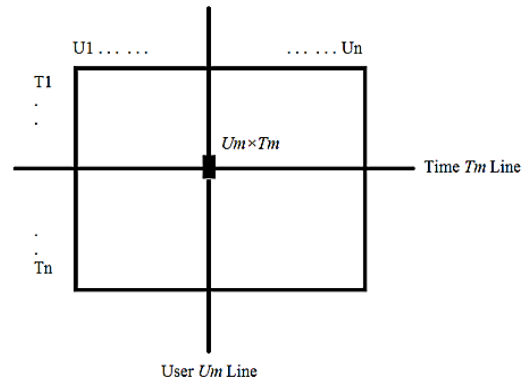
Not all techniques of visualization offer the required information. Different kinds of visualizations offer different features. With the right visualization technique of large data, it is possible to effectively support human perception, cognition, reasoning, with database operations and computational methods which are crucial in the case of large amounts of data. In addition it is important to realize that, even if there is certainty about the data, errors can occur in the process of turning the data into a picture. We conducted a case study on [‡]*below* dataset to demonstrate how uncertainties may occur due to lack of appropriate technique of visualization. We considered problems of Mini-Challenge 1 and used it's available park visitors' 14.5M movements and 4.1M communications datasets for this case study. As part of initial processing we filtered out park visitors' check-ins dataset and visualized to have an understanding of the situation.

The Mini-Challenge 1 describes an incident of vandalism at Dino World (an amusement park) during a weekend (Friday, Saturday, Sunday) of June 2014. Park officials and law enforcement figures are interested in understanding just what happened during that weekend to better prepare themselves for future events. They are also interested in understanding how people move and communicate in the park, as well as how patterns changes and evolve over time, and what can be understood about motivations for changing patterns.

[‡] <http://vacommunity.org/VAST+Challenge+2015>



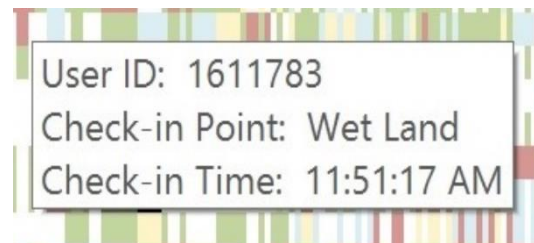
(i)



(ii)



(iii)



(iv)

Figure 3.4: (i) The park map, (ii) Visualization paradigm, (iii) Temporal view, (iv) Spatial view.

Visualization Paradigm

We developed two kinds of visualizations to show park visitors' check-ins over time – *Temporal* and *Spatial*. As shown in Figure 3.4(iii) we used blue color shades for temporal visualization to represent check-in frequencies of park visitors over time. We also used color codes of park map to visualize check-ins of park visitors at different areas over time as shown in Figure 3.4(iv). All user check-ins at different areas have been visualized temporally into a user vs time matrix represented as $U \times T$, where U =user and T =time as shown in Figure 3.4(ii).

map clustering Check-ins path view network

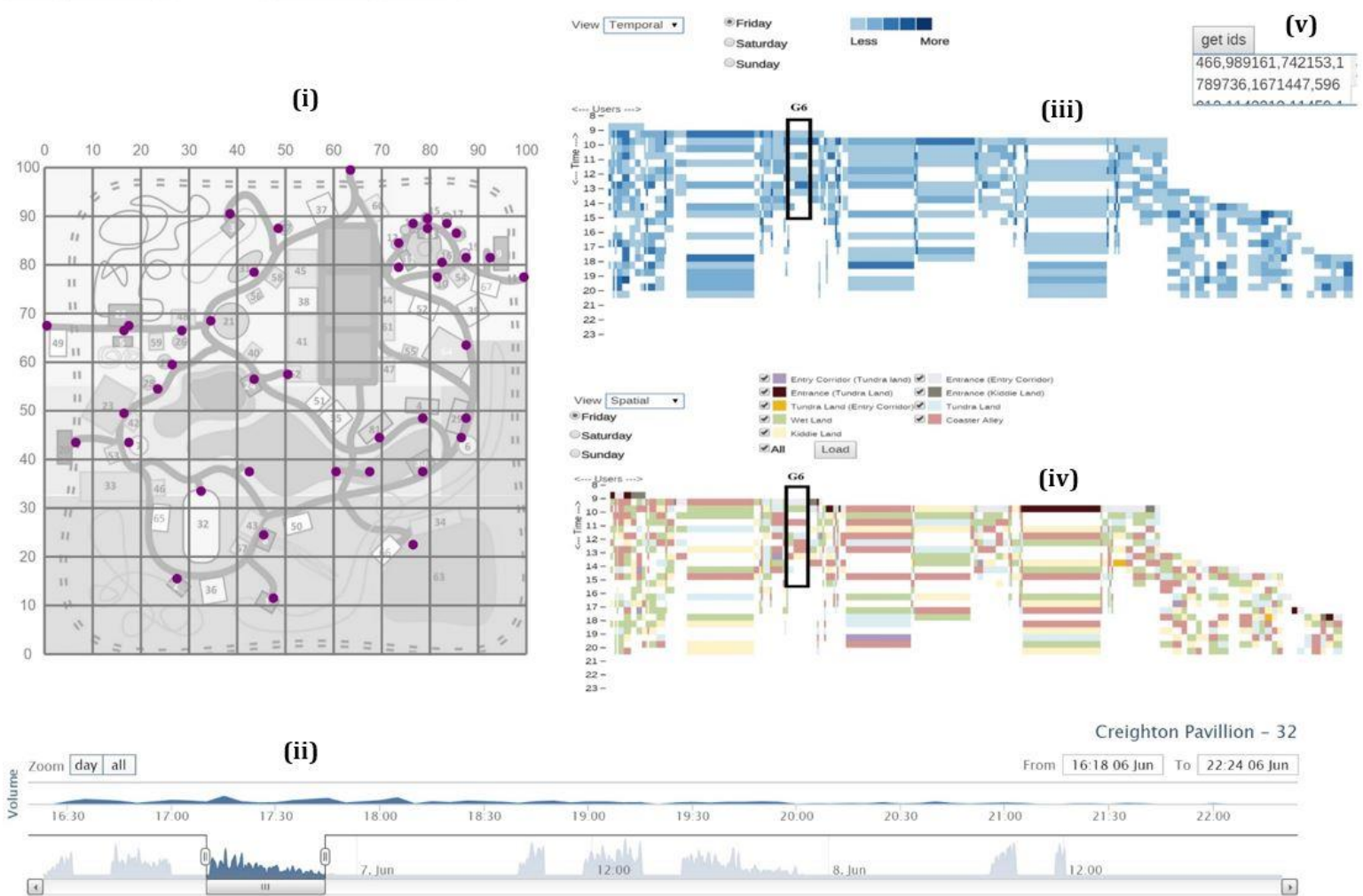


Figure 3.5: Four visualizations of the VAST dataset showing how the park visitors check-in. **(i)** Purple coloured dots represent different check-in points, **(ii)** Temporal high-chart view of overall check-ins, **(iii)** Frequencies of temporal check-ins visualizations with colour shades [less->more], **(iv)** Check-ins using spatial colours of different park areas as shown on park-map, **(v)** Filtered view of group check-in ids.

Data Visualization and Exploration

We visualized all visitor's check-ins data by using above paradigm. These visualizations give an idea of the structure of data. Let's say it uncovers things like – whether there is any cluster into the data, whether the variables are correlated with each other, similarities among them or if there are any outliers. From the above 'Temporal' and 'Spatial' views few groups like **G6** have been identified based on their criteria of being the same group. The criteria includes same kinds of activities (i.e, check-in frequencies, movement patterns, check-outs etc.) through-out the whole day.

These kinds of smaller/bigger group activities can be found quite a lot of time through-out the whole visualization.

3.3.2.4 Findings

Findings on Uncertainties

“Uncertainty is the dissimilarity between a given representation of reality and the known or unknown reality, where the unknown reality simply means you do not know what the reality actually is that you are representing” – the definition proposed by Plewe [77] has a similarity of fact into current visualizations. This is a true scenario of crime related intelligence analysis. Due to incomplete representation of data there might be flaws in logic, vague or misapplied similarities to unrelated events resulting failures of imaginations to find a viable solution. We call this as *‘Determinacy Problem’* which has two types:

- i. **Spatial Determinacy** – exact location of the event happening.
- ii. **Temporal Determinacy** – actual time of the event happening.

These determinacy problems lead to the uncertainty of space and time which means *‘Don’t know about when and where’*. As described in Mini Challenge 1 –

A news article was published in the newspaper on June 10, 2014 with the title ‘Mayhem at DinoFun World’ - by Mako Harrison, staff reporter by saying that – ‘The crime forced partial closure of DinoFun World and local police were on the scene shortly after the vandalism was discovered by park visitors. Security guards are being questioned to eliminate the possibility of an inside job. Creighton Pavilion-32 was closed and locked up tight before each show as stated by park Chief of Security Barney Wojciehowicz’.

This information gives a start of analyzing the crowd for initial understanding of the fact. Our spatio-temporal visualizations of above show groups of people who checked-in together and got split after a while. Our visualizations reveal more patterns of such activities by filtering out the data of **Coaster Alley** where **Creighton Pavilion-32** is situated as shown in acima to make an initial plot of the situation and make a judgement on the published news. Our visualization approach

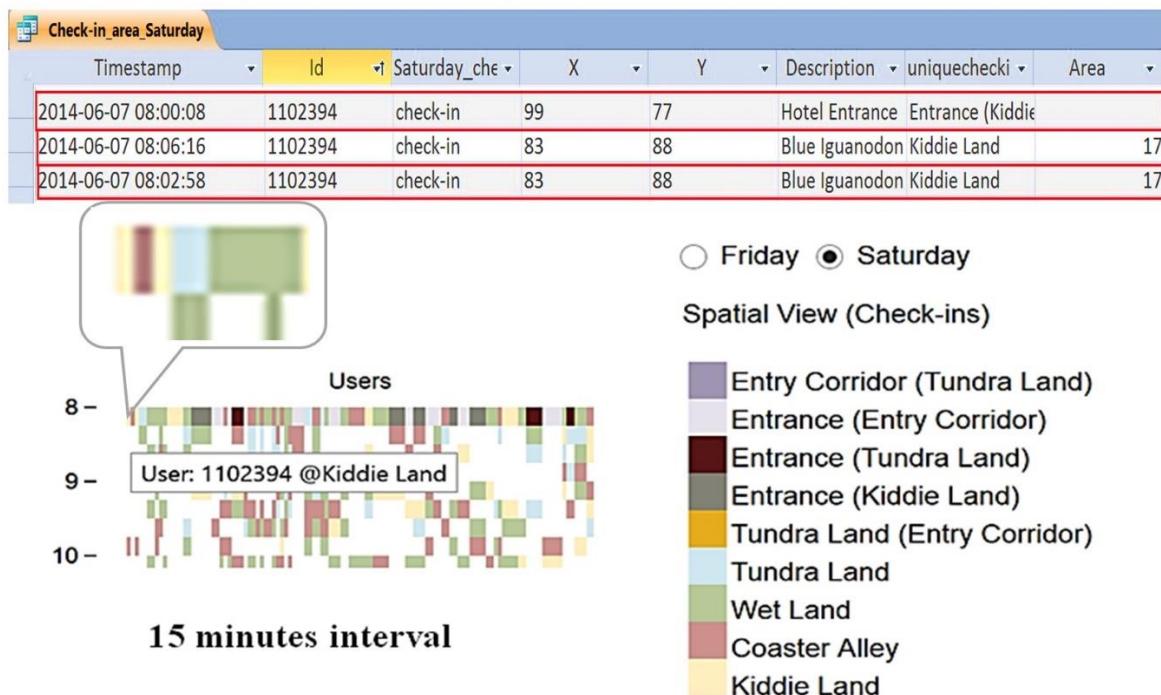


Figure 3.6: Spatial Determinacy Problem.

considered data of every 15 minutes' interval as a criterion of sampling and disambiguating dataset prior to visualization. We found that it is raising the 'Issue Uncertainty' for structuring, filtering and organizing dataset resulting to 'Decision Uncertainty'. As shown in Figure 3.6, three check-in events have been recorded for user id 1102394 within 15 minutes interval into movements table whereas current visualization only visualizes the most recent check-in point. We denote this as 'Spatial Determinacy Problem'. Such sampling strategy has raised another issue of missing particular temporal data of an event. We denote this as a 'Temporal Determinacy Problem'. As shown in below, no check-in event has been plotted by the high-chart whereas a check-in record has been found in the movements table as displayed on spatial selection panel. Both of these determinacy problems raise concerns about 'Personal Uncertainty' of the analyst to test the assumptions or hypotheses s/he has been operating with.

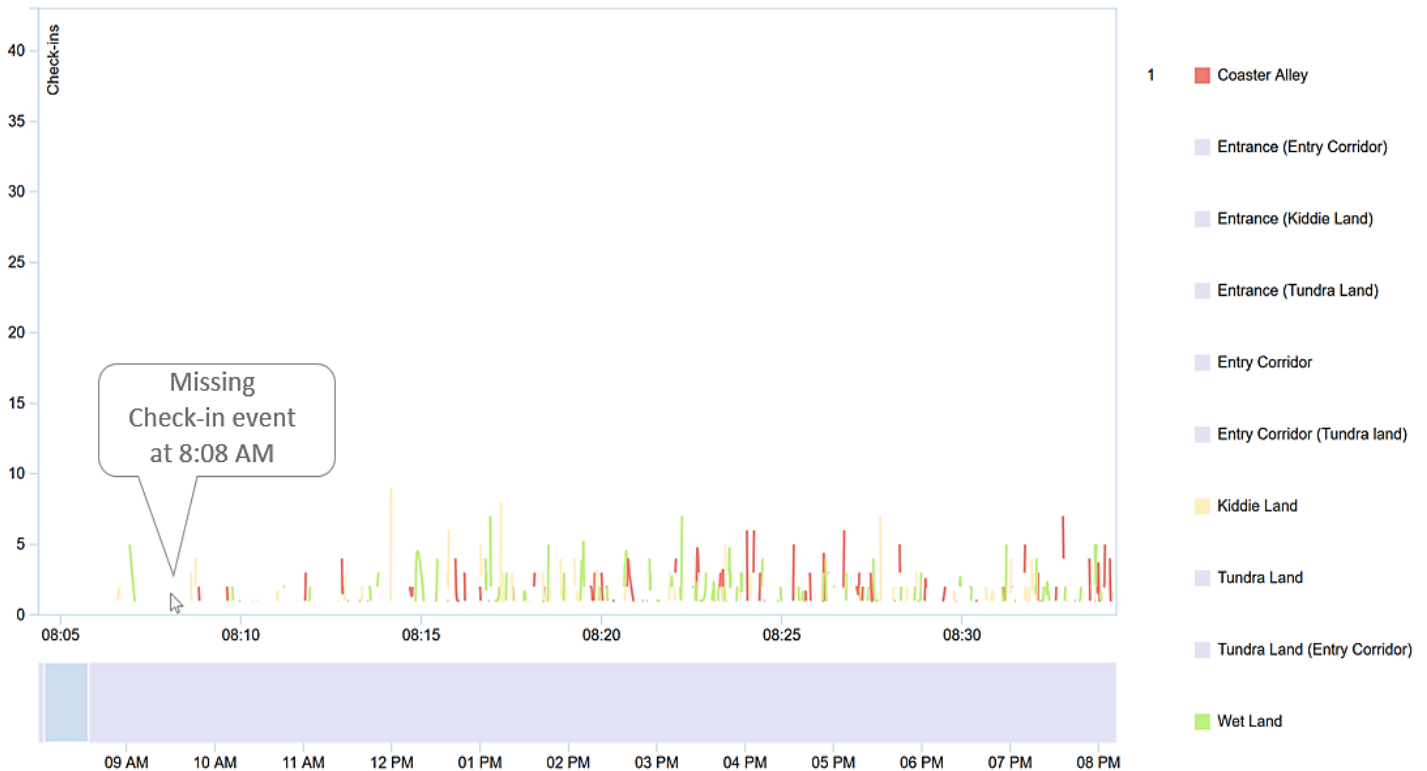


Figure 3.7: Temporal Determinacy Problem.

Findings on Visualization Biases

The VAST2015 dataset visualization as shown in above shows different patterns of movements, although they are not developed by using any statistical distribution theories i.e, frequency distribution for temporal view [above] and spatial distribution for spatial view [above]. So, these may create clustering illusions to analysts leading to cognitive biases while trying to find out patterns from these plotted data by using $U \times T$ visualization paradigm. Smaller/bigger group activities like group **G6** can be found quite a lot of time through-out the whole visualization. This is a visualization bias where a user is typically unaware of the data values, but is more aware of the position of graphic points from the display.

Human Trust-Building under Visualization Biases and Uncertainties

By considering the unawareness issues of analysts on errors and limitations of visualizations, we have found that human trust building may get affected due to visualization uncertainties such as spatio-temporal determinacy problems and has negative impacts on analytic processes due to visualization biases. Muir's [78] description of trust relations between human and machine includes the concept of trust calibration that is influenced by such factors. Analysts have to calibrate their trust not only towards the system but also towards the system outputs, or the findings and insights that have been gained by using the system. The trust in these parts may increase or decrease based on the understanding and awareness of errors or uncertainties that are hidden behind the final system outputs.

Provenance for Handling Uncertainties, Biases and Awareness

By making intelligible to analysts the datasets, data configuration and modelling on which their findings are based, provenance techniques can be leveraged to help mitigate uncertainty and distrust between human and machine. On the one hand, data provenance, that is information on the types of data that were used as well as details on quality of collected data, enables analysts to track, record and communicate processes in order to raise awareness of uncertainties. On the other hand, analytic provenance, that is the analytic context under which insights were made, enables analysts to review the analysis process or to infer trust levels based on his/her behaviour or interaction with the system. In the following we will describe how these methods can enhance analysis processes that include uncertainties.

- **Data Provenance** - Uncertainty Quantifications for each of the parts within the visual analytics pipeline are the foundation for handling and communicating uncertainties. These uncertainty measures can be Propagated and Aggregated in order to provide a combined measure that can be related to the system outputs. Furthermore, capturing the process of data transformations and uncertainty information enables the Visualization of Uncertainties. Finally, provenance techniques enable the exploration of uncertainties and an

understanding of how specific data items or dimensions are impacted by different uncertainties.

- **Analytic Provenance** - Analytic provenance methods for capturing, tracking, managing or organizing evidence found using a system should be enriched with trust cues about the included uncertainties in order to support *uncertainty aware sensemaking*. Further, trailing human interaction and behaviour might help to infer an analyst's trust level (e.g., which items are of interest or trusted). This information could be leveraged to provide hints about potential problems and biases. Finally, analytic provenance enables the analyst to track and review their analysis as a post-analysis activity in order to detect, assess and mitigate biases.

To find out the answers of 5WH (*Who, When, Where, What, How*) questions in criminal intelligence by using dynamically changing, incomplete, inconsistent data and visual analytic techniques; rises challenges on trustworthiness of outcome. As analysts are unaware of inherent uncertainties, so they may waste their time by following wrong and uncertain leads. We found from our case study that unawareness of errors and limitations into visualization systems introduce determinacy problems and creates issue uncertainty. Such personal uncertainties of analysts may hinder their decision making process. To make analysts aware of uncertainties at every stage of data analysis – background information on how the data were collected or processed (data provenance) and facilities to record, organize, revisit their processes (analytic provenance) will aid analysts in this regard. It is argued that, where uncertainties are fully understood and accounted for in a data analysis, there is greater trust in the acquired knowledge. This notion of trust is perhaps particularly important in crime analysis, where analysts must provide evidence with sufficient clarity and confidence for officials to use in strategic and operational decision-making.

Uncertainty in visualization is an inevitable issue for sensemaking in criminal intelligence. Analysts perceive the data as they go along with the system while finding out insights from crime related datasets. So, accuracy and precision of adopted visualization techniques have got a greater role in trustworthiness of the outcome. We have presented above a case study to introduce the concept of '*uncertainty in*

visualization' and its relevance along the way of data exploration and perceive those during crime analysis. Our findings show how uncertainties in visualization pipeline influence cognitive biases, human awareness and trust-building during crime analysis and how provenance can enhance analysis processes that include uncertainties.

3.3.3 Capturing Data

How can analytic provenance be captured?

To keep track of the data exploration process and insights, visual analytics systems need to offer to the analyst ways to track their historical operations. Such visualizations help to externalize knowledge that they may have about the challenge. This will reduce the cognitive overload imposed on the analyst by freeing essential mental resources and offering a new perspective on the “*Captured*” information.

Analytical provenance is the means for providing insight into data processing operation in question. So, for criminal intelligence analysis it is one of the best means to provide necessary support to explain in a clear way how decisions or choices were made, what they were based on, how steps in a selection process were made, provide information grounds to justify and answer claims of bias or discrimination, and show compliance. All these are enablers of fairness and lawfulness of the data processing activities from the legal framework. Transparency in criminal intelligence analysis is an important requirement for maintaining respective LEP (*Legal, Ethical, and Privacy*) guidelines. This is the property that all operations on data including legal, technical and organizational setting and the correlating decisions based on the results can be understood and reconstructed at any time. So, Transparency can be regarded as the underlying foundation of the analytical provenance. As well as analytical activities performed by analysts should be recorded for supporting ‘*Accountability*’ for a particular action of the analysis process. Analytical provenance data has got greater influence in this regard.

Capturing analytical provenance has also got a significant role in criminal intelligence analysis, because the legal directive foresees an obligation to provide competent legal

authorities with information about the processing operation upon request. Competent authorities are any public authority or any other entrusted body by national law to exercise public authority and public powers for the purposes of the prevention, investigation, detection or prosecution of criminal offences or the execution of criminal penalties, including the safeguarding against and the prevention of threats to public security. Analytical provenance data can help to validate and support any conclusion gathered from the visual analytics process.

3.3.3.1 Requirements Analysis

To have a better understanding of the requirements for analytical provenance in criminal intelligence analysis, we organized a focus group discussion with police analyst end users of the project above. Based on our initial understanding of capturing analytical provenance, we developed an analytical state capturing prototype and demonstrated to police analysts during the focus group. We adopted Walker et al.'s, [22] proposed technique of saving analytical states as bookmarks for implementing our prototype. The purpose of such prototype demonstration in the focus group was to gather requirements for a much larger system as well as to evaluate the prototype. The focus group involved three groups of police analysts and each group had two people.

We tested two techniques of capturing analytical states by using our developed prototype – (1) Capturing a URI, and (2) Capturing event properties to save and restore analytical states automatically. We also tested these techniques on two separate visualizations, using the †*below* dataset for Geo-Spatial Temporal (GST) crime analysis and ‡*above* dataset for Call Data Records (CDRs) analysis. This system automatically logs information about the user's interaction with system as well as saves corresponding state data into database and shows the preview of the analytical state at front-end along with meta information on tooltips by using which a captured state can be restored again. The event based approach out of these two techniques that we followed to develop our initial prototype, provided us better results for capturing analytical states even at a granular level.

System Requirements

† GST Analysis: Canadian Crimes by Cities during 1998-2012 (<http://open.canada.ca/>)

Based on the prototype development experience and realization from the focus group demonstration, we identified following system requirements for supporting criminal intelligence analysis.

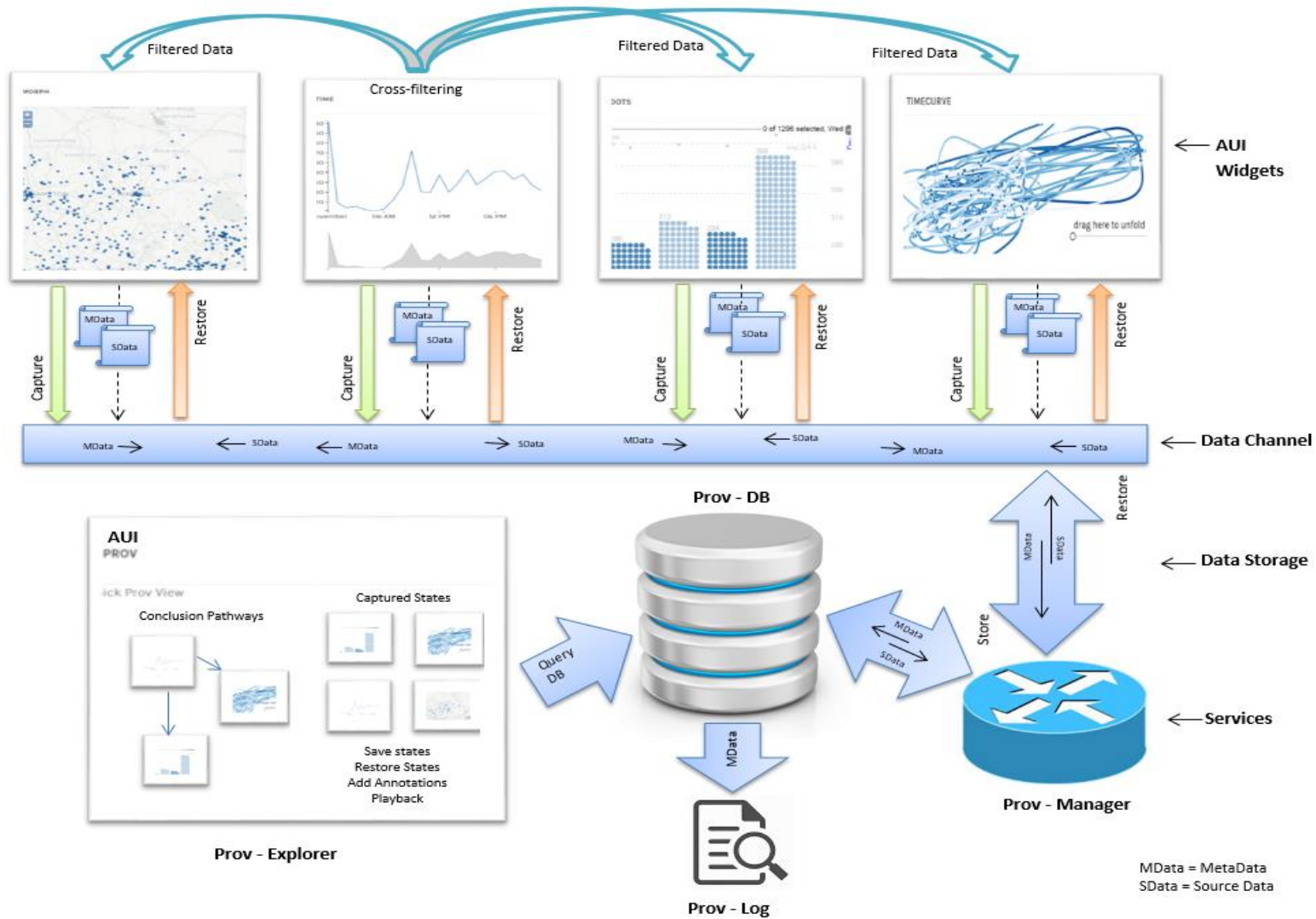
- **SysReq1:** different techniques should be supported for capturing and recording analytical provenance information.
- **SysReq2:** a standard mechanism should be referred to the discovery of an analytic provenance state object and a representation model should be used.
- **SysReq3:** different levels of granularity should be used in describing analytical provenance of complex state objects.
- **SysReq4:** analytical provenance data needs to be stored, logged, and versioned to allow capturing of states.
- **SysReq5:** the system needs to scale with large amounts of recorded analytical provenance data and lots of analyst end-users.
- **SysReq6:** analytical provenance information needs to be able to be easily queried.
- **SysReq7:** different levels of security are needed to provide access to analytical provenance data.

Police Analyst Requirements

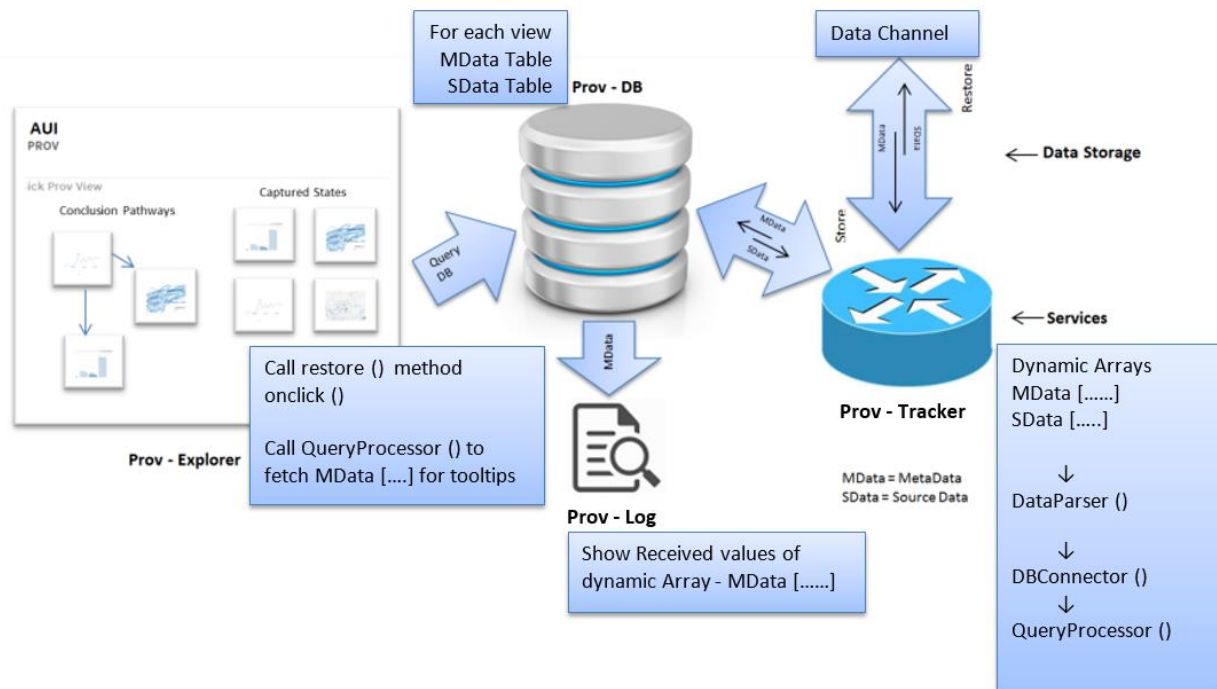
The police analysts currently record their thoughts in their diaries or spreadsheet manually and found this process cumbersome and ineffective. The police analysts found the demonstrated concept of analytical states capture and restore, and automatic state suggestion system could be effective for their work-flow. Based on the focus group we identified five potential end-users for an analytical provenance capturing system to support criminal intelligence. These include police analysts, analyst trainers, re-searchers, managers, and auditors. We now outline the identified requirements of the five-end users based on the focus group.

- **AnaReq1:** analysts need to see different representation techniques for visualizing analytical provenance data.

- **AnaReq2:** analysts need to be able to compare different analytical provenance information.
- **AnaReq3:** analysts need to validate whether captured analytical provenance information is of adequate quality for evidence.
- **AnaReq4:** the provenance information needs to show whether laws, rules and regulations have been correctly adhered to.
- **AnaReq5:** analysts must be able to step-back and step-forward through the states they have captured in the past to see what actions they performed in the system.
- **AnaReq6:** analysts need to be able to record a set of macro states to perform a collection of operations on different sets of data. We also call this *Repetitive Replicating Playback (RRP)* as shown in above.
- **AnaReq7:** analysts need to be able to annotate provenance information about different states.
- **AnaReq8:** analysts (based on role) must be able to turn off automatic logging of the provenance capture method.
- **AnaReq9:** trainers should be able to use the system to train new analysts.
- **AnaReq10:** auditors should be able to use the system to examine the kinds of activities analysts are performing and to generate reports.
- **AnaReq11:** managers need to be able to monitor what their police analyst colleagues are working on and see summaries of information.
- **AnaReq12:** researchers need to be able to use the system in conjunction with analysts to understand how to effectively perform criminal intelligence analysis.



(i)



(ii)

Figure 3.8: PROV – (i) System architecture, (ii) Internal system function calls.

3.3.3.2 System Design

The AUI system has been developed by following modular software design technique, consisting of many widgets and heterogeneous platforms as shown in above. In modular architecture, functionalities are separated into independent, interchangeable modules such that each contains all necessities for its own execution for distinct purposes. We progressed developing a widget named as 'PROV' and proposed a protocol for AUI to capture and visualize analytical provenance information. The protocol as shown in below supports such system to generically capture/restore analytical provenance states or workflows both automatically and manually by tackling heterogeneous data and development environments. The whole architecture has been divided into following functional sections:

- **AUI Widgets** – The widgets are analyst's visual interface for their scientific computations mostly built using different Javascript libraries on GOOGLE WEB TOOLKIT (GWT) framework by following MVP (*Model, View, Presenter*) design pattern. They have been integrated into the shell presenter of the AUI system that inherits widget attachment information from an *Abstract Presenter*, so that attachment of widgets can be tracked at any time. Few groups of widgets support interactive cross-filtering among themselves for the computation purpose as shown in above.
- **Data Channel** – The AUI system has been built by using Errai GWT-based framework for supporting uniform, asynchronous messaging services across the client and server end through a REMOTE PROCEDURE CALL (RPC) service. The data channel into above is the presenter of messages generated by the interactions during the analysis process. As shown into above and 3.8(ii), these messages are consisting of two types of data i.e, METADATA (MDATA) generated upon user's interactions and STATEDATA (SDATA) are accumulated states data of different widgets after interacting.
- **Provenance Service** – As shown in below, provenance service is the middle-tier server which co-ordinates with the tier-1 requests from clients and

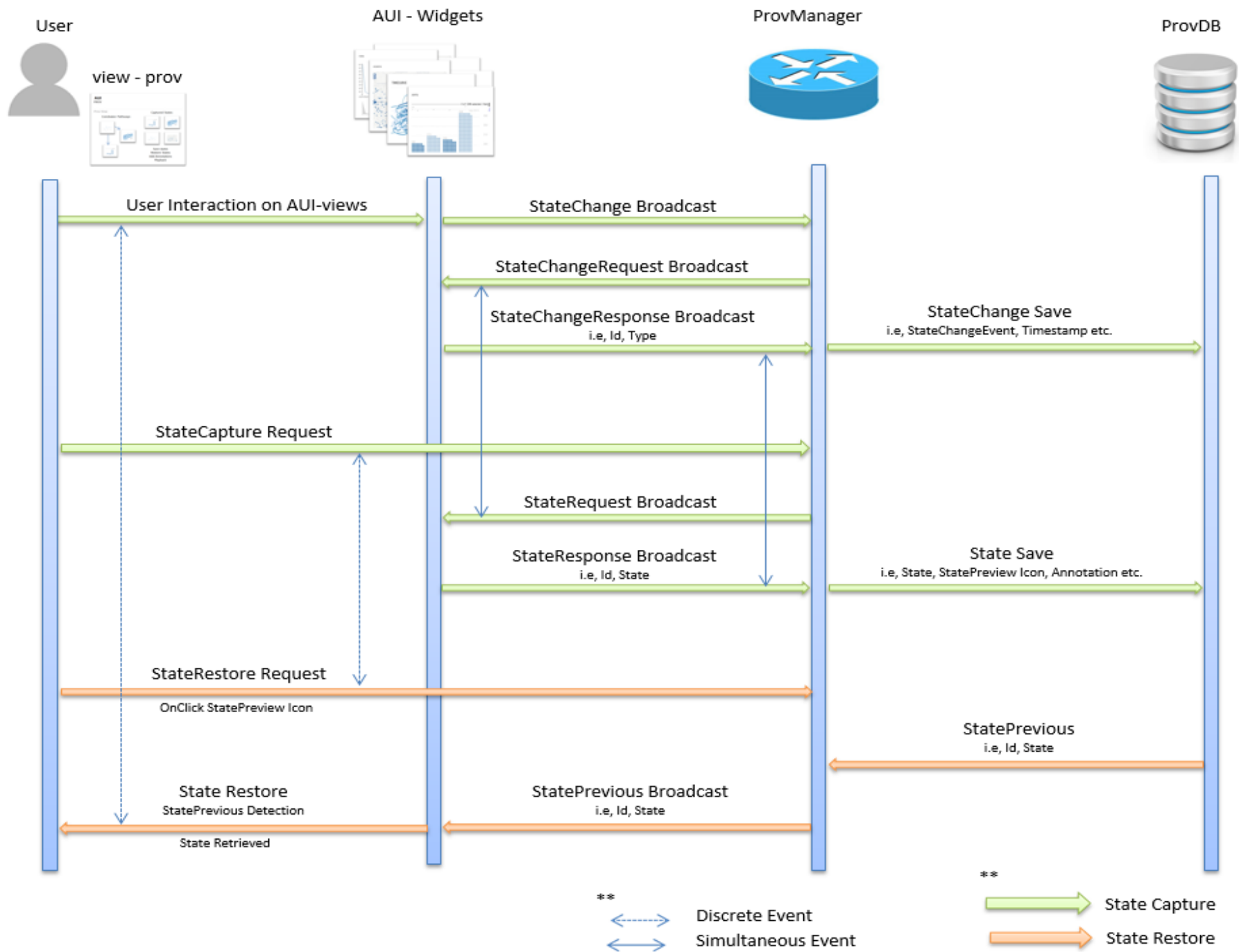


Figure 3.9: PROV - Simplified state event sequence diagram upon interactions on visualizations into Analyst's user Interface (AUI).

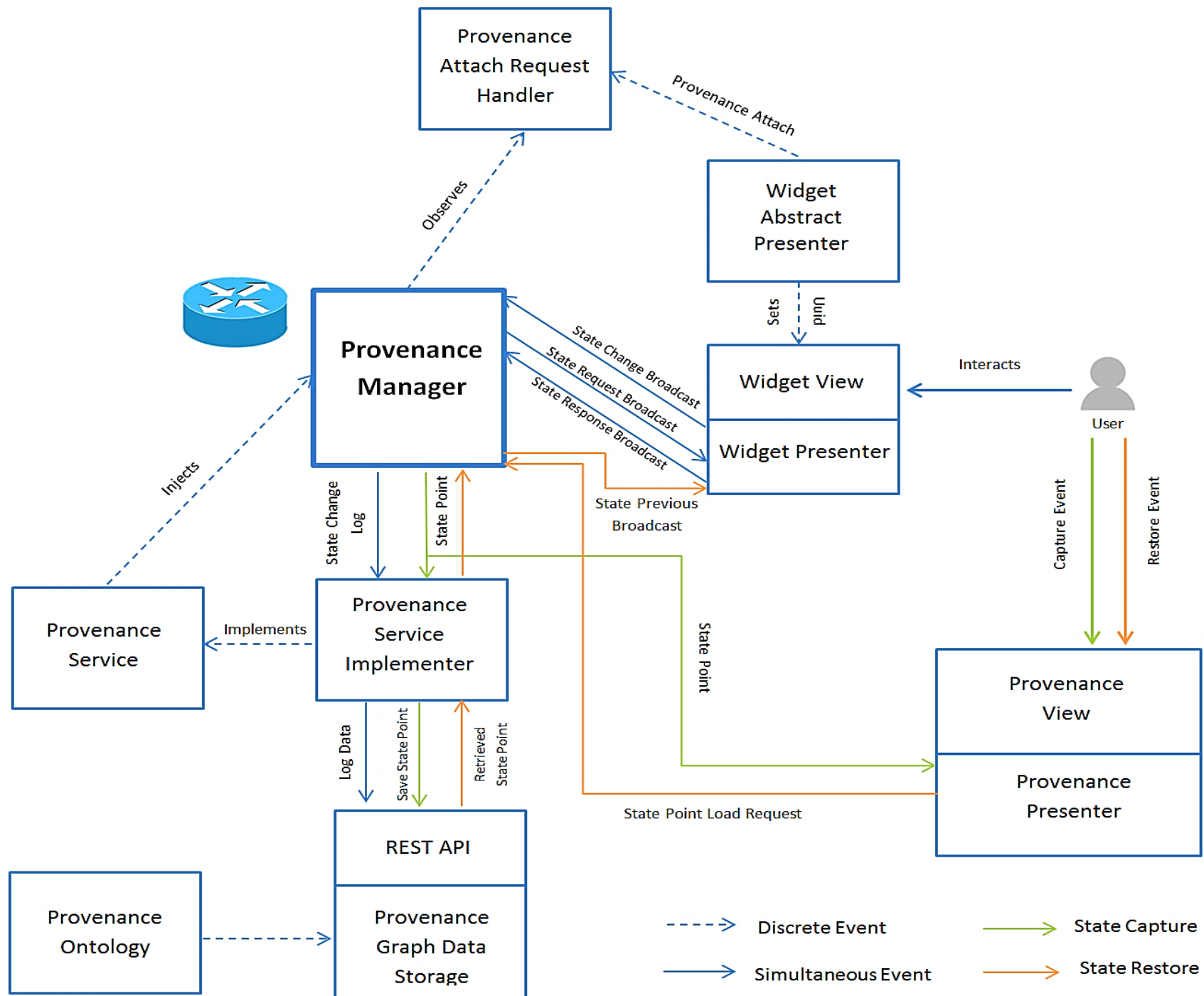


Figure 3.10: PROV - Provenance data flow diagram of Analyst's User Interface (AUI) back-end.

tier-3 data storage system. Provenance Service has got two vital roles: 1) Provenance Service Implementer and 2) Provenance Manager.

- **Provenance Service Implementer** – SAVE/QUERY SERVICES for provenance data i.e, log data and state point information as shown in below, are implemented by this role player into our system.
- **Provenance Manager** – While user interactions on AUI widgets occur, the interacted widgets initiate provenance service by broadcasting STATECHANGE message to Provenance Manager. A STATEREQUEST message is broadcasted by the provenance manager to receive state & state change information from different widget presenters through a STATERESPONSE broadcast message. This is how the provenance manager becomes aware of the state changes of AUI system. Not only state changes but also the provenance manager observes attachment requests into provenance system from different widgets through a request handler so that it can provide information on demand. These are all discrete events not dependent on user interactions as shown in above.
- **State Point Capture** – The analyst fires an event for capturing his/her intended analysis state. The most recent STATE POINT received into Provenance Presenter from Provenance Manager gets saved into data storage and creates an image as state point preview into provenance view PROV as shown in figure 3.9.
- **State Point Restore** – The analyst clicks on the state point preview to restore his/her previous analysis state. A State Point LOADREQUEST with it's corresponding id is sent to After receiving enquired state from data storage, Provenance Manager broadcasts this as a STATEPREVIOUS message (figure 3.9& 3.10) so that it is received by the widget to restore the analysis state back to the analyst.
- **Provenance Data-Storage** – All provenance data are stored into and queried from a virtuoso universal server in the RDF graph data format by using our developed REST (*Representational State Transfer*) API for the AUI system.

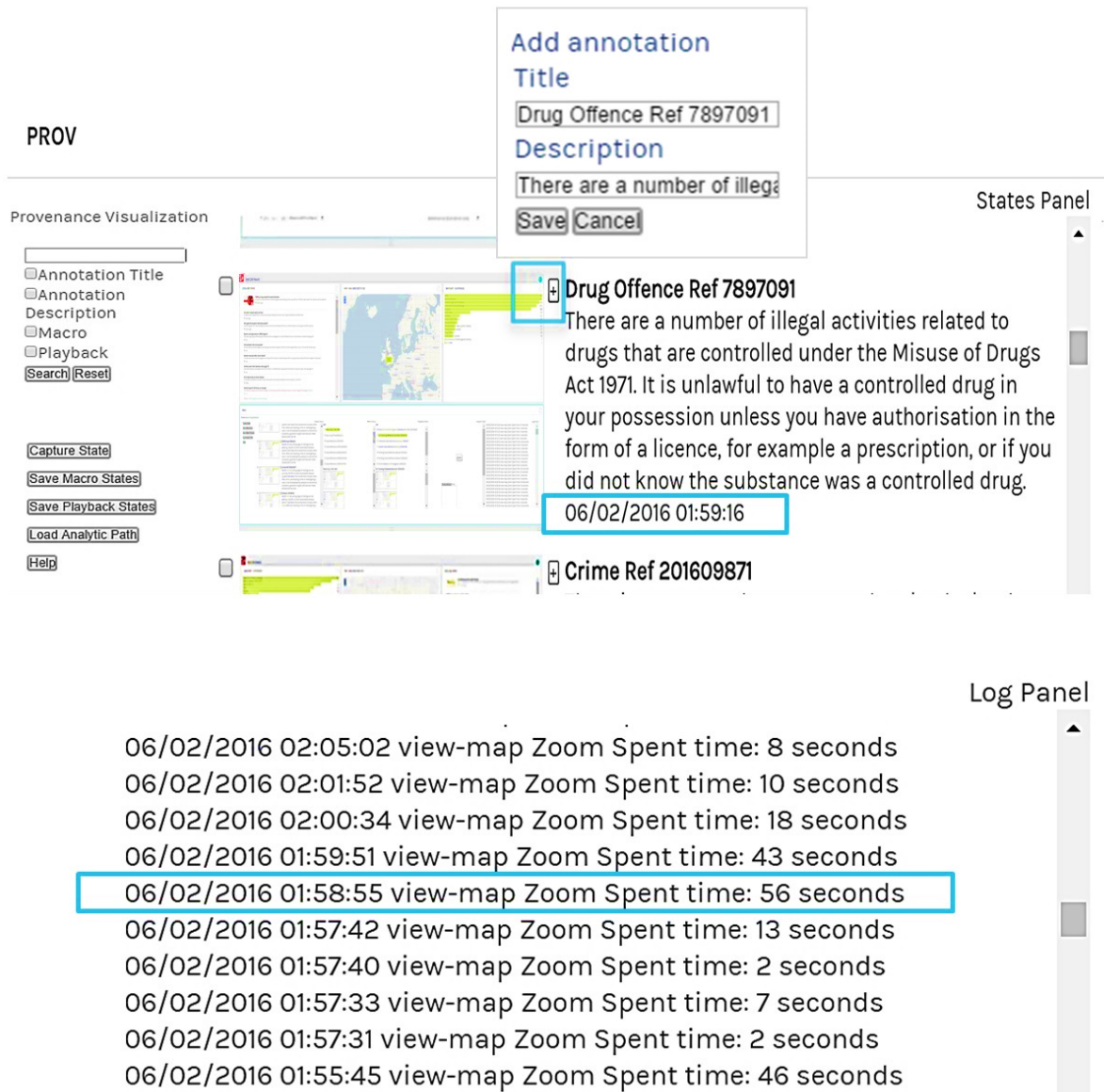


Figure 3.11: Manually captured **states panel** with annotation add/edit and automatically captured **log panel** for Analyst’s User Interface (AUI).

Provenance data gets stored into PROV-DB along with our proposed analytic provenance ontology.

