A Three Tier Forensic Model for Automatic Identification of Evidence of Child Exploitation by Analysing the Content of Chat-Logs

Md. Waliur Rahman Miah

A thesis submitted in fulfilment
for the degree of Doctor of Philosophy

Principal Supervisor
Professor John Yearwood

Associate Supervisor
Dr. Siddhivinayak Kulkarni

School of Engineering and Information Technology
Faculty of Science and Technology
Federation University Australia

June 2016

Abstract

Detection of child exploitation (CE) in Internet chatting by locating evidence in the chat-log is an important issue for the protection of children from prospective online paedophiles. The un-grammatical and informal nature of chat-text makes it difficult for existing formal language processing techniques to handle the problem. The methodology of the current research avoids those difficulties by developing a multi-tier digital forensic model built on new ideas of psychological similarity measures and ways of applying them to chat-texts.

The model uses text classifiers in the beginning to identify shallow evidence of CE. For locating the particular evidence it is required to identify the behavioural pattern of CE chats consisting of documented CE psychological stages and associate the perpetrators' posts to them. Similarities among the posts of a chat play an important role for the task of differentiating and identifying these stages. To accomplish this task a novel similarity measure is constructed backed by a dictionary with terms associated with each CE stage. Using the new similarity measure in a hierarchical agglomerative algorithm a new clusterer is built to cluster the posts of a chat-log into the CE stages to learn whether it follows the CE pattern. Inspired by the field of recognition of textual entailment a new soft entailment technique is developed and implemented to locate the specific posts associated with the CE stages. Those specific posts of the perpetrator are extracted as the particular evidence from the chat-log.

It is anticipated that the developed methodology will have many future practical implementations. It would assist in the development of forensic tools for digital forensic experts in law and enforcement agencies to conveniently locate evidence of

online child grooming offences in a confiscated hard disk drive. Another future implementation would be a parental filter used by parents to protect their children from potential online offenders.

Statement of Authorship

Except where explicit references are made, the text of this thesis contains no material published elsewhere or extracted in whole or in part from a thesis by which I have qualified for or have been awarded another degree or diploma. No other person's work has been relied upon or used without due acknowledgement in the main text and bibliography of the thesis.

Md. Waliur Rahman Miah

Publications

During my PhD candidature some parts of the material presented in this thesis have been published in the following papers:

Miah, Md. W. R., Yearwood, J. and Kulkarni, S. (2015). Constructing an inter-post similarity measure to differentiate the psychological stages in offensive chats. *Journal of the Association for Information Science and Technology*, *Vol.* 66(5): (p. 1065–1081). doi: 10.1002/asi.23247. John Wiley & Sons, Ltd.

Miah, M. W. R., Yearwood, J., & Kulkarni, S. (2011). Detection of Child Exploiting Chats from a Mixed Chat Dataset as a Text Classification Task. In Proceedings of the Australasian Language Technology Association Workshop 2011, (p. 157-165), Canberra, Australia. ACM.

Acknowledgements

All praises is due to the Almighty Allah for giving me the ability to do this research. I would like to acknowledge all the supports provided by different people in different aspects for the successful completion of this thesis.

At first, I would like to express my sincere gratitude to my principal supervisor, Professor John Yearwood. His valuable and brilliant guidance, and, encouragement and continuous support have made it possible for me to complete this thesis. His profound knowledge and expertise have provided me opportunities to learn many new things during my PhD candidature. I also learnt from him how to be patient and keep hope when experiencing the tough side of research. He also instructed me on producing high quality academic writings. It has been my very great privilege to know and work with John and have had him as my supervisor, and I look forward to having more opportunities to learn from and work with him in the future.

I especially thank my wonderful associate supervisor Dr. Siddhivinayak Kulkarni for his great support and guidance. He also helped me in presenting my works and writing research papers. I feel very lucky to work with Sid along with my principal supervisor.

I do acknowledge the financial assistance from the Federation University Australia. I must thank the School of Engineering and Information Technology, Faculty of Science and Technology for providing me with the logistic support I needed in order to complete my Ph.D. study. I am grateful to the staff of the school, the faculty and the research services, and, many other people at the University who have assisted me in many official ways.

I am also grateful to the Department of Computer Science and Engineering, Dhaka University of Engineering and Technology (DUET), Gazipur, Bangladesh for the approval of a higher-study leave for my Ph.D. study.