3.3.3.3 Analytic Provenance Visualizations

We adopted UIMD (*Understand, Ideate, Make, Deploy*) design process model introduced by Mckenna et. al. [79] to implement AUI’s analytic provenance visualization system ‘PROV’ for supporting police intelligence analyst’s (end-users)

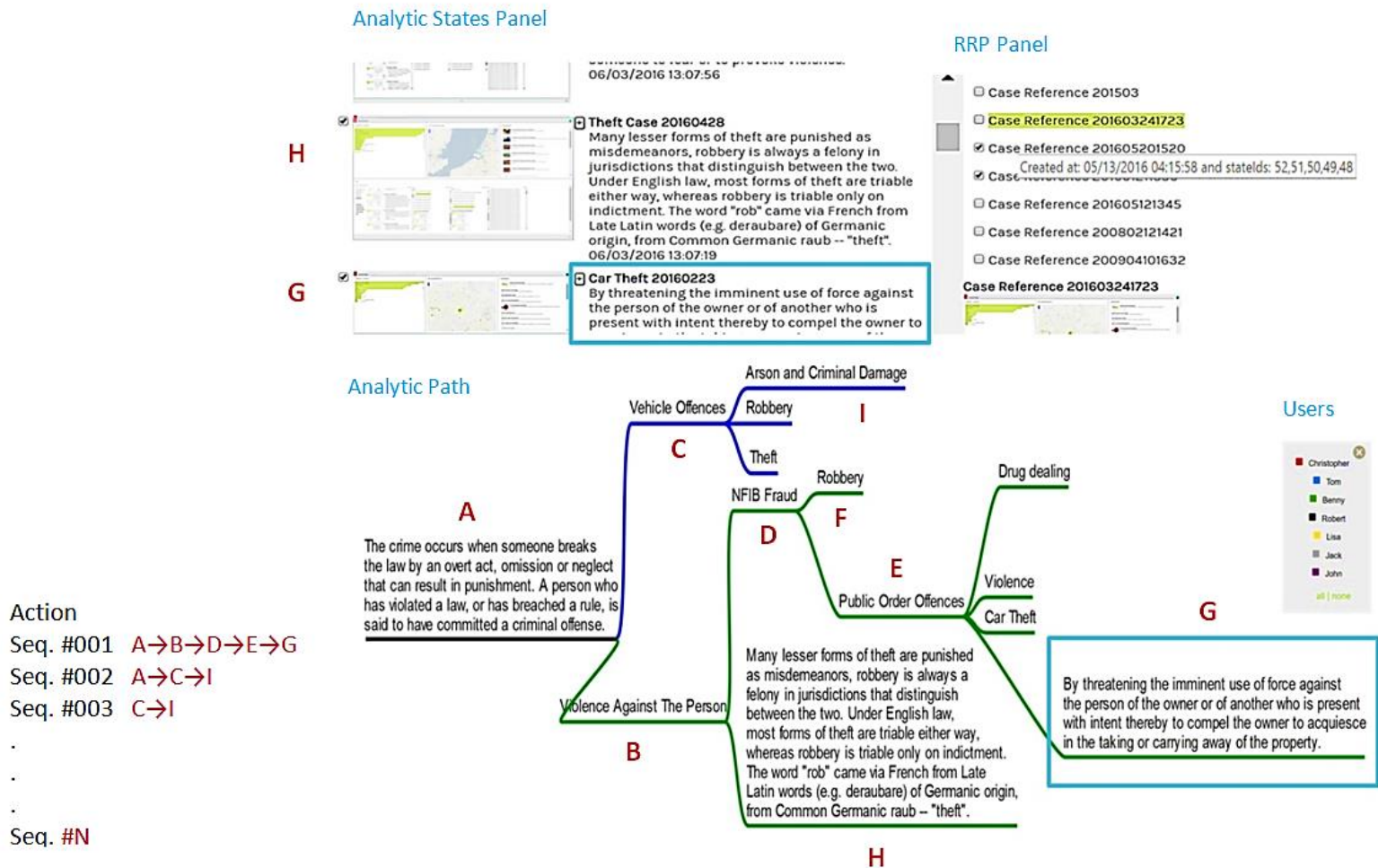


Figure 3.12: Analytic Path showing annotations set by analysts with captured states & their relationships based on interactions with colour coded users (analysts) information. States can be selected from States Panel & RRP (Repetitive Replicating Playback) list of Analyst’s User Interface (AUI) of the project *VALCRI to load analytic path for understanding intersections of analytical states captured by different analysts during their analysis process.

visual judgmental process on crime data. We performed requirements analysis (*AnaReqs*) to understand the challenge and ideate; designed a system to implement and deploy inside AUI. This system has several visual interactive panels for analytical captured states representation, multi-ways querying, workflow playing back and analytical process mapping. These visualizations have been built on our developed provenance data manipulating protocol (above) to query/access database and event based analytical states capturing method. The widget visualizations for crime analysis inside AUI system have been built on anonymized real crime dataset. We applied the same methodology as our earlier study prototypes on AUI system to capture analytic states.

Captured Analytic States Representation

To meet police analyst requirement *AnaReq1*, our developed provenance visualization system can capture different analytical states of AUI (*Analyst's User Interface*) and saves them as snapshots to show their previews (above). As well as to meet *AnaReq7* and *AnaReq8*, currently annotations can be added and viewed again on tooltips upon interactions with saved analytic states (above). Provenance data can be captured either manually by the analysts or automatically by the system as a log.

3.3.4 Recovering Data

How can we make sense of analytic provenance?

3.3.4.1 Analytic Path

Historical log alone is not sufficient for analyzing the analytical process with visualization tools. Often, there are relationships between the results and other elements of the analysis process which are vital to understanding analytic provenance.

To understand the relationships among reasoning steps we implemented '*Analytic Path*' (above) as a tool for visualizing analyst's activities through interactions with the visualizations. Intelligence analysis is not practiced exclusively as a solitary activity. So, in a collaborative environment of criminal intelligence analytic provenance can add considerable value, where it must be communicated and shared among teams. Additionally, by allowing communication and sharing of information, visual representations of analytic provenance data will support analyst's ability to identify and work with the desired information. So far the application of analytic provenance system supports sensemaking for individuals. In case of more than one analyst working together for a specific problem, automatically recorded interactions can help to understand their thinking processes.

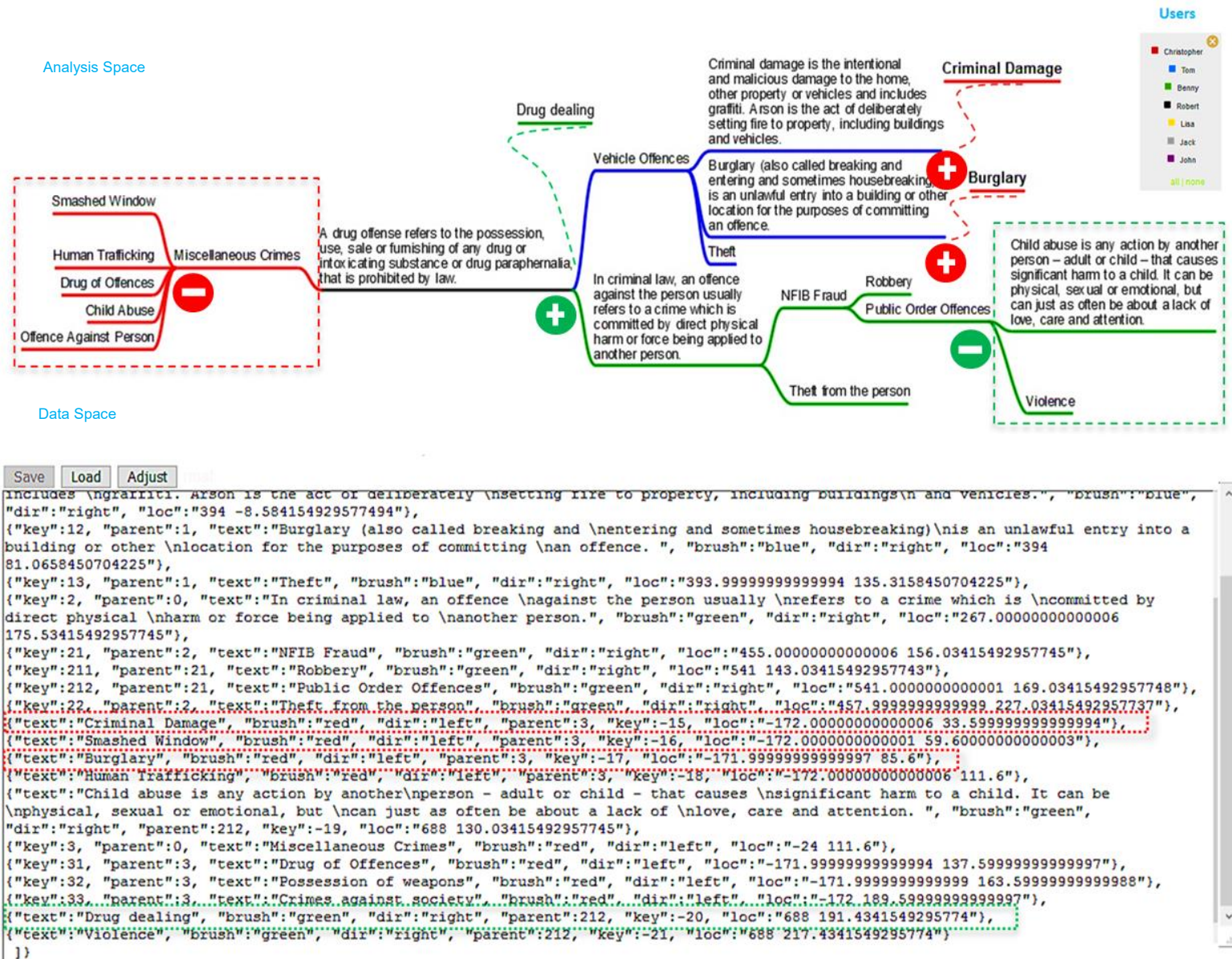






Figure 3.13: Schematization of analytic path in a visuo-spatially manner.

The tool 'Analytic Path' supports saving of mapped analytical states into and loading back from data storage. It allows combining such multiple maps together to make a visual story of the group analysis process. It also supports adding, deleting, editing or rearranging different branches with users' colour codes, consisting of annotations set by analysts along with captured states. This is known as 'Schematization' of the reasoning process as shown in Figure 3.13.

3.3.4.2 Schematization

Schematization is the process of organizing findings in some way that can trigger new insight. Pirolli and Card [4] suggest the need for tools to facilitate the schematization such as visualizing findings based on their temporal or spatial information. According to their sensemaking model, analysts seek information, search and filter for more relevant one, read and extract evidential information, and organize it into some schema. However, we find that the sensemaking loop is well elaborated through different sensemaking activities in the '*Data-Frame Model*' proposed by Klein et. al. [80]. These sensemaking activities are '*Connect*' data to a frame, '*Elaborate*' a frame, '*Question*' a frame, '*Preserve*' a frame, and '*Reframe*'.

According to data-frame model's terminology, the analyst tries to match some data to create an initial frame. When encountering new data, the analyst can either add it to the frame to elaborate the frame (if it fits to the frame) or remove existing data (if it cannot fit the frame any more). The analyst starts questioning the frame when they detect inconsistencies between data, or poor quality data in the frame. Then, they need to decide between preserving the frame by looking for more data, or reframing it by comparing it with other frames, or seeking a completely new frame [80]. Our developed '*Analytic Path*' visualization supports schematization in a visuo-spatial manner for iterative and dynamic nature. Takken et. al [81] found that when people directly manipulate data, for example, by moving individual pieces of information to create temporary groups or sequences, or eliminating pieces of information from a group; this can enhance their sense-making and analytical reasoning ability by helping them discover new explanatory relationships created by the rearranged pieces of information. They named the technique '*Tactile Reasoning*' which is an interaction technique that supports analytical reasoning by the direct manipulation of information objects in the Graphical User Interface (GUI). We have implemented the analytic path (above) in a visuo-spatial manner because it enhances the ability to process and interpret visual information about where data objects are in space and overcomes the shortcomings of existing tools to visually represent and support schematization in a collaborative environment. We visualized all of these analyst's

reasoning provenance information (annotations) as bookmarks on a time glider to provide their temporal information (below). The colour coded lines help to distinguish each analyst's reasoning contribution. As shown in above, analysts can integrate   and disintegrate   data to create a 'new frame'. They can drag the data and add it to an existing frame (*elaborate frame*). As well as they can disintegrate a piece of annotation data and drop it onto the void space (*preserving a frame*). The *dotted rectangles* into above shows the *parent nodes* of data pieces 'drug dealing', 'criminal damage' and 'burglary' into 'analysis space' and more about their spatial information into 'data space'. As well as *dotted curves* into analysis space shows their new nodes to be added with. Additionally analysts can release an existing frame into the void space to make all it's attached pieces of data free for rearrangement to add/delete new/existing pieces of annotation data. If the analyst thinks that a frame is completely wrong, s/he can construct a new frame (*reframing*). The analysis space can be panned, zoomed in/out and adjusted to scale and view the whole analysis in detail by using keyboard short keys. And schematization of annotation data can be saved into data space.

3.3.5 Reusing Data

How can a user's insight be reapplied to a new data or domain?

3.3.5.1 Repetitive Replicating Playback (RRP)

The research goal in analytic provenance is to be able to automatically reapply a user's insights to a new data or domain. In most analytical environments, analysts often utilize multiple tools simultaneously which renders the use of existing methods inadequate. A comprehensive and cohesive encoding, recovering, and reusing process is therefore necessary to support the analysts in their natural working environments.

One of the requirements from *Police Intelligence Analyst* (end users of the project above) as described into *AnaReqs* section was to record process and compare different provenance information (*AnaReq2* and *AnaReq6*). During the focus group they said "...we would like to be able to record a number of actions if some tasks are

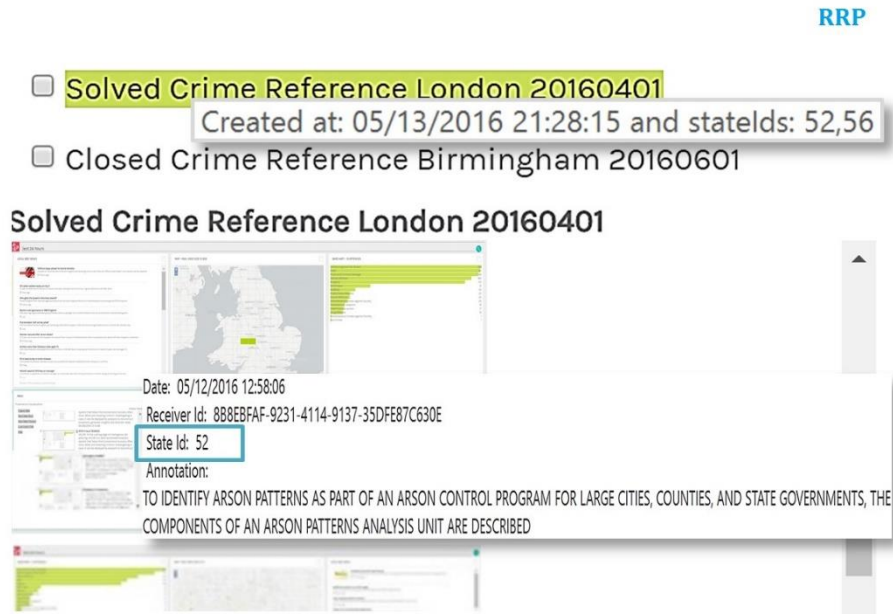


Figure 3.14: Repetitive Replicating Playback (RRP) System shows results with source state id information after running batch of saved group of states.

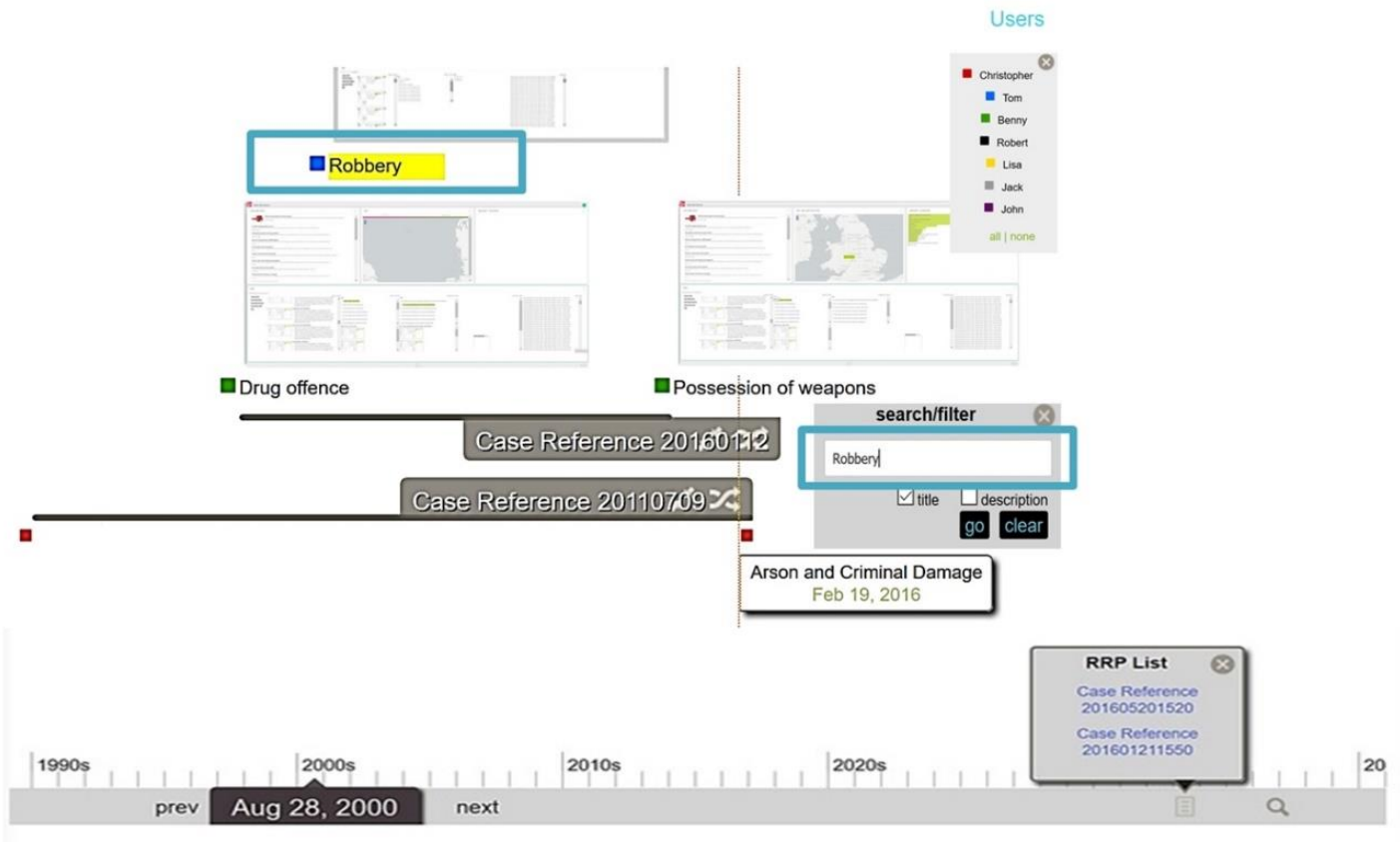


Figure 3.15: Visual representation of saved RRP batches of captured states and tracing those back by time gliding or by selecting colour coded users (analysts) or by keyword searching and selecting from RRP list.

more standardized. (start/stop recording). Therefore want to use the same tasks/ask the same questions for another case with similar data...". To develop such record/replay system we have proposed a model as shown into above in line with analyst's requirements. We call this Repetitive Replicating Playback (RRP) system.

The 'Analyst's User Interface' (AUI) for the project is consisting of many widgets developed under heterogeneous platforms. There are certain challenges to find out methods for reapplying insights to a new dataset by using such environment. North et al. [14] identified that analysts often utilize multiple tools simultaneously which renders the use of existing methods inadequate. Our proposed RRP system supports this problem very well for recording/replaying *WorkFlow (WF)* in a heterogeneous environment. The RRP cycle is consisting of 'replay', 'compose', 'reuse', 'retain' and 'compare' steps. We tested this model by implementing into AUI for replicating WFs on different set of crime data and compare result set with selected previous states to gain new insight. After 'replaying' each RRP state (consisting of previously captured group of states), we find automatically captured states (as result) to 'compare' with their corresponding previous states. Composition of such RRP states can be modified by adding, deleting or reshuffling all of their contained previously captured analytic states. We can 'compose' new RRP state by making selections from States and RRP panels (above) and save ('retain') them for future use.

One of the challenges to 'reuse' captured analytical states is to be able to formulate queries that retrieve and employ traces in order to fulfill an analyst's information needs in an user-friendly way. It involves formulating ad-hoc traceability queries, allowing interactive filtering of retrieved analytic states and ad-hoc query refinement. We visualized all RRP states (consisting of previously captured analytic states) on a timeglider as batches of captured analytic states organized in a temporal order (above). States can be searched/filtered by types and/or users. As well as states sequences can be represented (highlighted in yellow colour) to show the analysis steps temporally. Also states can be traced back by using temporal information (gliding the timeline or using calendar). Our developed RRP (*Repetitive Replicating*

Playback) system includes W3C PROV-AQ: Provenance Access and Query standard factors i.e, *Recording* – represent, denote; *Querying* – identify, pingback; *Accessibility* – locate, retrieve.

3.4 Evaluation

We conducted an evaluation with our police analyst end-users to elicit subjective feedback on our prototype. They were the participants (P) of this evaluation process. We wanted to evaluate how the provenance visualizations support analysis and reasoning about data for deriving relevant knowledge in criminal intelligence. The evaluation involved qualitative focus groups. We had three groups of analysts who participated in pairs. Each pair was from a different police organization. The procedure of the focus group involved demonstrating the prototype, illustrating the visualizations for different tasks, and obtaining feedback. Each group had 30 minutes for the demonstration and feedback. We had separate observers during the focus groups that recorded notes, ideas, and feedback from the end-users. We now report on the feedback as recorded by the observers, based on five questions as follows.

Question 1: What is the purpose and value of the component?

P1: The goal of the tool is to track the analyst's work and save what has been done enabling the later retrieval. It will capture the data from the interface and information on the users' interactions. The analyst will be able to "bookmark" a particular stage in the process and save it as part of the provenance record.

P2: Track and capture what has actual been done. Bookmark and save as a provenance component The idea is to be able to capture data from the interface and user interactions.

P3: Capture different states in time. Retrace earlier states system and user provenance.

P4: Capturing analyst's use of the system. Capturing anything happening from analyst's point of view and data point of view. Provenance refers to a historical record of the process taken or employed. Tool offers a capture mechanism.

Question 2: Is the purpose and value clear to End Users?

P1: Yes. The end users confirmed it is useful for the analyst to save the current state and be able to get back to it. Also, it is helpful if the analyst can automatically save the progress and get back to states they didn't save. Finally, the tool would allow the supervisor to trace the reasoning of the analyst.

P3: Once conceptual architecture, was displayed and explained, the Users really understood the component.

Question 3: What do End Users like or dislike about the component?

P1: The end users enjoyed the ability to get back to a certain significant point in time. They are interested in process playback not typical video playback (i.e. the re-run of the process which provides clear understanding of its various stages and components, not an overall view of what has happened and when).

P2: Feel that it could be of use to them as they don't do something like this already. Able to go back through the provenance log to a specific point and save is good. Would like to be able to record a number of actions if some tasks are more standardized.

(start/stop recording). Therefore want to use the same tasks/ask the same questions for another case with similar data (i.e. Macros). Want to be able to see previous action. recommendation system could be useful.

P3: Generally very useful. Particularly if you have several pieces of work in progress and you may not work on one for 2/3 weeks and have to come back to it. "Adding notation would be brilliant". Good basis for training for inexperienced analysts. Being able to pause the training and show what an experienced analyst would do. Ability to find new information and go back to and older state and add the information. The ability to demonstrate that you have previously saved states that are similar.

P4: Analysts have no way of capturing the data except in log-book. This is difficult to do given workload so tool is very helpful to this end. Like the ability to annotate Guides Tool offers a good basis for training; follow in the footsteps of an analyst using the provenance tools Like the tool they can reconstruct the path the analyst took over time.

Question 4: What features or functionalities would End Users like added, changed or removed?

P1: The end users suggested the recoding of the set of actions, not states, and then applying it to a new case in the same data set, the feature resembling macros in Excel.

P2: Would like to be able to record a number of actions if some tasks are more standardized. (start/stop recording). Therefore want to use the same tasks/ask the same questions for another case with similar data (i.e. Macros). Want to be able to see previous action recommendation system could be useful.

P3: Ability to “play” through the saved states.

P4: Analysts log in for system of auditing and performance.

Question 5: Overall, is the End User group’s assessment positive, negative or neutral?

P1: The assessment is positive. The ability to record the macro would be useful for repetitive tasks. The ability to review the history of the analytic process would provide useful insight into the evolving judgment.

P2: If you’re doing the same task everyday then a Macro would be useful, but Guy wouldn’t find the history side of things as useful as Mark would. Essentially a hypothesis flow Playback definition – process playback – re-run the process.

P3: Overall, very positive.

P4: Positive.

Purpose and value

All the end-users understood the purpose of the prototype to log and track analytic workflow in the above [82] system. They claimed that this task would add value to what their current workflow processes are, as it would allow them to track what they were doing on a daily basis and analyze what they had done previously. They deemed these tasks to add value, as it is necessary to explore different analytic pathways, or to even pick up and validate the work of others.

Strengths and weaknesses

The biggest strength as reported by all the end-users was that the tool tracked the tasks they were performing as well as the ability to bookmark certain parts of the interface they were working with. The tracking and book-marking feature was found to be useful as they could come back to a previous state where they had been working and continue to work from that state.

Improvements

Different features were suggested by some of the analysts for addition to the prototype. They would like to see a team leader login part, which can monitor the activities of all analysts. The purpose for doing so is they can see at what stage an analyst is working on within a crime investigation and to get reporting features based on the progress of analysts. They would also like to be able to add outcome reports to different stages of the analytical path. Being able to summarize information through annotations and free text will enable analysts to record some of their thoughts when investigating a crime.

Satisfaction


The overall assessment of the prototype by all the end-users was very positive and they were satisfied with the progress of the prototype. All the analysts felt that the different provenance features could add value to what they are currently doing and to help make more effective decision making for criminal intelligence analysis.

3.5 Discussion

The key to this research on analytic provenance is the belief that by capturing user's interactions with a visual interface, some aspects of the transparency of user's reasoning processes can be retrieved. To correlate analyst's interactions with the visualizations for his/her reasoning process, the analytic provenance research needs to start with the understanding of how information is perceived by the user. We conducted a focus group discussion meeting with the police analysts to understand their needs for analytic provenance visualization. As the user interacts with visualization, the series of interactions can be considered as a linear sequence of actions. So, how can these analytic provenance information be captured – is still an open challenge. We have implemented our proposed protocol for managing huge analytic provenance dataflow for a large complex system like *Analyst's User Interface* (AUI) of the project above [82]. Once the user's provenance data has been captured, the challenge becomes making sense of the provenance. As noted by Jankun-Kelly et al. [83] history alone is not sufficient for analyzing the analytical process with visualization tools. Often, there are relationships between the results and other elements of the analysis process which are vital to understanding analytic provenance. Our provenance visualization system can also capture analytical relationships automatically. We have developed an analytic process mapping system named as '*Analytic Path*' to visualize those related process sequences for multiple analysts working in a group. One of the research goals in analytic provenance is to be able to automatically reapply a user's insights to a new data or domain. It refers to the utilization of specific knowledge of previously experienced, concrete problem situations or cases. By employing such repetitive process, the analyst can solve a new problem by finding a similar past case, and reuse it in the new problem situation. We have developed a '*Repetitive Replicating Playback* (RRP)' system, where analysts can use their previously saved group of analytic states, apply to new dataset and see the results. We have tested our proposed way of capturing event-driven analytical

provenance by developing visualization prototypes based on police intelligence analysts' requirements and found it supports the challenges of five interrelated stages of analytic provenance generically, as suggested by North, et al [14] i.e., *perceive, capture, encode, recover and reuse*.

According to Gotz, et al's[10] hierarchy of analytic behaviour, the sub-tasks at higher-level have more concrete states with rich semantics into provenance-aware analytic process comprising of interactions for understanding human intention and computational elicitation. Semantics of interactions that occur during switching among multiple visualizations hasn't been addressed into this work. Also this work hasn't addressed the coupling between cognition and computation through interactions during analytic processes. As well as for sensemaking or computational problem solving during crime analysis in criminal intelligence and the analytic processes, require insightful alignment with the visualizations for supporting analyst's thought processes. The current developed visualizations have got limited support in this regard, which we shall present into next few chapters. Our future endeavour for this work is to add few more features with our current system i.e., creating case specific new provenance capturing space with pluggable annotation system and tag them. Also developing a document trail system by using attached crime reports with the annotations will be useful as identified by the police intelligence analysts. We also progressed to develop an ontology for analytical provenance as presented into future work section. This is currently absent into W3C standard for describing and integrating analytical states from different sources. As well as visualizing evolution of ontology is a crucial issue to understand the way knowledge evolves form one state to another during analyst's analytical process and that is a potential area of research.



Sensemaking Behavioural Markers



4

chapter

4.1 Chapter Overview

Recovering cognitive reflection on analytic reasoning processes from extended log data or only by observing is a difficult task. In the previous chapter we proposed RRP model as a way of composing and replaying those cognitive step sequences in the form of captured group of analytic states. But a gap exists between the cognitive constructs and manipulations or interactions humans employ to think and reason about data as identified by many researchers. We hypothesize in the current chapter that ***'Behavioural Markers (BMs)' can act as attributes for bridging between human cognition and analytic computation through interactions during fluid transitions between mental and analytic processes at micro-analytic level.*** To test this hypothesis we have considered following research questions:

RQ3: What are the constructs of Behavioural Makers (BMs)?

- To form an exhaustive list of behavioural constructs for criminal intelligence, we have presented a systematic approach to identify set of mostly relevant BMs by considering human factors and cognitive engineering principles.

RQ4: How to translate reasoning processes to Behavioural Markers (BMs)?

- Sequences of captured analytic actions need to be structured for meaningful representation of BMs. We have shown how the use of network graph visualization in this context can be a useful exploratory process, rather than exhaustive to observe and gain understanding of which empirical action combinations may provide meaningful sequence for targeted BM.

RQ5: How to externalize thinking processes from the constructs of Behavioural Markers (BMs)?

- A compositional reduction mechanism based network graph analysis of interactions has been proposed in this chapter as a way of recognizing BMs within an automated framework. But such automation has limitation too as a computer has no ability to make an expert judgement in the same way that a human can through experience or intuition, on the thinking process of someone solving a task. So, we conducted a *'Cognitive Task Analysis (CTA)'* too to detect transitions between mental states depicted by interacted states of visualizations through analytic processes. This study was part of the evaluation of the design principles of *VALCRI's *'Analyst's User Interface (AUI)'*.

4.2 Introduction

The analytical reasoning process is viewed as a cognitive process allowing individuals to interpret information in context so as to derive knowledge to initiate specific actions [84]. The actions of reasoning process lead people to ask different questions and to focus on understanding underlying cognitive processes. We consider intelligence analysis as a fluid activity allowing humans to transition between mental and computer interaction states. It is a coupled human cognition and analytic computation activity that is enhanced through computer interactions which can be decomposed into human intention and computational elicitation activities. To detect and understand the transitions between mental and interaction states we propose '*Behavioural Markers (BMs)*' of analytical processes as the bridge between them in criminal intelligence.

Commonly '*Behaviour Markers (BMs)*' are known as observable '*Non-Technical skills (NTs)*' that contribute to superior or substandard performance within a work environment [139]. In criminal intelligence for the successful investigation, an analyst requires a variety of skills i.e., cognitive, interpersonal and technical. Some of the cognitive skills include thinking critically, reasoning well, evaluating inferences and using logic. Necessary interpersonal skills underpinning good analysis include communicating clearly and building relationships. Technical skills necessary for good analysis are mastering specific techniques and being a strong researcher. The cognitive, personal resource and social skills that complement a person's technical skills and contribute to an overall task performance are termed as '*Non-Technical Skills (NTs)*'. NT skills cover the cognitive and social sides of a person.

Visual Analytics tools in recent years have made an impact in the criminal intelligence and analysis communities. Capture of user interaction as a user history has been used to advance our understanding of tool usage and user goals in a variety of areas. User interaction histories contain information about the sequence of choices that analysts make when exploring data or performing a task. To understand how the analyses are being made, users require support of correlating lower-level events with tasks, and tasks with goals [10].

Until recently, most of the research has focused on techniques and methods for refining visual analytic tools, with the emphasis on empowering analysts to make discoveries faster and more accurately. Although this emphasis is relevant and necessary, we like other researchers argue that the process through which an analyst arrives at the conclusion is just as important as the discoveries themselves. Understanding how an analyst performs a successful criminal investigation will finally let us start bridging the gap between the art of analysis and the science of analytics. We found out from our proposed detection approach of '*Behavioural Markers* (BMs)' from analytical data that they can bridge such gap alongside of performance measurement. This part of research work is aimed to find out appropriate methods or techniques to evaluate a visual analytic tool named as Analyst's User Interface (AUI) of the project VALCRI* (Visual Analytics for Sensemaking in Criminal Intelligence Analysis). The goal is to determine the extent to which imagination, insight, transparency, and fluidity and rigour are enhanced on the assumption that improving these, will likely improve analysts' ability to solve crime or be better at performing criminal intelligence analysis. The overarching aims of this research are based on research questions RQ3, RQ4 and RQ5 as described into chapter overview (Section 4.1).

4.3 The Problem

As real-time and retrospective interviews of analysts sometime produce inaccurate characterizations of the analytic process, other means of collecting information on the methods and steps that comprise the analysis process e.g., logging of user interactions, has already been introduced in many systems. Endert et. al. [38] argue that manual user interaction data capture may present significant usability issues because the process forces users out of their cognitive flow or zone, which may place fundamental limitations on reasoning activities. Reasoning about data is an inherently cognitive activity, where the mental artifacts that we leverage to reason can manifest themselves at different semantic and symbolic levels of detail. Thus, a gap exists between the cognitive constructs and manipulations or interactions humans employ to think and reason about data [85].

Table 4.1: Behavioural Attributes.

Enquiry	Conjecture	Goal
Knowledge[P2,P5], Intellectual- Curiosity[P3,P4], Foresight[P5,P6], Curiosity[P5,P6], Intuition[P5,P14], Synthesis[P6], Associative Questioning[P18], Sense of Humour[P7], Information Manipulation[P7], Pattern Recognition[P8].	Reasoning Ability[P1], Skepticism P[1], Imagination[P1], Generate Conceptual Models[P2], High Level Reasoning Ability[P3], Inductive Reasoning[P3], Intellectual Flexibility[P3], Deliberateness[P3], Make Judgements[P4], Logical[P5], Imagination[P5], Visual Thinking[P6], Systematic Thinking[P6], Thinking Ability[P7], Creative Connections Establishment[P7], Critical Reasoning[P7], Critical Thinking[P9,P10], Judgmental[P10], Anchoring[P17], Laddering[P17].	Comprehension[P2], Innovation[P5], Exhibit AHA Thinking[P7].

We propose to use the concept of ‘*Behavioural Markers (BMs)*’ as attributes for bridging between human cognition and analytic computation through interactions during fluid transitions between mental and analytic processes at a ‘*micro-analytic*’ level. From a behavioural perspective, analytical reasoning process sequences provide information about underlying cognitive information that relate to measuring performance [86, 87, 88]. Exploration of the processes employed, in problem solving or in engaging with complex tasks, provides information about the cognitive skills which underlie successful resolution of the problems or tasks [89, 90, 91]. The cognitive skills can be demonstrated through behaviours, which are captured in the form of reasoning process, that is completed in an intelligence analysis task environment. We aim to detect these behaviours in the sensemaking loop [4] during an activity that involves high-level tasks and sub-tasks [10] performed by analysts through low level events in the information foraging loop within a typical task model for criminal intelligence analysis.

4.4 Development Approach

The typical method for the initial development of ‘*Behavioral Marker (BM)*’ systems is to carry out a literature review of previous domain specific research concerned with ‘*Non-Technical skills (NTs)*’, followed by interviews with Subject Matter Experts (SMEs) designed to extract the ‘*Non-Technical skills (NTs)*’ required to do their job effectively [28, 29 30]. We carried out a ‘*Systematic Literature Review (SLR)*’ to create

an initial list of behavioural concepts that have been shown to be necessary for effective performance in wider intelligence domain will surface the following; intelligence analysts, crime analysts, strategic analysts and so on.

We considered following cognitive criteria as included by the †International Association of Law Enforcement Intelligence Analysts (IALEIA) in their †Foundations of Intelligence Analysis (FIAT; IALEIA, n.d.) training:

- **Enquiry** - A unit of knowledge acquired from exploration and interacting with the data.
- **Conjecture** - A supposition made by the analyst, usually as a result of making a series of enquiries.
- **Goal** - A phase reflecting the formation of an exploratory objective.

Several electronic databases (PsychINFO, ScienceDirect, Web of Science, Google Scholar, and the Defence Technical Information Center) were used to identify research articles by using the search terms: criminal intelligence, behavioral markers, human factors, situation awareness, decision making, intelligence analyst, cognitive skills etc. We also included a series of research papers on '*How Analysts Think*' [P13, P14, P16, P17, P18] to find out the related behavioural concepts of imagination and insight generation in criminal intelligence. Through the studies of analysts' thinking strategies we found - Gerber et. al. [P13, P14] proposed that before insight occurs, there is a stage of intuitive reasoning that leads to some form of assessment based on little or no data, a confused situation, or just plain ambiguity, which suggests to the analysts that he or she has to make a considerable leap - stretching the limits of one's belief (what might be considered realist or plausible) - stepping out and taking the risk of failure or ridicule, to propose or suggest a likely outcome that is a novel way of problem solving. Wong et. al. [P16] explained that inferences could be inductive, deductive or abductive in nature, and carried out in non-sequential manner and is very often intertwined and chaotic and cyclic where one starts depends on what data is available, what goals they wish to satisfy (i.e., to gain traction vs to prove a point), and the claims they desire to make. Wong et. al. [P17] also discussed '*anchors*' and '*anchoring*' when dealing with missing or ambiguous data; '*ladders*' and '*laddering*' as

they leverage off over data; and how many of these processes are linked together is through '*leaps of faith*' - a strategy to cope with missing data - that provides insights that can be used as suppositions to gain traction. Qazi et. al. [P18] discovered strategies on '*associative searching*' when one thinks one has exhausted all options - he or she needs ways to activate new possibilities through semantic similarity. We classified all of these cognitive criteria according to †below; below, n.d. to form SLR PHASE-I [31] behavioural concepts as shown into above. We also presented all of these behavioural concepts as a '*Means-Ends Abstraction Hierarchy*' in below to interrelate those and illustrate a decomposition approach to identify '*Behavioural Markers (BMs)*'. The means-ends chain (MEC) has been used as a hierarchy of '*goals*' that will afford the analysts with the ability to solve crimes effectively. The goals can be grouped into following three levels:

- **Action Goals:** These are concerned with acts of analyst themselves which include design, approach and strategies of MEC.
- **Outcome Goals:** These are immediate effects of actions related to components of MEC.
- **Consequences:** These are indirect effects from outcomes which help analyst to step on sub-concepts/concepts of MEC.

We arranged a workshop to discuss these concepts, considered for the evaluation task of the project VALCRI* and extracted related cognitive behaviours. There were about 30 researchers of criminal intelligence domain including ex-police, ex-intelligence analysts and other developers were present in the workshop. The whole team initially was divided into several groups and then each concept was gone through one by one. Each person in the group said some words that they associated with the concept. We put each concept on post-its and organized them thematically like an affinity diagram at the end.

†IALEIA - <https://www.ialeia.org/>

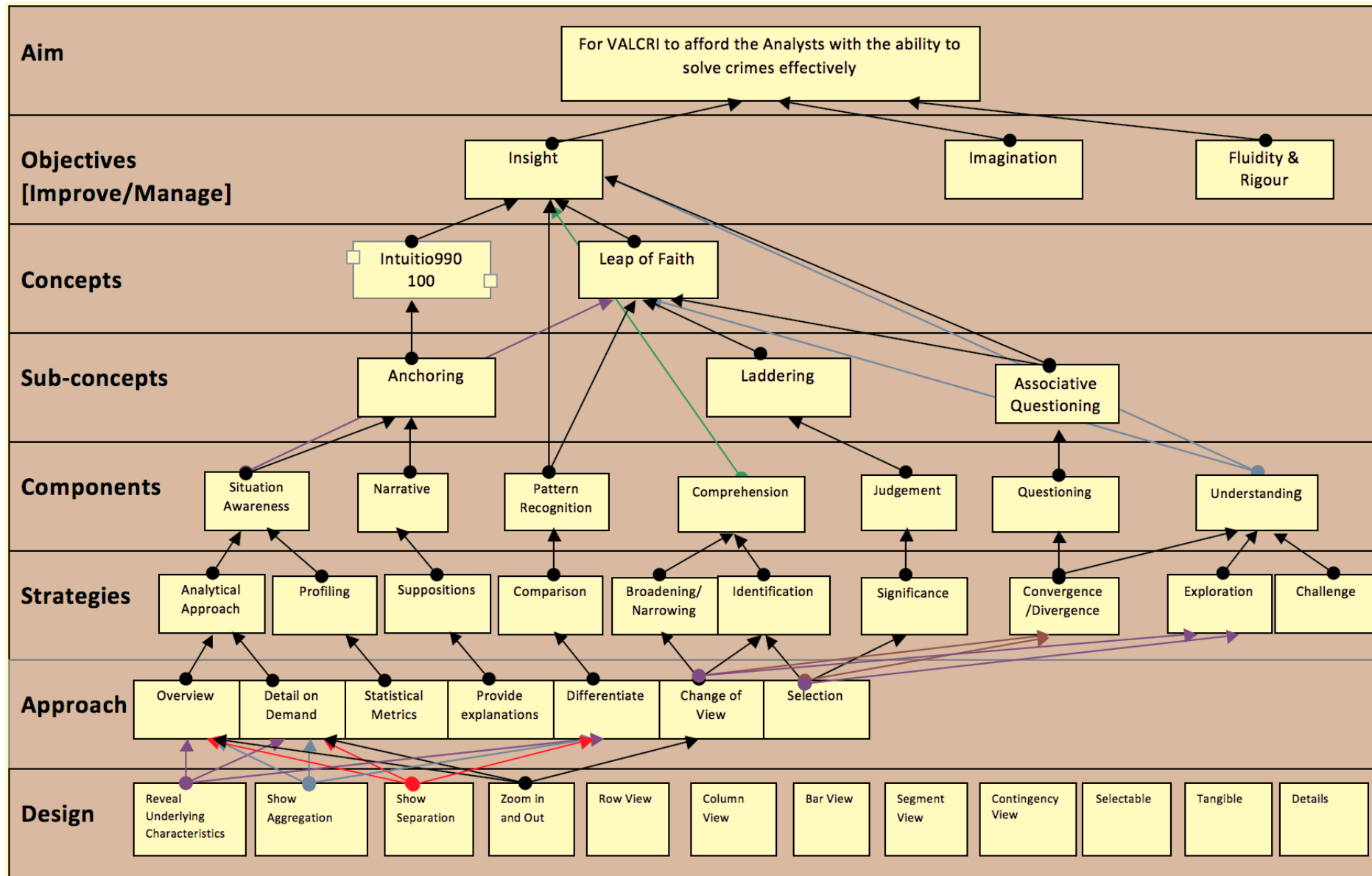


Figure 4.1: Means-ends abstraction hierarchy to illustrate the decomposition approach to identify Behavioural Markers (BMs).

During the workshop we also conducted focus group discussions among present researchers to formulate clear aims based on challenges of criminal intelligence analysis i.e., concepts of how do analysts think, what are the expert analytic reasoning and problem-solving strategies and how do they make sense of situational data etc.

Thus in Phase- II we formed an exhaustive list of '*Behavioural Markers (BMs)*' for criminal intelligence analysis as shown into below. Our aim is to identify a set of mostly relevant '*Behavioural Markers (BMs)*' by considering human factors and cognitive engineering principles that underlie the design of user interface, visualization and interaction on criminal intelligence analysis system.

So far, we have presented a decomposition approach for '*Insight*' into above and showed how it's constructs at different MEC levels can engage analysts to gain and achieve their final goal. On the otherhand, it decomposes their approaches from top towards the bottom level of hierarchy. We also have described a quantitative technique into next section of this research to detect constructs of '*Imagination*' and explained those by using cognitive engineering theory.

Table 4.2: Observable behavioural markers and their constructs for criminal intelligence analysis.

Categories	Antecedents	Processes	Outcomes
Imagination	<p>Passion, Inspired, Moral</p> <p>Motivation Openness, Focused, Inspiration, Motivation, Playfulness, Curiosity, Freedom.</p>	<p>Divergent Thinking Openness, Curiosity, Creative Play, Exploring, Experimenting, Idea Generation, Free Thinking, Freedom, Outlier Thinking, Thinking Outside the Box, Inventing, Going Beyond Given Information, Traditional Assumptions, Unusual Interpretation, Fluency, Flexibility.</p> <p>Mental Modelling Analytical Reasoning, Metaphorical Thinking, Analogical Reasoning, Moral Reasoning, Contrarian Thinking, Probability Reasoning, Questioning, Abstraction of Terms, Changing Potential Output, Comparison, Finding Alternate Objects, Generating Hypotheses, Scenario Building, Inferring Possibilities.</p>	<p>Idea Generation, Novelty, Inventive, Abstraction of Terms, Acceptance.</p>
Insight	<p>Incubation, Flair, Reason, Belief in Truth, Getting out of an Impass.</p> <p>Means to support insight Visualized information, Visualizing information.</p> <p>Managing Complexity Untangling complexity, Mess finding.</p>	<p>Ideational Developing New Ideas, Developing New Perspective, Evolving Perception, Revelation, Intuition, Understanding a Situation, Perceiving Information, Laddering, Creating a New Pattern, Associative Questioning, Leap of Faith.</p> <p>Problem Solving Recognition and Discovery, Problem Reformulation, Reframing, Uncovering.</p>	<p>Consequences Relevance Enhanced Perception, Being Able to Explain, Contribution to Plausible Narrative, Evidence for Hypothesis Building, Verifying Hypothesis, Contradiction of Previous Beliefs, Questioning Assumptions.</p> <p>Outputs Awareness, Understanding, Enhanced Perception, Unexpected Understanding, Sudden Jump in Understanding, Understanding Hypothesis, A Solution of Unknown Provenance, New Knowledge, New Pattern, Possibility, Discard Options, Breakthrough.</p> <p>Giving Insight Seeing Something in a Different Light, Unexpected Understanding, Eureka Moment, Recognition and Discovery, Without Conscious Thoughts, Internal and Conscious.</p>

Table 4.2: Observable behavioural markers and their constructs for criminal intelligence analysis (contd.)

Categories	Antecedents	Processes	Outcomes
Transparency	<p><u>Proper Motivation</u> Making Awareness Visible.</p>	<p>Structured Analysis, Critical Thinking, Assessment of Source Quality, Open Source, Ease of Access, See Through, Observability, Recording of Provenance, Externalization of Reasoning, Externalization of Assumptions.</p>	<p><u>Accountability and Legal Compliance</u></p> <p>Showing Compliance, Accountability, Legal Clarity, Legal Certainty, Fairness, Honesty, Truth.</p>
	<p><u>Techniques</u></p> <p>Usability, Visibility and Configurability of Algorithmic Parameters, Immune to Changes by Unauthorized Persons, Showing Info Outside Threshold, Define User Access, User Manuals.</p>	<p><u>Precision on Communication</u></p> <p>Communication of Uncertainty, Communication of Complexity, Communication of Probability, Communication of Limitations, Communication of Analytic Confidence.</p>	<p><u>Effects</u></p> <p>Contradiction of Privacy, Structured Analysis, Analytic Provenance, Making Awareness Visible, Critical Thinking, Acknowledging Alternatives, Ability to Understand and Reconstruct Operations or Decisions.</p>
		<p><u>Engagement of Multiple Stakeholders</u></p> <p>Individual and Collaborative Roles, Different Stakeholders.</p>	<p><u>Auditability</u></p> <p>Feedback, Easy to Access, Open Source, Disclosure, Traceability, Ability Know and Track Back, Verifiability, Showing Information Outside of Threshold, Direct Manipulation.</p>
			<p><u>Provenance</u></p> <p>Audit, Traceability, Disclosure of Algorithmic Reasoning, Accountability, Elements & Paths between Premises & Conclusions in Reasoning.</p> <p><u>Precision</u></p> <p>Counters Misuse, Not Ambiguous, Not Beguiling, Clarity, Accuracy, Certainty, See Through, Applicability, Acknowledging Alternatives, Quality of Information.</p>

Table 4.2: Observable behavioural markers and their constructs for criminal intelligence analysis (contd.)

Categories	Antecedents	Processes	Outcomes
Rigour	<p><u>Visual Support</u> Clear Distinction Between Facts and Suppositions, Narration.</p> <p><u>Analytic Support</u> Application of Analytic Techniques, Helpfulness, Decision Point, Seeing the Process of Deepening Analysis.</p>	<p><u>In Analysis</u> Structured Analytic Technique, Consideration of Multiple Hypothesis, Critical Thinking, Accuracy of Judgement, Stick to Rules & Procedures, Principle, Order, Responsibility, Due Diligence, Attention to Detail, Information Validation, Adherence to Standards, Rigour of Provenance, Certainty, Assessment of Sources & Quality, Timeliness, Substantiate.</p> <p><u>In the Communication of Analytic Findings</u> Communication of Analytical Provenance, Communication of Analytic Confidence, Communication of Assumptions, Communication of Probabilities, Communication of Uncertainty, Rigour of Argument, Evidenced, Substantiate, Trust Calibration, Confirmative Hypothesis, Decision Point, Information Validation.</p>	<p><u>Compliance</u> Due Diligence, Responsibility, Legal Compliance, Adherence to Standards, Assessment of Sources and Quality, Comprehensiveness, Thorough, Thoughtfulness, Attention to Detail, Exhaustive, Certainty, Stick to Rules and Procedures, Order, Rigour of Process, Principles, Rigour of Provenance.</p> <p><u>Fit for Purpose</u> Timeliness, Relevance, Commitment.</p> <p><u>Transparency</u> Clear Distinction Between Facts & Suppositions, Clarity of Reasoning, Transparency, Externalization of Reasoning Process, Seeing the Process of Deepening Analysis, Rigour of Provenance, Communication of Analytical Provenance.</p>
Fluidity	<p><u>Visual Support</u> Adaptable UI, Intuitive Interactions, Rapidly Reversible Interaction, Low Cognitive Load, Dynamic, Content Related Adaptation, Ease of Use, Multiple Views to Blend, Transposition of Data, Variability of Logical Relationships, Fast Analytic Response Time.</p> <p><u>Analytic Support</u> Transposition of Data, No Data Wrangling, Ease of Representing Relationships, Holistic View of Data.</p>	<p>Intuitive Interactions, Variability of Logical Relationships.</p> <p><u>Withholding Commitment</u> Circumspect, Tentative, Malleability, Explorable Data Analysis, Ease of Transition, Consideration of Multiple Hypothesis, Playfulness.</p>	<p>Variability of Logical Relationships, Context Related Adaptation, Ease of Use, Divergent Thinking, Explorable Data Analysis, Playfulness, Malleability.</p>

4.5 Behavioural Markers (BMs) Detection

4.5.1 Quantitative Approach

From a quantitative behavioural developmental theory perspective [92], behavioural constructs are events that have the potential to be directly observed. We have identified a set of behavioural constructs shown in above. This was achieved by identifying the occurrence in the recorded analytic process data by considering the context of the situations that these behaviours were observed (i.e. before and after actions and conditions). Within the field of criminal intelligence, process data from the task interface allows for the collection of information that may be indicative of observable behaviours. So, the underlying research challenge is to convert such analytic process related data into behavioural constructs. Such as – *'fluency'*, specifically during the data-finding process, can be defined as the ability to generate many different pieces of data. Fluency in data finding is the indicative of a behavioural construct known as *'creativity'*. To detect the events, we aim to follow a compositionally reductive framework. This strategy helps to break down or reduce the events into simpler, more quantitatively manageable constructs. Ideally, these smaller components have a more directly observable set of markers for a certain analytic behaviour. To illustrate this concept of applying compositional reductionism to complex tasks, suppose we need to measure *'imagination'*. It can be considered in terms of creativity. Creativity in the literature can be approximated as *'divergent thinking'*, and researchers have attempted to measure divergent thinking through concepts such as *'fluency in data finding'* or *'flexibility unshifting between approach'* [93]. This idea of reducing complex construct into simpler, easier to measure constituent cognitive components can be conceivably applied to complex problem solving tasks. The reductionist approach gives an overview of *'Behavioural Markers (BMs)'* and their role for the scientists to recognize them when certain behaviours have occurred into analytic process data stream. We aim to test presence of such behaviours and their constructs

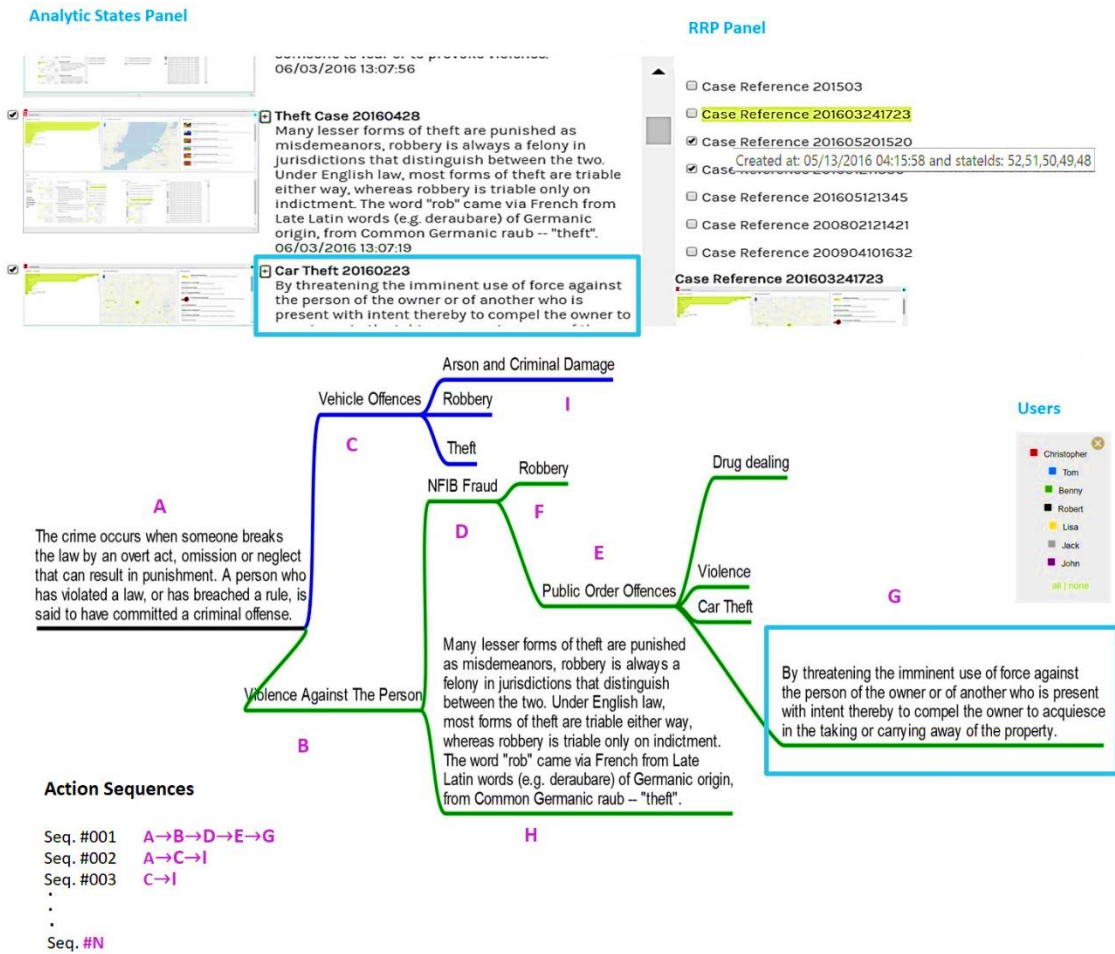


Figure 4.2: An analytic path showing annotations set by analysts with captured states and their relationships based on interactions with colour coded users (analysts) information. States can be selected from States Panel and RRP List of Analyst’s User Interface (AUI) to load analytic path for understanding intersections of analytical states captured by different analysts during their analysis process.

as identified into above through an automatic action sequence computation approach on captured analytic reasoning dataset.

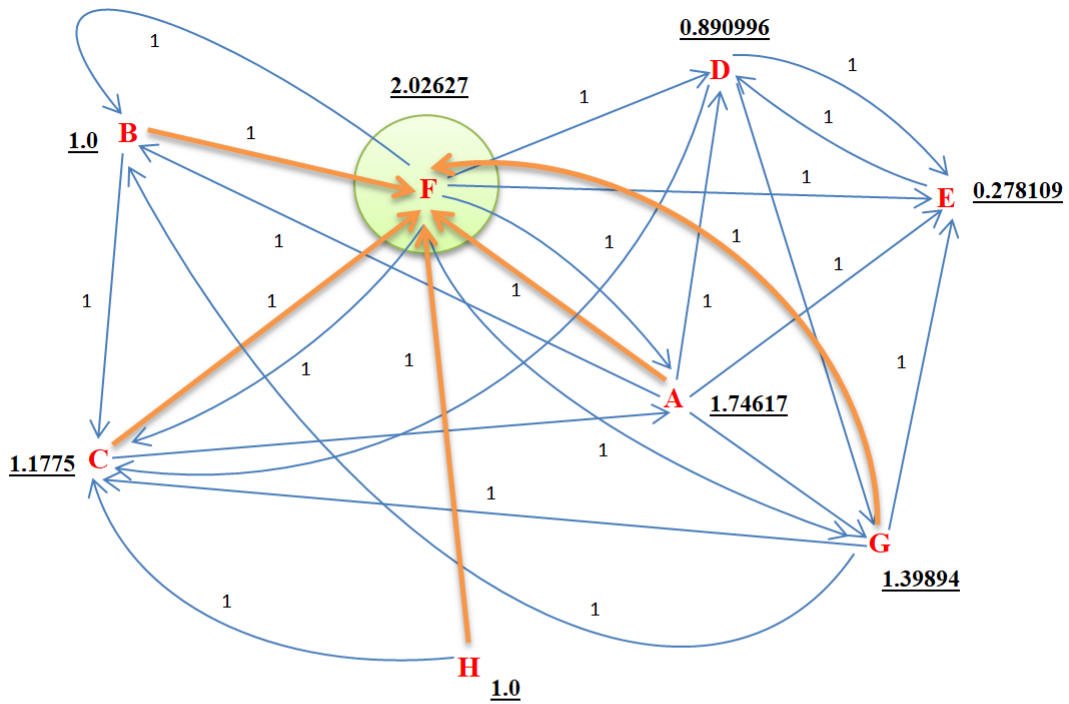
4.5.1.1 Action Sequence Computation

For recognizing ‘Behavioural Markers (BMs)’ within an automated framework, the streams of actions during analytic process can be meaningful markers for complex behaviours. Current approaches such as – finite state systems for fixed manipulable elements, a priori establishment of fixed sequences for clearly defined tasks,

exhausting all possible sequences for tasks with unpredictable human elements, are available for information computation about behavioural and cognitive processes and their implications for large scale complex analysis. An a priori approach is suitable for large scale data, but not suitable for complex tasks with human elements. In the 'exhaustive approach', number of sequences increase exponentially and very rapidly reaches infeasible levels. Bakeman et. al. [95] suggested that exploratory aspects of sequential analysis can provide empirical data that can ground later interpretations of observed behaviours because '*as we gain experience with the phenomena we are investigating, we learn which variables are important to us*'. The use of a network graph visualization in this context can be a useful exploratory process, rather than exhaustive method, to observe and gain understanding of the data, where empirical action combinations may provide meaningful sequence for targeted behavioural constructs. But the sequences need to be converted into a structure that is more suitable for network analysis and visualization. Some sequences might be observed more often while others are only observed in very rare occasions.

We developed an analytic visual state capturing, restoring and retracing prototype during our previous research study [94] on analytic provenance visualization for criminal intelligence as shown in Figure 3.2. The prototype shows captured analytic states with inserted annotations by the analysts into '*Analytic States Panel*' which are records of their reasoning provenance.

The RRP (*Repetitive Replicating Playback*) panel supports to create a composition of captured analytic logical states which can be applied back again on different other scenarios. All such captured visual analytic states can be replayed back again and visualized as a colour coded users' actions network known as "*Analytic Path*" to show analysts' higher level subtasks [10] through low level action sequences i.e., *Seq.#001* $A \rightarrow B \rightarrow D \rightarrow E \rightarrow G$, *Seq.#002* $A \rightarrow C \rightarrow I, \dots$, *Seq.#N* as shown in above. The incidence nodes of "*Analytic Path*" network can be computed by following our proposed compositionally reductive framework for the contextual information of complex analysis. To illustrate the idea – let's assume $P(S)$ is a semantic state composition function $P(S)$, where S is an analytic state. So, $P(S) = S$.



Action Sequence Graph $G = (V,E)$ where -

- $V =$ Nodes
- $E =$ Edges between pairs of nodes
 - **indegree $\text{deg}^-(V)$ = number of head ends adjacent to a vertex**
 $\text{deg}^-(V) = 5$
 - **outdegree $\text{deg}^+(V)$ = number of tail ends adjacent to a vertex**
 $\text{deg}^+(V) = 6$

Figure 4.3: Indegrees [$\text{deg}^-(V)$] of action sequence graph indicative of restoring previous analytic states.

For the *Seq. #001*, it can expressed as -

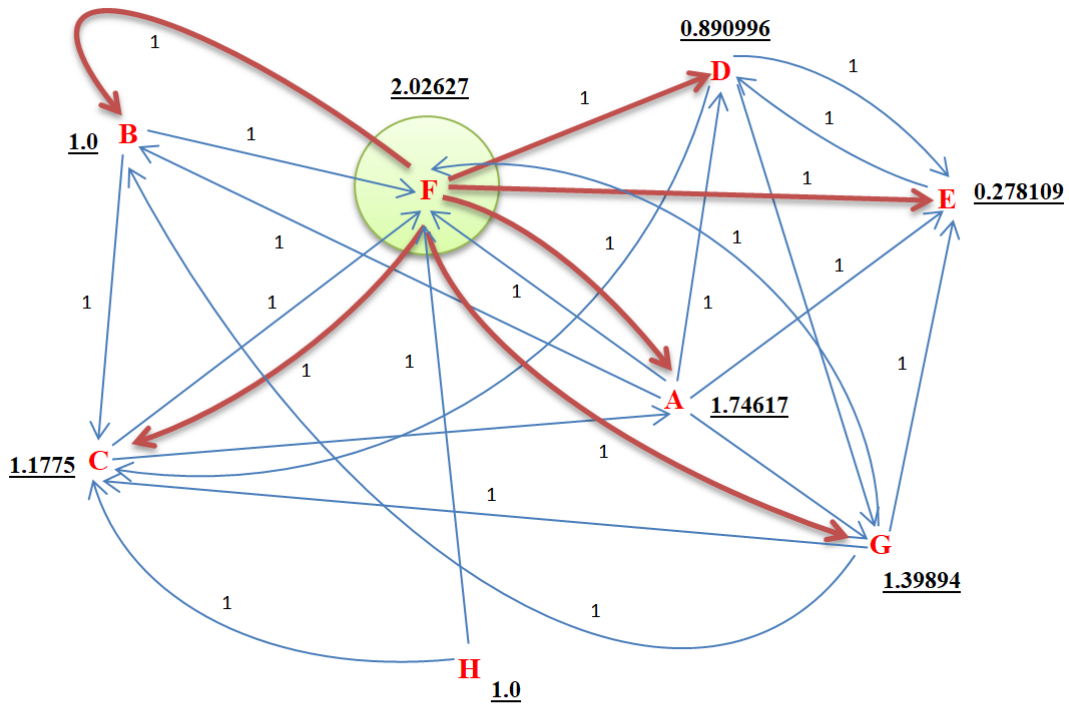
$$P(S_A) = S_A$$

$$P(S_B) = S_B$$

$$P(S_D) = S_D$$

.....

$$P(S_n) = S_n, \text{ where } n \text{ is the number of nodes.}$$



Action Sequence Graph $G = (V,E)$ where -

- $V =$ Nodes
- $E =$ Edges between pairs of nodes
 - indegree $\text{deg}^- (V) =$ number of head ends adjacent to a vertex
 $\text{deg}^- (V) = 5$
 - **outdegree $\text{deg}^+ (V) =$ number of tail ends adjacent to a vertex**
 $\text{deg}^+ (V) = 6$

Figure 4.4: Outdegrees [$\text{deg}^+(V)$] of action sequence graph indicative of generating more alternative approaches.

Thus we computed n th state S_n as $P: S_{A,B,D, \dots, n-1} \rightarrow S^n$. Composition function of different analytic states can be expressed as -

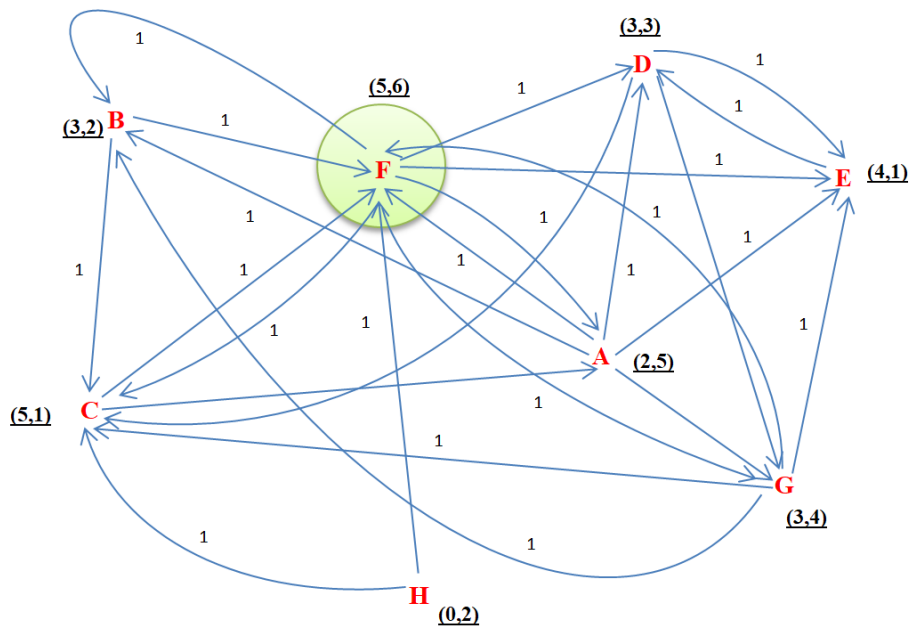
$$P(S_A) \circ P(S_B) = P \circ P(S_A, S_B) = \{S_A, S_B\} = S_{A,B} \quad P: S_A \rightarrow S_B$$

$$P(S_B) \circ P(S_D) = P \circ P(S_B, S_D) = \{S_B, S_D\} = S_{B,D} \quad P: S_B \rightarrow S_D$$

.....

$$P \circ P(S_{A,B,D, \dots, n-1}, S_n) = \{S_A, S_B, \dots, S^n\} \quad P: S_{A,B,D, \dots, n-1} \rightarrow S^n = S^{ST},$$

where S^{ST} is a Sub-Task State [10] through low level actions or events. All other low



Action Sequence Graph $G = (V,E)$ where -

Node (V)	(deg ⁻ V, deg ⁺ V)	Centrality
A	(2,5)	1.74617
B	(3,2)	1
C	(5,1)	1.1775
D	(3,3)	0.890996
E	(4,1)	0.278109
F	(5,6)	2.02627
G	(3,4)	1.39894
H	(0,2)	1

Figure 4.5: Calculating centrality or approximate importance of an action sequence graph.

level action sequences *Seq. #002, Seq. #003, ..., ..., Seq. #N* can be computed in the same way.

To determine which sequences are more valid measures of ‘Behavioural Markers (BMs)’, we consider our identified behavioural constructs of above and this would entail some form of network analysis; so each low level actions (representing an analytic state) can be defined as a ‘node’ and the links that make up a sequence across the nodes can be defined as ‘edges’. Eigenvector centrality is one method of computing the centrality or approximate importance of each node in a graph network. As shown in above, adjacency and centrality matrices for the action sequence graph have

been computed. The weight of each edge has been considered as '1' for the simplicity of this computation. The eigenvector of the adjacency matrix have been computed too such that all of its elements are positive and to identify the prevalent nodes for pathways of actions. **Node F** (above) shows higher importance and associated edges i.e, *indegree* $deg^-(V)$ and *outdegree* $deg^+(V)$ as shown in above and above respectively. It indicates that it has been taken more often and therefore may imply that the analysts are finding more sensible choices for shifting from one approach to another (*Flexibility*) or generating more alternative approaches (*Fluency*). Creativity is manifested through the flexibility, fluency and originality of responses to a task [96] which can be approximated as '*Divergent Thinking*' or alternately '*Imagination*'. This is how we can also calculate the '*Analytic Path*' of above to find out such cognitive constructs computationally.

The main challenges of recognizing such '*Behavioural Markers (BMs)*' within an automated framework include the limitation that a computer has no ability to make an expert judgement in the same way that a human can. For example, a human may be able to reflect, either through experience or intuition, on the thinking process of someone solving a task and be able to interpret correctly. Another important challenge lies in interpretation of interaction and analytic process data to extract markers of behaviours from them. For the reductionist approach, data reduction can be accomplished through coding and manual interpretation. This is extremely labour intensive and best for qualitative analysis. Direct observation through video, physical observation, participant interview, audio recording are needed.

4.5.2 Qualitative Approach

4.5.2.1 Methodology

We used AUI to study a group of police analysts to identify if the design principles used for the system encourages anchoring, laddering and associative questioning in fluidly which supports sense-making and insight generation. We assume that

improving these, will likely improve analysts' ability to solve crime or be better at performing criminal intelligence analysis.

The study was aimed to evaluate the performance of participants performing representative tasks using AUI components (*summative evaluation*) of the project *VALCRI and to elicit data that might inform further development (*formative evaluation*). The considered *independent variables* were - system with two levels, *VALCRI vs state-of-the-art, leading to two conditions, *VALCRI (experimental) and state-of-the-art (control). In the state-of-the-art condition participants were aimed to perform a task using the system that was representative of the current state-of-the-art in police work. The study used a repeated measures design (same subjects). Counterbalancing was set to be used to control for effects of condition order and task.

The *dependent variables* included imagination, insight, transparency, fluidity and rigour. The analysts were given a task such as '*...You are an analyst responsible for analyzing crime in the town of Tormington. You have been tasked with analyzing burglary dwelling offences in the town with a view to preventing future crimes of this type...*' to test dependent variables. Where possible, these were considered to measure the constructs objectively and subjectively (e.g., using participant self-reports). Where objective measurement was not possible, subjective measures were aimed to use.

4.5.2.2 Participants

For the study we recruited eight (n=8) police analysts who are the end users of the project *VALCRI. All participants were competent using computers for their daily work. There were four females and four males.

4.5.2.3 Procedure

AUI was loaded with an anonymized real crime dataset which was created for the project *VALCRI. We were interested in observing how senior police analysts would use the AUI to address realistic tasks similar to the once they are faced within their day to day operations.

Table 4.3: Description of behavioural constructs.

Categories	Behavioural Constructs	Description
Imagination	Curiosity	<ul style="list-style-type: none"> • A strong desire to know or learn something.
	Creative Play/ Playfulness	<ul style="list-style-type: none"> • Playfulness Cognitive spontaneity, joy, and a sense of humour in approach. • Play involving make-believe, an ‘as if’ stance, fantasy, and symbol substitution.
	Idea Generation	<ul style="list-style-type: none"> • Ideational flexibility - The number of themes or categories within an examinee’s or respondent’s ideation. • Ideational fluency - The total number of ideas given on any one divergent thinking exercise. • Ideational Originality - The unusualness or uniqueness of an examinee’s or respondent’s ideas.
	Creative Problem Solving	<ul style="list-style-type: none"> • Fluency in data finding or information retrieving (the number of different data or information generated); • Fluency in problem finding (the number of alternate problem statements produced); • Flexibility in problem finding (the number of categories created by the generated alternate problem statements); and • Quality of the problem statement (the complexity of each group’s final problem statement judged by two experts in terms of the degree to which the needs and motives of all those involved in the problematic situation including the owner, goal, and constraints identified in the final problem statement were satisfied)
Insight	Intuition	<ul style="list-style-type: none"> • ‘Affectively charged judgments that arise through rapid non-conscious and holistic associations’ [97]. • ‘A belief in something without evidence’ [98]. • ‘that reflects affective reactions’ [99].
	Leap of Faith	<ul style="list-style-type: none"> • ‘An interpretation of intuitive judgments that arise from experience consistent with perception of a current situation’ [P14] • ‘Sudden unexpected thoughts that solve problems’ [100]. • “An unexpected shift in the way we understand things” [46].

Table 4.3: Description of behavioural constructs (contd).

Categories	Behavioural Constructs	Description
Transparency	Anchor	<ul style="list-style-type: none"> • Anchors are key data elements that serve to create understandings that guide subsequent inquiry.
	Anchoring	<ul style="list-style-type: none"> • A specific understanding of a situation, given data, prior knowledge, general understanding of the world and the type of problem or crime, and the goals at the time. It provides the cognitive traction to enable reasoning to start. If one does not understand what the data means nor how the data might be created, one is not able to start. The analysts know that, and use assumptions to create plausible explanations to “pin” down, or anchor, no-data, ambiguous data or data that is unclear about how they fit in. The correctness and accuracy of these frames can be corrected or modified later when more is known about the situation [P17].
	Associative Questioning	<ul style="list-style-type: none"> • Asking of questions that attempt to discover what other information or knowledge that may or may not directly relate to the subject, but may lead to interesting insights P[18].

Participants were offered a training session of about 2-3 hours long before the (two) days of main evaluation sessions. At the end of the training session participants were given time to familiarize themselves with the shown functionality of the system.

Each participant performed the task. We had three analysts at each session in both days. Each analyst was in a room with a facilitator and an observer for note taking. Throughout the study participants were encouraged to think-aloud as a way to understand what a participant was thinking, observing, and doing to help trace the participants decision-making process. Participants notified the facilitator when they commenced and ended the task. At the end of each task, a semi-structured interview took place. A participant session lasted about 2 hours.

Questionnaires were developed to assess each of the constructs of above through participant self-reports, elicited post-task. It was intended that these questionnaires

will incorporate more task/domain specific sub-constructs (e.g. ‘inferring a modus operandi’ as a type of insight).

4.5.2.4 Data collection

Participants were notified of the study procedure, and gave consent for video and audio recording. *Multiple Cognitive Task Analysis* (CTA) methods were used to extract and understand the participants’ decision process during the tasks. Methods such as think-aloud elicitation during the task with full resolution video capture of the screen (video capture), user observation (video capture and field notes), semi- structured interviews (video play-back and review of field notes), and questionnaires were used as data collection methods.

4.5.2.5 Study setup

Two attached display monitors (vertically and horizontally), keyboard and mouse were connected and the AUI software was loaded into it. Participants were provided with A4 sheets, pen, pencils and MS Office to use if needed.

4.5.2.6 Assessment Method

The objective of the assessment phase was to inform selection of the most effective behavioural constructs in criminal intelligence. The set of behavioural constructs as shown in above is too large for a usable checklist and so a structured method of selecting most relevant and usable markers for inclusion was suggested. Effective constructs of behaviour are clear concepts which are described simply and related to task performance. The behaviour can be measured as a frequency (the absence or presence of the marker) or on a scale.

Simple three-point scales (for example observed, not observed, not applicable) are often used on ‘*Behavioural Marker (BM)*’ checklists in order to improve the clarity of the concept and to ensure reliability between different assessors (Fletcher et al., 2001). In our study we considered two axiomatic dimensions to inform selection: the *detectability* of the behaviour through AUI analytical tasks and the *relevance* of the behaviour to AUI performance. The most relevant cognitive criteria was

Table 4.4: Observed analysis techniques followed by crime analysts.

Types	Observed Analysis Techniques
Filtration	Analysts remove extraneous data to focus on what’s important. This applies to individual pieces of data (an analyst may have to filter out dozens of paragraphs from an officers’ narrative to find the information important to the analyst) and to large data sets (an analyst searching for patterns of nighttime burglary filters out daytime incidents and non-burglaries).
Categorization	Analysts categorize, classify, or cluster pieces of data into logical groups—robbery or burglary, purse snatching or carjacking, Beat 4 or Beat 5, offender or victim—to help identify, analyze, and communicate information. Some categories are obvious, such as the year (2008 or 2009); others require more intellectual effort, such as when an analyst creates situational types (domestic assault, gang assault, or road rage assault).
Aggregation	Analysts count, summarize, average, or otherwise aggregate data into categories. Examples of aggregation include taking 20,000 police calls for service and showing the number during each shift; arranging 365 auto thefts by counting the make and model of cars stolen; and showing the average dollar value stolen by crime type for 16,500 property crimes.
Comparison	Analysts may compare individual incidents to determine if they are related. They may also compare large data sets (e.g., 2008 crimes vs. 2009 crimes, crimes in Birmingham vs. crimes in London, Beat 4 calls vs. Beat 3 calls) to determine trends and deviations from the norm. Most crime statistics are meaningless unless compared to previous time periods or other geographic areas.
Correlation	The term “correlation” is sometimes used informally to denote any observed relationship between two variables (e.g., robberies seem to cluster around subway stops, burglaries decline during school hours.)
Causality & Explanation	This process takes correlation a step further by determining whether one factor causes another. Did the new shopping mall “cause” auto burglaries to increase in the area? Have increases in the price of heroin “caused” an increase in burglary?
Projection	Analysts can use existing data to project or predict the future. If we’ve already had 19 robberies this month, how many are we likely to have by the end of the month? If the offender’s activities continue as they have in the past, where and when is he likely to strike next?

explained in the ‘*development approach*’ section 4.4 and classified criteria according to †above, n.d. [31] as shown into above. But for this study, we present the detection strategies of those relevant cognitive constructs into above.

4.5.2.7 Results

We considered intelligence analysis as a fluid activity involving transitions between mental and interaction states through analytic processes. The analytic processes that analysts employ are the indicatives of ‘*Behavioural Markers (BMs)*’ through techniques of analysis they follow. Few of such analysis techniques that we observed are described into above.

We observed the participant **P3** at experiment station **ES3** started with gaining an overview of the available data. “... *what’s going on within data range?...*” She filtered the data according to provided date range and tried to focus on. This was similar to the *OFD (Overview-Filter-Detail)* strategy identified by Kang et al. [5]. We observed her ‘**Questioning**’ about data while performing ‘**Comparisons**’ of time chart data. She was performing visual comparison of information by using Index Cards, Map etc while trying to understand the crime situation. “*I don’t know*”, we found her feeling ‘**Uncertain**’ about the scenario while she was adopting ‘**Associative Thinking**’ and trying to find alternate data. At a certain point she was trying find out ‘**Similarities**’ among MO. She was trying to generate hypotheses by ‘**Self Questioning**’. She was trying to ‘**Build Scenario**’ of the problem by using CCT, S3, Map and Bar Chart views. She tried to understand clustering but failed to grasp it. She was struggling find out a clue but couldn’t succeed.

We observed another participant **P4** at experiment station **ES4**. We found him saying “...*too many information. Only the significant information will be helpful*”. However, we found ‘**Free Thinking**’ happening while he was exploring the data.

AT experiment station **ES4**, we observed another participant **P6** in the afternoon session. He was also found ‘**Exploring**’ and ‘**Free Thinking**’ while he started his analysis. We found that he had curiosity on ‘**Alternative Options**’. He was trying to

'Build Scenario' by using plausible suspects list for unsolved crimes. He was struggling because of too much information and got lost.

Participant **P7** at experiment station **ES1** has been observed in the following day. She followed the strategy *HTK (Hit the Keyword)* as identified by Kang et al. [5]. She went through the problem document and circled keywords. She was **'Thinking of Comparing'** solved vs unsolved crimes. We observed **'Analytical Thinking'** during her task as she was looking for where the large range of crimes occurred. She was also trying to **'Find out Correlations'** between holidays' data. "...It's good to know for the police officers either houses/flats are prevailing...." – such **'Idea Generation'** has also been observed while she was performing her analysis by using CCT and S3. We found her **'Curious'** to look for where the holidays occurred and she was trying to **'Anchor'** while using those CCT and S3 views.

We aimed to detect transitions between mental states depicted by interacted states of visualizations through analytic processes. To be more precise we gathered analyst's subjective feedbacks i.e, positive(+)/negative(-) and analyzed those to find out presence of **'Behavioural Markers (BMs)'**. Several of the results indicate different BMs as followed:

- **P7 at ES1:** '(+) *It works fast and gives you the impression of giving you the right results'* but '(-) *you have to go for a new search when you are looking for a special period. You cannot pick a special period meaning the period before the offender was caught'*.

Participant **P1** was **'Curious'** about knowing crime situation before the offender was caught and clustering solved/unsolved crimes to understand what has lead the situation.

- **P3 at ES3:** '(+) *Clean and simple design, flow of queries through all views on your screen, introduces new ways of thinking about data'* but '(-) *need to be able to analyze by time of day, want to choose offences as a group from the map'*.

Participant **P3** reported that the system generated number of different data or information which means the supportive is in **'Fluency'** of data finding or

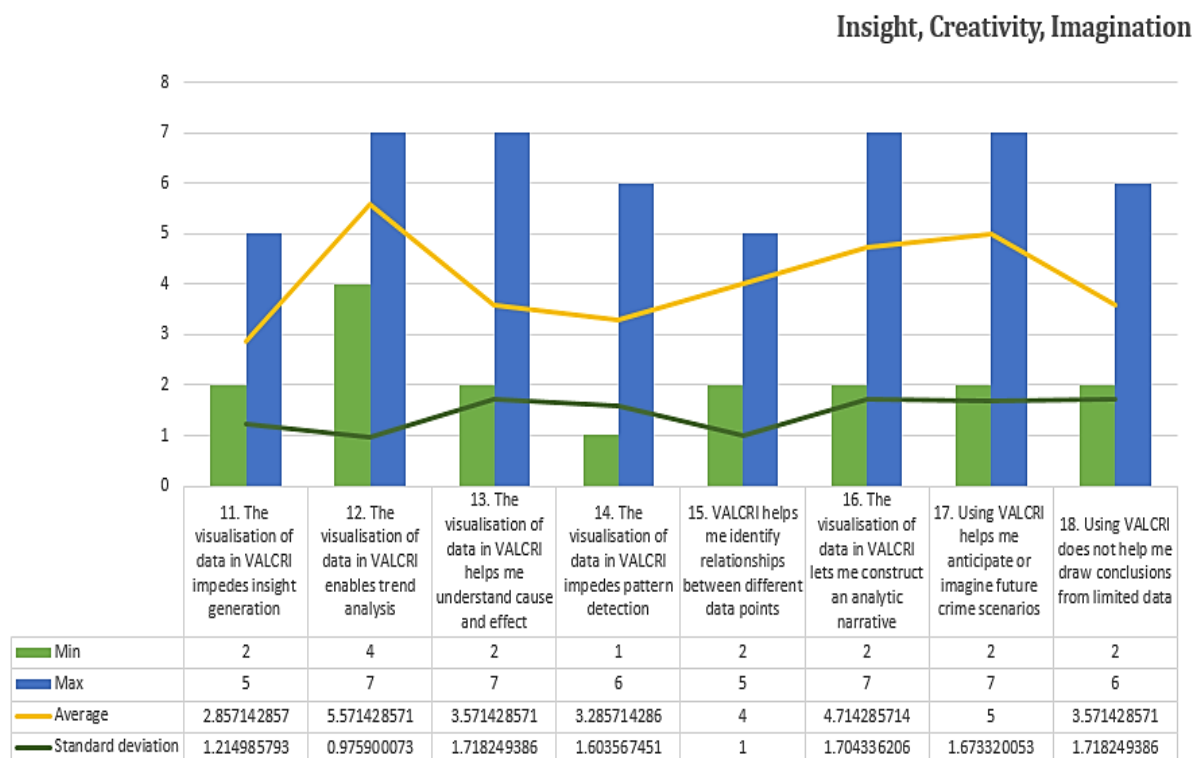


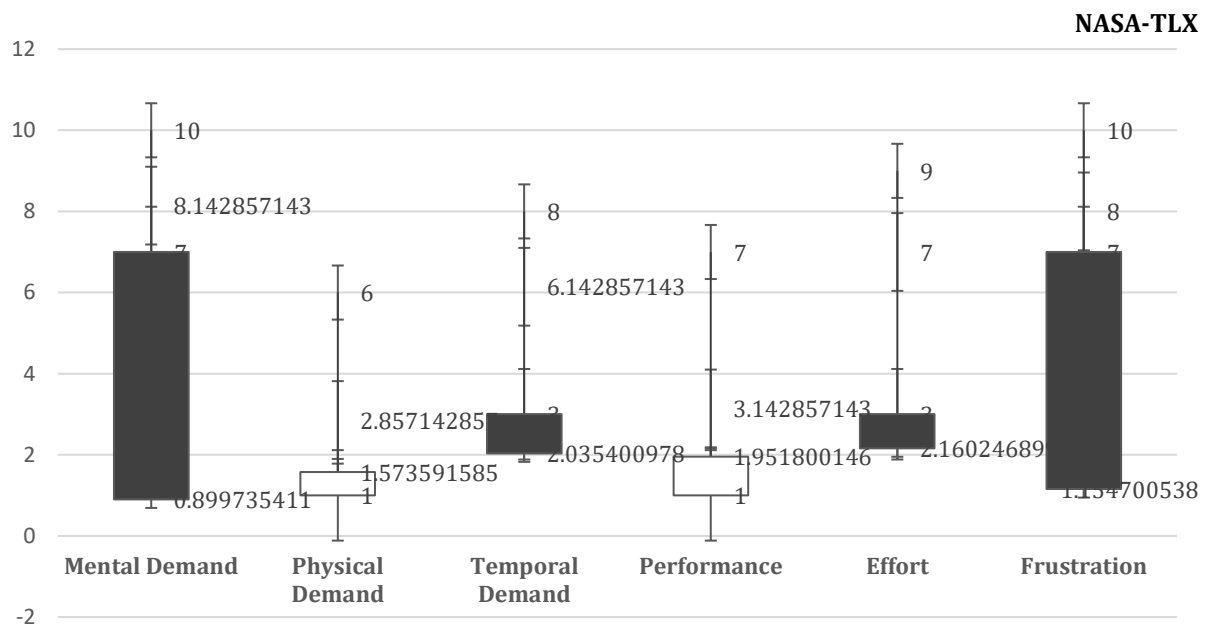
Figure 4.6: VALCRI’s phase-1 system evaluation: User feedback approach based on open-ended questionnaire to identify how it’s AUI system encourages or hinders insight, creativity and imagination.

information retrieving as part of *‘Creative Problem Solving’* and was looking for *‘Ideational Flexibility’* by trying to aggregate those data by time of day etc.

- **P5 at ES5:** *‘(+)* Almost direct result from a question, interactivity between time and space’ but *‘(-)* difficult to switch between offender/victim, sometimes got confused with selections’.

Participant **P5**’s difficulty to find more sensible choices for shifting from one approach to another and generating more alternative results demands the *‘Flexibility’* during analytical approaches.

Questionnaires were developed to assess each of the behavioural constructs of Tabove through participant’s self-reports (subjectively), elicited post-task and interview with the aim that these will assess those constructs in generic terms. Each question was scaled between (*strongly disagree*) 1<->7 (*strongly agree*). As shown in above, we have found during the phase-1 evaluation of VALCRI’s AUI system that most



	Mental Demand	Physical Demand	Temporal Demand	Performance	Effort	Frustration
Min	7	1	3	1	3	7
Max	10	6	8	7	9	10
Average	8.142857143	2.857142857	6.142857143	3.142857143	7	8
Standard deviation	0.899735411	1.573591585	2.035400978	1.951800146	2.160246899	1.154700538

Figure 4.7: VALCRI’s phase-1 system evaluation: NASA-TLX Mental Workload Rating Scale.

of the participants’ agreement ratings, $3.29 \leq r \leq 5.00$ with an overall average value of 4.07. The overall average min value for all delivered questionnaires for participants was 2.12 and max value 6.25 with the standard deviation, $\sigma = 1.45$.

Taking into consideration, we have understood through this experiment that constructs of analyst’s cognitive activities which we call ‘*Behavioural Markers (BMs)*’ are ingrained into their analytic activities. As described above, we observed reasonable number of constructs of those BMs while participants were working on their tasks at different experiment stations (ESs).

We also measured mental workload at NASA-TLX rating scale in order to assess effectiveness and other aspects of performance of VALCRI’s AUI. As shown in above,

following subjective subscales as described by Rahman et. al. [101] were considered while serving questionnaires to all participants:

- **Mental Demand (MD):** How much mental and perceptual activity was required (e.g. thinking deciding, calculating, remembering, looking, searching, etc)? Was the mission easy or demanding, simple or complex, exacting or forgiving?
- **Physical Demand (PD):** How much physical activity was required (e.g., pushing, pulling, turning, controlling activating, etc.)? Was the mission easy or demanding, slow or brisk, slack or strenuous, restful or laborious?
- **Temporal Demand (TD):** How much time pressure did you feel due to the rate or place at which the mission occurred? Was the pace slow and leisurely or rapid and frantic?
- **Performance (PF):** How successful do you think you were in accomplishing the goals of the mission? How satisfied were you with your performance in accomplishing these goals?
- **Effort (EF):** How hard did you have to work (mentally and physically) to accomplish your level of performance?
- **Frustration (FR):** How discouraged, stressed, irritated, and annoyed versus gratified, relaxed, content, and complacent did you feel during your mission?

We have found by using above subscales that participants' average mental demand rating MD=8.14 which is reasonably high. This suggests that **fluency in data finding** equates to **ideational fluency**. The lowest standard deviation SD=0.9 shows strong acceptance of all participants regarding MD ratings. However, the average performance rating PF=3.14 makes it clear that gaining an **insight** or reaching at stage of decision making was not as successful as expected despite of higher effort EF=7. So, to improve performance we also gathered data for positive(+) and negative(-) aspects of AUI alongside our observations and recommendations from all participants for further development.

4.6 Discussion

This research aims to explain how human cognition leads to interactions and vice versa to achieve certain goal. Recovering cognitive reflection on analytic reasoning processes from extended log data or only by observing is a difficult task. For example, knowing when one reasoning process ends and another begins may be unclear from a sequence of interactions alone. We call it '*Cognitive Steps Sequencing Problem*'. Endert et al. [38] contend that a new methodology to couple these cognitive and computational components of visual analytic system is necessary. During our previous stage of research as described into Section 3.3.5.1 and Section 3.3.4.2, we proposed analytic state composition and schematization techniques to tackle this problem with captured analytic data. At this stage of research we have showed that analysts' cognitive and adopted analysis steps can be bridged by using their captured analytic reasoning data. For this, we have considered markers of analyst's cognitive behaviours (known as *Behavioural Markers*) as attributes for bridging human cognition and analytic computation through interactions. To detect these '*Behavioural Markers (BMs)*' from captured analytic data, we have proposed a computational technique known as "*Compositional Reductionism*". Such technique provides a simple solution to overcome tedious effort of qualitative approach for detecting analyst's cognitive aspects from sequential actions into log data. Although computational technique is an automatic approach, it still lacks ability of making an expert judgement in the same way that a human can. For this reason, we also adopted a qualitative approach to detect analyst's '*Observable Behaviours*'. For this purpose '*Cognitive Task Analysis (CTA)*' method was used to extract and understand the participants' decision process during the tasks. Methods such as think-aloud elicitation during the task with full resolution video capture of the screen (video capture), user observation (video capture and field notes), semi-structured interviews (video play-back and review of field notes), and questionnaires were used

as data collection methods. We extracted a good number of observable BMs through this CTA study. We have found most of the participants' agreement ratings, $3.29 \leq r \leq 5.00$ with an overall average value of 4.07 where overall average min value for all delivered questionnaires for participants was 2.12 and max value 6.25 with the standard deviation, $\sigma = 1.45$. We also have found that participants' average mental demand rating MD=8.14 which is reasonably high. This is explicitly indication of '*fluency in data finding*' resulting to '*ideational fluency*'. The lowest standard deviation SD=0.9 shows strong acceptance of all participants regarding MD ratings. However, the average performance rating PF=3.14 makes it clear that gaining an '*insight*' or reaching at stage of decision making was not as successful as expected despite of higher effort EF=7.

To measure at what extent a system like AUI can enhance '*insight*', '*creativity*' and '*imagination*', following methodology can be adopted by using means-ends hierarchy as shown in Figure 4.1 but leaving this as future work.

Measuring Insight:

SCALE: % of Insights gained by demonstrating instances of [Concepts = {Intuition, Leap of Faith}], [Sub-Concepts = {Associative Questioning}] and [Components = {Pattern Recognition, Comprehension, Understanding}]

Measuring Intuition:

SCALE: % of instances where Intuition was gained by demonstrating instances of [Sub-Concept = {Anchoring}]

Measuring Anchoring:

SCALE: % of instances where Anchoring was made possible through the use of [Components = {Situation Awareness, Narrative}]

Measuring Situation Awareness:

SCALE: % of instances where Situation Awareness was demonstrated through the use of [Strategies = {Analytical Approach, Profiling }]

Measuring Analytical Approach:

SCALE: % of tasks that where the Analyst incorporated [Approach = {Overview, Detail on Demand}] as a starting point or method

Measuring Profiling:

SCALE: % of tasks where the Analyst incorporated [Approach = {Statistical Metrics}] as a starting point or method.



Sensemaking Task Inference



5



chapter

5.1 Chapter Overview

By considering the shortcomings of adopting a qualitative approach to detect 'Behavioural Markers (BMs)', we hypothesize in the current chapter that ***inferring chains of low-level analytic actions can be of assistance for understanding multi-tasking behaviour***. This will lead us to understand what the user is trying to do. The scope of this chapter include - understanding how conventional techniques (currently used in many systems) perform to infer user's tasks, conducting experiments to demonstrate and improving those. We have run experiments with different machine learning techniques to improve inference making results both into known and unknown scenarios. The aim of running those experiments are to test the above hypothesis by considering following research questions:

RQ6: How can meaningful units of task execution be produced from captured interaction logs?

- RQ6 addresses the fundamental problem of finding out the way of breaking down a search session into meaningful chunks to detect user's task switch points. We have discussed how can those search sessions be chunked according to contextual and hierarchical levels. We have also shown how those action chunks be utilized for an unknown scenario which is a bit complex problem.

RQ7: How precisely multi-task switches be inferred during execution of interactive tasks?

- RQ7 targets to prove/disprove the above hypothesis and evaluate the results. We have contributed to contextualize user's search session to better understand user's intention by inferring their task switch points which is not mostly considered by conventional search engines. We have developed machine learning models to find out how precisely this can be accomplished. We also have shown how can those models be tuned on user's search dataset for improving results both in known and unknown scenarios. Few visualizations have been developed to show semantic information of those action chunks and internal model operations for inference making by using trained semantic and contextualized dataset.

Details on machine learning models and produced results of this chapter can be retrieved from-

<https://github.com/Vis4Sense/ProvenanceLearning/tree/master/ML%20Algorithms/J>

5.2 Introduction

Identifying when users switch tasks involves detailed analysis of human multitasking behaviour. The process helps to create insight into individual people. We observed during our previous CTA study (which detected analytical behavioural markers (BMs) [102] of criminal intelligence analyst participants) that when an interesting insight was identified, analysts got engaged in an insight phase during which they either documented their discovery, by taking notes or capturing a visualization snapshot, or did both. Such cues can be utilized to punctuate the sequence of user performed actions, marking important semantic boundaries in the recorded history of user activities. But it is much difficult in case of an unstructured and unannotated sequential list as that does not contain enough structure to infer the analytical activity. Dragunov et al. [103] found through their *TaskTracer* system study that a computer assistant can infer information about users' tasks and goals which they achieved by analyzing the context in which the user performs one action or another. One approach is to use machine learning to learn user actions and predict the likely future ones as proposed by Keim et. al. [104].

Due to enormous growth of captured interaction events as the analysis unfolds, it is extremely difficult to infer chain of actions and organize those into semantically meaningful segments of activities. Punctuating the sequence of performed actions, marking important semantic boundaries in the recorded history of user activities, is a challenge as it may be unclear from the sequence of interactions when one reasoning process ends and another begins [105]. To distinguish, it is very important to understand what the users are trying to search and why are they searching? If a search engine knew it, then that could provide users with a better experience that is tailored to their goals [106]. But analytical processes are not a simple sequence of logical choices leading inexorably to a goal. Instead, the process involves exploratory analysis where analysts try a range of options and assess which is the most successful and backtracking when results show that a particular line of inquiry is fruitless [107]. So, the *'why'* of user search behaviour is actually essential to satisfying the user's information need. *'How do combinations of multiple actions signal to accomplishment*

of user's higher-level sub-tasks? , *'How to find out the actions that form meaningful chains of user's individual insights?'* – these are questions that can be considered as a first step of research to understand user's search behaviour and marking important semantic boundaries into recorded history of user activities. To delve into this research further the following issues can be considered:

1. *'How to classify the user activity types?'* - such as online shopping, travel planning, deciding which university to attend, and socialising etc. There are many existing classification methods can be applied, and the challenge is to:
 - i. Collect and label training/learning samples;
 - ii. Select and create the *features* that will be used as input for classifiers.
2. A more fundamental problem is finding out – *'how to break a search session into meaningful chunks'*. Each session is essentially a sequence of URLs that a user visits, and some of the URLs are more relevant to the task than rest of the sequence. For example, during an online shopping session, the users may check their emails or social networks from time to time. As a result, the parts corresponding to emails or social networking activities are 'chunks' that are different from those for the shopping.
 - i. *Chunking is context dependent* - In the same shopping example, if all the URLs are about shopping, then the 'chunking' may be the different stages of the shopping, such as the product research stage, the product comparison stage, and the best-price hunting stage. Each of these stage can be a 'chunk' itself.
 - ii. *Chunking is also hierarchical* - still with the shopping example, the first level of chunking may be between shopping vs. non-shopping (such as checking email) activities. The second level can be different stages of shopping (for the shopping part) and the different types of social activity (for the non-shopping part). This can keep going, for example, the 'product research' stage of the shopping activities can be further

broken down into 'reading online review', 'check forum discussion', and so on.

- iii. *Chunking can be binary* - Not considering the hierarchy, chunking can be treated as a binary classification problem, i.e., whether there should be a break after each step in a search session. This has the same training sample and feature engineering issue as those for the activity classification problem.

As in the task tier, sub-tasks [10] are often tightly coupled to the domain or application in which the user is working. For example, a task is very appropriate for an investment analyst working with financial tools, a travel agent (working with a set of travel and transportation tools) would not likely perform the same task. For this, domain independent (e.g., independent of the tool-specific sequence of clicks, drags, and key-press events required to perform a specific action) user’s behaviour in terms of analytic actions needs to be modelled. So, we hypothesize that – ‘*Inferring chains of low-level analytic actions can be of assistance for understanding user’s multitasking behaviour*’.