In this occasion I would like to respectfully remember my loving mother who left the world during my PhD study. Her love and care made it possible for me to come this far. Losing her is such a void in my life that can never be filled. May the Almighty Allah let her rest in peace in the paradise.

Finally, I would like to thank my father, my brothers and sister, my loving wife and son, and, my other relatives and friends for all their patience, love, support and encouragement throughout my study.

Contents

Abstract i

Statement of Authorship iii

Publications v

Acknowledgements vii

Contents ix

List Of Figures xiii

List Of Tables xvii

1. Introduction 1

1.1 Background and Motivation 1

1.2 Research Problem 3

Research Questions 4

1.3 Contributions 5

1.4 Organization of this Thesis 6

2. Literature Review 9

2.1 Social and Psychological Issues Surrounding Online Chatting 10

2.1.1 What is Online Chatting? 10

2.1.2 The Problem of Online Child Exploitation 10

2.1.3 The Pattern of Child Exploitation 14

2.2 Legal Aspects of Child Exploitation 19

2.2.1 Legislations against Child Exploitation 19

2.2.2 Authenticity of Chat as Evidence 21
2.3 Analysis of Chat Messages....24
2.4 Text Classification Techniques for Chat-text 26
2.4.1 Background 27
2.4.2 Formulation of the Problem of Chat Classification 32
2.4.3 Classifiers for Chat-text Categorization 33
2.4.4 Naïve Bayes (NB) Classifier....37
2.4.5 Decision Tree Classifier 40
2.4.6 Classification via Regression 42
2.4.7 Psycholinguistic Features for Chat-Text Classifiers 49
2.5 Text Similarity Measures and Clustering the Chat-text....50
2.5.1 Background 50
2.5.2 Existing works on Sentence Similarity Measures....52
2.5.3 Text Clustering....54
2.5.4 *K*-means Clustering....56
2.5.5 Expectation Maximization Clustering....58
2.5.6 Hierarchical Clustering 60
2.6 Recognition of Textual Entailment (RTE) in Chat....66
2.6.1 Background 66
2.6.2 Definition of RTE....67
2.6.1 Different Sources of Entailment....68
2.6.2 Classification of Entailments 70
2.6.3 Existing Works on RTE....72
2.7 Existing Works on CE Detection 75
2.8 Chapter Summary....81

3. A Similarity Measure and a Clustering Approach for the Child Exploitation Domain....83
3.1 Structure of CE Chats and Need for a New Similarity Measure 83
3.2 CE Psychological Dictionary....86
3.2.1 Criteria and Rationale behind the Dictionary 87
3.2.2 Building the Dictionary....92
3.3 CE Psychological Similarity Measure 93

3.4 PsyHAC Clustering 100
3.5 Chapter Summary 103

4. Soft Entailment for the Child Exploitation Context 105
4.1 Why Entailment in CE detection? 105
4.2 Hard vs Soft Entailment 107
4.3 Logics for Soft Entailment 110
4.4 Our Approach to Soft Entailment 112
4.4.1 Construction of the CE Psychological Domain Vector Space Model 112
4.4.2 Recognition of CE Entailment (RCE) 119
4.4.3 Limitations of RCE 127
4.5 Chapter Summary 127

5. The Design of a Three Tier Model 129
5.1 Overall Approach 129
5.2 Tier One: Shallow Evidence Analysis 131
5.2.1 Methodology for the CE vs. Non-CE Classification Phase 132
5.2.2 Methodology for the Predator vs. Victim Classification Phase 133
5.3 Tier Two: Particular Evidence Detection 136
5.3.1 Methodology for the Clustering Phase 136
5.3.2 Methodology for the Entailment Phase 140
5.4 Tier Three: Evidence Producing 141
5.5 Chapter Summary 143

6. Experiments and Results 145
6.1 Collection of Data 145
6.1.1 Difficulties in Finding a Formal Benchmark 145
6.1.2 Data Used in this Research 146
6.2 Evaluation of the CEDM System 154
6.2.1 Evaluation Metrics for Chat-Text Classifiers 155
6.2.2 Evaluation Metrics for Chat-Post Clusterer 160
6.2.3 Evaluation Metrics for Entailment and Evidence Producing 165

6.3 CE vs. Non-CE Classification Experiments 166
6.4 Predator vs. Victim Classification Experiments 172
6.5 Clustering Experiments 175
6.6 Entailment Experiments 183
6.7 Experiments and Results of Combined phase 188
6.8 Chapter Summary 195