5.3 Approach and Experiments

5.3.1 Experiment 1

To understand how machine can perceive user’s intent by considering their interaction preferences and finding out relevance, we have applied conventional data filtering techniques on a ²*Deskdrop* log ³dataset of 12 months developed by ¹*CI&T* focused in companies using Google G Suite.

5.3.1.1 Dataset

The ³dataset contains about 73k logged users interactions on more than 3k public articles shared in the platform. The dataset includes following features:

1. <https://ciandt.com/us/en-us>
2. <https://deskdrop.co/>
3. <https://github.com/yunshuipiao/sw-kaggle/tree/master/recommend-system/datasets>

- **Item attributes:** Articles' original URL, title, and content plain text are available in two languages (English and Portuguese).
- **Contextual information:** Context of the users visits, like date/time, client (mobile native app / browser) and geolocation.
- **Logged users:** All users are required to login in the platform, providing a long-term tracking of users preferences (not depending on cookies in devices).
- **Rich implicit feedback:** Different interaction types were logged, making it possible to infer the user's level of interest in the articles (eg., comments > likes > views).
- **Multi-platform:** Users interactions were tracked in different platforms (web browsers and mobile native apps).

5.3.1.2 Pre-processing

In above, users are allowed to view an article many times, and interact with them in different ways (eg. like or comment). Different weights have been associated for different types of interactions to assume the interest of a user on a specific article. Thus, to model the user interest on a given article, we aggregate all the interactions the user has performed in an item by a weighted sum of interaction type strength (i.e, *VIEW*: 1.0,....., *BOOKMARK*: 2.5,....., etc) and apply a log transformation to smooth the distribution. To avoid *user cold-start problem*, in which it is hard to model user a profile due to none or less available data, we kept only users in the dataset with atleast 5 interactions.

5.3.1.3 Evaluation

We have used the *holdout approach* for *cross-validation*, in which 20% random sample are kept aside during the training process and used for evaluation. We also have chosen to use *Top-N accuracy metrics*, which evaluates the accuracy of the top filtered data provided to a user, comparing to the items the user has actually interacted in test set. This evaluation method works as follows:

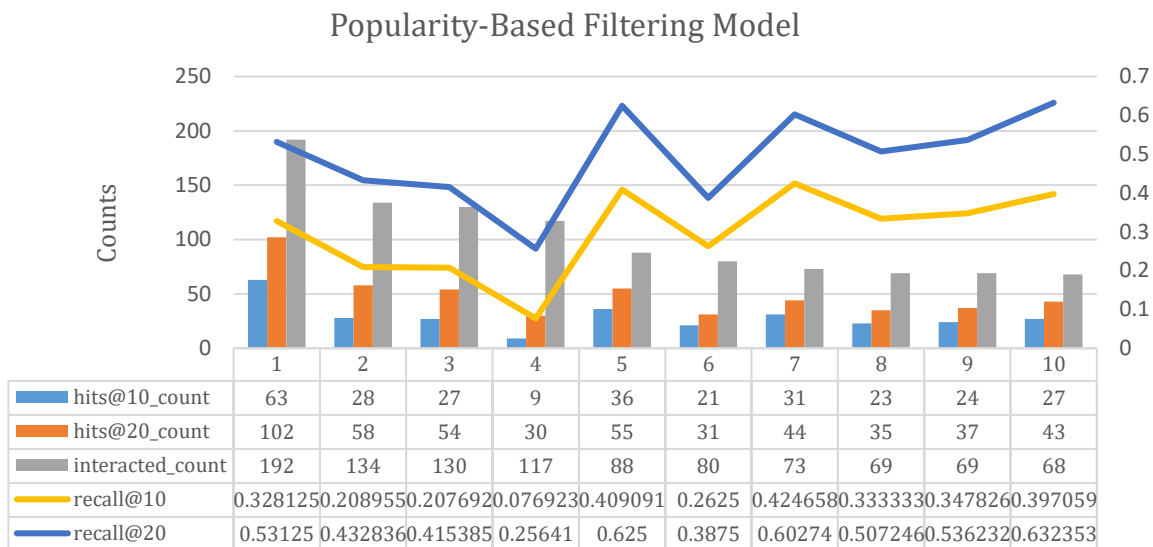


Figure 5.1: Evaluation results of Popularity Model on first 10 out of 1139 users having global metrics - recall@10: 0.39128211683497377 and recall@20: 0.5739486130640419.

- For each user
 - For each item the user has interacted in test set
 - Sample 100 other items the user has never interacted.
 - Ask the model to produce a ranked list of filtered items, from a set of one interacted item and 100 non-interacted items
 - Compute the *Top-N accuracy metrics* for this user and interacted item from the ranked list
- Aggregate the global *Top-N accuracy metrics*

The *Top-N* accuracy metric chosen is *Recall@N* which evaluates whether the interacted item is among the top *N* items in the ranked list of 101 filtered data for the user. We have then tested with following conventional data filtering techniques:

Popularity-Based Filtering Model

This model is not a personalized technique often known as ‘*wisdom of crowds*’. It considers the most popular items that the user has not previously consumed. Total weighted values of *eventTypes* corresponding to each content have been calculated

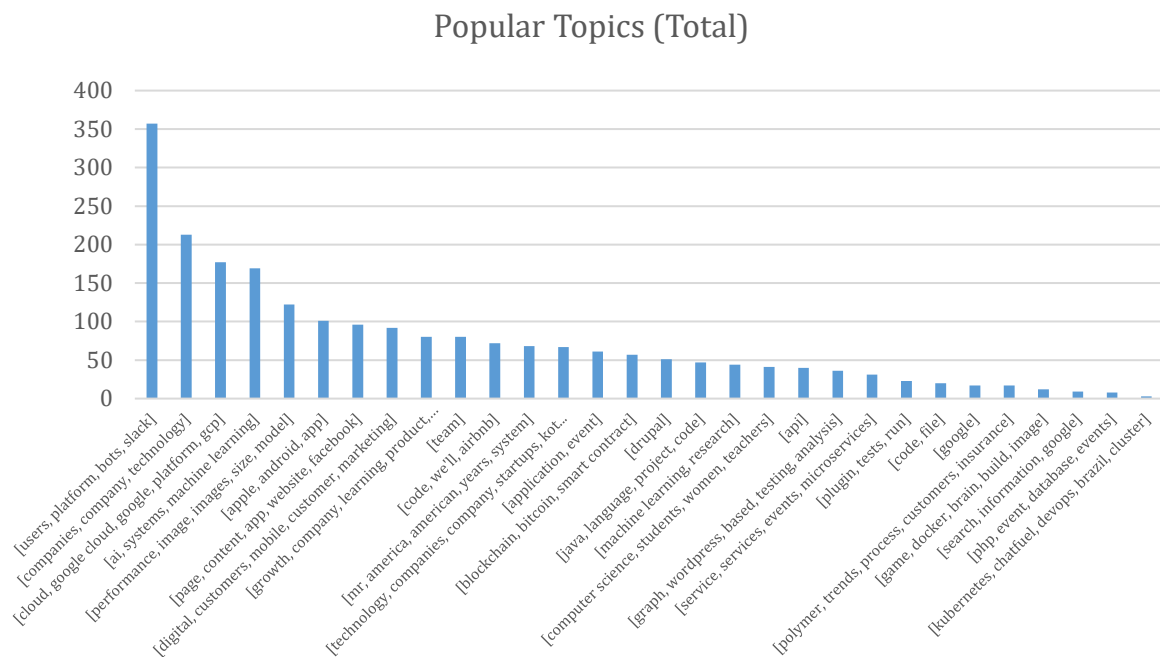


Figure 5.2: Popular Topics.

here. It achieved the $Recall@10$ of 0.39 which stands for 39% of interacted items in test set were ranked by popularity model among the top-10 items. For $Recall@20$ was 0.57 higher than the previous (above).

Content-Based Filtering Model

Content-based filtering approach leverages description or attributes from items the user has interacted to filter similar items. It depends only on the user's previous choices, making this method robust to avoid the *cold-start* problem. For textual items, like articles, news and books, it is simple to use the raw text to build item profiles and user profiles. We have used a very popular technique in *information retrieval* (search engines) named Term Frequency – Inverse Document Frequency ($TF-IDF$). An example of calculated term frequencies is shown in above. This technique converts unstructured text into a *vector structure*, where each word is represented by a position in the vector, and the value measures how relevant a given word is for an article. As all items will be represented in the same *Vector Space Model*,

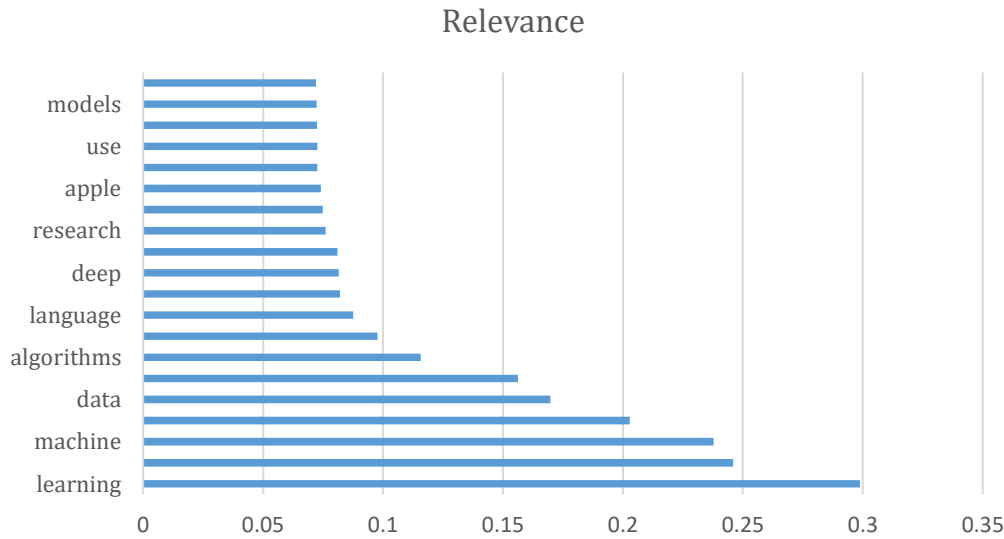


Figure 5.3: Relevance measure of top 10 tokens for user profile [-1479311724257856983].

it is to compute *cosine similarity* between articles. To model the *user profile*, we have taken all the *item profiles* the user has interacted and averaged them. The average is weighted by the interaction strength, in other words, the articles the user has interacted the most (eg., viewed or bookmarked) will have a higher strength in the final user profile. For an example – we chose a user profile [-1479311724257856983] to compute relevance (above) of each token (unigram or bigram) and know user’s interests.

With this personalized approach of content-based filtering model, we have achieved *Recall@10* to about 0.26, which means that about 26% of interacted items in the test set were ranked by this model among the *top-10* items (from lists with 100 random items). And *Recall@20* is about 0.40 (40%). The lower performance of the *Content-Based* model compared to the *Popularity* model may indicate that users were not too fixated of investigating similar contents while exploring different articles.

Collaborative Filtering Model

This model makes prediction based on preferences of many other users. *Collaborative Filtering (CF)* has two main implementation strategies:

Memory-based: This approach uses the memory of previous user's interactions to compute users similarities based on items they've interacted (*user-based approach*)

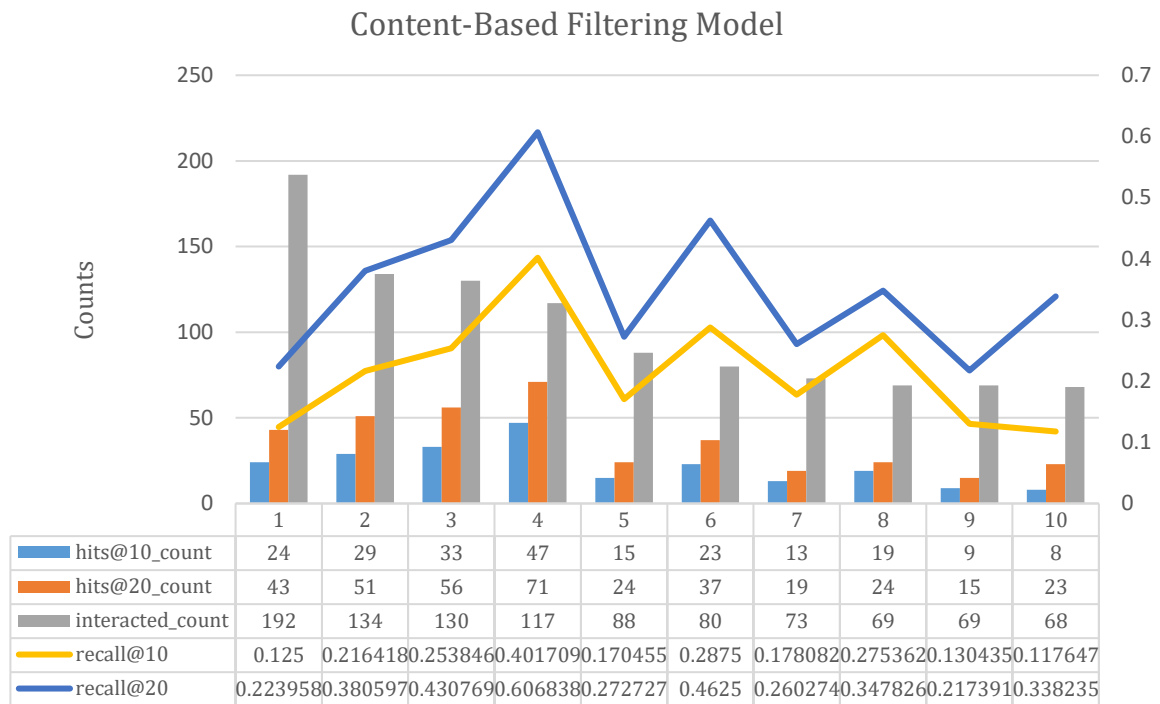


Figure 5.4: Evaluation results of Content-Based Filtering Model on first 10 out of 1139 users having global metrics - recall@10: 0.2614420864229097, recall@20: 0.3975965226284838.

or compute items similarities based on the users that have interacted with them (*item-based approach*). A typical example of this approach is - *User Neighbourhood-based CF*, in which the *top-N* similar users (usually computed using *Pearson correlation*) for a user are selected and used to filter items those similar users liked, but the current user have not interacted yet. This approach may not scale well for many other users.

Model-based: In this approach, models are developed using different machine learning algorithms to filter relevant data for users. There are many *model-based CF* algorithms i.e., *Neural Networks (NN)*, *Bayesian Networks (BN)*, *Clustering Models*, and *Latent Factor (LF)* models such as *Singular Value Decomposition (SVD)* and, *Probabilistic Latent (PL)* semantic analysis.

Latent Factor (LF) model for Collaborative Filtering (CF)

We have used a LF model named *Singular Value Decomposition (SVD)* for this section to evaluate CF technique for data filtering. LF models compress *user-item* matrix into

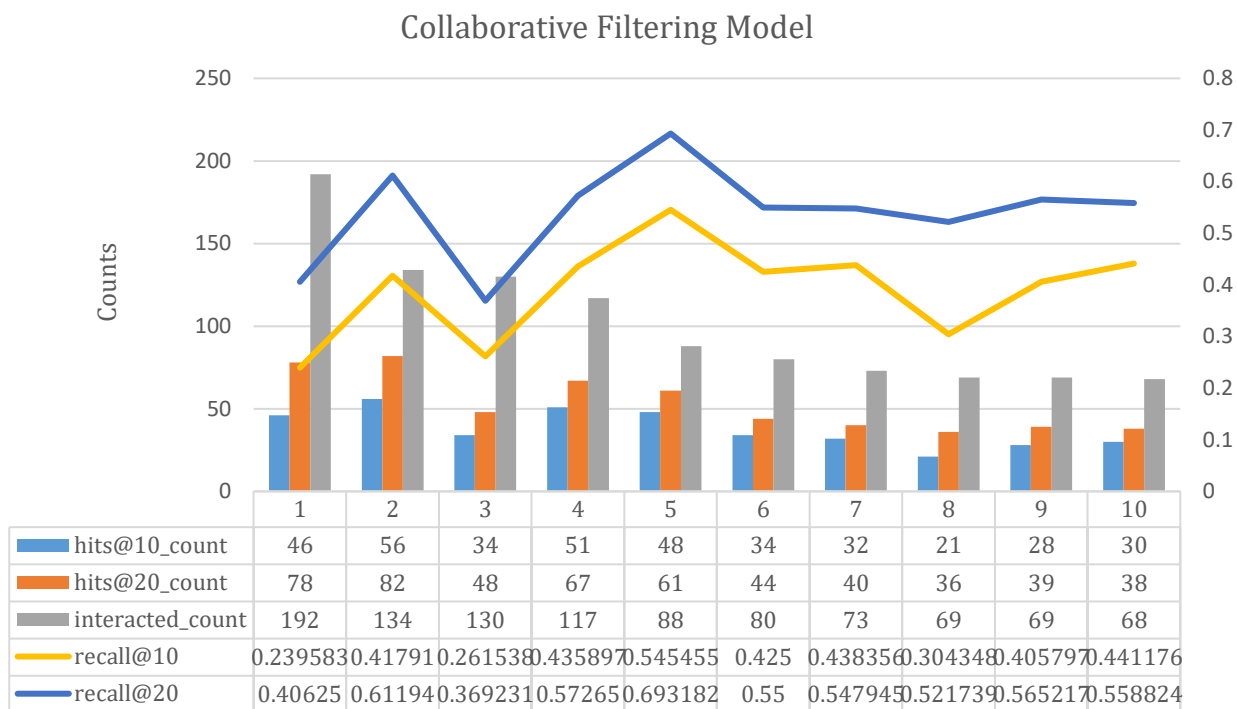


Figure 5.5: Evaluation results of Collaborative Filtering Model on first 10 out of 1139 users having global metrics - recall@10: 0.46803886474047557, recall@20: 0.6146765533111737.

a low-dimensional representation in terms of latent factors. Such reduced representation can be utilized for either *user-based* or *item-based* neighbourhood algorithms. We shall take the advantage of using much smaller matrix in lower-dimensional space instead of having a high dimensional matrix containing abundant number of missing values. This is known as *Matrix Factorization (MF)*. An important decision for *MF* is the number of factors to factor the *user-item* matrix. The higher the number of factors, the more precise is the factorization in the original matrix reconstructions. Therefore, if the model is allowed to memorize too much details of the original matrix, it may not generalize well for data it was not trained on. Reducing the number of factors increases the model generalization. After the factorization, we try to reconstruct the original matrix by multiplying its factors. The resulting matrix is not sparse any more. For example, it has generated predictions for items the user has not yet interacted with, which we will exploit for data filtering at later stage.

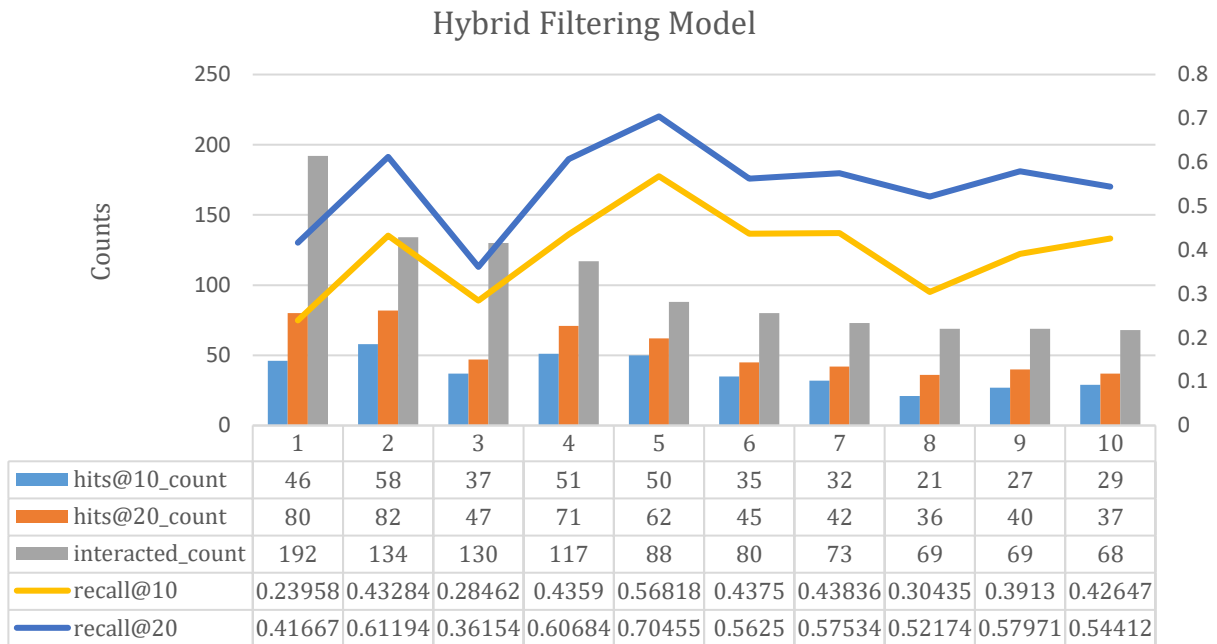


Figure 5.6: Evaluation results of Hybrid Filtering Model on first 10 out of 1139 users having global metrics - recall@10: 0.4796727179749425, recall@20: 0.6259268729225262.

Evaluating the Collaborative Filtering model (SVD matrix factorization) as shown in above, we observe that *Recall@10* is 0.47 (47%) and *Recall@20* is 0.61 (61%), which are much higher than *Popularity* and *Content-Based* models.

Hybrid Filtering Model

Predictive accuracy is substantially improved when blending multiple predictors. This is known as *Hybrid* approach that combines *Collaborative*, *Content-Based* and other approaches. This approach is an ensemble that takes the weighted average of normalized *Collaborative Filtering (CF)* scores with the *Content-Based (CB)* scores, and ranks them by resulting score. The combined prediction scores (above) by *contentId* obtained from *Collaborative* and *Content-Based* approaches and sorted by hybrid score returned better results for *Recall@10* as 0.48 (48%) and *Recall@20* as 0.63 (63%).

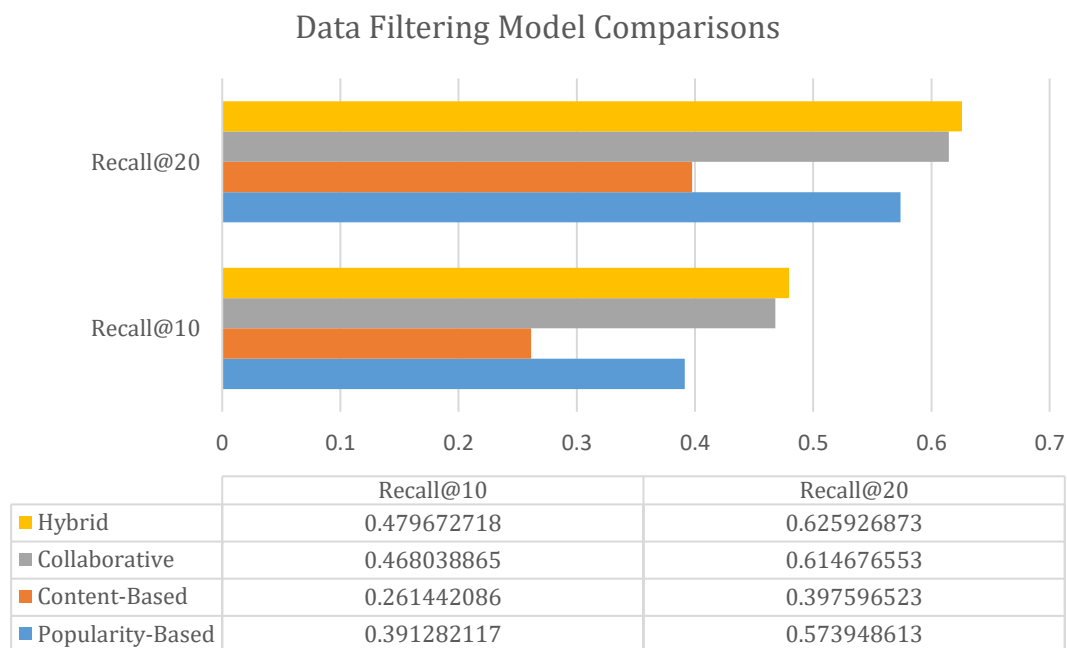


Figure 5.7: Comparison of Top-N accuracy values calculated as Recall@N from 100 random test data by using data filtering models.

5.3.1.4 Discussion

To understand how conventional information filtering techniques perform to infer user’s tasks, we ran an experiment by associating weights for different types of interactions to assume the interest of a user on a specific article and applied those on ³Google G-Suite data having 3K publicly shared articles and 73K logged user interactions. We have found through the above experiment that –

- *Hybrid* and *Collaborative* approaches show substantially better accuracies than other approaches where *Hybrid* model is the best to retrieve relevant data. *Content-Based* model shows the least performance according to its global metrics (above).
- *Recall@N* increases as *Top-N* increases. We have found Recall@20 as always higher than Recall@10 for all above models.
- In case of *Popularity* model *Recall@N* values drop as hit(*N*) counts decrease and rise as hit(*N*) counts increase (above). This is because *popularity* model often considers the most popular items based on “*wisdom of crowds*”.

- For *Content-Based* model we also have observed proportional relationships among *Top-N* and *Recall-N* values (above). This is because for building user and item profiles *Term Frequency – Inverse Document Frequency (TF-IDF)* was used for raw text dataset. However, the lowest performance compared to other models may indicate that users were not fixed into similar contents while exploring different articles.
- *Hybrid (above)* and *Collaborative (above)* models are the top most performing models. *Top-N* and *Recall@N* are also proportionally related for both models. This is because, in the case of *Collaborative* approach the evaluation results are dependent on *user-item matrix factorization* and the reconstructed matrix generates predictions for not yet interacted items which are later exploited for filtering *Top-N*. The *Hybrid* model also includes *Content-Based* and other approaches.

We have learned from the above *Recall@N* values after applying conventional data filtering techniques that for better understanding of user's intention and retrieving relevant data it always requires higher *hit(N)* counts. This is a drawback in the case of *user cold-start* problem where none or a very few number of relevant data items can be retrieved due to the lack of information. We could leverage the *contextual information* for such scenario to model user profiles.

5.3.2 Contextual Attention

The results of above conventional information filtering were achieved by associating weights for different types of interactions to assume the interest of a user on a specific article rather than considering his/her contextual attention of search.

5.3.2.1 The Context

To understand the context it is important to discover the underlying sequences of actions and the transitions among those. This is difficult in case of unstructured/unannotated sequential list, because it does not contain enough structure to infer the analytical attentions. We have considered meaningful '*Action*

Chunking as a way to provide sequence streams of actions during analytic process that will mark the semantic boundaries between user's sensemaking tasks and other actions. We also have explained that chunking can be '*Contextual*' as well as '*Hierarchical*'. To know the context an '*Extractive*' or '*Abstractive*' technique can be leveraged to rank different texts and then distinguish between sensemaking and non-sensemaking tasks after computing relevance of tokens/words extracted from sentences. For an example – *TextRank* algorithm is an extractive and unsupervised technique where *word embeddings* (i.e., GloVe algorithm) are done to form *cosine similarity matrix* and then converted into a graph with sentences as vertices and similarity scores as edges for *TextRank* calculation. However, to form the hierarchy, *Word2Vec* algorithm has been used to learn word associations from a large corpus of text. It is a NLP technique that uses neural network model. To distinguish between sensemaking and non-sensmaking tasks, it is important to identify synonymous words with top relevant words/tokens. The underlying assumption of *Word2Vec* is that two words sharing similar contexts also share a similar meaning and consequently a similar vector representation from the model. From this assumption, *Word2Vec* can be used to find out the relations between words in a dataset and compute the similarity/dissimilarity between them.

5.3.2.2 The Attention

To identify top relevant words/tokens, we have adopted '*Attention Mechanism*' which takes two sentences, turns them into a matrix where the words of one sentence form the columns, and the words of another sentence form the rows, and then it makes matches, identifying relevant context. Not all words contribute equally to the representation of the sentence meaning. Hence, attention mechanism can be used to extract such words that are important to the meaning of the sentence. We have used such '*Word Attentions*' to label text corpus of articles dataset to mark their categories. To experiment how accurately such text categories be predicted by applying popular classification models, we implemented those and found accuracies for – Logistic Regression: 0.70, Random Forest: 0.60, Linear SVC: 0.71, Multinomial Naive Bayes: 0.55 and SGD Classifier: 0.67. Classification results that we found by following this

approach were not promising. However, such '*Local Attention*' of words are not enough to understand user's intention and accurately chunk user's tasks at different levels as a whole. It is also important to consider '*Global Attention*' which implies we attend all the input words rather than local attention which only attends a subset of words. Besides local and global attentions we also considered '*intra-attentions*' sometimes known '*self-attention*' which is a mechanism relating to different word positions of a single sequence. Google proposed a sequence transduction model entirely relying on self-attention mechanism known as '*Transformer*'. It applies the *self-attention* mechanism which directly models relationships between all words in a sentence, regardless of their respective position. For an example – 'I arrived at the bank after crossing the river', to determine that the word 'bank' refers to the shore of a river and not a financial institution, the '*Transformer*' can learn to immediately attend to the word 'river' and make this decision in a single step. *Self-Attention* is computed not once but multiple times in the *Transformer's* architecture, in parallel and independently. It is therefore referred to as '*Multi-head Attention*' which allows the model to jointly attend the information from different representation subspaces at different positions.

5.3.3 BreakPoints for Action Chunking

We have considered meaningful '*Chunks of Actions*' as a way to provide sequence streams of actions during analytic processes that will mark the semantic boundaries between user's sensemaking tasks and other actions. It involves a detailed analysis of when users switch tasks, is critical to a deeper understanding of human multitasking behaviour. A particular goal of this research involves how accurately can such multi-task switches be inferred during execution of interactive tasks. We named those task switches as '*BreakPoints*' of user's analytical tasks.

5.3.3.1 Definition

A '*BreakPoint*' is the moment between two meaningful units of task execution, and reflects internal transitions in perception or cognition [108, 109]. These are the points in a task sequence where the user can most conveniently switch tasks [110].

5.3.3.2 Detecting BreakPoints

One common method for detecting break-points is to match users' ongoing interactions to specifications of tasks defined a priori. Although this allows break-points to be easily detected within tasks that are fairly prescribed. But it is much more difficult to leverage these types of static specifications to detect break-points within tasks that have highly variable interactions, i.e., free-form tasks or multi-user's interactions during similar task execution. In case of an unknown task although previous user's data can be utilized in suggestive manner for multiple users, however it may not be useful due to cognitive and perceptual variances. So, detecting breakpoints in such scenario even becomes more strenuous.

Through this research, we seek to overcome this problem by understanding how to detect breakpoints and differentiate their granularity without requiring any task specification. Granularity refers to the degree of perceptual difference of the actions surrounding a breakpoint. A basic question is – *'how many granularities of breakpoints are detectable and meaningful during task execution?'* Number of granularities will depend on – *'what are the meronyms of breakpoints?'*, *'What is the smallest constituent unit of break-points?'*, *'How partly or broadly shall we consider break-points?'* etc. From studies of event perception and task interruption [108], there is evidence for at least three perceptually meaningful granularities; *'Coarse'*, *'Medium'*, and *'Fine'* [108, 111]. For example, when booking for holidays online, *'Fine'* may be switching different sites to check various options; *'Medium'* may be searching in different categories i.e, flights, hotel, places of attractions, conveyances, food restaurants etc and *'Coarse'* may be switching to activities other than holiday booking i.e., check emails or social networking from time to time. Bogunovich et. al. [110] have denoted "Coarse" activities as secondary tasks and found those are the good indicators of multitasking breakpoints. Those are cognitive breaks and users may occasionally take such short breaks when a cognitive subtask is completed and before beginning a new subtask. Finding out what are relevant and irrelevant tasks to achieve the goal, are a bit challenging. Because even a machine learning model will not be able to know what exactly a user's main task is but can predict it with a certain level of accuracy. Increasing accuracy will require extracting and mapping more predictive features to

breakpoints. Iqbal et. al. [111] determined some candidate features based on analysis of observer's explanations and event logs. They then asked users whose interaction data was originally annotated by observers and thus tested the accuracy. We endeavour to test our assumption that computing semantic similarities (relevant tasks) and dissimilarities (irrelevant tasks) in an automatic data-driven manner will be helpful to detect breakpoints without any prior task specification through couple of experiments described into following sections.

As we have defined *breakpoint* as the moment between two meaningful units of task execution, and reflected on internal transitions in perception or cognition, so understanding semantic similarity/dissimilarity will help us to pin point where a user does different things during a targeted task with a specific goal but it does not necessary express their cognitive transition. The '*Contextual Attention*' method into NLP as described above can be one method of understanding user's intention by using '*Multi-Headed Self-Attention*' mechanism, where there are no pre-defined tasks. In case of a priori defined tasks, the method of detecting breakpoints can be matching users' ongoing interactions to specifications of task. Although this allows break-points to be easily detected within tasks that are fairly prescribed. It is much more difficult to leverage these types of static specifications to detect breakpoints within tasks that have highly variable interactions, i.e., free-form tasks or multi-user's interactions during similar task execution. In case of an unknown task, apart from the *contextual attention* analysis, user's data can also be utilized in suggestive manner for multiple users, however it may not be useful due to cognitive and perceptual variances. So, detecting breakpoints in such scenario even becomes more strenuous. We already have applied conventional information filtering techniques on ³Google G-Suite data for multiple users and filter top ranked similar data to understand how accurately machine can perceive user's intents. We have found - Recall@10 of 0.39 (which stands for 39% of interacted items in test set were ranked among the top-10 items), Recall@20 of 0.57 for '*Popularity Model*'; Recall@10 of 0.26, Recall@20 of 0.40 for '*Content Based Model*'; Recall@10 of 0.47, Recall@20 of 0.61 for '*Collaborative Model*'; Recall@10 of 0.48, Recall@20 of 0.63 for '*Hybrid Model*'; from the experiment

of previous section which are not promising results. Finding out which tasks are relevant and which are irrelevant, can pinpoint the *breakpoint* at higher level of hierarchy as explained earlier. Knowing relevant or irrelevant tasks are exploiting commonalities and differences across tasks which can be achieved by following two methods:

- **Task Abstraction** – Tasks may be grouped according to some general metric which can be used for *task abstraction*. Iqbal et. al. [111] determined some candidate features as metric based on analysis of observer's explanations and event logs. For such abstraction to be automated, one must hypothesize a mechanism by which low-level operations or actions can be inferentially mapped to higher-level intents. Bors et. al. [112] define task abstraction as the idea that low-level operations can be grouped into sets that can themselves be usefully considered as unified, purposeful units of action. These units of action may then be grouped into still larger units of action and so on. Hence, any given coherent sequence of operations can be described in terms of an abstraction hierarchy. They have proposed a conceptual '*Abstraction Mapping Mechanism (AMM)*' to enable adhoc parsing of interaction streams into abstract tasks and inferring upcoming actions.
- **Identifying Unrelated Tasks** - Commonalities and differences can then be described in terms of *Task Abstraction* hierarchy to distinguish between relevance and irrelevance of tasks. For an example - '*coarse*' level break point indicates semantic change between two chunks of action. This is the same as our original definition for break point. Common techniques for finding semantically similar text corpus can be '*Topic modelling*'. The underlying assumption of '*Word2Vec*' is that two words sharing similar contexts also share a similar meaning and consequently a similar vector representation from the model. From this assumption, '*Word2Vec*' can be used to find out the relations between words in a dataset and compute the similarity/dissimilarity between them.

5.3.4 Experiment 2

5.3.4.1 Supervised Learning for BreakPoint Detection

Inferring user's intent into a task model is all about understanding multi-tasking behaviour, understanding search language and building context of interacted contents based on attention. In 2018, Google released their '*neural network-based*' technique for '*Natural Language Processing (NLP)*' named as '*Bidirectional Encoder Representations from Transformers*' or BERT [55] in short. It allows the language model to learn word context based on surrounding words rather than just the word that immediately precedes or follows it. It builds a network with attention known as a '*Transformer*' network which includes self and multi-head attention mechanisms [53]. This is how BERT is useful for understanding user's intent.

5.3.4.2 Dataset

To capture log dataset we used '*Chrome Browsing History View*' software developed by NirSoft⁴. It is a utility that reads the history data of different browsers and displays it in one table. The browsing history table includes information shown in below. The process of data collection lasted for a week and considered following steps:

- We collected data samples under several categories such as – online shopping, search holiday destinations, Net banking, Hobbies or Interests, News etc.
- Google search engine was used find out topics of interests or directly visited to known sites by typing the url on browser.
- As part of sensemaking activities we implemented contextual, hierarchical and binary search criteria as described into Section 5.2, so that those concepts remain inherent into dataset.
- We also performed some other non-sensemaking tasks outside of those categories such - browsing social network sites, checking emails etc.
- Thus we collected a month's 53K chrome history log having different search criteria hidden into dataset.

4. https://www.nirsoft.net/utils/browsing_history_view.html

Table 5.1: Chrome browsing history information.

Attribute	Description
URL	Address of the webpage user is currently visiting
Title	Title of the visited webpage
Visited On	Refers to the initial/start time any webpage is loaded
Visit Count	The number of times the user has navigated to the webpage
Typed Count	The number of times the user has navigated to the webpage by typing in the address
Referrer	It is the webpage that sends visitors to another site using a link
Visit ID	Corresponds to unique id for each website visit
Profile	User’s default device profile
URL length	String length of the url
Transition Type	Types of interactions to navigate to a particular url i.e, link, reload, typed, form submit, auto bookmark, manual subframe etc.
Transition Qualifiers	It further defines the transition i.e, chain_start: the beginning of navigation chain, chain_end: the last transition in a redirect chain, client_redirect: redirects caused by meta refresh on the page, server_redirect: redirects sent from the server by HTTP headers etc.
History File	Location of local browser history data.

5.3.4.3 Pre-processing

The raw data collected for the analysis consisted of many attributes initially. So, proper feature selection was done, and attributes which were relevant to the aim and interconnected were selected. We deleted ‘*History File*’ and ‘*Profile*’ columns from the dataset. Additionally, certain derived attributes were also added to the dataset’s attributes list. These were ‘*Domain*’ and ‘*ElapsedTime*’.

We also added one more column at the end named as ‘*Breakpoint*’. We used ‘0’ as indicating non-breakpoint and ‘1’ as indicating breakpoint to label the whole dataset according to known search criteria during sensemaking sessions. Thus we gave couple of passes to make sure that the breakpoint markings are correct. This is how we prepared the training dataset to use for developed machine learning models of this experiment as described into next few sections.

5.3.4.4 Implementation

We used 53K pre-processed log dataset and trained BERT transformer model for almost 25 hours to infer ‘*where the breakpoints are*’. Because of BERT’s complexity, it is difficult to interpret the meaning of its learned weights. Deep-learning models in general are notoriously opaque, and various visualization tools have been developed to help make sense of them. Jesse Vig et. al. [113] modified *Tensor2Tensor*⁵ attention visualization tool to make it work with BERT. We have used the tool to explore different attention patterns of BERT base (12 layers - transformer blocks, 12 attention heads). We have experimented on different input values by following breakpoints <- > non-breakpoints transitions to understand attentions of different words contributing towards it’s classifications. As shown in Figure 5.8, the tool visualizes attention as lines connecting the position being updated (left) with the position being attended to (right). Colours identify the corresponding attention head(s), while line thickness reflects the attention score. At the top of the tool, the user can select the model layer, as well as one or more attention heads (by clicking on the colour patches at the top, representing the 12 heads) [113]. Following two texts have been chosen from an actual breakpoint into our original dataset to explain different patterns.

Text A = "xnet.unisys.com Default Reload Chain Start Chain End" (BreakPoint = 0)

Text B = "youtube.com My Favourite Dishes-YouTube Default Link Chain Start" (BreakPoint = 1)

Pattern 1 - In this pattern, most of the attention at a particular position is directed to the next token in the sequence. From below we see an example of this for *layer 2, head 0*. (The selected head is indicated by the highlighted square in the colour bar at the top.) The figure on the left shows the attention for all tokens, while the one on the right shows the attention for one selected token (*‘my’*). In this example, virtually all of the attention is directed to *‘favourite’* the next token in the sequence. On the left, we also can see that the [SEP] token disrupts the next-token attention pattern, as most of the attention from [SEP] is directed to [CLS] rather than the next token. Thus this pattern appears to operate primarily within each sentence [113].

5. <https://github.com/tensorflow/tensor2tensor/tree/master/tensor2tensor>

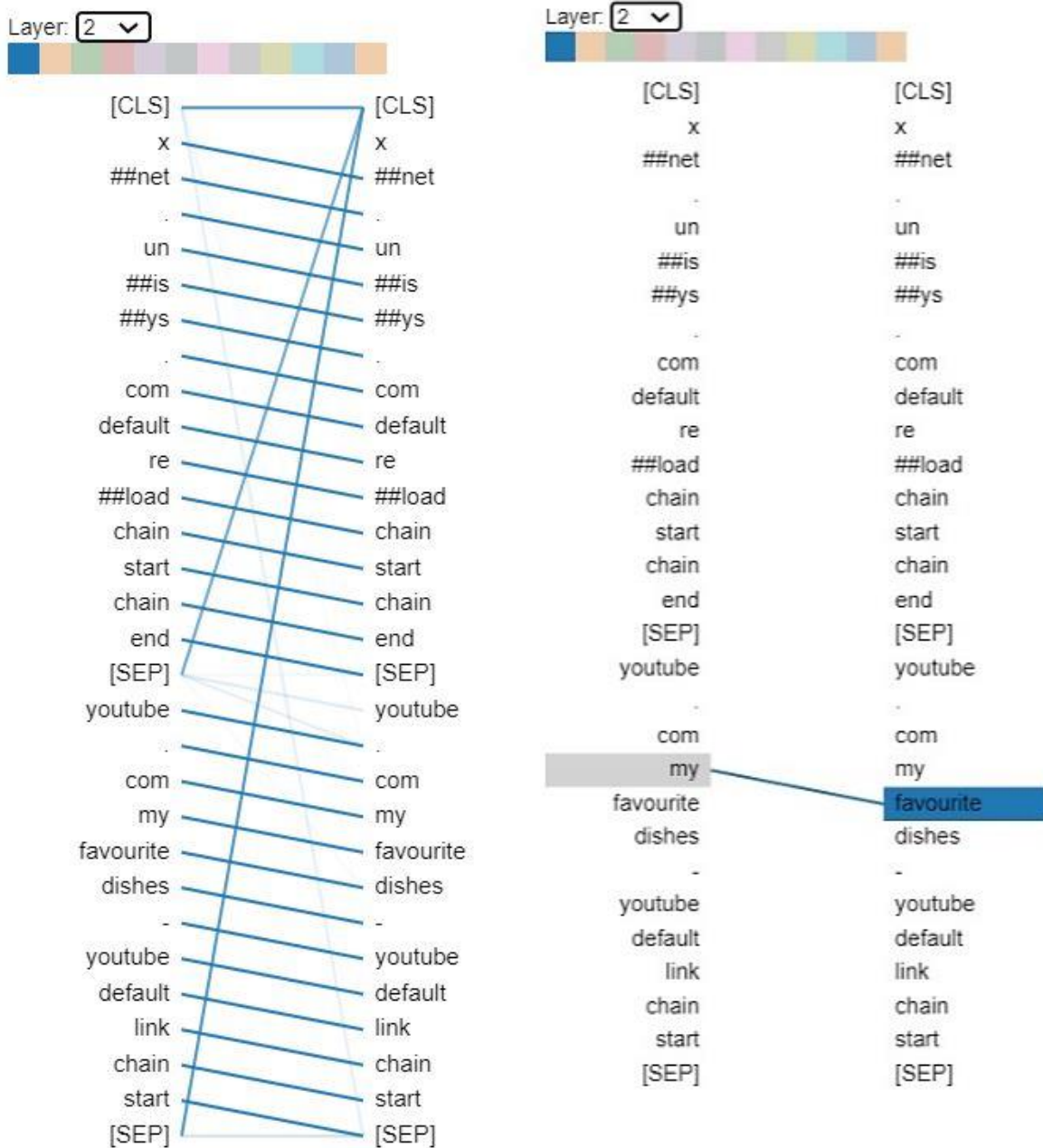


Figure 5.8: Attention to next word at layer 2 head 0, found from Jesse Vig et. al.'s [113] modified *Tensor2Tensor* attention visualization tool. **Left:** attention weights for all tokens. **Right:** attention weights for selected token ('my').

Pattern 2 – In this pattern, much of the attention is directed to the previous token in the sentence. As shown in below, the attention for 'dishes' is directed to the previous word 'favourite'. This pattern is not as distinct as the previous one; some attention is also dispersed to other tokens, especially the [SEP] tokens [113].

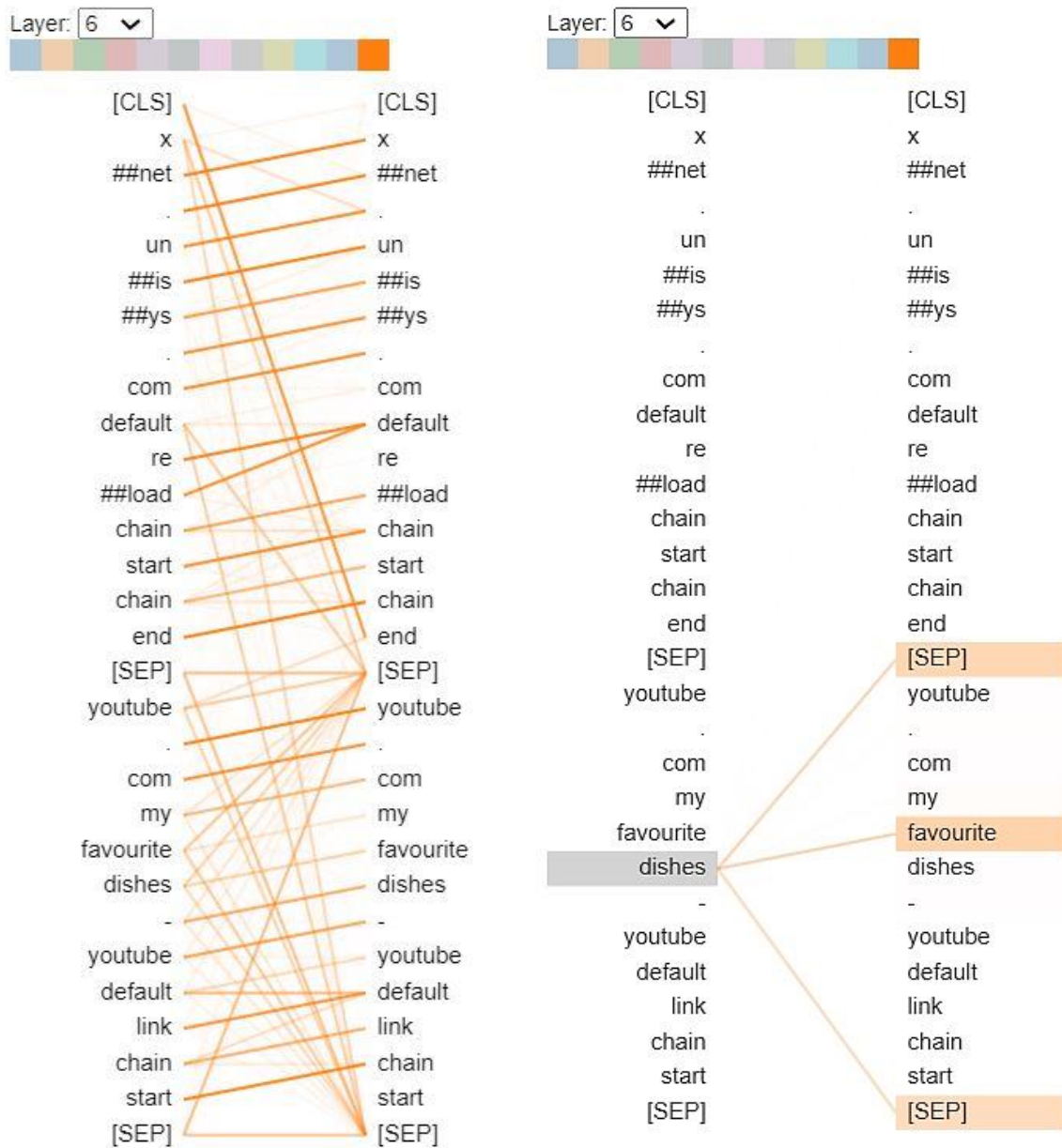


Figure 5.9: Attention to previous word at layer 6 head 11, found from Jesse Vig et. al.’s [113] modified *Tensor2Tensor5* attention visualization tool. **Left:** attention weights for all tokens. **Right:** attention weights for selected token (‘dishes’).