7. Conclusion 199
7.1 Overview 199
7.2 Research Contributions 200
7.3 Limitations and Future Directions 204
7.4 Prospective Applications 207

8. Appendix 209
Appendix A 209
A Perpetrator's Posts of a CE chat-log 209
Appendix B 213
Example of Hypothesis and Surrogates 213
Appendix C 216
Proof of "If CEPsySim(P_a, P_b) = 1, 216
then CEPsySim(P_c, P_a) = CEPsySim(P_c, P_b)" 216
Appendix D 218
Detailed Computation of Evaluation Metrics for Classification of the posts of a Chatlog 218
Appendix E 225
List of Selected Acronyms 225
Appendix F 229
List of Resources Used in the Experiments 229

9. Bibliography 231

List Of Figures

Figure 1.1: A snippet of a child exploiting type chat. 2

Figure 2.1: A model of luring communication theory. 17

Figure 2.2: Linear classifier; There are infinite number of hyperplanes that separate two linearly separable classes. 34

Figure 2.3: A non-linear classification problem. 35

Figure 2.4: Linear regression as a straight line. 44

Figure 2.5 : Input training instance X, coefficient β , and output indicator matrix Y.. 46

Figure 2.6: An example of a chat-posts data set with a clear cluster structure. 55

Figure 2.7: A K-means example for K = 2 in **R**2.. 57

Figure 2.8: A dendrogram showing hierarchical agglomerative clustering (HAC). 61

Figure 2.9: The different notions of cluster similarity used by the four HAC algorithms. 63

Figure 2.10: Three iterations of centroid clustering. Each iteration merges the two clusters whose centroids are closest. 65

Figure 3.1: Psychological structure of a CE chat. 84

Figure 3.2: Total number of terms (N) vs threshold T. 90

Figure 3.3: Average number of category types for each term (C_{avg}) vs threshold T. ..91

Figure 3.4: Comparison of average pairwise similarity among the chat-posts.............98

Figure 3.5: Discriminating power (δ) of different similarity techniques.........................99

Figure 4.1: Procedure of a system to decide whether a chat contains the act of Child Exploitation..106

Figure 4.2: Parse tree output from Stanford parser for two strings with the same meaning. 108

Figure 4.3: Term vector space model to CE Psychological domain vector space model. P is a test chat-post in the vector space...112

Figure 4.4: Determining inverse category frequency (iCF). 115

Figure 4.5: A list of examples of the evidential hypotheses..119

Figure 4.6: Algorithm to make surrogates of hypothesis 122

Figure 4.7: Determining number of hypotheses-surrogates and entailment-threshold. 125

Figure 5.1: Overall system architecture of three tier CE evidence detection model. 130

Figure 5.2: Classification of chat-logs into CE and Non-CE 132

Figure 5.3: Method for Predator vs Victim classification task..134

Figure 5.4: Procedural framework for clustering-phase. 137

Figure 5.5: Average NMI values vs values of R in the training set. 138

Figure 5.6: Entailment by RCE..141

Figure 6.1: Example of direction changing portion of a SF chat. 149

Figure 6.2: Distribution of predators' posts over the four categories of BF, IE, GR and AP. 153

Figure 6.3: Recall and Accuracy of different classifiers for CE vs. Non-CE Classification. 171

Figure 6.4: Comparison of evidence entailment among GR, AP and IE context. 187

Figure 6.5: Comparison between the combined-phase and entailment-phase for the GR contextual evidence detection. 192

Figure 6.6: Comparison between the combined-phase and entailment-phase for the AP contextual evidence detection. 192

Figure 6.7: Comparison between the combined-phase and entailment-phase for the IE contextual evidence detection. 193

Figure 6.8: Comparison between the combined-phase and entailment-phase for the results averaged among GR, AP and IE contextual evidence detection. 193

Figure 7.1: Exponential change of inverse category frequency (iCF) 206

List Of Tables

Table 3.1: An excerpt of DF-table 87

Table 3.2: Number of terms (N) in the dictionary according to minimum number in MaxDF (M). 88

Table 3.3: An excerpt of CEPsy dictionary. 92

Table 3.4: CEPsy dictionary entries for the terms in RVS (P_1, P_2). 95

Table 3.5: Randomly selected 40 posts. 97

Table 4.1: Excerpt of the table corresponding to the CE Psy Domain Vector Space Model (CEPDVSM) 118