Pattern 3 – In this pattern, attention is paid to identical or related words, including the source word itself. In the example below (below), most of the attention for the first occurrence of ‘youtube’ is directed to itself and to the second occurrence of ‘youtube’. This pattern is not as distinct as some of the others, with attention dispersed over many different words [113].

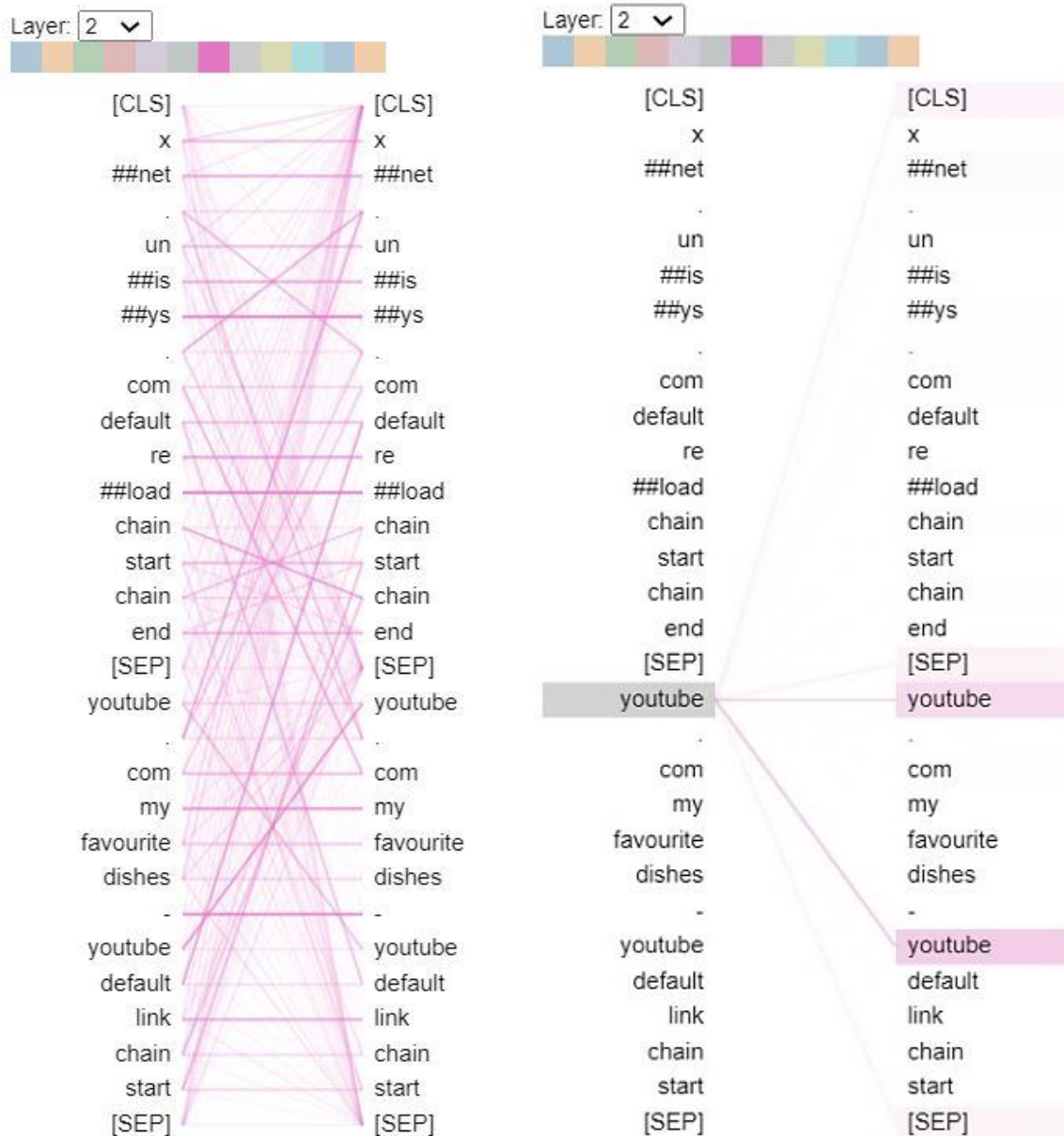


Figure 5.10: Attention to identical/related tokens at layer 2 head 6, found from Jesse Vig et. al.'s [113] modified *Tensor2Tensor* attention visualization tool. **Left:** attention weights for all tokens. **Right:** attention weights for selected token ('youtube').

Pattern 4 – In this pattern, attention is paid to identical or related words in the other sentence. For example, most of attention for 'chain' in the second sentence is directed to 'chain' in the first sentence (below). One can imagine this being particularly helpful for the next sentence prediction task (part of BERT's pre-training), because it helps identify relationships between sentences [113].

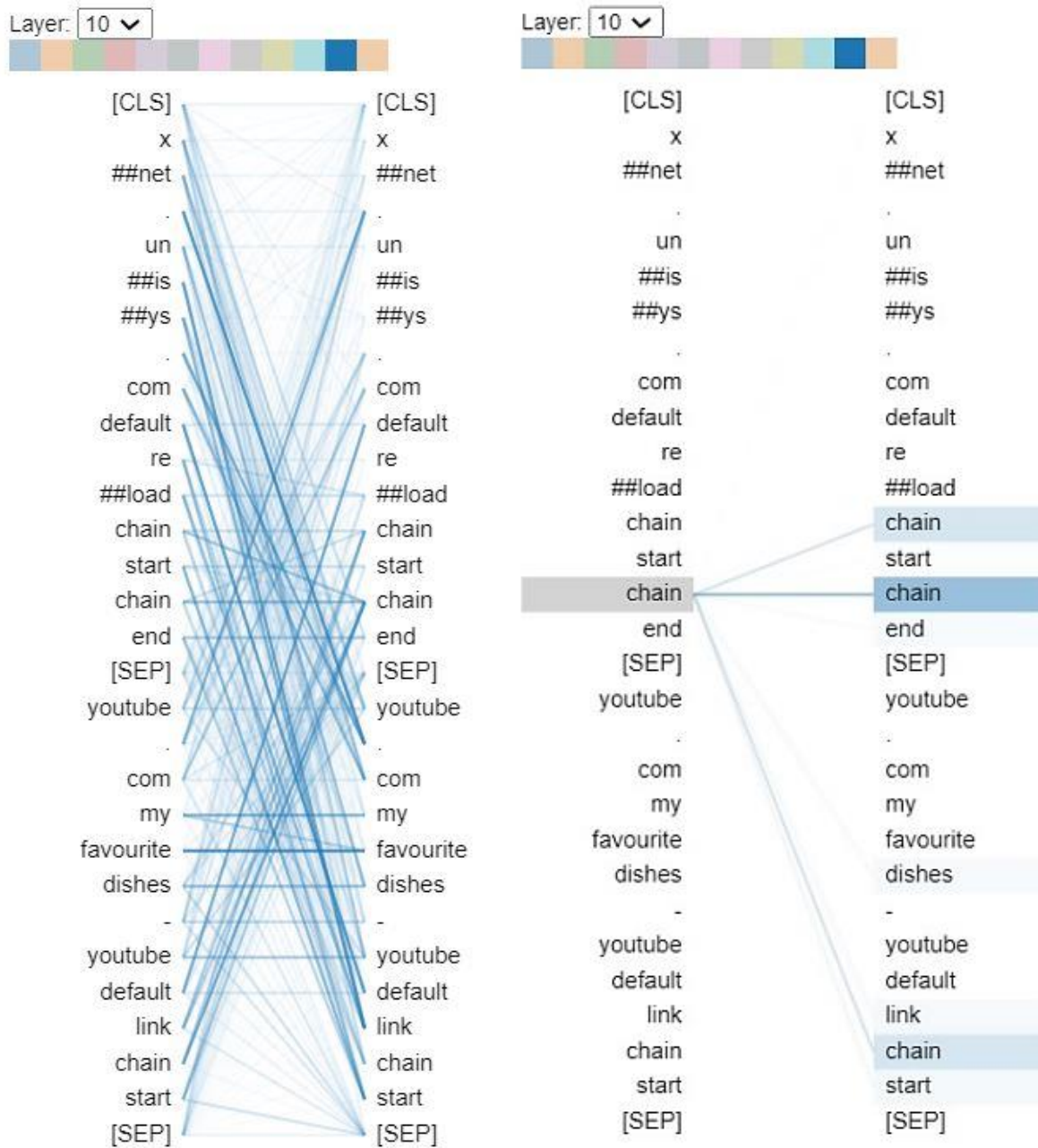


Figure 5.11: Attention to identical/related words in other sentence at layer 10 head 10, found from Jesse Vig et. al.'s [113] modified *Tensor2Tensor5* attention visualization tool. **Left:** attention weights for all tokens. **Right:** attention weights for selected token ('chain').

Pattern 5 – In this pattern, attention seems to be directed to other words that are predictive of the source word, excluding the source word itself. The example in below, most of the attention from 're' is directed to '##load', and most of the attention from '##load' is focused on 're' [113].

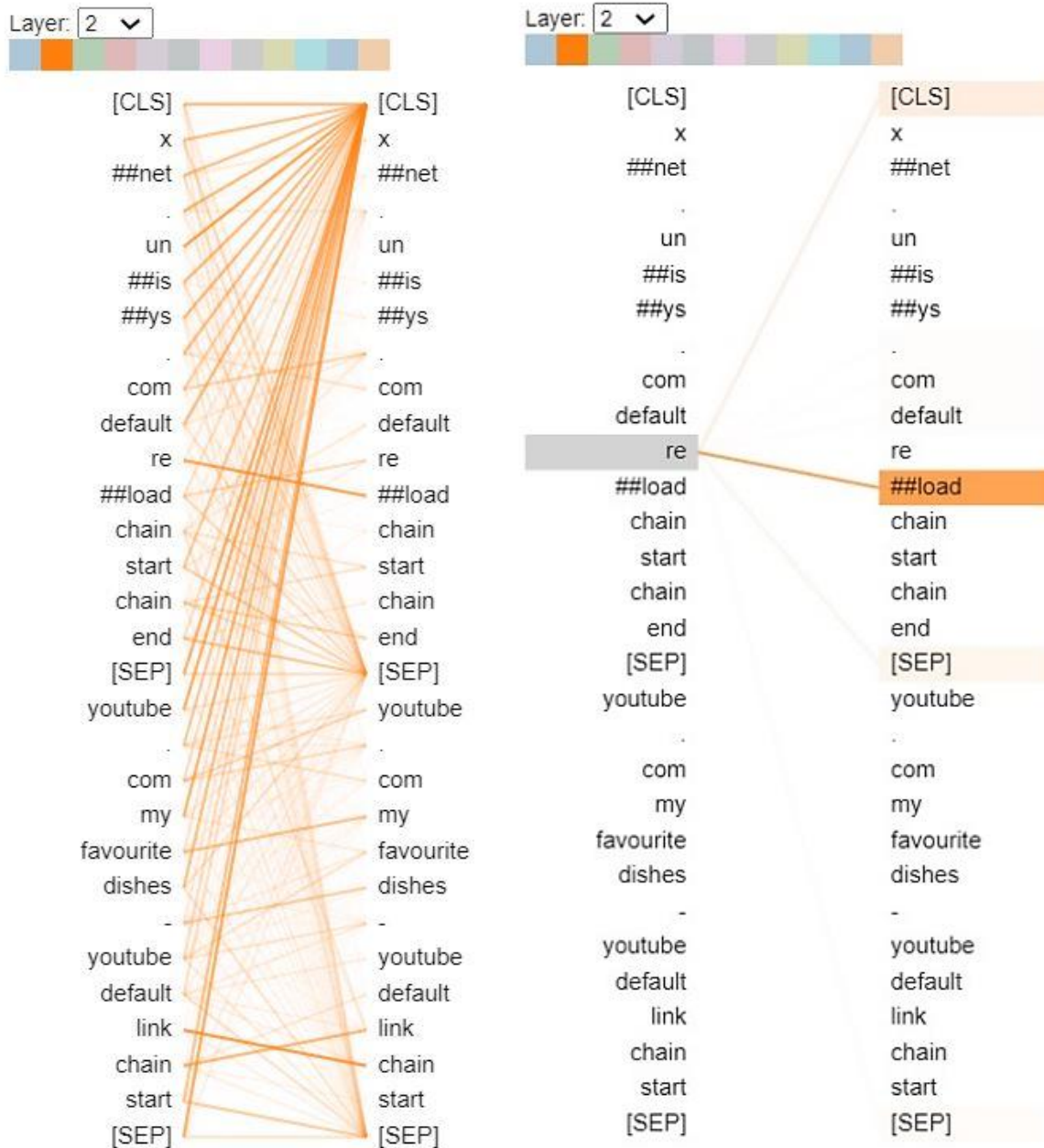


Figure 5.12: Attention to other words predictive of word at layer 2 head 1, found from Jesse Vig et. al.'s [113] modified *Tensor2Tensor5* attention visualization tool. **Left:** attention weights for all tokens. **Right:** attention weights for selected token ('re').

Pattern 6 – In this pattern, most of the attention is directed to the delimiter tokens, either the [CLS] token or the [SEP] tokens. In the example as shown in below, most of the attention is directed to the two [SEP] tokens.

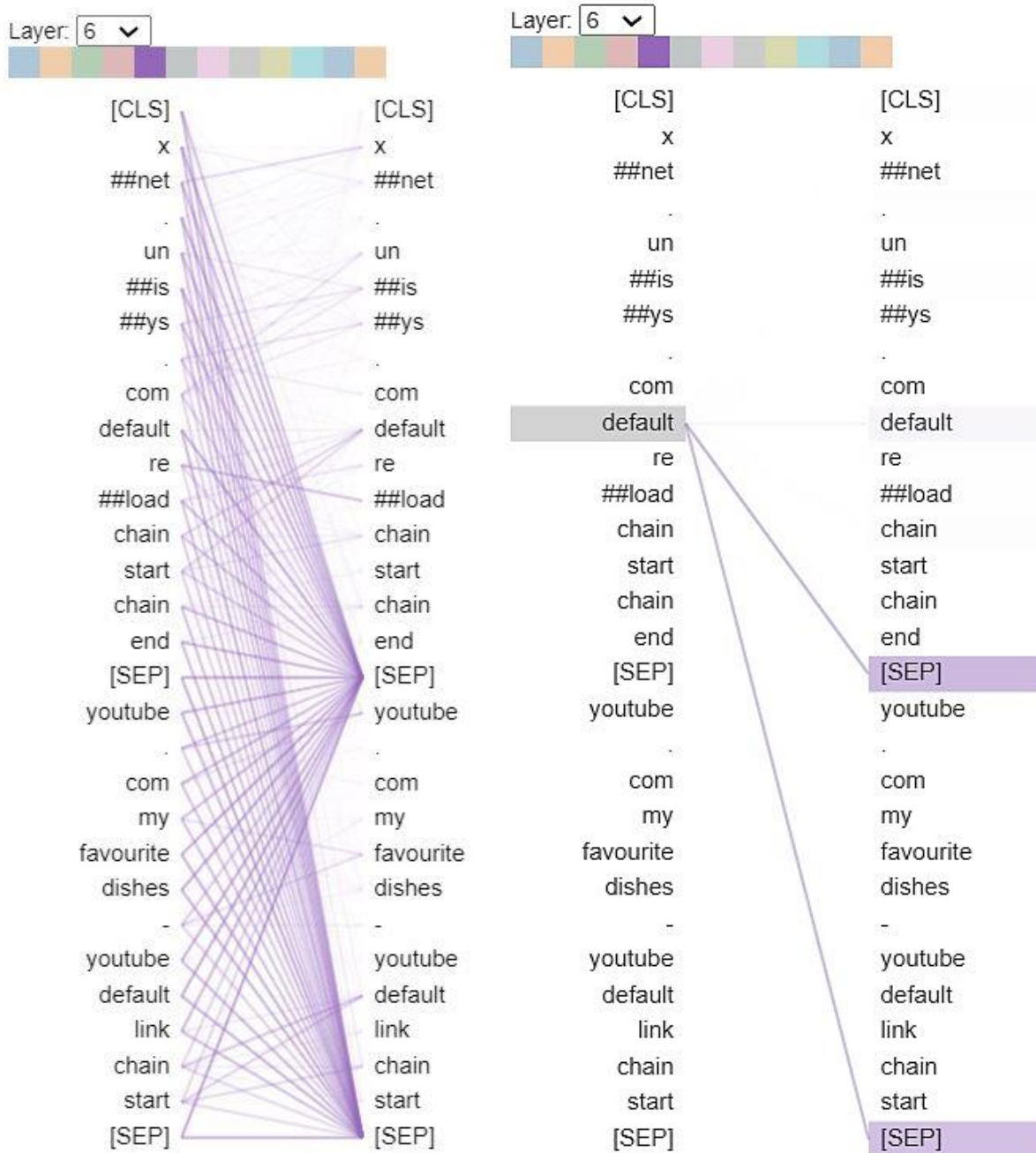


Figure 5.13: Attention to delimiter tokens at layer 6 head 4, found from Jesse Vig et. al.’s [113] modified *Tensor2Tensor5* attention visualization tool. **Left:** attention weights for all tokens. **Right:** attention weights for selected token (‘default’).

Clark et. al. [114] describes this pattern as a ‘no-op’: an attention head focuses on the [SEP] tokens when it can’t find anything meaningful in the input sentence to focus on.

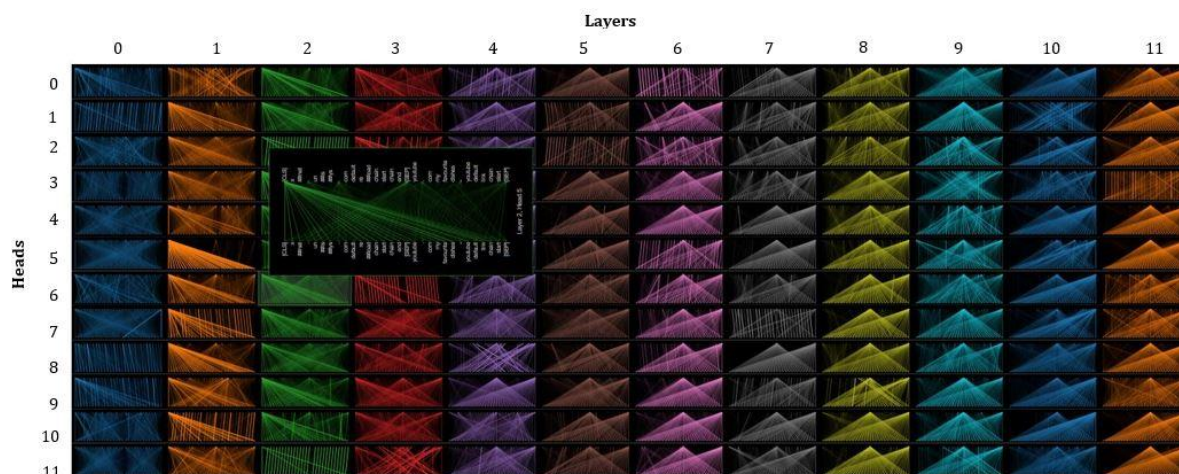


Figure 5.13: BERT base model visualizations found from Jesse Vig et. al.'s [113] modified *Tensor2Tensor5* attention visualization tool, for 12 layers and 12 heads resulting in a total of $12 \times 12 = 144$ distinct attentions for Text A and Text B.

Multi-Head Attention Patterns

Visualizations of patterns 1-6 show one attention mechanism within the model. BERT actually learns multiple attention mechanisms, called *heads* [Figure 2.23(ii)], which operate in parallel to one another. Multi-head attention enables the model to capture a broader range of relationships between words than a single attention mechanism. BERT also stacks multiple layers of attention, each of which operates on the output of the layer that came before. Through this repeated composition of word embeddings, BERT is able to form very rich representations as it gets to the deepest layers of the model. Because the attention heads do not share parameters, each head learns a unique attention pattern [113]. As shown in Figure 5.13 — BERT Base has 12 layers and 12 heads, resulting in a total of $12 \times 12 = 144$ distinct attention mechanisms. Thus attention in all of the heads can be visualized at once. Each cell in the in the BERT Base model visualizations show the attention pattern for a particular head (indexed by row) in a particular layer (indicated by column), using a thumbnail form of the attention-head view from earlier. The attention patterns are specific to input Text A and Text B. From the visualizations, we can see that BERT produces a rich array of attention patterns.

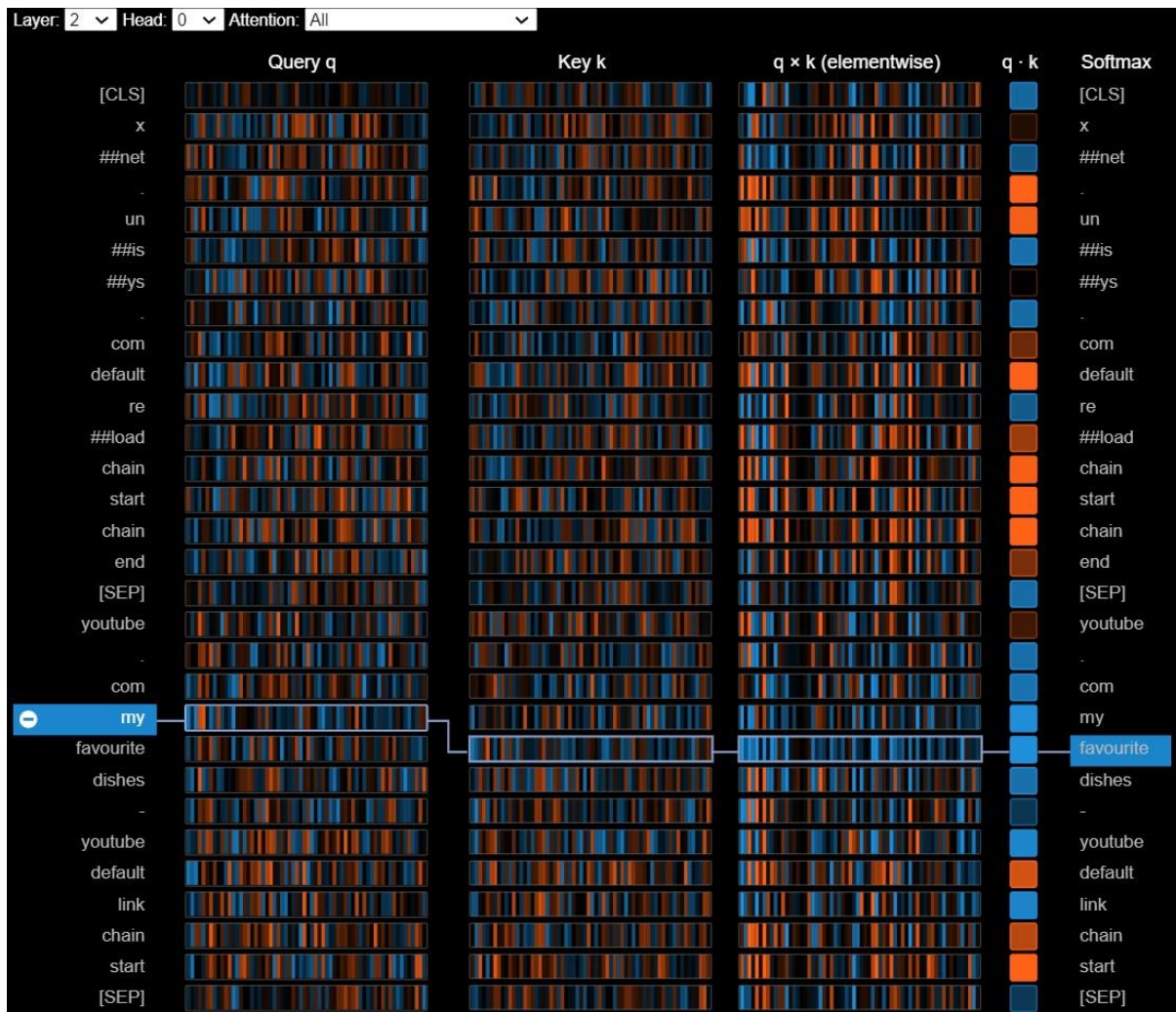


Figure 5.14: Text A and Text B focused next word attention pattern at layer 2 head 0 of the BERT-base pre-trained model, found from Jesse Vig et. al.’s [113] modified *Tensor2Tensor5* attention visualization tool.

Explaining BERT Attention Patterns

BERT uses a *compatibility function*, which assigns a score to each pair of words indicating how strongly they should attend to one another. To measure compatibility, the model first assigns to each word a *query vector* (q) and a *key vector* (k). The *compatibility score* is just the dot product ($q \cdot k$) of the *query vector* (q) of one word and the *key vector* (k) of the other. To turn these compatibility scores into valid attention weights, we must normalize them to be positive and sum to one (since



Figure 5.15: Elementwise and dot products of query (q) and key (k) vectors for next word attentions at layer 2, head 0.

attention weights are used to compute a weighted average). This is accomplished by applying the *softmax function* over the scores for a given word. The dot-products are scaled by dividing by the square root of the vector length ($\frac{1}{\sqrt{d_k}}$) [53]. There is a precursor to the dot product, which is the elementwise product ($q \times k$) between the query vector (q) of the selected word and each of the key vectors (k). It shows how individual element in the query and key vectors contribute to the dot product. Based on these calculations we have presented a few visualizations in this section to show how attention weights are computed from query and key vectors. As shown in above, it traces the computation of attention from the selected word on the left to the complete sequence of words on the right. ‘Positive’ values are coloured as ‘blue’ and ‘negative’ values as ‘orange’, with colour intensity representing magnitude. Like the BERT attention-pattern views presented earlier, the connecting lines indicate the strength of attention between the connected words.

Next Word Attentions

We have found from above that most of the attention at a particular position is directed to the next token in the sequence at layer 2, head 0. The reason can be - adjacent words are often the most relevant for understanding a word’s meaning in context. Traditional *n-gram* language models are based on this same intuition.

We see from above that the product of the query vector for ‘my’ and the key vector for ‘favourite’ (the next word) is strongly positive across most neurons. For tokens other than the next token, the key-query product contains some combination of positive and negative values. The result is a high attention score between ‘my’ and ‘favourite’.

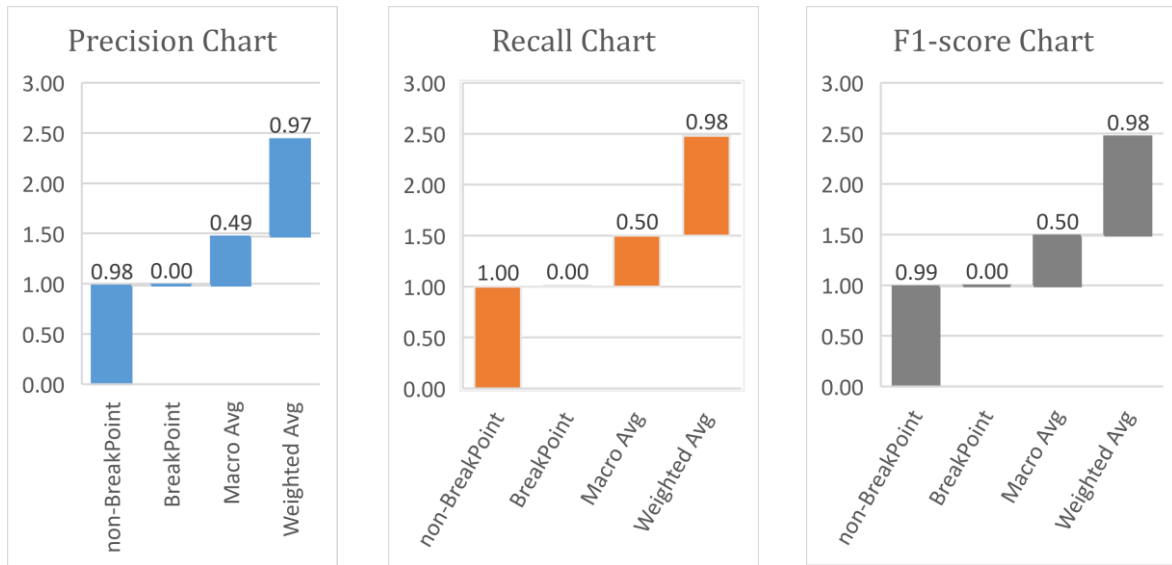


Figure 5.16: BERT classification reports.

5.3.4.5 Accuracy Scores

BERT classification reports as shown into above show better results of inference making than conventional approaches as described into Section 5.3.1. Both precision and recall for non-breakpoints are quite high which means BERT model predicts this class quite well. It's due to our imbalanced (breakpoint – 1.69% and non-breakpoint 98.31%) dataset. It becomes difficult for BERT to determine attention scores by using small subset of neurons which has greater impact on classification results. From the report obtained, the precision is 0.98 and recall 1.00 which depicts the predicted values for non-breakpoints are almost similar to their originals. This is because BERT transformer is able to generate large number of neurons of query and key vectors and feed forward to generate output probabilities. The overall accuracy obtained is – 98%.

5.3.4.6 Discussion

From experiment 1 (Section 5.3.1) we have found that it always requires higher hit (N) counts and with an exception to content based approach as shown in above,

hybrid model outperforms other conventional models. So, we have aimed to utilize contextual information into current Section 5.3.2 for experiment 2 to test either it can produce better results by overcoming the cold-start problem that is a drawback of experiment 1 (Section 5.3.1). We also have aimed to test either it supports domain independent user's behaviour modelling in terms of analytic actions or not.

We have explained how concepts of '*context*' and '*intention*' can be leveraged to detect the transition point of two meaning units of tasks. We have denoted this transition point as '*Breakpoint*'. Through our study we have found that breakpoints can be present at different granular levels known as '*meronyms*' as well as at different '*hierarchy*' or change of '*context*'.

The first part (Section 5.3.4.1) of experiment 2, adopts a supervised approach where we processed the training dataset by using concepts of context and hierarchy. We have trained a 'neural network' based model known as BERT which builds a network with attention and capable of learning word context on surrounding words rather than the word that immediately precedes or follows it. We have visualized the internal operations of BERT neural network known as '*Transformer*' to understand attentions of different words contributing towards its classification of breakpoint(1)/non-breakpoint(0). We have found 6 patterns of attentions at its different heads and layers. We also have visualized another 144 distinct attentions for the two input sample text at Mutli-head layers of BERT. We also have explained how attention weights are computed from query and key vectors. Their product contains some combination of positive and negative values and thus we know strength of attentions into visualizations i.e, '*positive*' values are coloured as '*blue*' and '*negative*' values as '*orange*'.

The classification report obtained, the precision 0.98 and recall 1.00 which depicts the predicted values for non-breakpoints are almost similar to their originals. This is because BERT transformer is able to generate large number of neurons of query and key vectors and feed forward to generate output probabilities. Obtained overall accuracy score is – 98% .

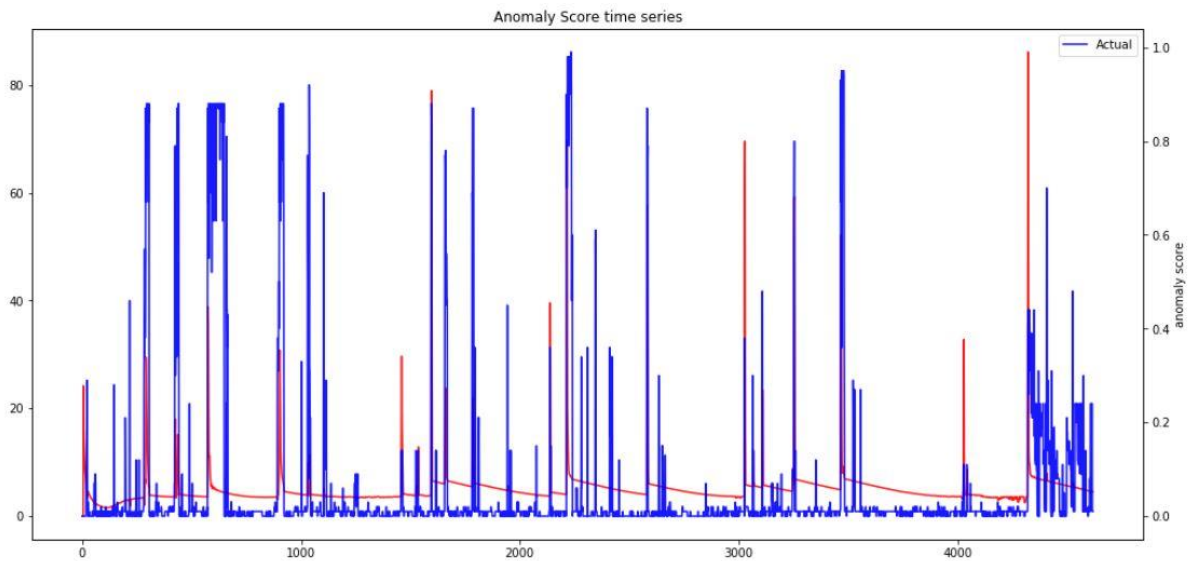


Figure 5.17: Detecting change points.

5.3.4.7 Unsupervised Learning for BreakPoint Detection

We have run experiment to infer 'where the break-points are' by applying supervised machine learning model and found promising results. But if we consider free-form tasks for multi-users or highly variable interactions for the similar goal, then a data driven automatic approach will be more feasible than the manual approach. In case of an unknown task, although previous user's data can be utilized in suggestive manner for multiple users, however it may not be useful due to cognitive and perceptual variances. Existing methods i.e, '*Bayesian multiple changepoints detection*' can only detect statistically boundaries by computing abrupt changes (i.e, mean, variance, spectrum etc) in the trends of a data sequence. Lee et.al. [115] showed that it is nearly impossible to detect human specified breakpoints by using existing *changepoint* techniques. We have tested their claim by applying '*ChangeFinder Algorithm*' on our captured log dataset and found that it can mostly detect change points (above) from trends of data but is not indicative of actual breakpoints in all cases. The 'blue' line shows the actual data plotting where the 'red' line plots are indicating change points among usual trends based on anomaly scores.

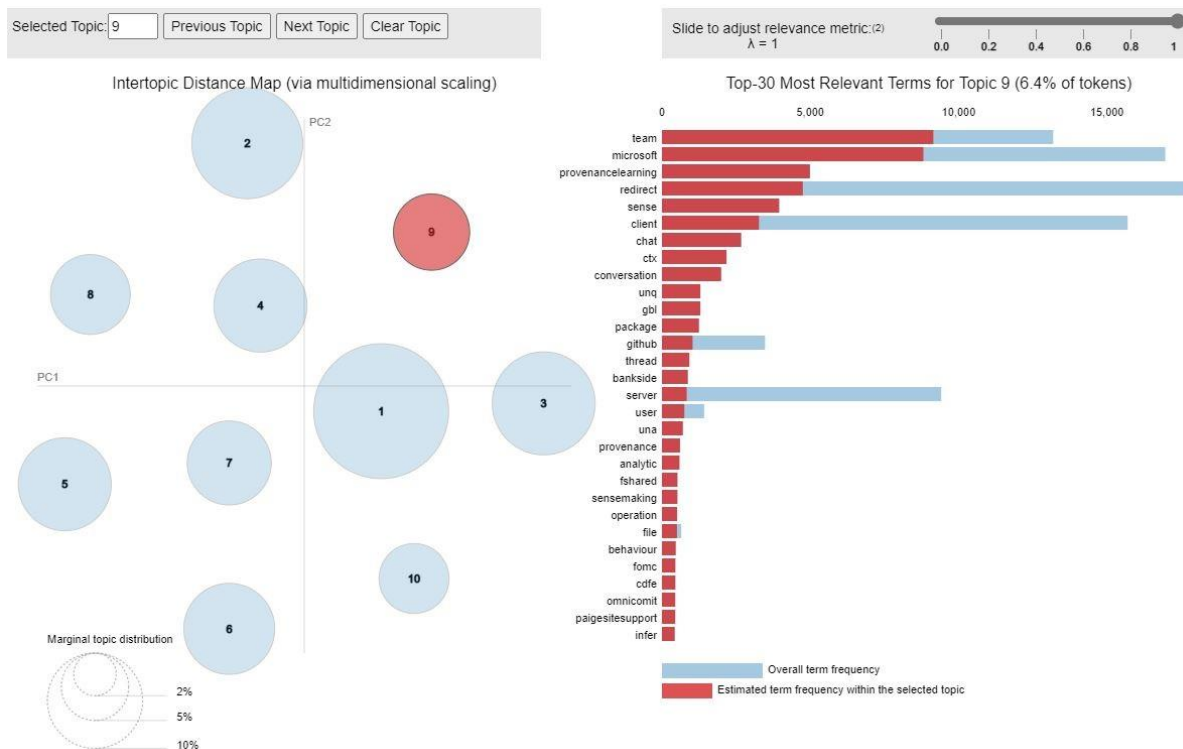


Figure 5.18: Topic predictions visualization by using LDA model while relevance metric $\lambda=1$.

As a computer does not have the ability to make an expert judgement in the same way that a human can, so detecting breakpoints accurately in an automated data-driven manner is a strenuous task. To detect human specified boundaries through learning the most representative features specific to the input time-series data and exploit these features for segmentation, Lee et.al. [115] utilized '*Autoencoder Model*' in deep learning techniques to automatically and effectively extract unique features specific to the input data without making any prior assumption. They calculated distance between two features corresponding to consecutive time windows by using '*Euclidean Distance*' and constructed a distance curve. They selected all the peaks (local maximum) in the curve as breakpoints. Iqbal et. al. [111] used '*Correlated Feature Selection (CFS)*' to extract predictive features and leveraged '*Multilayer Perceptron (MPL)*' to learn model for mapping predictive features to the breakpoint types. Their models were able to detect 69% - 87% of each breakpoint type across

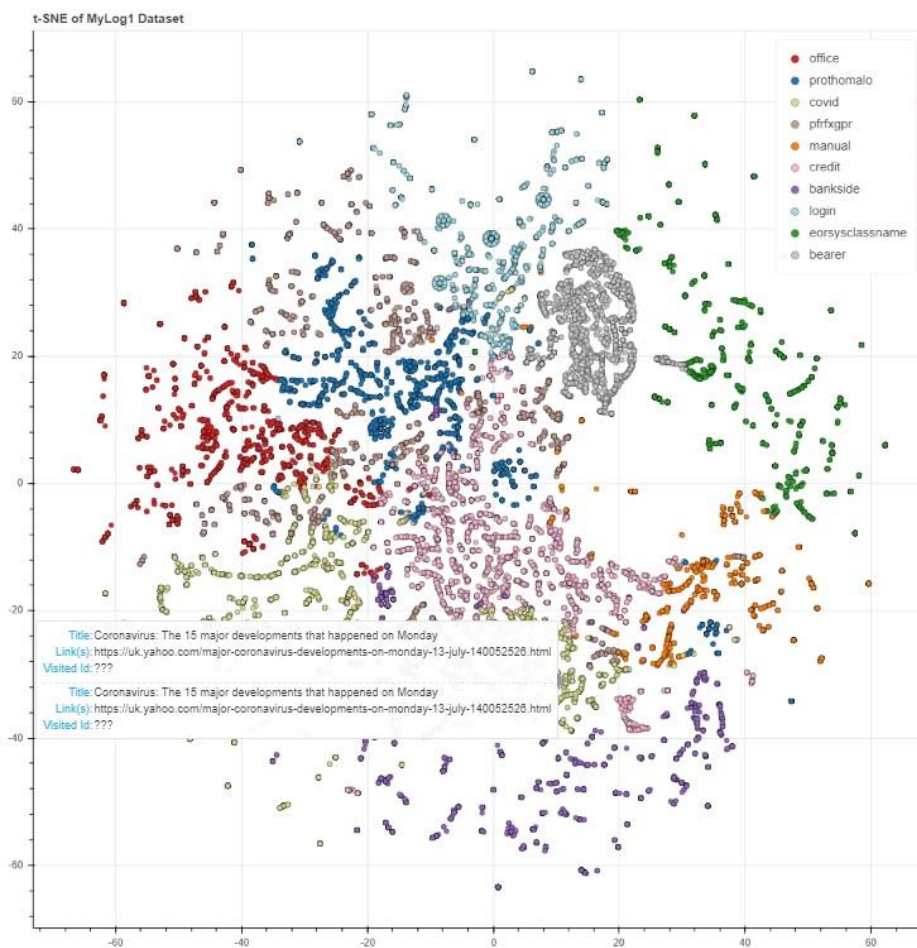


Figure 5.19: t-distributed stochastic neighbour embedding (t-SNE) visualization of inferred chunks in 2D.

tasks. They suggested more sophisticated analysis of the similarities among contents for further improvements.

5.3.4.8 Data Transformation

We have transformed the dataset to chunk semantically similar text corpus and tag those with a name. We transformed the dataset by -

- Pre-processing text input.
- Creating *Document-Term-Matrix (DTM)* based on *Bag of Words (BoW)* and *Term Frequency-Inverse Document Frequency (TF-IDF)*.
- Fitting the *Latent Dirichlet Allocation (LDA)* model on *DTM_TF* and *DTM_TFIDF*.
- Applying grid search to select the best model.

- Training *DTM_TF* and get *LDA_output* based on best model parameters.
- Determining the dominant chunk (based on the topic probabilities) for each row representing site visit.
- Inferring chunks according to their tokens (above).
- Visualizing chunks in 2D as shown in above by using *t-Distributed Stochastic Neighbour Embedding (t-SNE)* to cluster all site visits. Into t-SNE similar objects are modelled by nearby points with higher probabilities and dissimilar objects are modelled by distant points with lower probabilities.

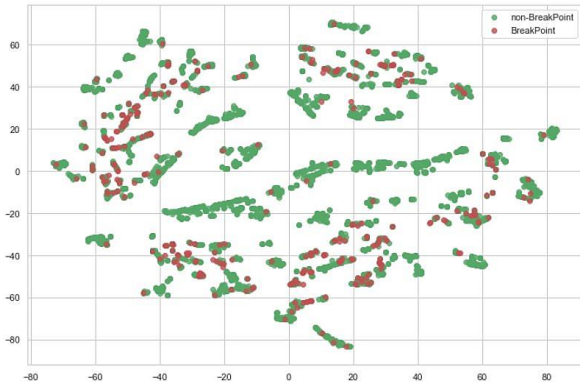
Although Lee et.al. [115] calculated distances between features corresponding to consecutive time windows to form a distance curve and marked all peaks as breakpoint, however from the t-SNE visualization (above) we have found that the breakpoints are not dependent on such feature distances.

5.3.4.9 Implementation

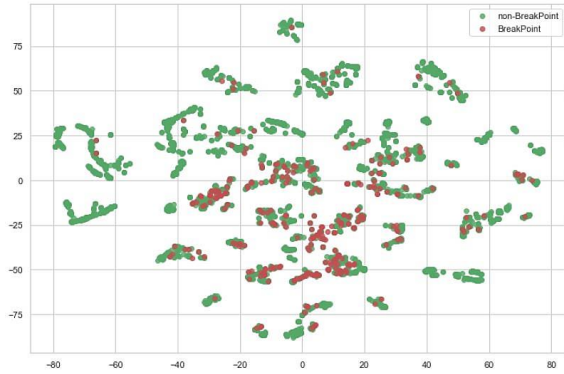
We have used ‘*autoencoders*’ to evaluate how accurately can breakpoints be inferred into unsupervised setting. The idea is to use the reconstruction errors as the limit to separate between non-Breakpoints (lower errors) and breakpoints (higher errors) while deforming back from latent representation to project the actual data with true labels. After applying ‘*autoencoders*’ on our transformed dataset the latent representation looks like below.

We have used the training dataset obtained from latent representations and applied *Logistic Regression* to test how accurately can all breakpoints be inferred into a Semi-Supervised setting. The classification reports have been shown in below. Although the model performs very well into a semi-supervised setting, however the ultimate goal is to evaluate the results of decoded data from it's latent representation as shown above.

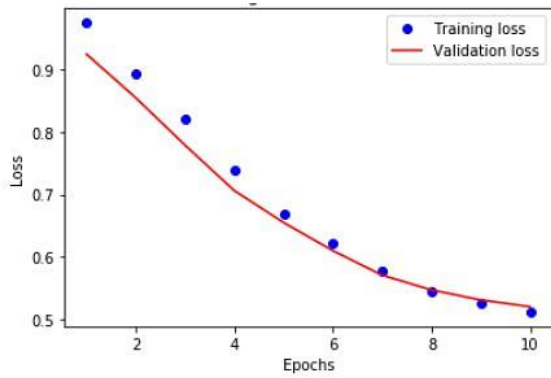
From plots of below we have found decreasing validation loss and increasing validation accuracy. But the model is a bit overfitting at some epochs while being in synch for few cases. We have found $F1=0.1916$ as the best after optimizing



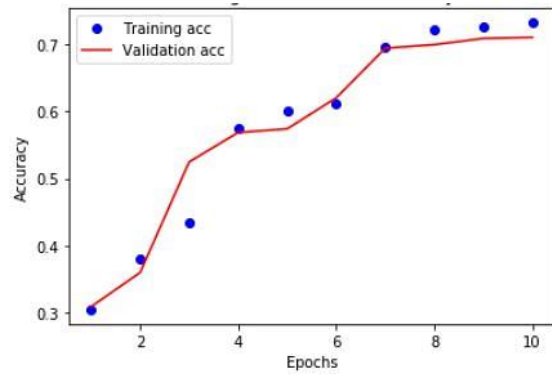
a. Original representation.



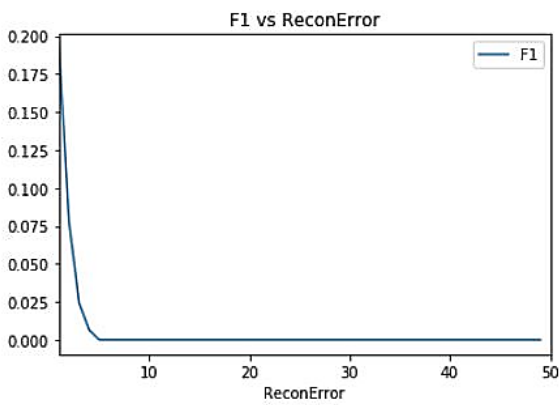
b. Latent representation.



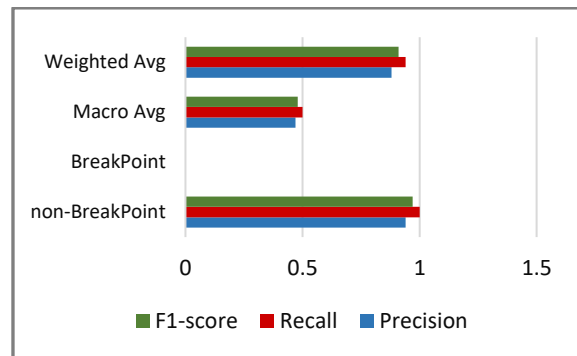
c. Training and validation loss.



d. Training and validation accuracy.



e. F1 vs reconstruction error.



f. Classification report for latent representation.

Figure 5.20: Autoencoder evaluation results for inferring breakpoints.

Table 5.2: Un-supervised model performances of inferring breakpoints.

Model Name	Time to train	Time to predict	ROC AUC score	F1 score
Isolation Forest	1.11 s ± 170 ms per loop (mean ± std. dev. of 7 runs, 1 loop each)	439 ms ± 74.7 ms per loop (mean ± std. dev. of 7 runs, 1 loop each)	0.520737	0.086956
KMeans	105 ms ± 17.3 ms per loop (mean ± std. dev. of 7 runs, 1 loop each)	14 ms ± 871 μs per loop (mean ± std. dev. of 7 runs, 100 loops each)	0.497240	0.043478
Local Outlier Factor (LOF)	113 ms ± 1.13 ms per loop (mean ± std. dev. of 7 runs, 10 loops each)	234 μs ± 13.6 μs per loop (mean ± std. dev. of 7 runs, 1000 loops each)	0.520737	0.086956
One-Class SVM	84.5 ms ± 1.9 ms per loop (mean ± std. dev. of 7 runs, 10 loops each)	55 ms ± 3.99 ms per loop (mean ± std. dev. of 7 runs, 10 loops each)	0.469442	0.076433

the reconstruction error. So, there is clearly scope to improve the performance of the model by introducing more complexities into it that can reduce the error as the plot below show decreasing F1 as shown in above. Besides 'autoencoder', we also have tested with few other un-supervised models as shown into above and compared their performances on accuracies of inferring breakpoints.

Although we have used a labelled dataset, but all of those algorithms do not see the labels while training. The labels in this study are only used to compare the model predictions to the actual values and to create performance metrics.

Hyperparameter Optimization

Hyperparameter optimization is the process of finding the best combination of model parameters (also known as 'tuned') in order to achieve maximum performance on the data. For an example, in the *Random Forest* algorithm *hyperparameters* are the number of estimators (*n_estimators*), maximum depth (*max_depth*) and criterion. We have adopted both 'manual' and 'automatic' approaches for *hyperparameter tuning* and used those data to infer breakpoints into our transformed dataset.

Manual Search (*n_estimators=50*)

We have used *Random Forest Classifier* as the model to optimize. In *Random Forest* each decision tree from ensembled uncorrelated decision trees, makes it's own prediction and the most frequent prediction is selected as model output.

Random Search (*max_depth=100, n_estimators=10*)

In this approach we have used random combinations of the values of the *hyperparameters* to find the best solution for the built model.

Grid Search (*max_depth=100, n_estimators=10*)

For grid search, we have setup a grid of *hyperparameters* and train/test our model on each of the possible combinations.

Bayesian Optimization with HYPEROPT (*max_depth=10, n_estimators=0*)

Bayesian optimization is a model based method for finding the minimum of a function for achieving lowest possible output value for better results with fewer iterations than random search. Bayesian optimization can be performed by using *Hyperopt library*.

Artificial Neural Networks (ANNs) Tuning

(*neurons= 100, optimizer='Adam', epochs= 50, batch_size=1024*)

We have tried to optimize some of ANN parameters i.e., how many neurons to use, which activation function to use etc. It is possible to apply grid/random search into deep learning models by using *KerasClassifier* wrapper.

Tree-based Pipeline Optimization Tool (TPOT)

(*max_depth=2, generations=10, n_estimators=10*)

TPOT is a python library having tree-based structure. It uses a version of genetic programming to automatically design and optimize a series of data transformations and machine learning models to maximize the classification accuracy. For example -

Generation 1 - Current best internal CV score: 0.9344215311731479
Generation 2 - Current best internal CV score: 0.9344215311731479
Generation 3 - Current best internal CV score: 0.9344215311731479
Generation 4 - Current best internal CV score: 0.9344215311731479
Generation 5 - Current best internal CV score: 0.9344215311731479
Generation 6 - Current best internal CV score: 0.9344215311731479
Generation 7 - Current best internal CV score: 0.9344215311731479
Generation 8 - Current best internal CV score: 0.9344215311731479
Generation 9 - Current best internal CV score: 0.9344215311731479
Generation 10 - Current best internal CV score: 0.9347234635402977

OPTUNA Framework (*n_trials = 200*)

We have used *Optuna* as a framework which is designed for the automation and acceleration of the optimization of our study. We have also aimed to find out the optimal set of hyperparameter values (i.e., classifier and svm_c) through multiple trials (*n_trials = 200*). Example of few trial runs are as followed:

[I 2020-11-05 23:45:25,230] A new study created in memory with name: no-name-4883cee5-ead0-4f32-84a9-be40f6c84166

[I 2020-11-05 23:45:25,237] **Trial 0** finished with value: 0.9415081042988019 and parameters: {'classifier': 'SVC', 'svc_c': 4312228.92901136}.

[I 2020-11-05 23:45:25,243] **Trial 1** finished with value: 0.9415081042988019 and parameters: {'classifier': 'RandomForest', 'rf_max_depth': 3.3536092595948013}. Best is trial 0 with value: 0.9415081042988019.

[I 2020-11-05 23:45:25,248] **Trial 2** finished with value: 0.9415081042988019 and parameters: {'classifier': 'SVC', 'svc_c': 14348.079711843067}.

.....

[I 2020-11-05 23:45:27,975] **Trial 198** finished with value: 0.9415081042988019 and parameters: {'classifier': 'RandomForest', 'rf_max_depth': 2.4047586401115297}.

[I 2020-11-05 23:45:27,992] **Trial 199** finished with value: 0.9415081042988019 and parameters: {'classifier': 'RandomForest', 'rf_max_depth': 5.216269406900703}.

5.3.4.10 Discussion

Part-1 of experiment-2 as described into Section 5.3.4.1 showed promising performance of inferring human specified breakpoints. But in case of unknown tasks or highly variable interactions of multi-users into free-form tasks, it will not be feasible to adopt previous approach due to cognitive and perceptual variances of different users. So, into part-2 of experiment-2 (Section 5.3.4.7), we have proposed a data-driven unsupervised approach where all used algorithms do not see the labels while training. All labels have only been used to compare the model predictions to the actual values and create performance metrics.

At this stage of experiment we transformed the dataset to chunk semantically and tag those with the aim to test either breakpoints are changepoints, outliers or far distant features among semantic chunks as mentioned into other related literatures [111, 115]. Then we have chunked dataset according to topic probabilities and modelled similar objects by nearby data points with higher probabilities and dissimilar objects by distant points with lower probabilities. After applying 'ChangeFinder Algorithm' we have found (as shown in above) that it can mostly detect changepoints among

Table 5.3: Classification reports after hyperparameter optimization.

Model Name		non-BreakPoint	BreakPoint	Accuracy	Macro Avg	Weighted Avg	
Hyperparameter Optimization	Manual Search <i>n_estimators=50</i>	0.95	0.50	0.94	0.72	0.92	Precision
		0.99	0.16		0.57	0.94	Recall
		0.97	0.24		0.60	0.94	f1-score
	Random Search <i>max_depth=100, n_estimators=10</i>	0.94	0.83	0.94	0.89	0.94	Precision
		1.00	0.06		0.53	0.94	Recall
		0.97	0.11		0.54	0.92	f1-score
	Grid Search <i>max_depth=100, n_estimators=10</i>	0.95	0.63	0.95	0.79	0.93	Precision
		0.99	0.14		0.57	0.95	Recall
		0.97	0.24		0.60	0.93	f1-score
	Bayesian HYPEROPT <i>max_depth=10, n_estimators=0</i>	0.94	0.00	0.94	0.47	0.89	Precision
		1.00	0.00		0.50	0.94	Recall
		0.97	0.00		0.48	0.91	f1-score
Artificial Neural Networks <i>neurons= 100, optimizer='Adam', epochs= 50, batch_size=1024</i>	0.94	0.00	0.94	0.47	0.89	Precision	
	1.00	0.00		0.50	0.94	Recall	
	0.97	0.00		0.48	0.91	f1-score	
TPOT (Tree Based Pipeline Optimization Tool) <i>max_depth=2, generations=10, n_estimators=10</i>			0.94			Overall	
OPTUNA Framework <i>n_trials = 200</i>			0.94			Overall	

usual trends based on anomaly scores but those are not always indicative of breakpoints. Those changepoints are statistical boundaries with abrupt changes (i.e., mean, variance etc) into trends of data. On the otherhand, we have found from the

latent representation of autoencoder [above] that breakpoints can be present among the nearby datapoints with higher probabilities. As well as we have found from above that there might be semantic overlaps of different chunk data points. For an example – the chunk ‘covid’ is overlapping with other chunks such as ‘office’, ‘credit’ and ‘prothomalo (a newspaper)’. It may be due to it’s impact on those issues although those are not semantically related issues. So, dissimilar objects of distant points with lower probabilities are not only the indicatives of breakpoints. In such scenario it is hard to infer breakpoints with high accuracy. To further experiment this finding, we have applied ‘*LOF (Local Outlier Factor)*’ algorithm which looks at the local neighbourhood of a data point and measures the local deviation of density of a given sample with it’s neighbour. We have found F1 accuracy: 0.086956 and ROC AUC score: 0.520737. So, the outlier model is in flip of a toss situation according to ROC AUC score while inferring those data points as breakpoints. We also have tested with some other unsupervised models i.e, K-means, One-Class SVM as shown in Table 5.2 but found worse results than LOF. The autoencoder model as we applied, also returned decreasing F1 score which means it could mostly infer non-breakpoints with lower errors. Because we have considered higher reconstruction errors as presence of breakpoints.

At the later stage of the experiment, we have tested with few manual and automatic model parameter tuning techniques known as ‘*Hyperparameter Optimization*’ in order to find the best combination and improve performance of breakpoints inference on the data. We have achieved success by adopting this technique. As shown in above, ‘*Artificial Neural Networks (ANN)*’ with it’s best combination of neurons and batch_size achieved F1 score of 0.91; *Tree based Pipeline Optimization Technique* with it’s best estimator achieved 94% as overall accuracy after 10 generations run. We also have tested on ‘*OPTUNA Framework*’ which is an automated optimization approach and also achieved 94% as overall accuracy after 200 trial runs.



Validations



6



chapter

6.1 Chapter Overview

We have conducted experiments to infer “*where the breakpoints are*” by applying both supervised and unsupervised machine learning models in chapter 5 and evaluated those based on their classification reports. However, reliability, accuracy, relevance of those results as part of transparent validation of outcomes are still in dark to judge. Knowing which features have positive/negative impacts on prediction results and how are those influencing the models are important for building human trust on machine produced results. To uncover all of those from black-boxes, we have endeavoured to use *eXplainable AI (XAI)* techniques at this stage of research.

The more explainable a model, the deeper the understanding that humans achieve in terms of the internal procedures that take place while the model is training or making decisions [58]. For an example – ‘*Random Forest*’ is one of the classifiers, used for different approaches of hyperparameter optimization in Section 5.3.4.9 for finding the best combination of model parameters for improving breakpoint inference making results. But it was not possible to understand those model outcomes in terms of 3WH questions as mentioned in Section 2.5 and ensure trust on those results. So, following research question has been considered to find out techniques of explaining machine learning model:

RQ8: How to validate inference making results for building trust on machine learning models and maintain transparency?

- To address the research question (RQ8), we have implemented some model explanation methods in this chapter to provide further explanations on model outcomes of previous chapter and validate those results. As an approach we have adopted ‘*post hoc model agnostic*’ method and attempted to find out best fit XAI techniques for it. ‘*SHAP (SHapley Additive exPlanations)*’ as a post hoc local and global interpretations, ‘*LIME (Local Interpretable Model-Agnostic Explanations)*’ as a local model agnostic algorithm, ELI5 as post hoc feature importance method have been used in this chapter. To better understand and interpret model’s decision making steps, we also have built decision trees and visualized the decision path.

Details on machine learning models’ explanation techniques and visualizations of this chapter can be found from-

<https://github.com/Vis4Sense/ProvenanceLearning/issues/20>

6.2 XAI with SHAP

The Shapley value is the average contribution of a feature value to the prediction in different coalitions. For each of coalitions prediction is calculated with or without the feature value. The feature value is the numerical or categorical value of a feature and instance; the Shapley value is the feature contribution to the prediction. The effect of each feature is the weight of the feature times the feature value. SHAP (*SHapely Additive exPlanations*) provides following salient propositions:

6.2.1 Prediction Explainer

6.2.1.1 Local Interpretability

The local interpretability enables to pinpoint and contrast the impacts of individual feature. The above explanation shows features each contributing to push the model output from the *base value* (the average model output over the training dataset we passed) to the model output. Features pushing the prediction '*higher*' are shown in '*red*', those pushing the prediction '*lower*' are in '*blue*'. Let's describe the plot in more detail:

- $f(x) = -4.02$ is the output prediction value of `X_train.iloc[0,:]` as shown in below.
- The base ($y_{\hat{}}$) = -3.182, which is the value that would be predicted if no features of the current output was known.
- Red/blue: Features that push the prediction higher (to the right) are shown in red, and those pushing the prediction lower are in blue.

It is predicted -1.28 for `X_train.iloc[421,:]`, whereas the `base_value` is -3.182 [below]. Magnitude of the pink feature values are larger in this scenario, causing better prediction than base value. Although '*Visit_Count*' is negatively related to the model predictor however it pushes the prediction to the right with higher magnitude. '*Elapsed_Time*' also has negative impact driving the prediction to the left.

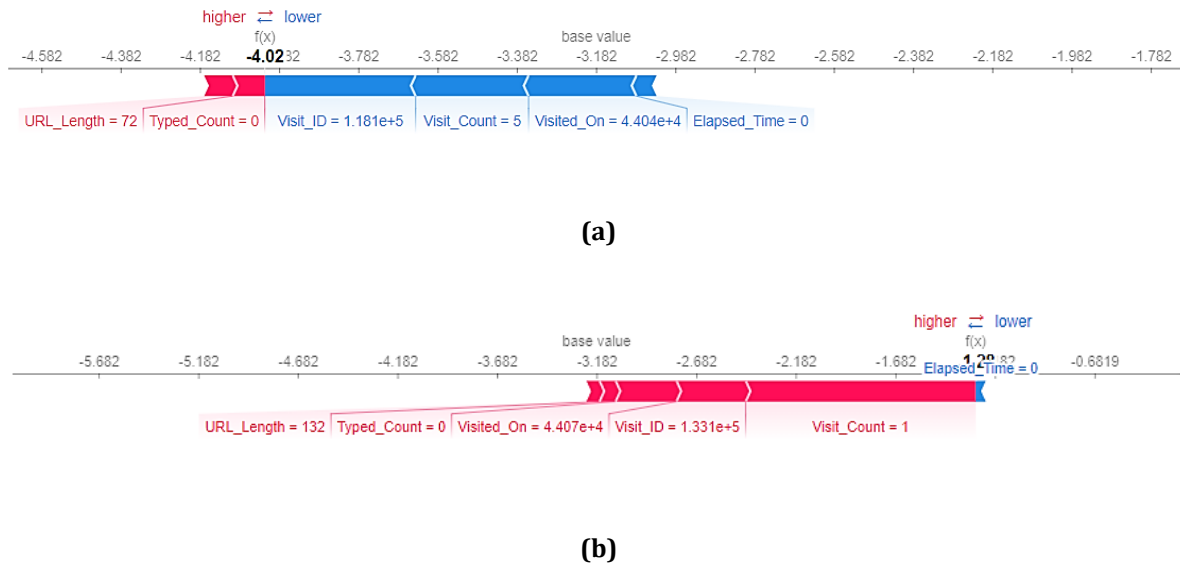


Figure 6.1: (a) $X_{train}.iloc[0,:]$, (b) $X_{train}.iloc[421,:]$.

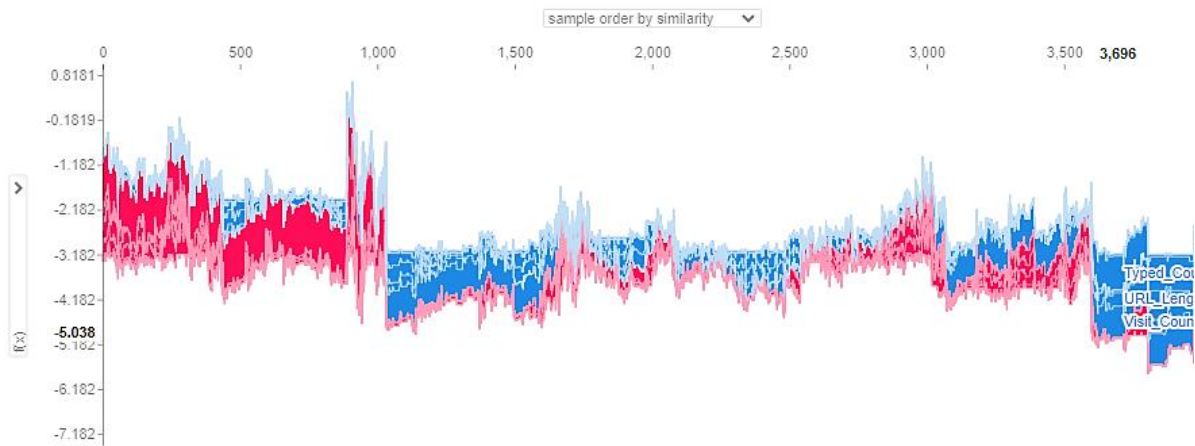
6.2.2 Model Explainer

6.2.2.1 Global Interpretability

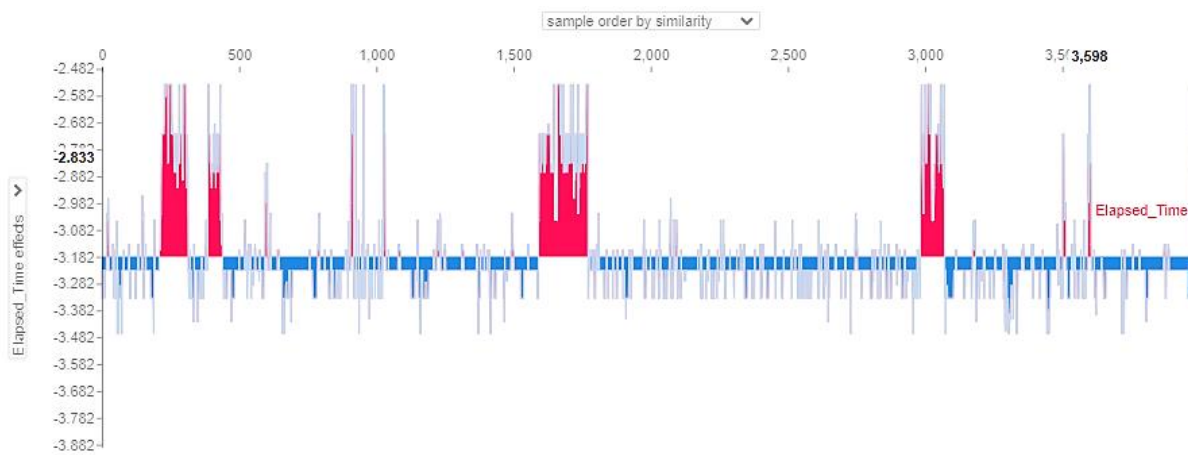
If we take many explanations (force plots) such as the one shown above, rotate them 90 degrees, and then stack them horizontally (combined force plots), we can see explanations for an entire data X_{test} as shown below [below]. It has the purpose of explaining the model as a whole. It essentially has all the X_{train} data plotted on the x-axis (in this case, ordered by similarity, but it can be changed from the drop-down box) and their prediction values plotted on the y-axis. Also, it has the individual contributions of each feature for each sample, based on feature value.

6.2.2.2 Elapsed_Time Effects

In this example, we selected sample number 3598 (x -axis) to see its 'Elapsed_Time' effects on predictions and found value of -2.833 (y -axis) as shown in below. This is how by just hovering over other samples, it is possible to see how feature values and their impact change, as well as the predictions.



(i)



(ii)

Figure 6.2: Shapley value - **(i)** Global interpretability, **(ii)** Elapsed Time Effects.

6.2.2.3 Dependence Plot

The dependence plot in Figure 6.3 shows how 2 features are related to one another in terms of their impact in the model, measured by a SHAP value (a measure of feature relevance in the model). When we specify a feature, 'Dependenc_Plot' function will automatically pick up another feature which has the strongest dependency in terms of relevance with the feature. We can see from the above plot that there is a non-linear relationship between 'URL_Length' and the target variable; the 'URL_Length' has strong relevance with the 'Visit_Count' although negative relationship exists as shown in below.

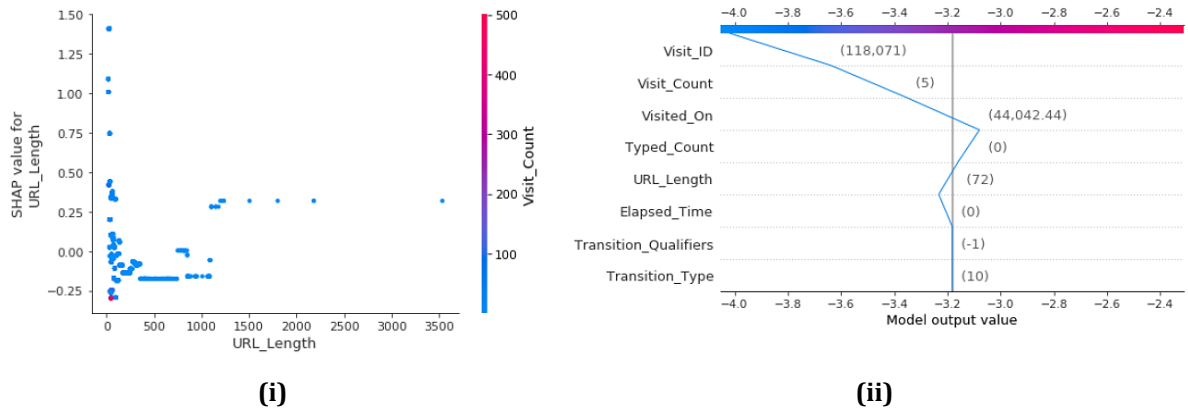


Figure 6.3: (i) Dependence plot, (ii) Decision plot.

6.2.2.4 Decision Plot

SHAP decision plots show how complex models arrive at their predictions (i.e., how models make decisions). Moving from the bottom of the decision plot to the top, SHAP values for `X_train.iloc[0, :]` feature are added to the model's base value. This shows how each feature contributes to the overall prediction. At the bottom of the plot, the observations converge at `explainer.expected_value [1] = -3.2` which is the model's base value as shown in above.

6.2.2.5 Summary Plot

To get an overview of which features are most important for a model we can plot the SHAP values of every feature for every sample. The plot below [below] sorts features by the sum of SHAP value magnitudes over all samples, and uses SHAP values to show the distribution of the impacts each feature has on the model output. The colour represents the feature value (red high, blue low).

From the plot below [below], we can see that *Visit_Count* was considered as one of the bottom ranked LightGBM features based on importance averaged over folds, however it's average impact on overall model output is the top most one based on mean SHAP value.

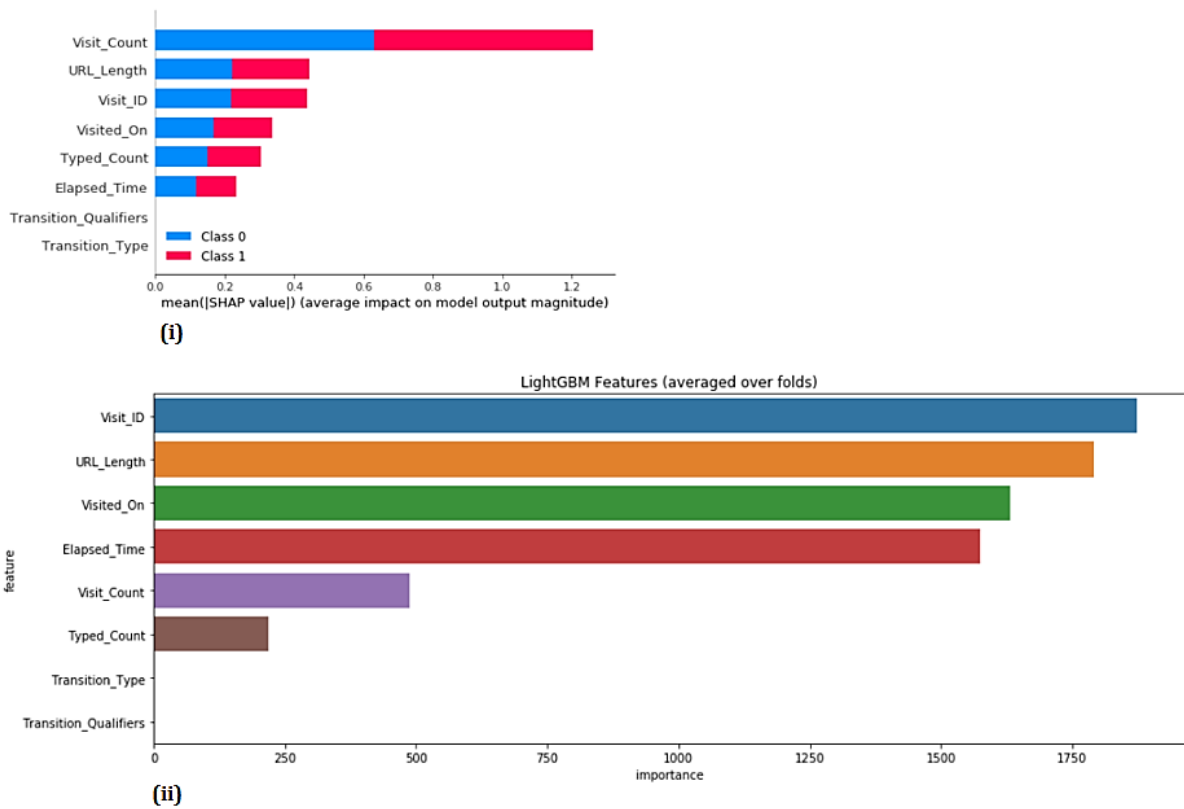


Figure 6.4: (i) Summary plot, (ii) LightGBM feature importance.

6.3 XAI with LIME

LIME (*Local Interpretable Model-Agnostic Explanations*), also known as 'local fidelity' can explain the predictions of any classifier in an interpretable and faithful manner by learning a model locally around the prediction. LIME uses a local surrogate model trained on perturbations of the data point we are investigating for explanations. Approaches taken by LIME to achieve this goal are as followed:

- For each prediction to explain, permute the observation n times.
- Let the complex model predict the outcome of all permuted observations.
- Calculate the distance from all permutations to the original observation.
- Convert the distance to a similarity score.
- Select m features best describing the complex model outcome from the permuted data.

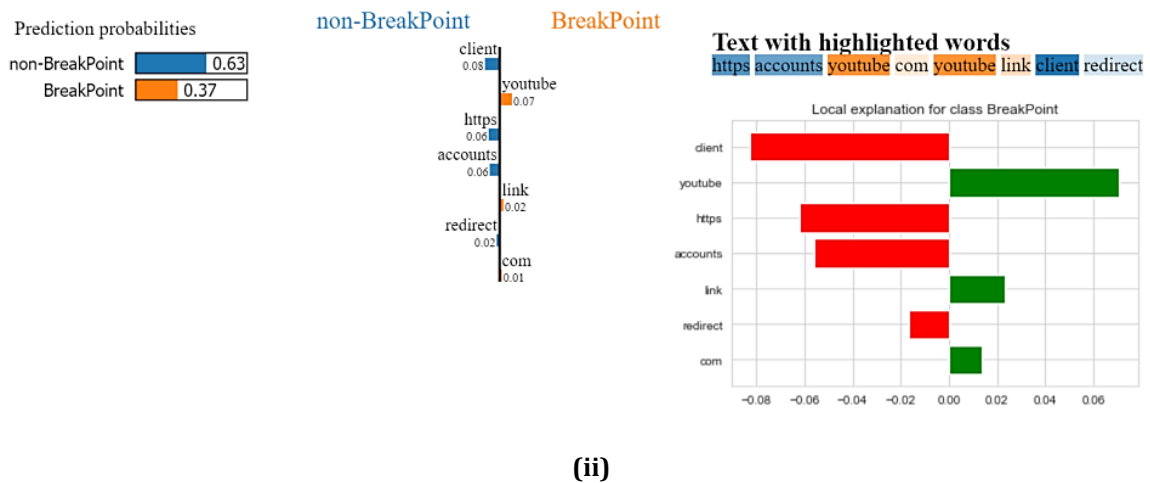
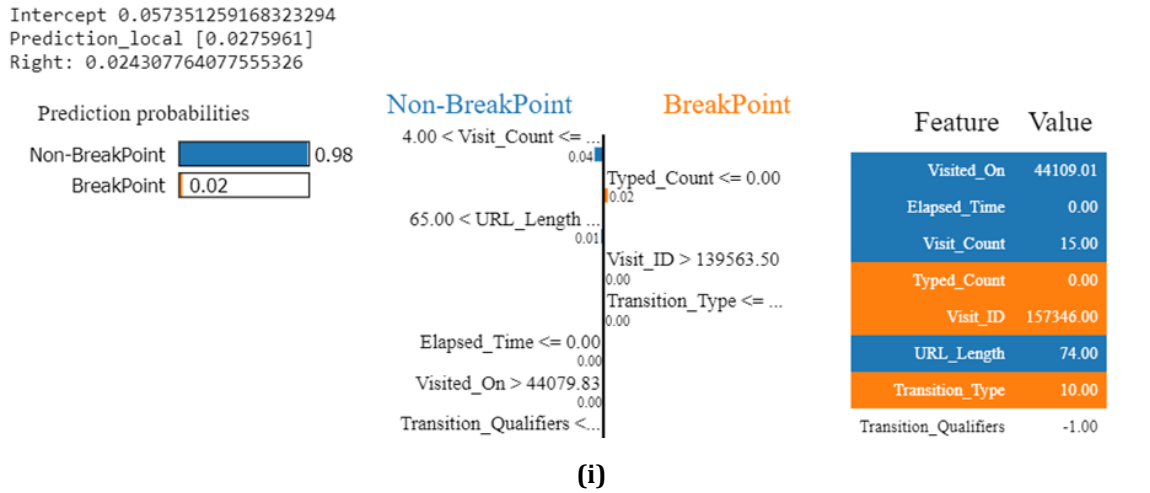


Figure 6.5: LIME - (i) Tabular representation and (ii) Text highlights for test.loc[310].

- Fit a simple model to the permuted data, explaining the complex model outcome with the m features from the permuted data weighted by its similarity to the original observation.
- Extract the feature weights from the simple model and use these as explanations for the complex models local behaviour.

We have used RF (*Random Forest*) model to classify breakpoints and considered X_test[310] as a test sample to understand it's features' local contribution weights towards the overall predictions. As our dataset is highly imbalanced, we also did over-/under-sampling for producing more reliable predictions on the text part our dataset.

We have used tf-idf to vectorize the text dataset and logistic regression model for classifying breakpoints. As shown in above, LIME has been used to visually interpret their weights in both tabular and highlighted text formats.

As shown in above, LIME model *prediction_local* can be obtained by adding the total of the coefficients with model intercept. *Right* is the original LightGBM prediction. *LIME for text* differs from *LIME for tabular data*. Variations of the data are generated differently: Starting from the original text, new texts are created by randomly removing words from the original text. The proximity of the variation to the original text can be calculated as 1 minus the proportion of words that were removed, for example if 1 out of 7 words is removed then the proximity will be $1 - 1/7 = 0.86$. As shown above, '*negative (blue)*' words indicate '*non-BreakPoint*', while '*positive (orange)*' words indicate '*breakPoint*'. The way to interpret the weights is by applying them to the prediction probabilities. For example, if we remove 'client', 'accounts' words from the text, we expect the classifier to predict '*non-BreakPoint*' with probability $0.63 - 0.08 - 0.06 = 0.49$. On the otherhand, features that have '*positive correlations*' with the target are shown in '*green*', otherwise '*red*' i.e, $youtube > 0$ is positively and $client < 0$ is negatively correlated with the breakpoint prediction for `test.loc[310]`.

6.4 Explaining Model's Decision Making

We have used *LightGBM* for the above classification task. To better understand and interpret the model's decision making, we have built decision trees as shown in below. Nowadays, the most used and the most performant types of machine learning algorithms are ensemble of *decision Trees (RandomForest, XGBoost, LightGBM)* for structured data and Deep Learning. We shall also use *dtreeviz* to provide explanatory visualizations for tree structure, leaf nodes information, prediction path etc.

The decision tree classifier *dtreeviz* visualization as shown in below, uses node size to give visual cues about the number of samples associated with each node. Histograms get proportionally shorter as the number of samples in the node decrease and leaf node diameters get smaller. The feature space (horizontal axis) is always the same width and the same range for a given feature, which makes it much easier to compare the feature-target spaces of different nodes. The bars of all histograms are the same width in pixels. Pie-charts for classifier leaves are indicating purities.

We have used a stacked histogram so that overlap is clear in the feature space between samples with different target classes. The height in the Y axis of the stacked histogram is the total number of samples from all classes; multiple class counts are stacked on top of each other.

6.4.1 Visualizations of Leaves Impurities

The goal for splitting a node (classification) is to create another two nodes as pure as possible. Decision Tree's performance depends on the performance of each individual leaf. So, it will be very helpful to understand/visualize what's happening into each leaf, because those as a whole lead to final prediction. A leaf contains information about the number of samples and its purity. Purity measures the distribution of target class values in each node. To measure the node purity, the most popular formulas are Gini (between [0, 0.5]) and Entropy (between [0, 1]).

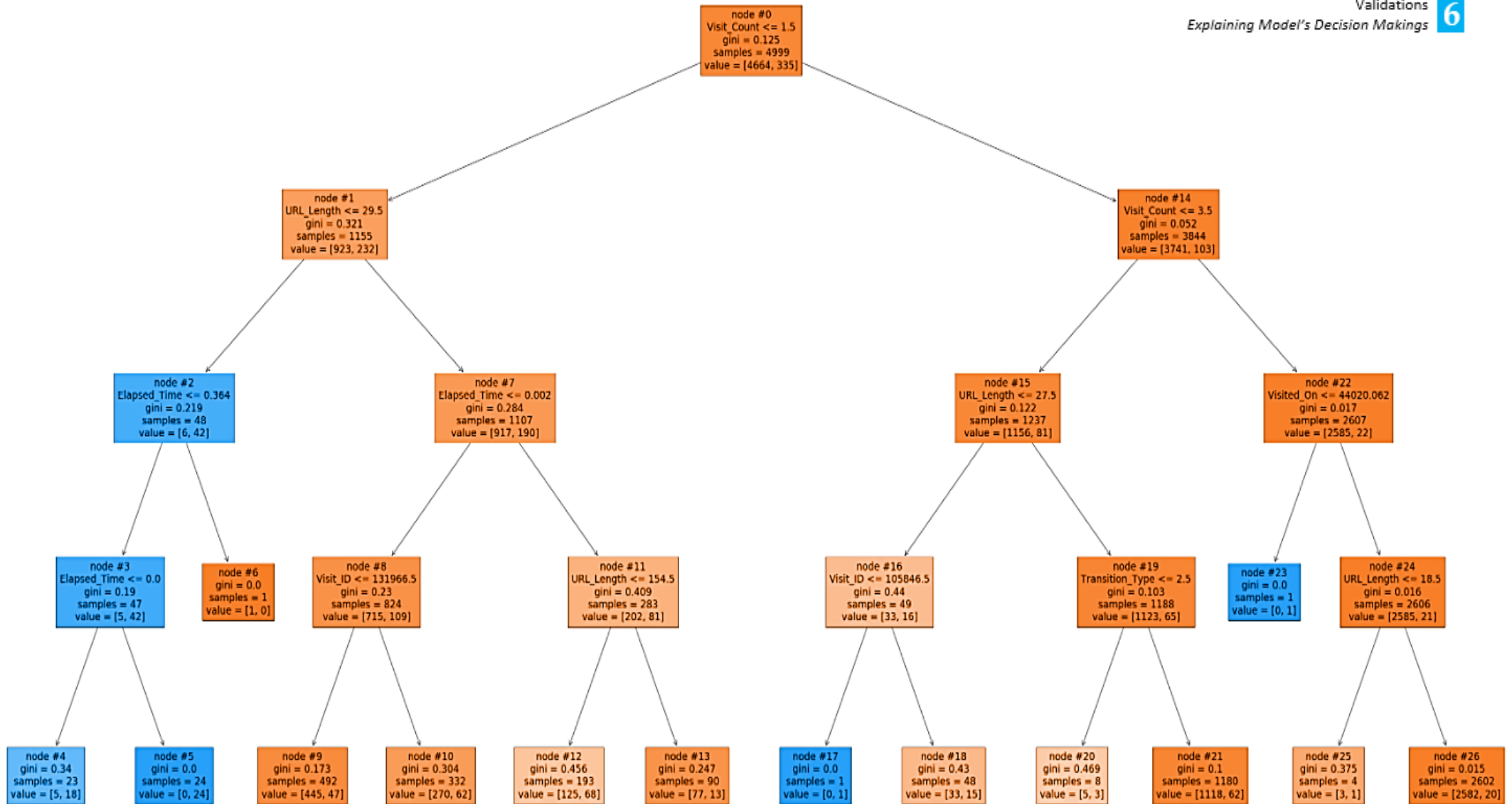


Figure 6.6: Scikit-learn visualization of decision trees for max_depth=4, random state = 310 (test.loc[310]).

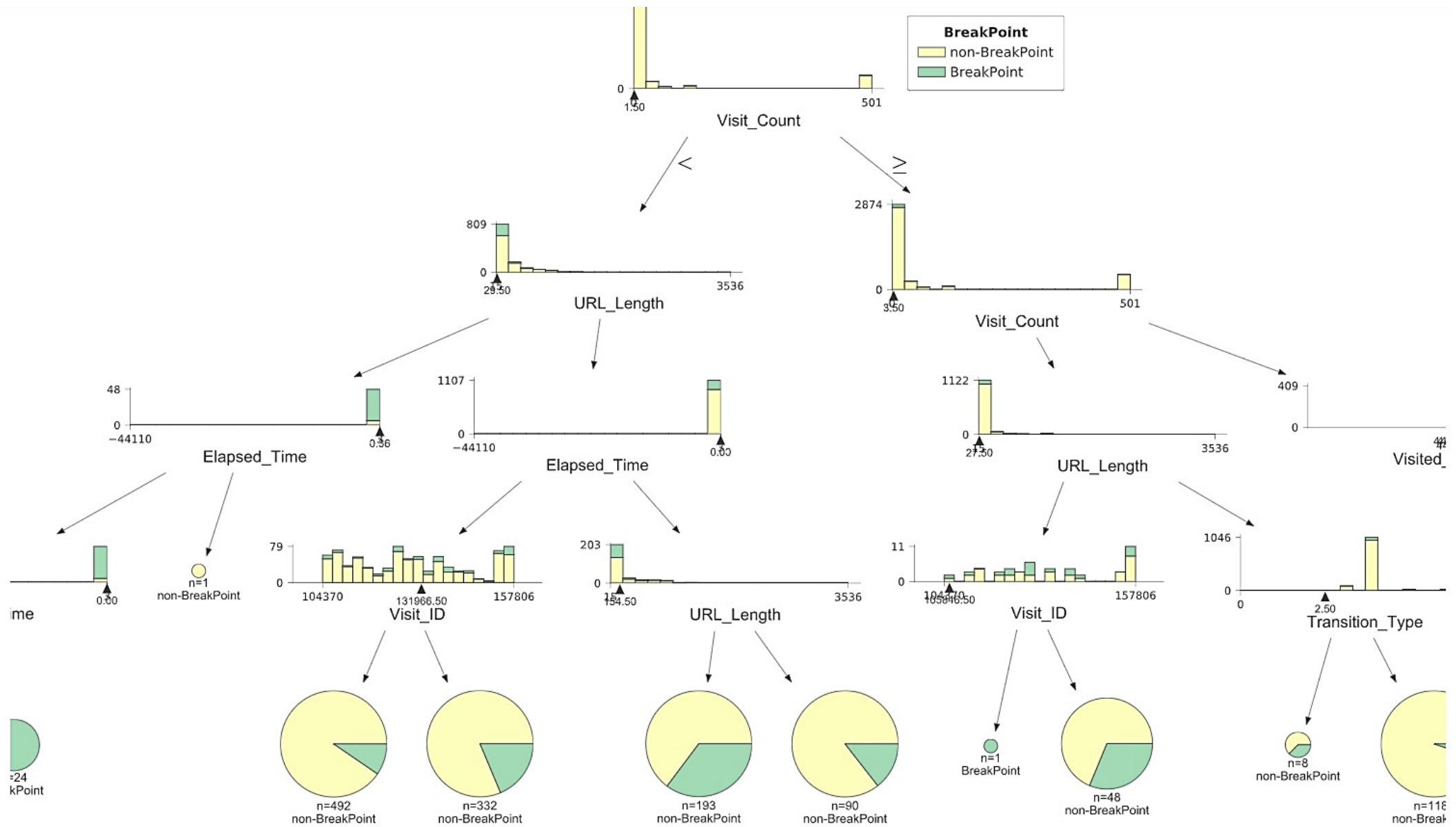


Figure 6.7: dtreeviz visualization of decision tree classifier for max_depth=4, random state = 310 (test.loc[310]).

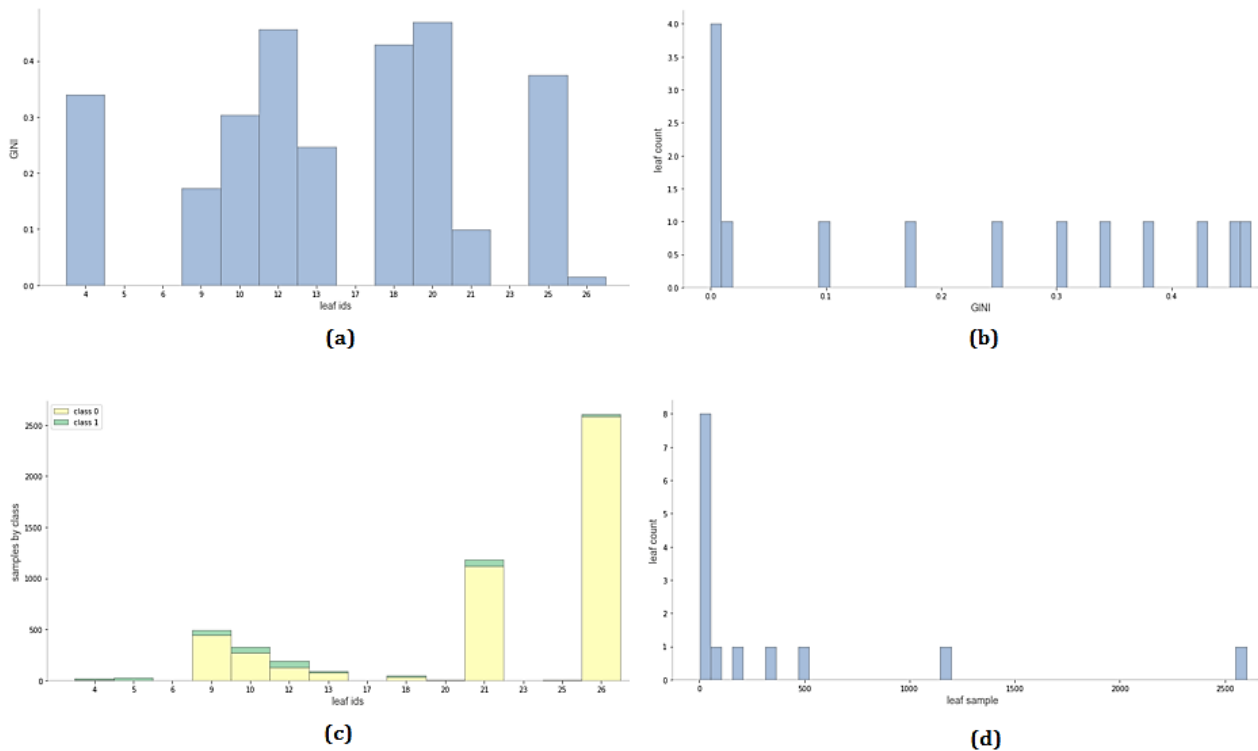


Figure 6.8: (a) Gini purity for each leaf, (b) Leaves purities distribution, (c) Number of leaves grouped by target class, (d) Leaves sample distribution.

As shown into scikit-learn visualization of decision tree (above), the ideal scenario of Gini is, when the value = 0 and that node contains only values from a single target class. If the value = 0.5 and the node contains values from all target classes in equal proportions, then it is the worst case scenario. Lastly, $0 < \text{Gini} < 0.5$ indicates how pure or impure are the node samples.

We have from above that quite a few leaves with purity close or very close to 0. This is the ideal scenario, but we need to take into account also the number of samples from these leaves. If we have a leaf with purity 0, but only very few samples into it, we cannot be very confident for its predictions, because it's based on few samples. It can be a clear sign of overfitting.

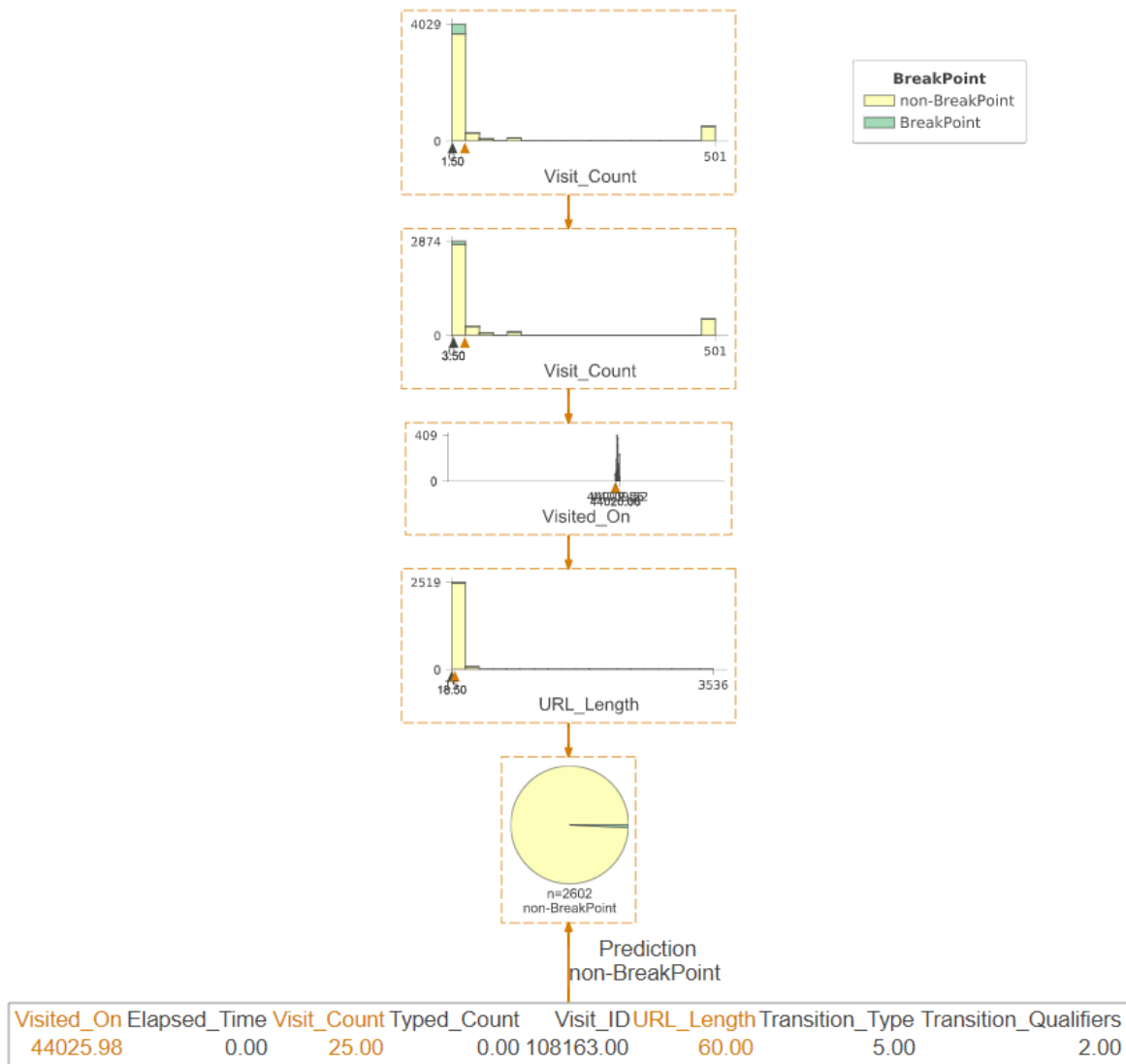


Figure 6.9: Prediction path visualization for random state = 310 (test.loc[310]).

6.4.2 Decision Tree Regressor

The decision of making strategic splits heavily affects a tree's accuracy. The decision criteria is different for classification and regression trees. Decision trees regression normally use *Mean Absolute Error (MAE)* to decide to split a node in two or more sub-nodes. Regression is calculated based on target variance values which measures how spread are the values out from their average. Leaves with low variance among the target values (regression) are much more reliable predictors.

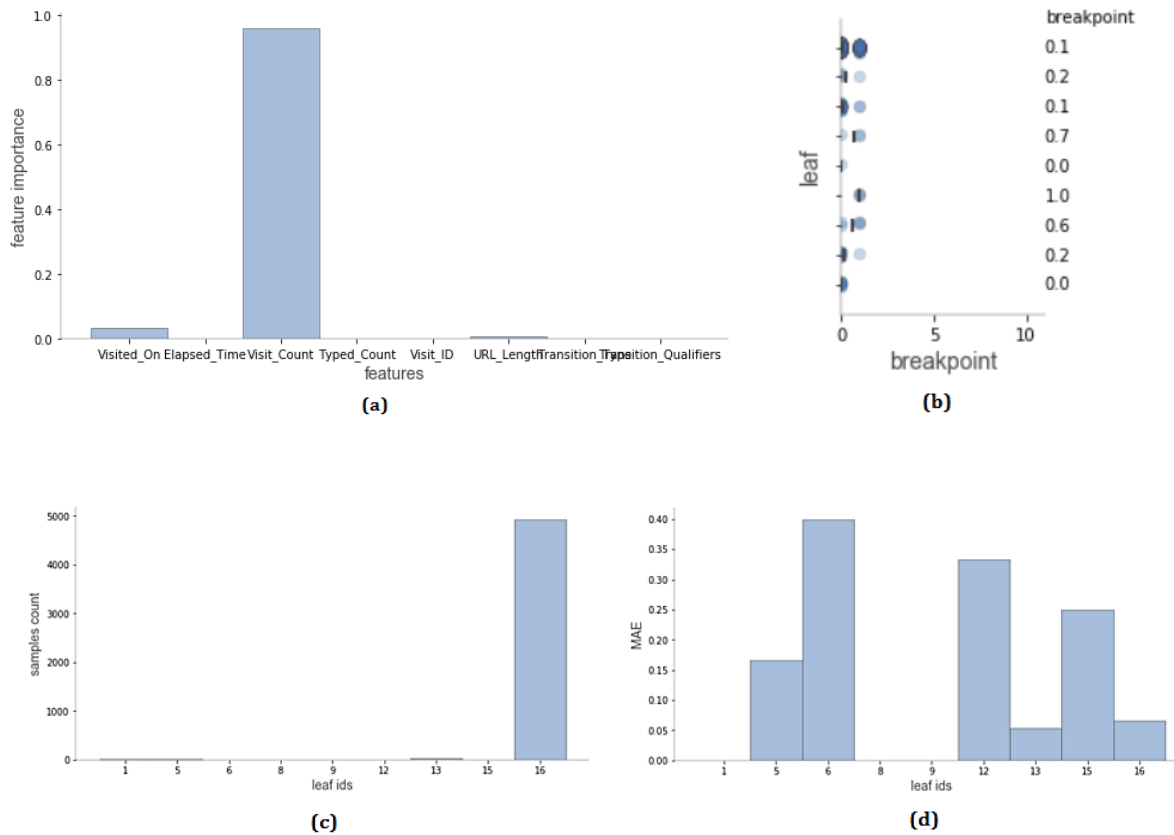


Figure 6.10: (a) Feature importance based on prediction path nodes, (b) Leaf target distribution for regression decision trees, (c) Number of samples from each leaf, (d) Mean Absolute Error (MAE) for each leaf.

As shown in the visualization of below, each splitting node represent a scatter plot between features and target value. The vertical dashed line shows the splitting value and the other two horizontal dashed lines show the mean target values for each subset. Leaf nodes visualize the target values from its samples and the horizontal line represents the target mean value, which is the leaf prediction. For our use-case, the splitting process tries to find another two subsets where each subset has low variance for *breakPoint* values. The 'blue circles' as shown in above represents leaf target distribution and the small 'vertical black lines' are leaf predictions.

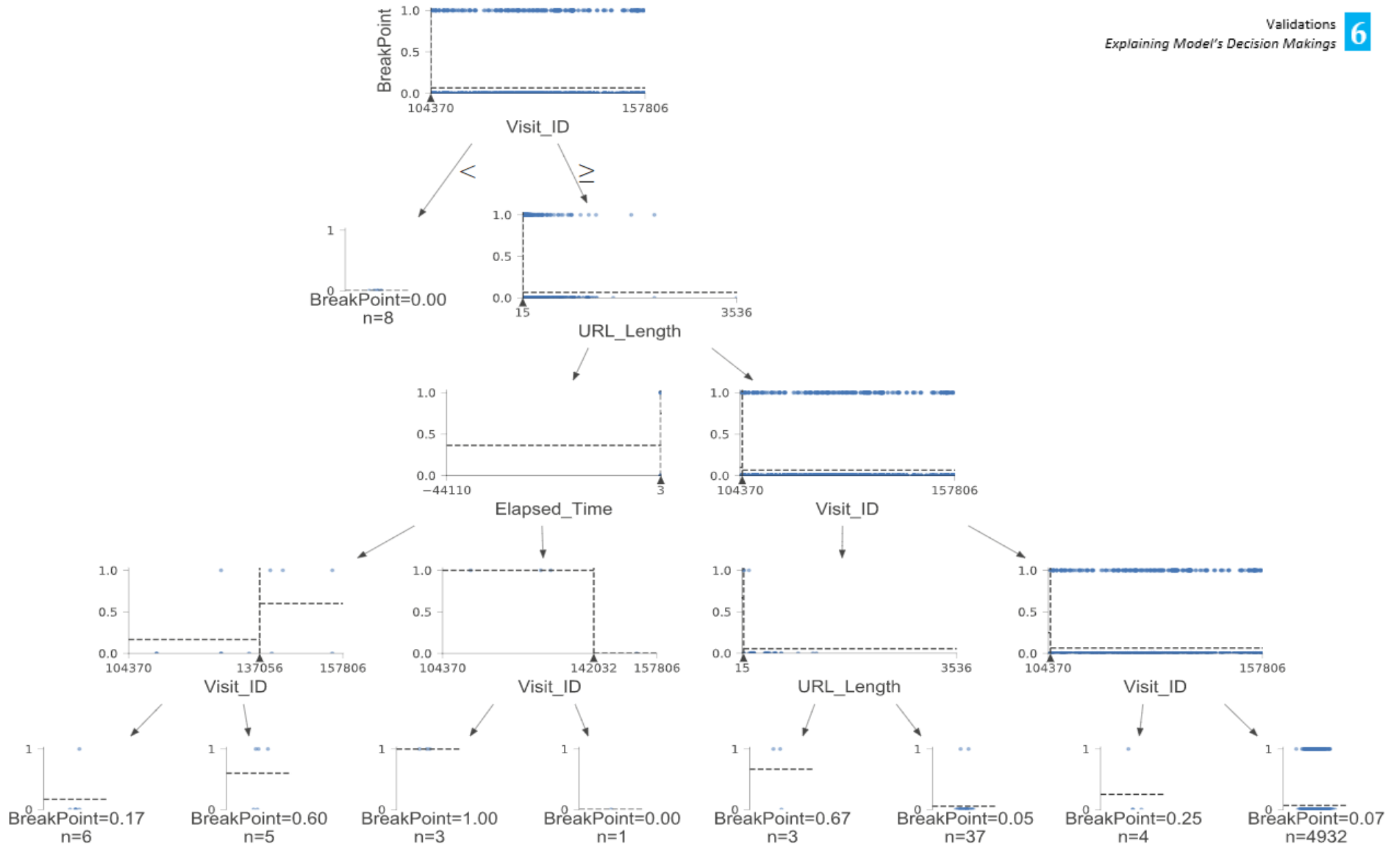


Figure 6.11: Decision tree regressor for max_depth=4, random state = 310 (test.loc[310]).

6.5 Feature Importance

As described in previous sections, the models can only classify every interaction as either breakpoint or non-breakpoint. What if, after the interaction is classified as breakpoint, the analyst would like to know why the model made this decision, i.e., how much each feature contributed to the final outcome? We have used ‘*Random Forest*’ as the classifier in previous section, which consists of a large number of deep trees, where each tree is trained on bagged data using random selection of features, so gaining a full understanding of the decision process by examining each individual tree is infeasible. Furthermore, even if we are to examine just a single tree, it is only feasible in the case where it has a small depth and low number of features. A tree of depth 10 can already have thousands of nodes, meaning that using it as an explanatory model is almost impossible. One way of getting an insight into a random forest is to compute feature importances, by permuting the values of each feature one by one and checking how it changes the model performance. The idea is that feature importance can be measured by looking at how much the score (i.e., accuracy, F1, R^2 , etc. - any score we’re interested in) decreases when a feature is not available. This method is also known as ‘*Permutation Importance*’ or ‘*Mean Decrease Accuracy (MDA)*’.

When considering a ‘*Decision Tree*’, it is intuitively clear that for each decision that a tree (or a forest) makes there is a path (or paths) from the root of the tree to the leaf, consisting of a series of decisions, guarded by a particular feature, each of which contributes to the final predictions. A decision tree with M leaves divides the feature space into M regions R_m , $1 \leq m \leq M$. The prediction function of tree can be defined as $f(x) = \sum_{m=1}^M c_m I(x, R_m)$ where M is the number of leaves in the tree, R_m is a region in the feature space, c_m is a constant corresponding to region m and finally I is the indicator function [116]. The definition is concise and captures the meaning of tree: the decision function returns the value at the correct leaf of the tree. But it ignores the operational side of the decision tree, namely the path through the decision nodes and the information that is available there. More on operational way, these

predictions can be defined through the sequence of regions R_m which gets divided by M leaves of a decision tree. Since each decision is guarded by a feature, and the decision either adds or subtracts from the value given in the parent node, the prediction can be defined as the sum of the *feature contributions* + the “BIAS” (i.e. the mean given by the topmost region that covers the entire training set). So, at this stage the prediction function for a tree can be rewritten as –

$$f(x) = c_{full} + \sum_{k=1}^K contrib(x, k)$$

Where K is the number of features, c_{full} is the value at the root of the node and $contrib(x, k)$ is the contribution from the k -th feature in the feature vector x [116]. Since the prediction of a forest is the average of the predictions of its trees –

$$f(x) = \frac{1}{J} \sum_{j=1}^J f_j(x)$$

Where J is the number of trees in the forest. From this, it is easy to see that for a forest, the prediction is simply the average of the bias terms plus the average contribution of each feature [35]:

$$f(x) = \frac{1}{J} \sum_{j=1}^J c_{j\ full} + \sum_{k=1}^K \left(\frac{1}{J} \sum_{j=1}^J contrib_j(x, k) \right)$$

This has resemblance to linear regression $y = c + m_1x_1 + m_2x_2 + \dots + m_nx_n$ where independent variables are ‘*features to predict*’ and the dependent variables are ‘*features to be predicted*’. The coefficients ‘ m ’ multiplying independent variables ‘ x ’ show the relationships between the dependent and independent variables. For the current part of experiment, we have used ‘*Ridge Regression*’ as a classifier to unleash the most/least important features contributing to actual prediction. *Ridge Regression (RR)* is an example of estimating coefficients of multiple-regression models in machine learning where independent variables are highly correlated.

y top features

Weight?	Feature
+4.632	<BIAS>
+1.046	Title__redirecting
+0.980	URL__3d
+0.865	Title__openam
+0.817	Title__gmail
+0.777	URL__state
+0.754	Title__working
+0.718	Title__youtube
+0.708	URL__sidt
+0.708	URL__setsid ssdc sidt
+0.708	URL__ssdc sidt
... 19504 more positive ...	
... 10057 more negative ...	
-0.709	URL__uk
-0.725	URL__login
-0.726	URL__com
-0.770	URL__http www
-0.836	URL__http youtube
-0.836	URL__http youtube com
-0.889	URL__org
-1.295	URL__https
-1.716	URL__http

(i)

y (score 3.873) top features

Contribution?	Feature
+4.632	<BIAS>
+0.410	Title: Highlighted in text (sum)
-0.026	Transition_Qualifiers: Highlighted in text (sum)
-1.144	URL: Highlighted in text (sum)

Title: vis4sense/provenancelearning: infer user sensemaking behaviour/task from analytic provenance
Transition_Qualifiers: client redirect
URL: https://github.com/vis4sense/provenancelearning

(ii)

Figure 6.12: ELI5 (i) Global feature importance, (ii) Local TextExplainer for train.values[1000] with Ridge classifier.

We also have used ELI5 (a python package) which allows to explain weights and predictions of scikit-learn linear classifiers and regressors, print decision trees as text or as SVG, show feature importances and explain predictions of decision trees and tree-based ensembles. ELI5 feature importance visualizations (above) on ridge classification shows that, 'redirecting' is the most important feature from column 'Title'.

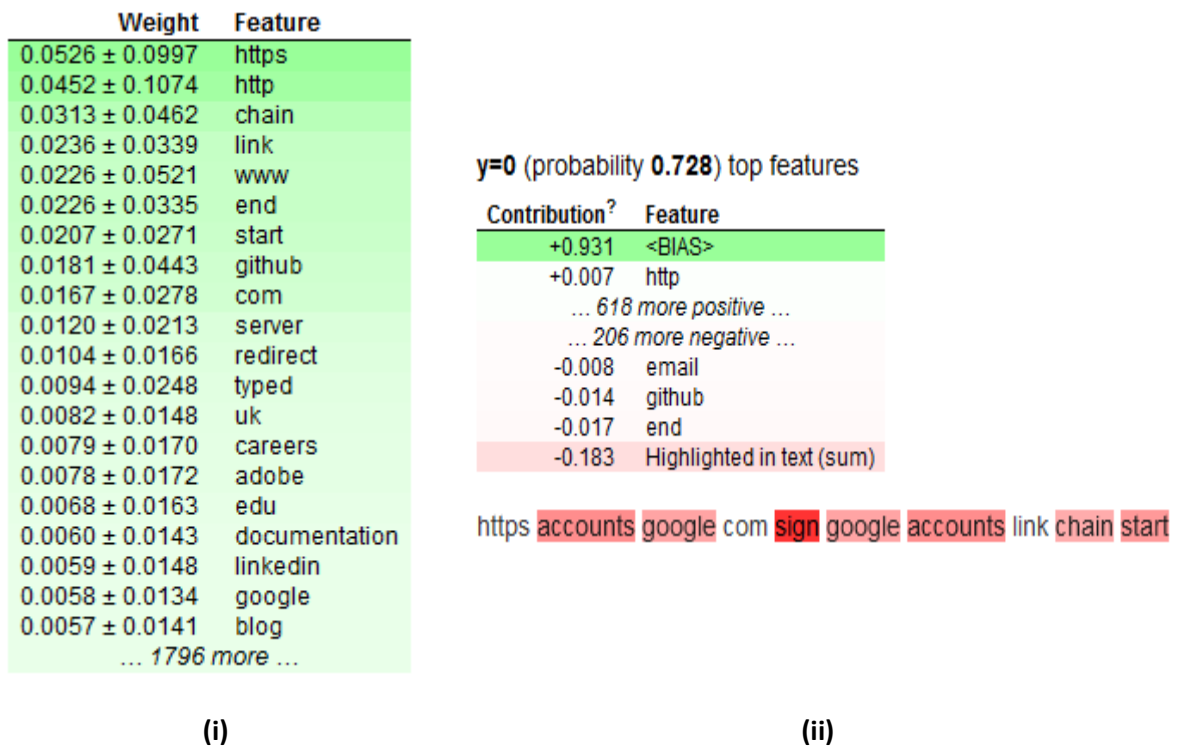


Figure 6.13: ELI5 (i) Global feature importance (ii) TextExplainer with Random Forest classifier.

The higher the position, the more critical the features are affecting the scoring. Some features in the bottom place is showing minus values, which means that the feature 'https' into column 'URL' has the least contribution towards prediction. For an example - if input includes 'redirecting', then it increases score +1.046. Another case is 'uk', it will decrease score -0.709. It is hard to approximate a black-box classifier globally (for every possible text), approximating it in a small neighbourhood near a given text often works well. ELI5 follows the LIME algorithm for explaining a text locally by generating distorted text, predict probabilities of it and then train the classifier which tried to predict the output of a black-box classifier on that text.

We also have implemented *Random Forest* to test how the scoring (accuracy, precision, recall etc) shift with feature existence or no. As shown in above, we can see that the displacement has the highest score with 0.0526. It means that when the permutation occurs to 'https' feature, it will change the accuracy of the model as big as 0.0526. The value after the plus-minus sign (i.e, ±0.0997) is the uncertainty

Decision tree feature importances; values are numbers $0 \leq x \leq 1$;
all values sum to 1.

Weight	Feature
0.2907	URL_Length
0.2489	Visit_ID
0.1808	Visited_On
0.1443	Elapsed_Time
0.1041	Visit_Count
0.0273	Transition_Type
0.0039	Typed_Count

```

Visit_Count <= 1.500 (24.2%)
  URL_Length <= 30.500 (1.2%)
    URL_Length <= 24.500 (0.5%) ---> 1.000
    URL_Length > 24.500 (0.7%)
      URL_Length <= 26.500 (0.2%)
      :
      :
      :
Visited_On > 44025.701 (4.0%)
  URL_Length <= 43.500 (0.3%)
    Transition_Type <= 3.500 (0.1%)
      Visit_ID <= 112088.500 (0.1%) ---> 0.000
      Visit_ID > 112088.500 (0.0%) ---> 1.000
    Transition_Type > 3.500 (0.2%) ---> 0.000
  URL_Length > 43.500 (3.7%) ---> 0.000
  Visit_ID > 118609.500 (11.5%)
    Visited_On <= 44043.572 (0.1%)
      Elapsed_Time <= 0.001 (0.1%) ---> 1.000
      Elapsed_Time > 0.001 (0.0%) ---> 0.000
    Visited_On > 44043.572 (11.4%)
    :
    :
    :

```

(a)

y=0 (probability **1.000**) top features

Contribution?	Feature	Value
+0.931	<BIAS>	1.000
+0.196	URL_Length	409.000
+0.110	Visit_ID	104869.000
-0.102	Elapsed_Time	0.051
-0.135	Visit_Count	1.000

(b)

y=0 (probability **0.993**) top features

Contribution?	Feature	Value
+0.934	<BIAS>	1.000
+0.057	Visit_Count	9.000
+0.002	URL_Length	62.000

(c)

Figure 6.14: ELI5 (a) Explanation as decision tree (partial view) (b) Feature importance of $X_{\text{test}}.\text{iloc}[310]$, (c) $X_{\text{test}}.\text{iloc}[1]$ for decision tree classifier.

value. The permutation importance method is inherently a random process, so it uses uncertainty value. above shows us how does *Random Forest* classifier predict the given data. It also shows how each feature contributes to the probability and the score (score calculation is based on the decision path) for non-breakpoint class ($y=0$). The classifier also introduces *<BIAS>* feature which is expected average score output by the model, based on the distribution of the training set. What is important into above visualization is that each feature contributed to the prediction result and hence the feature contribution affecting the weight result. Weight is after all the percentage of each feature contributed to the final prediction across all trees.

6.6 Discussion

Knowing which features have positive/negative impacts on the prediction of results and their influence on the models are important for building human trust on machine produced results. SHAP local explanation shows features each contributing to push the model output from the base value (the average model output over the training dataset we passed) to the model output. We have found that values of those contributing features can be negative themselves. For example – values of feature contributions $\phi_j(\hat{f})$ of instance `X_train.iloc[421,:]` as shown in above are negative while pushing the prediction to higher/lower. This is because the mean effect estimate $E(\beta_j X_j)$ is greater than the j -th feature effect $\beta_j x_j$, as explained into Section 2.5.3.2. So, as shown in above *'Visit_Count'* is negatively related to the model predictor however it pushes the prediction to the right with higher magnitude. But *'Elapsed_Time'* has negative impact driving the prediction to the left. This is just a local interpretation for that specific instance. SHAP provides global interpretation too i.e, *'Elapsed_Time'* also pushes the prediction globally to 'lower' (in 'blue') from it's base value -3.182. It is also possible to find out some relevant features from SHAP dependence plot, having similar impact of predictions. We have found *'URL_Length'*

has strong relevance with the *'Visit_Count'* as shown in above although negative relationship exists.

While SHAP explains feature influences on model outputs, it is also necessary to know most/least contributing features as well as importance of different word segments into a text towards breakpoint/non-breakpoint predictions. As shown in Figure 6.5, we have used LIME to visually interpret their weights in both tabular and highlighted text formats. LIME works by approximating instances locally with the interpretable model and performs perturbations on instances around the explained instance to weight those based on proximity (Π_x) measure as explained into Section 2.5.3.1. For text data, usually cosine similarity is used to measure those proximities. To explain such measures and related model outputs we have considered test.loc[310] as an example and found some negative weights, also known as coefficients. The true class for this instance is a non-breakpoint. We have found higher prediction probabilities of 0.98 for non-breakpoint after applying RF (*Random Forest*) as the classifier. *Visit_Count* > 4.00 is the top most coefficient in this case. *URL_Length* > 65.00 is the next most contributing feature inferring non-breakpoints. We have found strong relevance of these two features too according to SHAP dependence plot. We have over-/under- sampled to produce more reliable predictions on the text part of our dataset as well as it is highly imbalanced. We have found the word *'youtube'* having the highest coefficient value to lead the prediction towards *'breakpoint'* for test.loc[310]. On the otherhand *'youtube'* is positively and *'client'* is negatively correlated with the breakpoint prediction as shown in Figure 41(ii).

LIME has provided local interpretation and contribution of features specific to the instances. For calculating global importance of different features, we have used another method known as ELI5 that follows LIME algorithm. It calculates *'Permutation Importance'* also known as *'Mean Decrease Accuracy (MDA)'* by distorting texts and predict probabilities of those. Thus after applying RF (*Random Forest*) as classifier we have found *'github'*, *'linkedin'*, *'google'* etc, as the highest contributing features to the probability and the score for non-breakpoint class $y=0$ as

shown in Figure 6.13(i). The value after the ‘±’ sign is the uncertainty value. As the permutation importance method is inherently a random process, so it uses uncertainty values. Alike other approaches, we also have found ‘*URL_Length*’ as the highest contributing feature after calculating ‘*Decision Tree*’ feature importance as shown in above. The classifier also introduces <*BIAS*> which is the expected average score output by the model. In decision trees, each decision is guarded by a feature and it either adds or subtracts from the value given in the parent node and the prediction can be defined as the sum of ‘*the feature contributions + the BIAS*’.

When considering ‘*Decision Tree*’, each decision that a tree (or a forest) makes, there is a path (or paths) from the root of the tree to the leaf consisting of series of decisions, guarded by a particular feature, each of which contributes to the final predictions. We have presented such a prediction path visualization for test.loc[310] in above, which includes ‘*Visit_Count*’ and ‘*URL_Length*’ along the way to it’s prediction as a non-breakpoint. We have found from the dtreeViz in above that ‘*Visit_Count*’ has also been selected as root node by decision tree’s ASM (*Attribute Selection Measure*) heuristics. above shows quite a few leaves with $Gini \geq 0$, which are ideal scenarios of minimizing MAEs (*Mean Absolute Error*) to decide to split a node. The decision of making strategic splits heavily affects a tree’s accuracy. We also have found that decision criteria for classification and regression are different. For an example - ‘*Visit_ID*’ has been selected as the root with ‘*Decision Tree Regressor*’ as shown in above. We have observed low variances with non-breakpoint horizontal lines which actually measures how spread are the values out from their average. Leaves with such low variance among the target values are much more reliable predictors. Thus the whitebox approach of using decision trees has helped to understand and interpret the model’s decision making.

Lastly, explaining model’s decision making process, unfolding blackbox calculations of probabilities towards predictions, computing feature importance, understanding their local and global implications are important to show algorithmic transparency of machine learning outcomes for inferring breakpoints from uncertain log dataset. We

have aimed to explain evaluation results as achieved from Chapter 5 to prove their validities through human judgemental process by uncovering their execution steps, ranking features based on their importance and implications. It is necessary to explain those evaluation results as we could not compare those outcomes with the measurements of different behavioural constructs due to lack of available data from CTA as explained into Chapter 4. Alongside the transparency these explanations will also help to build trust on machine produced results. From Chapter 5, we have found supervised approaches achieve better results than unsupervised approaches and comparatively less difficult to explain those results. To achieve better results we also have shown how to tune model parameters into both manual and automatic approaches to find out the best combinations. For those approaches, alongside other ML algorithms we have used ensemble of decision trees (*Random Forest, LightGBM*) for structuring data and deep learning. We also have used regressions i.e, *Ridge Regression, Logistic Regression* and found difference between their decision criteria from '*Decision Tree Regressors*'. Thus we have shown influences and relevance of different features and their influences on ML model prediction results upon permuting those. But in future we aim to compare these results with the outcome of manually detected and calculated results of a CTA approach to distinguish between machine and human approaches.

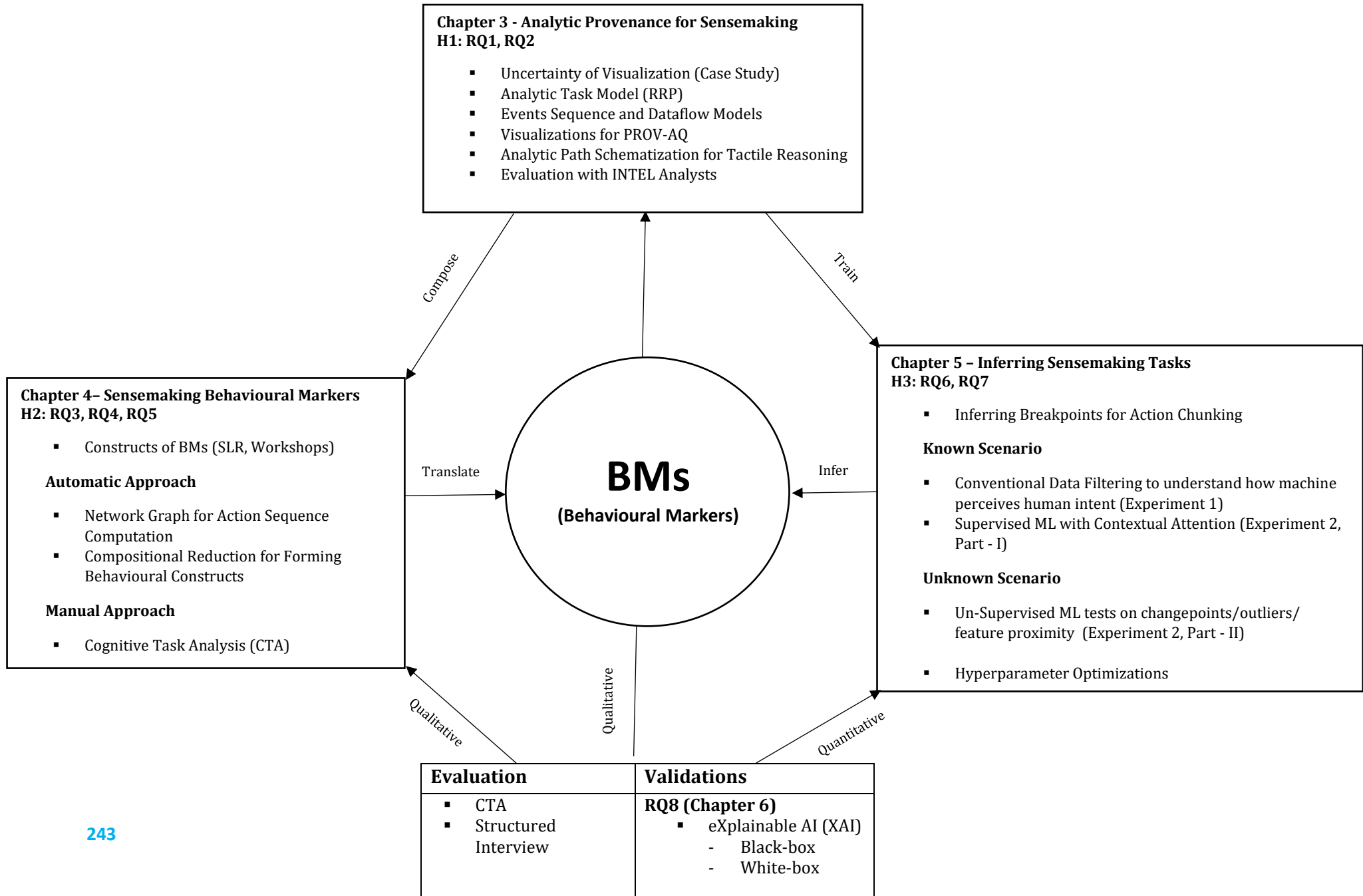
Conclusion

7 chapter

7 Research Review

This research aims to bridge the gap between cognitive constructs and manipulations or interactions human employ to think and reason by leveraging captured analytical provenance data. In this research, detecting '*Behavioural Markers (BMs)*' has been proposed as a way of establishing this bridge to execute a fine-tuned computation led cognition and vice-versa during sensemaking activities mainly into visual analytic systems. The overall contribution of this research to detect BMs has been presented in Figure 7.1 as a summary. As the first step, we have attempted to develop and evaluate a system to capture, visualize and utilize analyst's sensemaking interactions after conducting extensive research with a group of police intelligence analysts for finding out the requirements (Section 3.3.3.1) during real criminal intelligence analysis. In the next step, we have applied a composition→translation technique on captured analytic datasets to form constructs of BMs for externalizing analyst's thinking processes. We also have conducted a cognitive task analysis for detecting those BMs. Prior to this qualitative approach we have developed an exhaustive table of observable BMs and their constructs as shown in Table 4.2. BMs are consisting of different behavioural constructs which are basically different approaches followed by analysts. We have found from the CTA that different cognitive transition information are also nipped into captured analytic dataset which are the building blocks of behavioural constructs at the lower level. We have named those transitions points as '*Breakpoints*' and aimed to infer those by adopting machine learning (ML) techniques both in known and unknown task scenarios. As ML models are all black-boxes, we also have interpreted those quantitative evaluation results by using relevant '*eXplainable AI (XAI)*' techniques to validate machine produced results and maintain transparency.

Figure 7.1: Research Contributions.



7.1 Summary of Research Contributions

The whole research has been conducted based on several hypotheses(H) and associated research questions (RQs) as shown in above. So, overall thesis contributions have been made through finding out appropriate tools & techniques for addressing those RQs and test hypotheses(H) which are as followed:

7.1.1 Hypothesis 1

Capturing user's interactions with a visual interface can retrieve some aspects of the transparency of user's reasoning processes in intelligence analysis.

RQ1: How to develop a system that tackles large flow of heterogeneous analytical data and supports W3C PROV-AQ: Provenance Access and Query standard factors i.e, Recording - represent, denote; Querying - identify, pingback; Accessibility - locate, retrieve into a multi-modular environment?

RQ2: How to utilize captured analytic provenance data for sensemaking?

To test hypothesis 1 by considering RQ1 and RQ2 we have contributed into following stages of research and development:

We have proposed a new '*Analytic Task Model*' by gathering information from real police intelligence analysts about '*how do they think*' and '*what do they do*' to achieve their goals. The aim is to examine analytic provenance into that model's stages (Figure 3.3) and represent those to support judgements of computations made with large number of complex interconnected systems. This is important because failing to examine followed analytical techniques and lack of tool support may lead to faulty cavalier and superficial data analysis, making faulty claims with confidence that may cause poor decision making.

- While understanding how the analyst '*perceives*' visualization of data through a case study on criminal situation as described into Section 3.3.2.3, we have

found that inaccuracy may occur through the pipeline of data processing. We have denoted this as '*Uncertainty of Visualization*' which differs with the concept '*Visualization of Uncertainty*' on which lot of research have been carried out. We also have found that -

- Uncertainties in visualizations also lead to spatial (Figure 3.6) and temporal (Figure 3.7) determinacy problems which means '*don't know about when and where*'. We have found it raising '*Issue Uncertainty*' for structuring, filtering and organizing dataset resulting to '*Decision Uncertainty*' as described into Section 3.3.2.2.
 - Visualization bias may occur where a user is typically unaware of the data. We have found such clustering illusions from Figure 3.5 (iii, iv) whereas those were not developed by using any statistical algorithm.
 - We contend that analytic provenance methods for capturing, tracking, managing or organizing data can support such '*uncertainty-aware sensemaking*'.
- For '*capturing*' analytic provenance data we have contributed by designing a new system architecture (Figure 3.8), event sequence (Figure 3.9) and underlying data-flow (Figure 3.10) models for large modular AUI (*Analyst's User interface*) system used for the project *VALCRI. We have tested those models and found it supports capturing/restoring analytical provenance states or workflows both automatically and manually by tackling complexities of heterogeneous development environments.
- Prior to those model implementations, we conducted an extensive requirements analysis by arranging several focus group discussions with the end-users (police intelligence analysts) and formulated lists of system requirements (*SysReqs*) and police analysts' requirements (*AnaReqs*) as described into Section 3.3.3.1.
- For utilizing and making sense of captured analytic data we also have proposed a '*Repetitive Replicating Analytic Path Schematization*' approach.
- The first part of this approach enables analysts to study reasoning relationships from the default workflow visualization. We have captured these relationships from a collaborative environment of

intelligence analysis and visualized those as an '*Analytic Path*' (Figure 3.12). We have visualized (as shown in Figure 3.15) those set of related analytic states in a way so that those support W3C standard PROV-AQ as well.

- Secondly, the analytic path visualization supports '*schematizing in visuo-spatial*' manner (Figure 3.13) according to Klen et. al.'s [80] data-frame model to enable '*Tactile Reasoning*' [81] for the analysts.
- Lastly, this approach includes a new method of reusing and reapplying insights into different domain and retaining those for gaining a new insight. We have named this method as '*Repetitive Replicating Playback (RRP)*' system as shown in Figure 3.3(ii) and 3.12.

The contributions of this research are quite significant and valid for the area of criminal intelligence analysis as we carried out this whole research, development and evaluation with the real police intelligence analysts (end-users) intending to contribute to the provenance module 'PROV' of the project *VALCRI. The whole approach was non-trivial as we targeted to develop models and techniques so that those work generically for a large complex heterogeneous modular visual analytic system. We have found from the evaluation report on the module 'PROV' that few of the building blocks of '*Transparency*' i.e., source, process, accountability, series of events etc can be retrieved. All these are enabler of fairness and lawfulness of the data processing activities from the legal framework. However, for establishing insightful alignment with analyst's cognitive processes we aim to utilize some of these analytical data into next stage of our research.

7.1.2 Hypothesis 2

Behavioural Markers (BMs) can act as attributes for bridging between human cognition and analytic computation through interactions during fluid transitions between mental and analytic processes at micro-analytic level.

RQ3: What are the constructs of Behavioural Makers (BMs)?

RQ4: How to translate reasoning processes to Behavioural Markers (BMs)?

RQ5: How to externalize thinking processes from the constructs of Behavioural Markers (BMs)?

To test hypothesis 2 by considering RQ3, RQ4 and RQ5, we have contributed into following stages of experiments and reported results:

We have found most of the participants' agreement ratings, $3.29 \leq r \leq 5.00$ with an overall average value of 4.07 (Figure 4.6) and MD=8.14 (Figure 4.7), while assessing each of behavioural constructs through '*Cognitive Task Analysis*'. Taking into consideration we have understood that constructs of analyst's cognitive activities are ingrained into their analytic activities and higher MD rating is explicitly the indication of '*fluency in data finding*' resulting to '*ideational fluency*'. Its lowest standard deviation SD=0.9 (Figure 4.7) shows strong acceptance of all participants regarding MD ratings.

- The above results are important to test the current hypothesis, because these findings prove occurrence of transitions between mental and interaction states through analytic processes during fluid activity of intelligence analysis although the performance rating PF=3.14 was average despite of higher effort EF=7.0. Recovering such cognitive reflection on analytic reasoning processes from extended log data is difficult. Endert et. al. [38] contends that a new methodology is needed to couple these cognitive and computational components.
- We have proposed that markers of analyst's cognitive behaviour are the attributes for bridging human cognition and analytic computation through interactions. We have named those as '*Behavioural Markers (BMs)*' in this research.

- Detecting constructs of BMs are a bit non-trivial, because the beginning or end of reasoning process may be unknown . We call it as '*Cognitive Steps Sequencing Problem*'.
- At the first stage, through an systematic literature review we have selected some behavioural attributes as shown in Table 4.1. At the next stage, we have we have discussed those initial attributes with subject matter experts through arranging a workshop by considering human factors, cognitive engineering and interactions on visualizations in *VALCRI's AUI system. Thus we have formed an exhaustive list of constructs of BMs as shown into Table 4.2.
- For translating captured reasoning processes to BMs, we have visualized those into a network graph named as '*Analytic Path*' (Figure 4.2) to understand which action combinations may provide meaningful sequence for targeted BM.
- We have proposed a computational approach known as '*compositionally reduction*' that leads complex constructs breaking down or reducing into simpler, more quantitatively manageable constructs. Ideally, these smaller components have a more directly observable set of markers for a certain analytic behaviour. Thus it externalizes human thinking process as it continues reducing down as described into Section 4.5.1.

Endert et. al's [38, 85, 117] suggested '*semantic interaction*' concept of model steered interaction affordance to couple cognitive (NTS) and computational (TS) components, has limitations in case of erroneous move which may have negative influence on user's confirmation bias. Fisher et. al. [118] has raised a concern of having small memory footprint in case of such adaptive computation to provide the user with rich information throughout the user's exploration process. Our current approach of translating different cognitive constructs in terms of computational interactions can overcome these issues by pinpointing or inferring cognitive transitions with a goal to understand user's sensemaking behaviour. We aim to investigate this claim in the next piece of our research work.

7.1.3 Hypothesis 3

Inferring chains of low-level analytic actions can be of assistance for understanding multi-tasking behaviour.

RQ6: How can meaningful units of task execution be produced from captured interaction logs?

RQ7: How precisely multi-task switches be inferred during execution of interactive tasks?

RQ8: How to validate inference making results for building trust on machine learning models and maintain transparency?

To test hypothesis 3 by considering RQ6, RQ7 and RQ8, we have contributed into following stages of experiments and reported results:

i.

We have found $0.26 \leq \text{Recall}@10 \leq 0.48$ and $0.40 \leq \text{Recall}@20 \leq 0.63$ (Figure 5.7) from experiment 1 (Section 5.3.1), which show that it always requires higher hit(N) for understanding user's intention by using conventional methods.

- The above finding is important because it identifies the drawback of conventional methods in case of lack of information known as '*cold-start*' problem.
 - To evaluate *Recall@N* results we have used popularity-based, content-based, collaborative and hybrid filtering methods as discussed into Section 5.3.1.3.
 - Among those the '*Hybrid*' method returned better '*Recall@N*' values as 48% (N=10) and 63% (N=20) for 100 random test data by using the data filtering model.
 - Predictive accuracy of '*Hybrid*' model is comparatively better because it uses blending of multiple predictors such as – weighted average of normalized CF scores with the CB scores. So, applying this model is comparatively non-trivial than other approaches.

- We have proposed to exploit the contextual attention mechanism for improving performance of conventional methods and overcoming cold-start issue in case of lower hit(N).

ii.

We have used contextual attention information as explained into Section 5.3.2 to test either it supports domain independent user's analytic behaviour modelling or not. We have proposed '*breakpoints*' as the way of chunking users' stream of actions and understand their intention at different granular levels (hierarchical/contextual). After applying these concepts into part 1 of experiment 2 (Section 5.3.4.1), we have obtained, the precision 0.98 and recall 1.00 which depict the predicted values for '*non-Breakpoints*' are almost similar to their originals. The overall accuracy score is - 98%.

- The above finding is important because it solves the cold-start issue by building context of interacted contents as opposed to conventional way of associating static weights to different types of interactions as explained into Section 5.3.1.2.
 - The above result has been obtained by applying a neural network based '*Multi-Headed Self-Attention*' mechanism as shown in Figures 5.8 – 5.13.
 - This method helps to learn a word's context on surrounding words rather than the word immediately precedes or follows it.
 - For preparing training dataset we have adopted our proposed abstraction techniques i.e, context dependent, hierarchical and binary chunking for activity classification (Section 5.2).
- The overall process of implementing and explaining the results of above approach are challenging. Because the model is built on pre-trained 12-layer, 768-hidden, 12-heads, 110M parameters and takes longer time to finish the training process. Thus it generates 144 distinct attention patterns at it's multi-head layers only for two text inputs as shown in Figure 5.13.
- The above approach is universal for all settings of large text corpus as it will use the same pre-trained model. We have visualized strength (positive/negative)

calculations of attentions for feature inputs based on ‘*query*’ and ‘*key vector*’ products as shown in Figure 5.15.

- We have found that although computing contextual attention has shown promising results in inferring human specified breakpoints but it will not be feasible to adopt in case of unknown/free-form task due to cognitive and perceptual variances of different users.

iii.

To come round the problem of defined task, we have proposed a data-driven approach in part 2 of experiment 2 (Section 5.3.4.7) where all used algorithms do not see the labels while training but later used for performance metrics. We have aimed to test in this part of experiment either ‘*Breakpoints*’ are – (1) *Changepoints* or (2) *Outliers* or (3) *Distant Features*.

- For testing those assumptions, we have presented a data transformation technique to produce semantically similar text corpus and tag those with a name.
 - After applying ‘*ChangeFinder Algorithm*’ as shown in Figure 5.17, we have found that it mostly can detect ‘*Changepoints*’ among usual trends based on anomaly scores but those are not always indicatives of ‘*Breakpoints*’.
 - To test breakpoints being outliers, we have applied ‘*Local Outlier Factor (LOF)*’ algorithm which looks at the local neighbourhood of a data point and measures the local deviation of density of a sample with it’s neighbour. We have found F1 accuracy: 0.086956 and ROC AUC score: 0.520737 which shows flip of a toss situation in inferring data points as breakpoints.
 - By analyzing t-distributed stochastic neighbour embedding visualization (Figure 5.19) we have found semantic overlaps of dissimilar chunk data points. We also have found from the latent representation of ‘*Autoencoder*’ [Figure 5.20(b)] that breakpoints can be

present among closer proximities of data points with higher probabilities. So, dissimilar objects of distant points with lower probabilities are not always the only indicatives of breakpoints.

- We also have calculated the 95th percentile of breakpoint prediction values for K-Means and found F1 accuracy: 0.043478; for Isolation Forest and found F1 accuracy: 0.086956 as shown in Table 5.2.
- After decoding data from autoencoder's latent representation, we have found decreasing F1 score [Figure 5.20(e)] which means it mostly can infer non-breakpoints with lower reconstruction errors. We have found the model a bit overfitting at some epochs while being in synch for few other cases although having decreasing validation loss and increasing validation accuracy.
- These findings are important for the research in cognitive science as conventional clustering and pattern mining techniques for user behaviour modelling [39], task identification [40], clickstream modelling [39] don't fit well with inference making in case of cognitive and perceptual variances. Our finding on feature distances also opposes with Lee et. al's [115] distance curve's peaks denoting as breakpoints. But their finding matches with ours while detecting human specified breakpoints by using existing '*Changepoints*' detection technique. Our '*Contextual Attention*' approach outperforms Iqbal et. al's [111] approach to infer breakpoints where they found detection accuracy of 69% - 87% by using CFS and MPL techniques.
- Although we have achieved promising performance for the supervised learning on contextual attention but for unknown scenario our experiment setting on unsupervised learning did not perform well. From the later approach, we have found F1 accuracy between 0.043 to 0.087; ROC AUC between 0.47 and 0.52 (it means the model has no class separation capacity).
- For achieving improved performance of breakpoint inference, we have tested with an automatic model parameter tuning technique known as '*OPTUNA*' to find the best combination and achieved 94% as overall accuracy after 200 trial runs. As shown in Table 5.3, we also have tested with other '*Hyperparameter*

Optimization techniques i.e, *'TPOT', 'ANN', 'Bayesian Hyperpot'* and obtained similar results.

iv.

Lastly, We have implemented few eXplainable AI (XAI) techniques for interpreting model's decision making process, unfolding blackbox calculations of probabilities towards predictions, computing feature importance and understanding their local/global implications. The aim is to provide transparent validations of above explained evaluation results and building trust on machine produced results through human judgemental process.

- We have found the most performant types of machine learning algorithms are ensemble of *'Decision Trees (RandomForest, LightGBM)'* in hyperparameter tuning (Table 5.3) for inference making. So, we have used those algorithms for measuring feature contributions on model predictions in line with XAI techniques.
- After considering several sample instances we have found *'Visit_Count'* (number of times the user has navigated to the webpage) as one of the top locally contributing features. For example–
 - Calculated SHAP value of `X_train.loc[421,:]` in line with *LightGBM classifier* shows that it pushes the prediction to the right (higher) [Figure 6.1(b)] and *'URL Length'* (string length of the url) has the strongest relevance as shown by *'SHAP Dependence Plot'* [Figure 6.3(i)].
 - LIME tabular representation [Figure 6.5(i)] of `X_test.loc[310]` in line with *'Random Forest classifier'* shows similar results for it's true class *'non-breakpoint'*.
 - Alike SHAP, LIME approaches, we also have found *'Visit_Count'* and *'URL_Length'* as top contributing features after calculating *'Decision Tree'* feature importance as shown in Figure 6.10(a) along the *'Prediction Path'* (Figure 6.9) too.

- We also have found from test.loc[310] text sample after applying '*Random Forest*' as classifier that the word '*youtube*' is positively correlated with prediction as a '*breakpoint*' [Figure 6.5(ii)]. The ELI5 global feature importance in line with '*Ridge Classifier* (estimating coefficients of multiple-regression models where independent variables are highly correlated)' has also identified '*youtube*' as one of the top positively (having least negative) affecting word on model prediction [Figure 6.12(i)].
- While drawing the decision path we have found quite a few leaves with $Gini \geq 0$ which are ideal for minimizing '*MAEs (Mean Absolute Errors)*' as shown in Figure 6.8 (a,b). We also have found '*ASM (Attribute Selection Measure)*' and decision criteria for classification and regression are different. This is important because decision the decision of making strategic splits heavily affects a tree's accuracy.

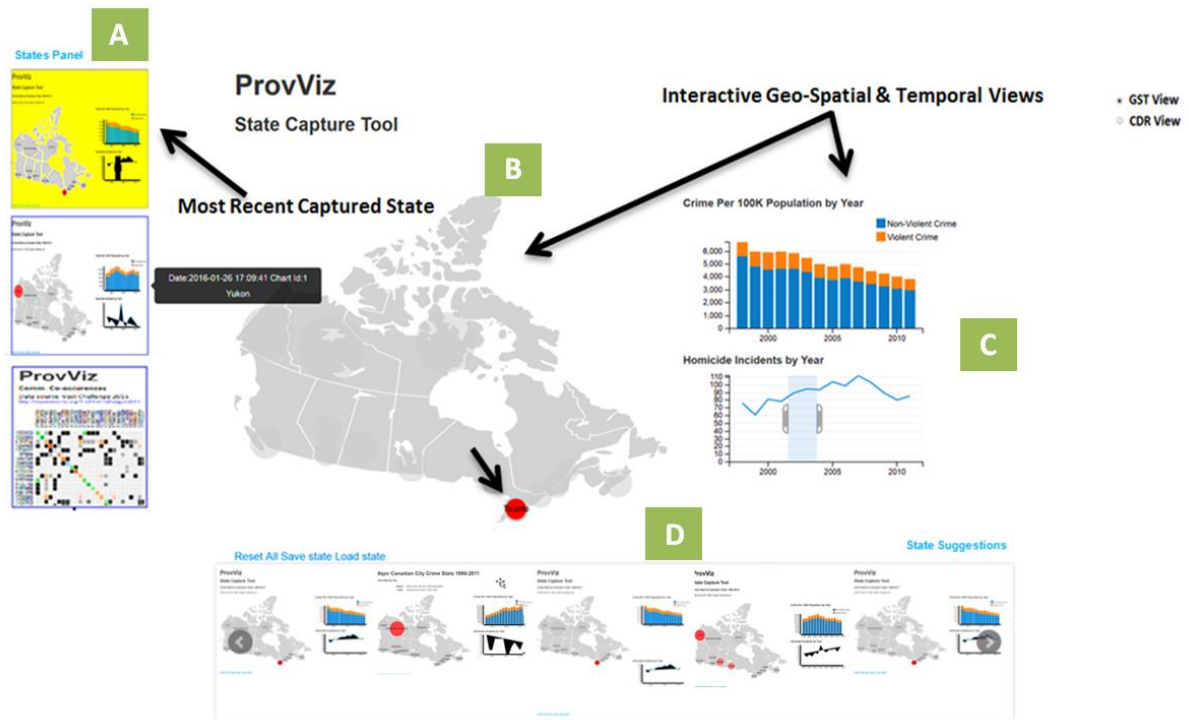
We have aimed to judge the validity of obtained evaluation results from Chapter 5 due to lack of required data from '*CTA (Cognitive Task Analysis)*' as described into Chapter 4 to measure constructs of BMs (*Behavioural Markers*) and compare those outcomes with evaluations results. The '*blackbox*' and '*whitebox*' calculations (knowns as XAIs) of probabilities towards predictions have unfolded most importantly the prediction path and it's contributing features for human judgement and build trust on machine produced results.

7.2 Additional Work and Scopes of Further Development

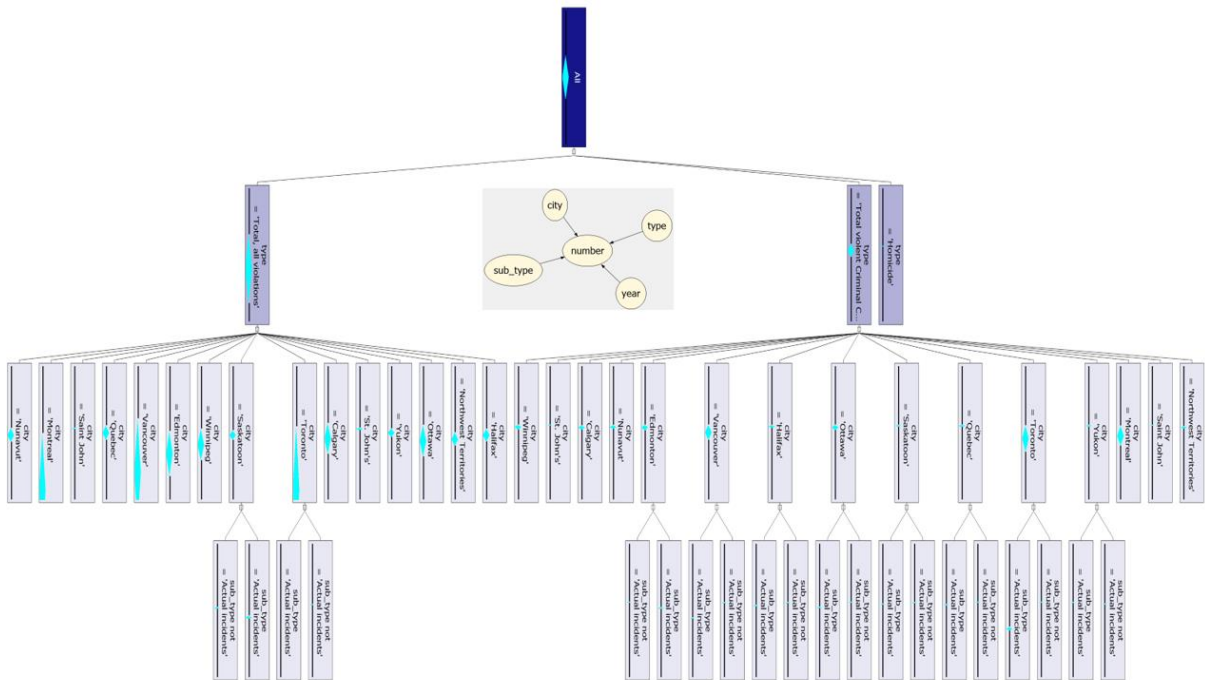
We have discussed limitations of our current approach in each chapter. However, there are scopes to improve or test other approaches still. We started with another project which can be a potential area to explore to extend and improve results of our current research in some cases.

Canadian Crimes By Cities 1998-2012

Data Source : <http://open.canada.ca>



(i)



(ii)

Figure 7.2: (i) ProvViz - An analytic state suggestion system (GST View): **A**) Automatic analytic state capture panel, **B**) Canadian Crime (by cities 1998-2012) Visualization on map. **C**) Bar chart and line chart for showing temporal crime statistics. **D**) Automatic state suggestion panel. (ii) Preliminary ontology development for ProvViz.

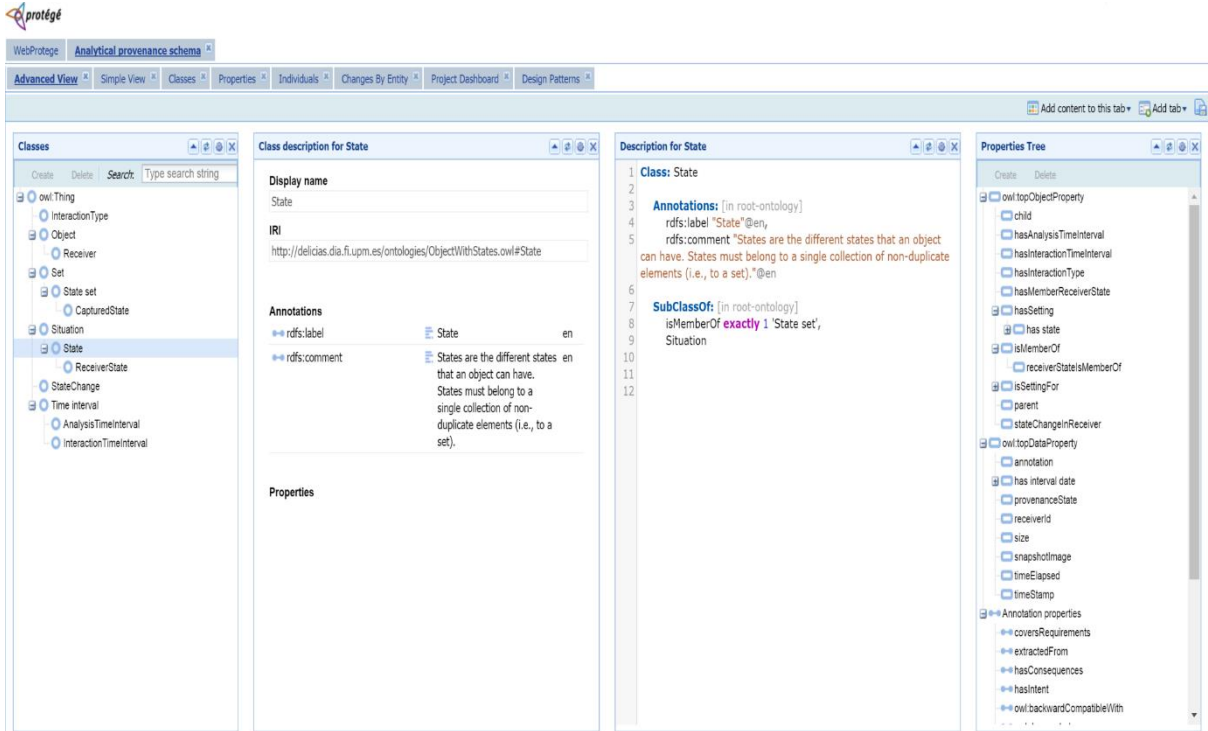
7.2.1 Ontological Approach for Data Provenance

We have followed computational approaches so far to elicit user's intent of reaching goal. But we have not considered implementation of any domain related knowledge for inference making which might be needed in areas like intelligence analysis for decision making. Implementing/changing/switching such knowledge as semantic data layer may be needed alongside understanding of user's analytical behaviour during intelligence analysis. For an example – switching from '*West Midlands policing policies*' to '*Belgium policing policies*' for the purpose of case based reasoning. On the otherhand, implementing/maintaining/changing such knowledge architecture is crucial at the code level during run time into any system. One of the potential solutions of this problem can be adopting an ontological approach to operate as underlying architecture of the dataset and make changes there when it is needed. To test the idea, we have developed a prototype [above] that captures analyst's interactions automatically. We have named this prototype as '*ProvViz*' that works interactively with the Canadian map's '*GST (Geo-Spatial and Temporal) view*' with total number of crimes visualizations for 1998-2012[†] (red circles) and . The bigger the circle is - means more crimes occurred in a city than the comparatively smaller ones.

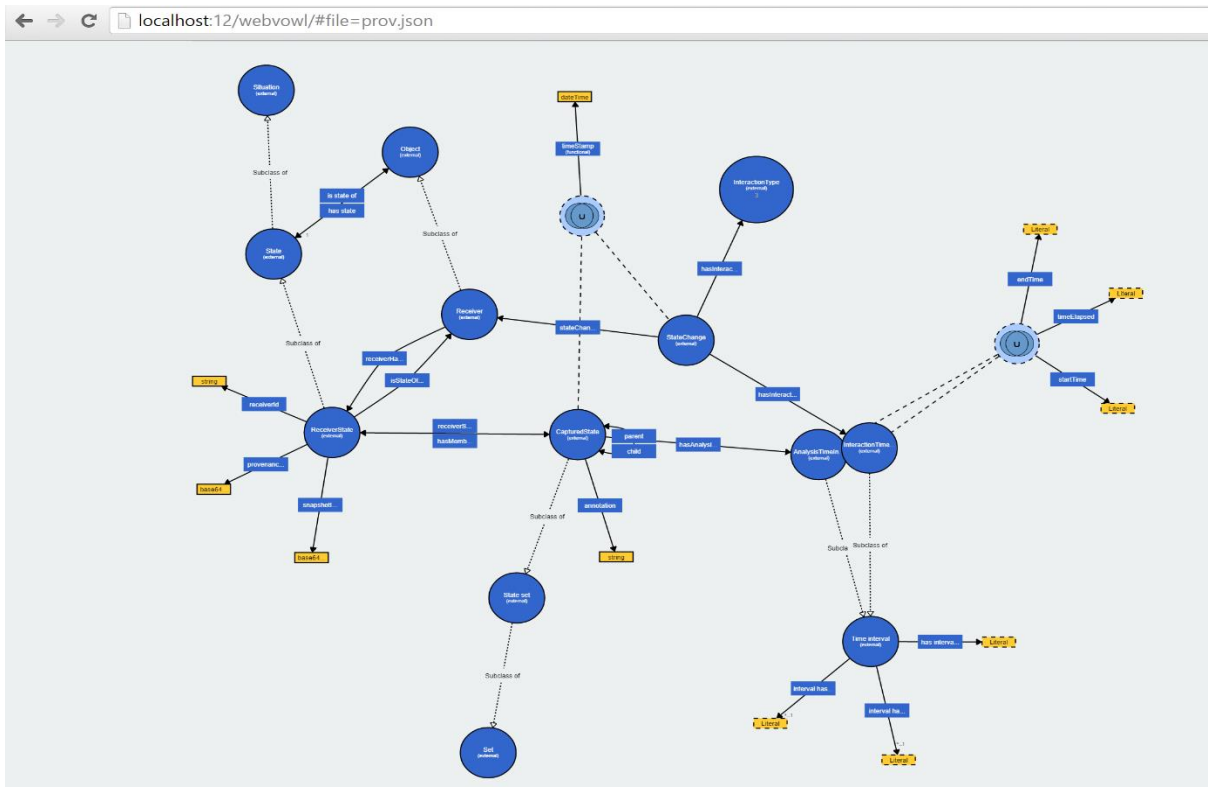
It also includes another view known as '*CDR (Call Data Records) view*', developed by using VAST Challenge 2015 dataset[‡]. The initial version of schema shows the relationships among different crime types and their subtypes occurring in Canadian cities. From the schema we can see that '*All*' crimes have '*Types*' and these types are related to different '*cities*' where it occurred. All cities have '*subtypes*' of crimes occurred in different times of the year. After interacting, this system suggests relevant states based on the computation of developed initial version of ontology as shown in above.

[†] GST Analysis: Canadian Crimes by Cities during 1998-2012
<http://open.canada.ca/>

[‡] CDRs Analysis: VAST Challenge 2015:
MC2 <http://vacommunity.org/VAST+Challenge+2015>



(i)



(ii)

Figure 7.3: Preliminary version of analytic provenance ontology for 'PROV' – (i) WebProtégé class view, (ii) WebVowl visualization.

We aim to further develop this simple ontology by creating a semantic representation of data for a large complex system like *VALCRI's AUI (Figure 3.2) so that it can discover associations of different crime types both geo-spatially and temporally. This will help to answer the question – ‘*why has the crime occurred?*’. We also aim to infer inconsistencies in data, derive new data as ontologies support automatic reasoning as well by using existing concepts, relations and additional axioms relevant for the specific domain [119].

7.2.2 Ontological Approach for Analytic Provenance

Alongside data provenance, we aimed to develop an analytic knowledge representation as well with an assumption that the system-led sensemaking may enhance support into sensemaking steps. The system can proactively or reactively help analysts by inferring ‘*what he/she is trying to do*’. There is no W3C ontology available to represent such analytical knowledge. Currently, only available PROV-DM or PROV-O are the conceptual data model that form a basis for the W3C provenance (PROV) family of specifications. Design of ontology for ‘*process provenance*’ can be useful to domain specific systems for automatic execution of procedures or guiding analysts to reach the goal.

As shown in above, we have attempted to develop a preliminary version of ontology by using captured process provenance data from *VALCRI's AUI. We have not tested it yet with such large complex system but aim to develop it further in future and store/retrieve captured analytic provenance data with its ontology.

We have chosen the ontological approach is because - Ontology is one of the approaches for knowledge representation. It supports reusability and share ability [121]. Ontologies enable us to share the domain and the knowledge between applications [120, 121]. Ontologies create machine-understandable descriptions of learning resources and provide the personalization and adaptively. Currently, data

mining and machine learning communities have developed a large set of algorithms and techniques to identify trends and patterns in different types of data. These range from simple association rule and clustering algorithms [122] to sophisticated models for pattern recognition [123]. Various visualization tools like - VIZREC [124] also has used such technique to develop visualization recommendation systems that can automatically identify and interactively recommend visualizations relevant to an analytical task. But those techniques are not always suitable for an intelligence analysis system where domain specific policies and procedures need to be implemented during sensemaking for achieving plausible conclusion compliant with law.

References

- [1] K. Xu, S. Attfield, T. J. Jankun-Kelly, A. Wheat, P. H. Nguyen and N. Selvaraj. Analytic provenance for sensemaking: A research agenda. In: *Computer Graphics and Applications*, IEEE, 35: 3, 2015.
- [2] R. Chang and J. Crouser. Visual Analytics and Provenance. Tufts University Course Description. Retrieved from <http://www.cs.tufts.edu/comp/250VA/>, 2012.
- [3] C. Görg, Y. A. Kang, L. Zhicheng and J. Stasko. Visual analytics support for intelligence analysis. *Computer*, 46(7), 30-38, 2013.
- [4] P. Pirolli and S. Card. The sensemaking process and leverage points for analyst technology as identified through cognitive task analysis. In: *Proceedings of the International Conference on Intelligence Analysis*, 2-4, 2005.
- [5] Y. A. Kang, C. Görg and J. Stasko. Evaluating visual analytics systems for investigative analysis: Deriving design principles from a case study. In: *Proceedings of the IEEE Symposium on Visual Analytics Science and Technology, VAST'09*, 139-146, 2009.
- [6] Y. A. Kang and J. Stasko. Characterizing the intelligence analysis process: Informing visual analytics design through a longitudinal field study. In: *IEEE VAST Oct. 2011*, 21-30, 2011.
- [7] Y. A. Kang, C. Görg and J. Stasko. How can visual analytics assist investigative analysis? Design implications from an evaluation. *IEEE Transactions on Visualization and Computer Graphics*, 17(5), 570-583, 2011.
- [8] C. Görg, Y. A. Kang, L. Zhicheng and J. Stasko. Visual analytics support for intelligence analysis. *Computer*, 46(7), 30-38, 2013.
- [9] G. Cybenko and B. Brewington. The Foundations of Information Push and Pull. Abstract. To appear in *Mathematics of Information, Institute for Mathematics and Applications Proceedings*. Retrieved from: <http://www.dartmouth.edu/~gvc/push.html>, 1997.
- [10] D. Gotz and M. X. Zhou. Characterizing user's visual analytic activity for insight provenance. In: *Proc. IEEE Symp. Visual Analytics Science and Technology (VAST)*, 123-130, 2008.
- [11] R. Heuer. Psychology of intelligence analysis. Center for the Study of Intelligence, Central Intelligence Agency, 1999.

- [12] A. Tversky and D. Kahneman. Judgment under Uncertainty: Heuristics and Biases. In: *Science* Vol. 185, Issue 4157, pp. 1124-1131
DOI: 10.1126/science.185.4157.1124, 1974.
- [13] F. Sørmo and J. Cassens. Explanation Goals in Case-Based Reasoning. In: *Proceedings of the ECCBR 2004 Workshops*, 165-174, 2004.
- [14] C. North, R. Chang, A. Endert, W. Dou, R. May, B. Pike, and G. Fink. Analytic provenance: process+ interaction+ insight. In: *CHI'11 Extended Abstracts on Human Factors in Computing Systems*, ACM, 33-36, 2011.
- [15] T. J. Jankun-Kelly, K. L. Ma, and M. Gertz. A model and framework for visualization exploration. *IEEE Transactions on Visualizations and Computer Graphics* 13(2), March/April 2007, 357–369, 2007.
- [16] D. Keim, G. Andrienko, J. D. Fekete, C. Gorg, J. Kohlhammer and G. Melancon. Visual analytics: Definition, process, and challenges. In: *Information Visualization: Human-Centered Issues and Perspectives*, 154-175, 2008.
- [17] J. Thomas and K. Cook. Grand challenges. In J. Thomas & K. Cook (Eds.), *Illuminating the path: The research and development agenda for visual analytics* (p 19-32). Washington, DC: *National Visualization and Analytics Center*, 2005.
- [18] Y. B. Shrinivasan and J. J. van Wijk. Supporting the Analytical Reasoning Process in Information Visualization. *ACM Human Factors in Computing Systems (CHI)*, Florence, Italy 2008.
- [19] W. Pike, J. Bruce, B. Baddeley, D. Best, L. Franklin, R. May and K. Younkin. The Scalable Reasoning System: Lightweight visualization for distributed analytics. *IEEE Symposium on Visual Analytics Science and Technology, VAST'08*, 131-138, 2008.
- [20] C. T. Silva, J. Freire and S. P. Callahan. Provenance for visualizations: Reproducibility and beyond. *Computing in Science & Engineering*, 9(5), 82-89, 2007.
- [21] R. Eccles, T. Kapler, R. Harper, W. Wright. Stories in GeoTime. *IEEE Symposium on Visual Analytics Science And Technology*, 7(1):19–26, 2007.
- [22] R. Walker, A. Slingsby, J. Dykes, K. Xu, J. Wood, P. H. Nguyen, D. Stephens, B. L. W. Wong and Y. ZHENG. An extensible framework for provenance in human terrain visual analytics. *IEEE Transactions on Visualization and Computer Graphics*, 19(12):2139– 2148, 2013.
- [23] C. Plaisant, R. Mushlin, A. Snyder, J. Li, D. Heller and B. Shneiderman. LifeLines: using visualization to enhance navigation and analysis of patient records. In: *Proceedings of the AMIA Symposium*, 08(98):76–80, 1998.
- [24] P. H. Nguyen, K. Xu, A. Wheat, B. L. W. Wong, S. Attfield and B. Fields. SensePath: Understanding the Sensemaking Process through Analytic Provenance. In: *IEEE Transactions on Visualization and Computer Graphics*, 20(1):41-50, 2016.

References

- [25] S. Gratzl, A. Lex, N. Gehlenborg, N. Cosgrove and M. Streit. From Visual Exploration to Storytelling and Back Again. *Eurographics Conference on Visualization (EuroVis)*, Volume 35, Number 3, 2016.
- [26] R. L. Helmreich, A. C. Merritt and J. A. Wilhelm. The Evolution of Crew Resource Management Training in Commercial Aviation. *International Journal of Aviation Psychology*, 9(1), 19-32, 1999.
- [27] R. Flin and L. Martin. Behavioral Markers for Crew Resource Management: A Review of Current Practice. In: *THE INTERNATIONAL JOURNAL OF AVIATION PSYCHOLOGY*, 11(1), 95–118, 2001.
- [28] G. Fletcher, R. Flin, P. McGeorge, R. Glavin, N. Maran and R. Patey. Rating nontechnical skills: developing a behavioral marker system for use in anaesthesia. In: *Cognition, Technology, and Work* 6, 165–171, 2004.
- [29] S. Yule, R. Flin, S. Paterson-Brown, N. Maran and D. Rowley. Development of a rating system for surgeons' nontechnical skills. In: *Medical Education* 50, 1098–1104, 2006.
- [30] L. Mitchell and R. Flin. Scrub practitioners' list of intra-operative nontechnical skills-SPLINTS. In: *Flin, R., Mitchell, L. (Eds.), Safer Surgery. Ashgate Publishing Ltd., Aldershot, England*, pp. 67–82, 2009.
- [31] R. Flin, R., P. O'Connor, M. Crichton. Safety at the Sharp End: Training Nontechnical Skills. In: *Ashgate Publishing Ltd., Aldershot, England*, 2008.
- [32] C. North. Toward measuring visualization insight. In: *IEEE Computer Graphics and Applications*, Volume: 26, Issue: 3, pages: 6-9, Electronic ISSN: 1558-1756, DOI: 10.1109/MCG.2006.70, 2006 .
- [33] P. Saraiya, C. North, and K. Duca. An Insight-Based Methodology for Evaluating Bioinformatics Visualizations. In: *IEEE TRANSACTIONS ON VISUALIZATION AND COMPUTER GRAPHICS*, VOL. 11, NO. 4, JULY/AUGUST 2005
- [34] G. Fletcher, R. Flin, P. McGeorge , R. Glavin, N. Maran and R. Patey. Anaesthetists' Non-Technical Skills (ANTS): evaluation of a behavioural marker system. In: *British Journal of Anaesthesia*, 90 (5): 580-8, DOI: 10.1093/bja/aeg112, 2003.
- [35] K. Reda, A. E. Johnson, J. Leigh, M. E. Papka. Evaluating user behavior and strategy during visual exploration. In: *BELIV'14*, Paris, France, DOI:10.1145/2669557.2669575, 2014.
- [36] L. L. Lacher, G. S. Walia, F. Fagerholm, M. Pagels, K. Nygard, J. Munch. A Behavior Marker tool for measurement of the Non-Technical Skills of Software Professionals: An Empirical Investigation. In: *Proceedings of the 27th International Conference on Software Engineering and Knowledge Engineering (SEKE)*, DOI:10.18293/SEKE2015-227, 2015.

- [37] N. Riem, S. Boet, M. D. Bould, W. Tavares and V. N. Naik. Do technical skills correlate with non-technical skills in crisis resource management: a simulation study. *British Journal of Anaesthesia* 109 (5): 723–8 (2012), Advance Access publication 31 July 2012 . doi:10.1093/bja/aes256.
- [38] A. Endert, C. North, R. Chang, M. Zhou. Toward Usable Interactive Analytics: Coupling Cognition and Computation. In: *IEEE Computer Graphics and Applications*, Volume:35, Issue:4, Page(s): 94 –99, DOI: 10.1109/MCG.2015.91, 2015.
- [39] G. Wang, X. Zhang, S. Tang, C. Wilson, H. Zheng, B. Zhao. Clickstream User Behaviour Models. In: *ACM Transactions on the WebJuly 2017 Article No.: 21* <https://doi.org/10.1145/3068332>, 2017.
- [40] W. Hua, Y. Song, H. Wang, X. Zhou. Identifying Users' Topical Tasks in Web Search. In: *WSDM'13: Proceedings of the sixth ACM international conference on Web search and data mining*, Pages 93–102, <https://doi.org/10.1145/2433396.2433410>, February 2013.
- [41] W. Wu, H. Li, H. Wang, and K. Q. Zhu. Probase: A probabilistic taxonomy for text understanding. In *SIGMOD*, 2012.
- [42] R. Jones and K. L. Klinkner. Beyond the session timeout: automatic hierarchical segmentation of search topics in query logs. In: *CIKM*, pages 699–708, 2008.
- [43] X. Li, C. Joshi, A. Y. S. Tan, R. K. L. Ko. Inferring User Actions from Provenance Logs. In: *TRUSTCOM '15: Proceedings of the 2015 IEEE Trustcom/BigDataSE/ISPA - Volume 01*, Pages 742–749, <https://doi.org/10.1109/Trustcom.2015.442>, 2015.
- [44] F. Zhang, W. He, X. Liu and P. G. Bridges. Inferring users' online activities through traffic analysis. In: *WiSec '11: Proceedings of the fourth ACM conference on Wireless network security*, Pages 59–70, <https://doi.org/10.1145/1998412.1998425>, 2011.
- [45] N. Kodagoda, S. Pontis, D. Simmie, S. Attfield, B. L. W. Wong, A. Blandford, C. Hankin. Using Machine Learning to Infer Reasoning Provenance from User Interaction Log Data: Based on the Data/Frame Theory of Sensemaking. In: *Sage Journals*, Volume: 11 issue: 1, Page(s): 23-41, doi.org/10.1177/1555343416672782, 2016.
- [46] G. Klein, J. K. Phillips, E. L. Rall, D. A. Peluso. A data-frame theory of sensemaking. In: *R. R. Hoffman (Ed.), Expertise out of context: Proceedings of the Sixth International Conference on Naturalistic Decision Making* (p. 113–155). Lawrence Erlbaum Associates Publishers, 2007.
- [47] C. C. Gramazio, J. Huang, D. H. Laidlaw. An Analysis of Automated Visual Analysis Classification: Interactive Visualization Task Inference of Cancer Genomics Domain Experts. In: *IEEE transactions on visualization and computer graphics*, vol. 14, no. 8, August, 2015.

References

- [48] E. T. Brown, A. Ottley, H. Zhao, Q. Lin, R. Souvenir, A. Endert, R. Chang. Finding Waldo: Learning about Users from their Interactions. In: *IEEE Transactions on Visualization and Computer Graphics (Volume: 20, Issue: 12)*, DOI: 10.1109/TVCG.2014.2346575, Dec. 31, 2014.
- [49] J. Shen, L. Li, T. G. Dietterich, J. L. Herlocker. A Hybrid Learning System for Recognizing User Tasks from Desktop Activities and Email Messages. In: *IUI '06: Proceedings of the 11th international conference on Intelligent user interfaces* January 2006 Pages 86–92 <https://doi.org/10.1145/1111449.1111473>, 2006.
- [50] D. Bahdanau, K. Cho, Y. Bengio. Neural Machine Translation by Jointly Learning to Align and Translate. In: *3rd International Conference on Learning Representations, (ICLR)*, <http://arxiv.org/abs/1409.0473>, 2015.
- [51] T. Luong, H. Pham, C. D. Manning. Effective Approaches to Attention-based Neural Machine Translation. In: *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, Pages:1412–1421, DOI:10.18653/v1/D15-1166, 2015.
- [52] J. Cheng, L. Dong and M. Lapata. Long Short-Term Memory-Networks for Machine Reading. In: *Journal: CoRR, Volume: abs/1601.06733*, eprint: 1601.06733, url: <http://arxiv.org/abs/1601.06733>, 2016.
- [53] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, Ł. Kaiser, I. Polosukhin. Attention is all you need. In: *NIPS'17: Proceedings of the 31st International Conference on Neural Information Processing Systems*, Pages 6000–6010, 2017.
- [54] P. Joshi. Article: How do Transformers Work in NLP? A Guide to the Latest State-of-the-Art Models, Analytics Vidhya, url: <https://www.analyticsvidhya.com/blog/2019/06/understanding-transformers-nlp-state-of-the-art-models/>, 2019.
- [55] J. Devlin, MW Chang, K. Lee, K. Toutanova. BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. *Journal: CoRR*, url: <http://arxiv.org/abs/1810.04805>, volume: abs/1810.04805, 2018.
- [56] A. Radford, K. Narasimhan, T. Salimans, and I. Sutskever. Improving language understanding with unsupervised learning. Technical report, OpenAI, 2018.
- [57] M. Peters, M. Neumann, M. Iyyer, M. Gardner, C. Clark, K. Lee, and L. Zettlemoyer. Deep contextualized word representations. In *NAACL*, 2018.
- [58] P. Linardatos, V. Papastefanopoulos, S. Kotsiantis. Explainable AI: A Review of Machine Learning Interpretability Methods. In: *Entropy 2021, 23, 18*. <https://dx.doi.org/10.3390/e23010018>, 2021.
- [59] F. Doshi-Velez, B. Kim, Towards a rigorous science of interpretable machine learning. *arXiv 2017, arXiv:1702.08608*, 2017.

- [60] A. Adadi, M. Berrada. Peeking inside the black-box: A survey on Explainable Artificial Intelligence (XAI). In: *IEEE Access*, 6, 52138–52160, 2018.
- [61] T. Miller. Explanation in artificial intelligence: Insights from the social sciences. In: *Artif. Intell.*, 267, 1–38, 2019.
- [62] L. H. Gilpin, D. Bau, B. Z. Yuan, A. Bajwa, M. Specter, L. Kagal. Explaining explanations: An overview of interpretability of machine learning. In: *Proceedings of the 2018 IEEE 5th International Conference on Data Science and Advanced Analytics (DSAA), Turin, Italy, 1–3 October 2018*; pp. 80–89, 2018.
- [63] M. T. Ribeiro, S. Singh, and C. Guestrin. Why Should I Trust You? In: *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining - KDD '16. New York, New York, USA: ACM Press*, pp. 1135–1144, 2016.
- [64] S. M. Lundberg, G. Erion, H. Chen, A. DeGrave, J. M. Prutkin, B. Nair, R. Katz, J. Himmelfarb, N. Bansal & S.I. Lee. From local explanations to global understanding with explainable AI for trees. In: *Nature Machine Intelligence*, volume 2, pages: 56–67, January 2020.
- [65] E. Strumbelj, and I Kononenko. Explaining prediction models and individual predictions with feature contributions. *Knowledge and information systems* 41.3 (2014): 647-665.
- [66] S. M. Lundberg, B. Nair, M. S. Vavilala, H. Mayumi, J. E. Michael, A. Trevor, D. E. Liston, D. KW. Low, SF. Newman, J. Kim & S.I. Lee. Explainable machine-learning predictions for the prevention of hypoxaemia during surgery. *Nat Biomed Eng* 2, 749–760, <https://doi.org/10.1038/s41551-018-0304-0>, 2018.
- [67] P. F. Vitiello and R. S. Kalawsky. Visual analytics: A sensemaking framework for systems thinking in systems engineering. In: *IEEE International Systems Conference SysCon 2012*, pp. 1-6, doi: 10.1109/SysCon.2012.6189489, 2012.
- [68] W. A. Pike, J. Stasko, R. Chang and T.A. O'Connell. The science of interaction: *Information Visualization*, 8(1): 263-274, 2009.
- [69] Wong, B. L. W., Xu, K. and Attfield, S. Provenance for intelligence analysis using visual analytics. In: *CHI 2011: Workshop on Analytic Provenance, Vancouver, BC, Canada, 07-08 May 2011*.
- [70] S. J. Attfield, S. K. Hara, and B. L. William Wong. Sensemaking in Visual Analytics: Processes and Challenges. In: *EuroVAST'10: The 1st European Symposium on Visual Analytics Science and Technology*, 2010.
- [71] J. Islam, K. Xu, B. L. W. Wong. Uncertainty of Visualizations for SenseMaking in Criminal Intelligence Analysis. In: *EuroRV3: EuroVis Workshop on Reproducibility, Verification, and Validation in Visualization*, ISBN 978-3-03868-066-6, p25-29, 2018.

References

- [72] J. Islam, C. Anslow, K. Xu, B. L. W. Wong. Analytical Provenance for Criminal Intelligence Analysis. In: *VALCRI WHITE PAPER SERIES, VALCRI-WP-2017-009*, <http://valcri.org/>, 2017.
- [73] B. L. W. Wong, C. Rooney, N. Kodagoda. Analyst User Interface: Thinking Landscape as Design Concept. VALCRI-WP-2017-002 AUI Thinking Landscape. <http://valcri.org/>, 2016.
- [74] K. Brodlie, R. A. Osorio, A. Lopes. A Review of Uncertainty in Data Visualization. In: *Expanding the Frontiers of Visual Analytics and Visualization, (2012)*. ISBN 978-1-4471-2803-8, pp 81-109, 2012.
- [75] E. Geoffrey, A. Dix. Decision Making Under Uncertainty in Visualisation? In: IEEE VIS2015. Chicago, USA, Oct 25, 2015 - Oct 30, 2015. In: *POTTER, Kristi, ed., Rüdiger WESTERMANN, ed.. VDMU (2015): Workshop on Visualization for Decision Making under Uncertainty*. IEEE VIS2015. Chicago, USA, Oct 25, 2015 - Oct 30, 2015.
- [76] D. Sacha, H. Senaratne, B. C. Kwon, G. Ellis & D. A. Keim. The role of uncertainty, awareness, and trust in visual analytics. In: *IEEE Transactions on Visualization and Computer Graphics (Proceedings of the Visual Analytics Science and Technology) 2016 Jan; 22(1):240-9*, doi: 10.1109/TVCG.2015.2467591, 2016.
- [77] B. PLEWE: The Nature of Uncertainty in Historical Geographic Information, *Transactions in GIS*, 6 (4), 431–456, 2002.
- [78] B. M. Muir: Trust between humans and machines, and the design of decision aids, *International Journal of Man-Machine Studies* 27(5-6), 527–539, 1987.
- [79] S. Mckenna, D. Mazur, J. Agutter, M. Meyer. Design Activity Framework for Visualization Design. In: *IEEE Transactions on Visualization and Computer Graphics*, Volume: 20, Issue: 12, ISSN: 1077-2626, Pages: 2191 – 2200, 2014.
- [80] G. Klein, J. K. Phillips, E. L. Rall, and D. A. Peluso. A Data-Frame Theory of Sensemaking. In: *R. R. Hoffman, editor, Expertise out of context: Proceedings of the sixth international conference on naturalistic decision making*, pages 113–155. Mahwah, NJ: Lawrence Erlbaum Associates, 2003.
- [81] S. Takken, B. L. W. Wong. Tactile reasoning: hands-on versus hands-off—What is the difference? In: *Cogn Tech Work* 17, 381–390. <https://doi.org/10.1007/s10111-015-0331-5>, 2015.
- [82] B. L. W. WONG, L. ZHANG, I. D. H. SHEPHERD. VALCRI: Addressing European Needs for Information Exploitation of Large Complex Data in Criminal Intelligence Analysis[C]//*European Data Forum*. 2014: 19-20.
- [83] T. J. JANKUNKELLY, K. L. MA, M. GERTZ. A model and framework for visualization exploration[J]. *IEEE Trans Vis Compute Graph*, 13(2):357-369, 2007.
- [84] KE Weick. Sensemaking in Organizations. Sage, Thousand Oaks, CA, 1995.

- [85] A. Endert, R. Chang, C. North, M. Zhou. Semantic Interaction: Coupling Cognition and Computation through Usable Interactive Analytics. Published in: *IEEE Computer Graphics and Applications*, Volume: 35, Issue: 4, July-Aug. INSPEC Accession Number: 15305788, 2015.
- [86] R. P. Cooper and T. Shallice. Hierarchical schemas and goals in the control of sequential behaviour, 2006.
- [87] G. W. Ryan. What do sequential behavioral patterns suggest about the medical decision-making process?: modeling home case management of acute illnesses in a rural Cameroonian village. *Social Science & Medicine*, 46(2), 209e225, 1998.
- [88] B. M. Yamauchi and R. D. Beer. Sequential behavior and learning in evolved dynamical neural networks. *Adaptive Behavior*, 2(3), 219e246, 1994.
- [89] H. F. O'Neil. Perspectives on computer-based performance assessment of problem solving. *Computers in Human Behavior*, 15, 255e268, 1999.
- [90] H. F. O'Neil, S. Chuang, and G. K. W. K. Chung. Issues in the computer-based assessment of collaborative problem solving. *Assessment in Education: Principles, Policy & Practice*, 10, 361e374, 2003.
- [91] E. Care and P. Griffin. An approach to assessment of collaborative problem solving. *Special issue: assessment in computer supported collaborative learning*. Research and Practice in Technology Enhanced Learning, 9(3), 367e388, 2014.
- [92] M. L. Commons, E. J. Trudeau, S. A. Stein, F. A. Richards and S. R. Krause. The existence of developmental stages as shown by the hierarchical complexity of tasks. *Developmental Review*, 8, 237e278, 1998.
- [93] A. N. Fontenot. Effects of Training in Creativity and Creative Problem Finding Upon Business People. In: *the Journal of Social Psychology*, 133(1), 11-22, 1992.
- [94] J. Islam, C. Anslow, K. Xu, B. L. W. Wong and L. Zhang. Towards analytical provenance visualization for criminal intelligence analysis. In: *Computer Graphics and Visual Computing (CGVC'16), 15-16 Sept 2016, Bournemouth University, United Kingdom*. ISBN 9783038680222. [Conference or Workshop Item] (doi:[10.2312/cgvc.20161290](https://doi.org/10.2312/cgvc.20161290)), 2016.
- [95] R. Bakeman, and J. M. Gottman. Observing interaction : an introduction to sequential analysis. New York : Cambridge University Press, 2nd Ed, 1997.
- [96] E. P. Torrance. The nature of creativity as manifest in its testing. In: R. J. Sternberg (Ed.), *The nature of creativity* (pp. 43–73). New York: Cambridge University Press, 1988.

References

- [97] E. Dane and M. G. Pratt. Exploring Intuition and its Role in Managerial Decision Making. In: *Academy of Management Review*, Vol. 32, No. 1, <https://doi.org/10.5465/amr.2007.23463682>, 2007.
- [98] E. Salas, K. Wilson, S. Burke, D. Wightman. Does Crew Resource Management Training Work? An Update, an Extension, and Some Critical Needs. *Human Factors*, 2006.
- [99] S. Merritt. Affective Processes in Human-Automation Interactions. *Human Factors*, 2011.
- [100] R. M. Hogarth. *Educating Intuition*. University of Chicago Press, 2001.
- [101] N. I. A. Rahman, S. Z. M. Dawal and N. Yusoff. Subjective responses of mental workload during real time driving: A pilot field study. In: *IOP Conf. Series: Materials Science and Engineering 210* 012076 doi:10.1088/1757-899X/210/1/012076, 2017.
- [102] J. Islam, B. L. W. Wong and K. Xu. Analytic provenance as constructs of behavioural markers for externalizing thinking processes in criminal intelligence analysis. In: *Community-Oriented Policing and Technological Innovations*. Leventakis, Georgios and Haberfeld, M. R., eds. SpringerBriefs in Criminology . Springer, pp. 95-105. ISBN 9783319892931. [Book Section] (doi:10. /978-3-319-89294-8_10), 2018.
- [103] A. N. Dragunov, T. G. Dietterich, K. Johnsrude, M. McLaughlin, L. Li, J. L. Herlocker. TaskTracer: a desktop environment to support multi-tasking knowledge workers. In: *IUI '05: Proceedings of the 10th international conference on Intelligent user interfaces*, Pages 75–82, January 2005.
- [104] D. Keim, T. Munzner, F. Rossi, M. Verleysen. Bridging Information Visualization with Machine Learning. *Dagstuhl Reports, Volume 5, Issue 3, 10.4230/DagRep.5.3.1*, ISSN: 2192-5283, 2015.
- [105] D. Gotz and M. X. Zhou. Characterizing users' visual analytic activity for insight provenance. *Information Visualization*, 8(1):42–55, Jan 2009.
- [106] D. E. Rose, D. Levinson. Understanding User Goals in Web Search. In: *WWW '04: Proceedings of the 13th international conference on World Wide Web*, Pages 13–19, <https://doi.org/10.1145/988672.988675>, 2004.
- [107] A. Endert, W. Ribarsky, C. Turkay, W Wong, I. Nabney, I Díaz Blanco and F. Rossi. The State of the Art in Integrating Machine Learning into Visual Analytics. In: *Computer Graphics Forum*, Wiley, 2017, 36 (8), pp.458 – 486, DOI: 10.1111/cgf.13092, 2018.
- [108] J. M. Zacks and B. Tversky. Event Structure in Perception and Conception. In: *Psychological Bulletin*, volume: 127(1), pages: 3-21, 2001.
- [109] D. Newtonson. Attribution and the Unit of Perception of Ongoing Behaviour. In: *Journal of Personality and Social Psychology*, 28 (1): 28-38.

- [110] P. Bogunovich and D. Salvucci. Inferring Multitasking Breakpoints from Single-Task Data. In: Proceedings of the Annual Meeting of the Cognitive Science Society, 32. <https://escholarship.org/uc/item/720422kn>, 2010.
- [111] S. T. Iqbal and B. P. Baily. Understanding and Developing Models for Detecting and Differentiating Breakpoints during Interactive Tasks. In: CHI 2007 Proceedings • Tasks, ACM 978-1-59593-593-9/07/0004, 2007.
- [112] C. Bors, J. Wenskovitch, M. Dowling, S. Attfield, L. Battle, A. Endert, O. Kulyk, and R. S. Laramee. A Provenance Task Abstraction Framework. In: IEEE Computer Graphics and Applications, volume: 39, issue: 6, pages: 46-60, DOI: 10.1109/MCG.2019.2945720, 2019.
- [113] J. Vig. A Multiscale Visualization of Attention in the Transformer Model. In: Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics: System Demonstrations, pages 37–42, 2019.
- [114] K. Clark, U. Khandelwal, O. Levy, C. D. Manning. What Does BERT Look At? An Analysis of BERT’s Attention. In: Proceedings of the 2019 ACL Workshop Blackbox NLP: Analyzing and Interpreting Neural Networks for NLP, DOI: 10.18653/v1/W19-4828, 2019.
- [115] W. Lee, J. Ortiz, B. Ko, R. Lee. Time Series Segmentation through Automatic Feature Learning. Volume: abs/1801.05394, <http://arxiv.org/abs/1801.05394>, 2018.
- [116] T. Hastie, R. Tibshirani, J. Friedman. The Elements of Statistical Learning: Data Mining, Inference, and Prediction (2nd edition), Springer, February 2009.
- [117] A. Endert, P. Fiaux, C. North. Semantic Interaction for Visual Text Analytics. In: CHI’12, May 5–10, 2012, Austin, Texas, USA. ACM 978-1-4503-1015-4/12/05, 2012.
- [118] D. Fisher, I. Popov, S. Drucker and M. C. Schraefel. Trust Me, I’M Partially Right: Incremental Visualization Lets Analysts Explore Large Datasets Faster. In: Proceedings of the SIGCHI Conference on Human Factors in Computing Systems. ACM 1673–1682, 2012.
- [119] T. Gruber. A translation approach to portable ontology specifications. Knowledge Acquisition, 5(2), 199-220, 1993.
- [120] Yu, Z., Y. Nakamura, et al. Ontology-based semantic recommendation for context-aware e-learning. In: Ubiquitous Intelligence and Computing, 898-907, 2007.
- [121] S. Shishehchi, S. Banihashem, et al. A proposed semantic recommendation system for e-learning: A rule and ontology based e-learning recommendation system, IEEE, 2010.
- [122] J. Han, M. Kamber and J. Pei. Data mining: concepts and techniques: concepts and techniques, Elsevier, 2011.

References

- [123] C. M. Bishop. Pattern Recognition and Machine Learning (Information Science and Statistics). In: *Springer-Verlag New York, Inc., Secaucus, NJ, USA*, 2006.
- [124] M. Vartak, S. Huang, T. Siddiqui, S. Madden, A. Parameswaran. Towards Visualization Recommendation Systems. In: *ACM SIGMOD Record, Volume 45 Issue 4, Pages 34-39*, 2016.
- [125] B. Shneiderman. The Eyes Have It: A Task by Data Type Taxonomy for Information Visualizations. In: *Proceedings of the IEEE Symposium on Visual Languages, pp. 336-343*, 1996.
- [126] V. Khatri, S. Ram, R. T. Snodgrass, G. M. O'Brien. Supporting User-Defined Granularities in a Spatiotemporal Conceptual Model. In: *Journal of Annals of Mathematics and Artificial Intelligence, Volume 36, Issue 1-2, pp 195-232*, 2002.
- [127] C. C. Xi, J. H. Faghmous, A. Khandelwal, V. Kumar. Clustering Dynamic Spatio-Temporal Patterns in the Presence of Noise and Missing Data. In: *Proceedings of the Twenty-Fourth International Joint Conference on Artificial Intelligence, IJCAI*, 2015.
- [128] G. Fletcher, R. Flin, P. McGeorge. Anaesthetists' Non-technical Skills: Development of Behavioural Marker Taxonomy. In: *Technical Report (97/SCR/1) to the Scottish Council for Postgraduate Medical and Dental Education. University of Aberdeen*, 2001.
- [129] MA Runco (ed.). *Divergent Thinking*. Norwood, NJ: Ablex, 1991.
- [130] C. Rooney, S. Attfield, B. L. W. Wong, S. Choudhury. INVISQUE as a tool for intelligence analysis: the construction of explanatory narratives. In: *International Journal of Human-Computer Interaction, 30 (9) . pp. 703-717. ISSN 1044-7318 [Article] (doi:10.1080/10447318.2014.905422)*, 2014.
- [131] L. Xin, J. Chaitanya, Y. S. T. Alan, Ryan K. L. Ko. Inferring User Actions from Provenance Logs. In: *IEEE International Conference on Trust, Security and Privacy in Computing and Communications, Electronic ISBN: 978-1-4673-7952-6, DOI: 10.1109/Trustcom.2015.442*, 2015.
- [132] M. Levandowsky and D. Winter. 1971. Distance between sets. *Nature* 234, (34-35), 1971.
- [133] LR James, RG Demaree, G. Wolf. Estimating within-group interrater reliability with and without response bias. *J Appl Psychol*; 69: 85-98, 1984.
- [134] LR James, RG Demaree, G. Wolf. An assessment of within group interrater agreement. *J Appl Psychol*; 78: 306-9, 1993.
- [135] PJ Johnson, TE Goldsmith. The importance of quality data in evaluating aircrew performance. US Federal Aviation Authority Technical Report. *Available from Federal Aviation Authority website: www.faa.gov/avr/afs/aqphome*, 1998.

- [136] TE Goldsmith, PJ Johnson. Assessing and improving evaluation of aircrew performance. *Int J Aviat Psychol*; 12: 223-40, 2002.
- [137] J. Islam, and Wong, B. L. W. Wong. Behavioural markers: Bridging the gap between art of analysis and science of analytics in criminal intelligence. European Intelligence and Security Informatics Conference (EISIC). In: 2017 European Intelligence and Security Informatics Conference, 11-13 Sept 2017, Dekelia Air Base, Attica, Greece. ISBN 9781538623855. [Conference or Workshop Item] (doi:10.1109/EISIC.2017.30), 2017.
- [138] A. George, Presidential Decisionmaking in Foreign Policy: *The Effective Use of Information and Advice* (Boulder, CO: Westview Press), Chapter 2, 1980.
- [139] B. Klampfer, R. Flin, R. L. Helmreich. Enhancing performance in high risk environments: recommendations for the use of behavioural markers. Ladenburg: Daimler-Benz Shiftung, Source: www.psyc.abdn.ac.uk/servo2:10, 2001.

A. Systematic Literature Review

- P1 J. Clauser and S. M. Weir. Intelligence research methodology: An introduction to techniques and procedures for conducting research in defense intelligence. Washington: Defense Intelligence School (Google Books), 1976.
- P2 R. V. Katter, C. A. Montgomery and J. R. Thompson. Human processes in intelligence analysis: phase 1 overview (pp. 74). Alexandria: U.S. Army Research Institute for the Behavioral and Social Sciences, 1979.
- P3 M. Fischl and A. C. Gilbert. Selection of Intelligence Analysts: DTIC Document, 1983.
- P4 S. R. Schneider. The criminal intelligence function: toward a comprehensive and normative model. *IALEIA Journal*, 9(2), 403-427, 1995.
- P5 I. Wing. The characteristic of successful and unsuccessful intelligence analysts. In: *The Journal of the Australian Institute of Professional Intelligence Officers*, 9(2), 4-11, 2000.
- P6 A. Wolfberg. To transform into a more capable intelligence community: A paradigm shift in the analyst selection strategy (pp. 32). Washington: National War College, 2003.
- P7 D. T. Moore, L. Krizan and E. J. Moore. Evaluating Intelligence: A Competency-Based Model. *International Journal of Intelligence and Counter-Intelligence*, 18(2), 204-220. doi: 10.1080/08850600590911945, 2005.
- P8 D. M. Allen. Building a better strategic analyst: a critical review of the U.S. army's all source analyst training program (pp. 59). Fort Leavenworth: United States Army Command and General Staff College, 2008.

References

- P9 N. Quarmby and L. J. Young. *Managing intelligence the art of influence*. Sydney: The Federation Press, 2010.
- P10 J. Richards. *The art and science of intelligence analysis*. Oxford: Oxford University Press, 2010.
- P11 P. F. Walsh. *Intelligence and intelligence analysis*. London: Routledge, 2011.
- P12 J. Corkill and A. Davies. The contemporary Australian intelligence domain. In: *The Journal of the Australian Institute of Professional Intelligence Officers*, 21(2), 37-53, (UnPub), 2013.
- P13 M. Gerber, B. L. W. Wong and N. Kodagoda. How analysts think: decision making in the absence of clear facts. Adaptation of the RPD model and the decision ladder to analysts' decision making. In: *Proceedings of the 7th European Intelligence Security Informatics Conference, EISIC 2016, on Counterterrorism and Criminology*, 17-19 August, 2016, Uppsalla, Sweden (pp. To be published): SAGE Publications, 2016a.
- P14 M. Gerber, B. L. W. Wong and N. Kodagoda. How analysts think: Intuition, Leap of Faith and Insight. In: *Proceedings of the Human Factors and Ergonomics Society 60th Annual Meeting*, 19-23 September 2016, Washington, D.C., USA (pp. 173-177): SAGE Publications, 2016b.
- P15 N. Selvaraj, S. Attfield, P. Passmore and B. L. W. Wong. How Analysts Think: Think-steps as a Tool for Structuring Sensemaking in Criminal Intelligence Analysis. In: *Proceedings of the 7th European Intelligence Security Informatics Conference, EISIC 2016, on Counterterrorism and Criminology*, 17-19 August, 2016, Uppsalla, Sweden (pp. To be published): SAGE Publications, 2016.
- P16 B. L. W. Wong and N. Kodagoda. How analysts think: Inference making strategies. In: *Proceedings of the Human Factors and Ergonomics Society 59th Annual Meeting*, 26-30 October 2015, Los Angeles, USA (pp. 269-273): SAGE Publications, 2015.
- P17 B. L. W. Wong, and N. Kodagoda. How analysts think: Anchoring, Laddering and Associations. In: *Proceedings of the Human Factors and Ergonomics Society 60th Annual Meeting*, 19-23 September 2016, Washington, D.C., USA (pp. 178-182): SAGE Publications, 2016.
- P18 N. Qazi, B. L. W. Wong, N. Kodagoda and R. Adderley. Associative Search through Formal Concept Analysis in Criminal Intelligence Analysis. In: *Proceedings of 2016 IEEE International Conference on Systems, Man, and Cybernetics SMC 2016*, October 9-12, 2016, Budapest, Hungary: IEEE Press, 2016.