Table 6.1 : Information summary of posts count. 153

Table 6.2: A confusion matrix for a single-class classifier. 156

Table 6.3: A confusion matrix for a multi-class classifier. 156

Table 6.4: The contingency table for category c_i. 157

Table 6.5: Confusion matrices for different classification experiments 168

Table 6.6: Summary of result metrics for CE vs. Non-CE classification. 171

Table 6.7: kNN Results for Predator vs. Victim classification. 174

Table 6.8: Results from Different Classifiers for Predator vs. Victim classification. 174

Table 6.9: Comparison of NMI results of different clusterers for the test-set. 178

Table 6.10: Pairwise statistical comparison of different clusterers with PsyHAC clusterer. 179

Table 6.11: Results of association of clusters to CE evidence. 180

Table 6.12: Comparison of accuracies between PsyHAC and ChatCoder2. 182

Table 6.13: Results of GR evidence entailment. 185

Table 6.14: Results of AP evidence entailment. 185

Table 6.15: Results of IE evidence entailment... 186

Table 6.16: Result of combined-phase for GR evidence. 189

Table 6.17: Result of combined-phase for AP Evidence. 189

Table 6.18: Result of combined-phase for IE Evidence. 190

Table 6.19: Macro-averaged results of combined-phase. 190

Table 6.20: Micro-averaged results of combined-phase. 190

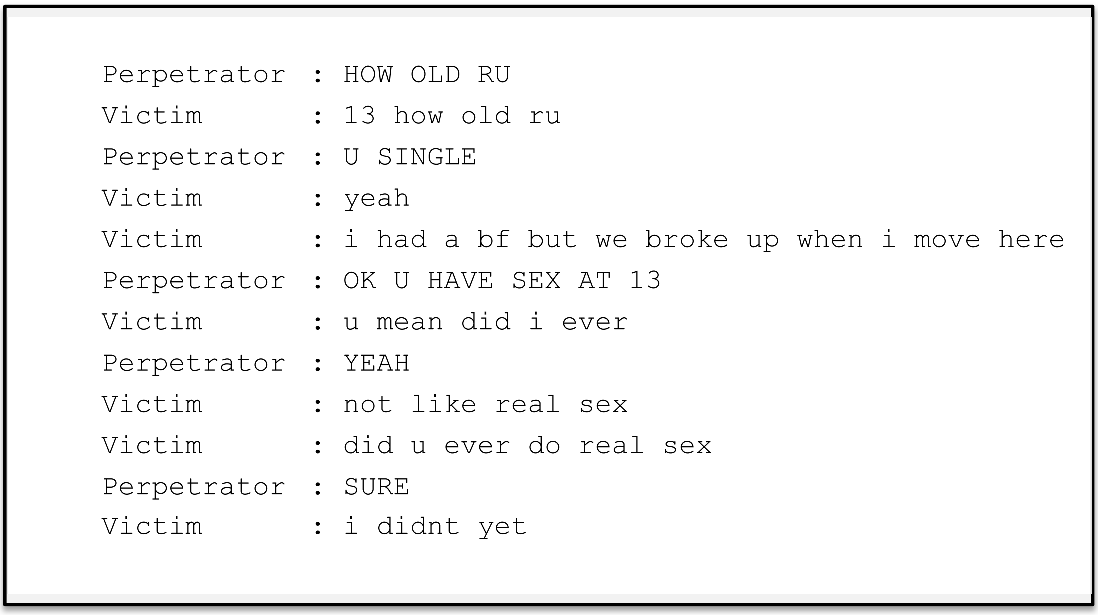
Table 6.21: Statistical Comparison between Combined and Entailment Phases 194

Chapter 1

Introduction

1.1 Background and Motivation

The Internet and a range of communicating devices are increasingly available to the people in the modern day around the world. Along with adults, youngsters and children are also accessing the Internet for information, education and social involvement as well as a source of perfectly innocent fun, games and connecting to friends through online chatting and social networking. Using online chat-rooms one can make friends from far places of the world where one may not be able to visit. Online chatting has become a popular tool for personal as well as group communication. It is cheap, convenient, virtual and private in nature. In online chatting one can hide one's personal information behind the monitor. This makes it a source of fun on one hand which can become a threat on the other hand. The privacy and virtual nature of this medium increased the chance of some heinous acts which one may not commit in the real world. O'Connell (2003) points out that the Internet affords greater opportunity for adults with a sexual interest in children to gain access to children. Communication between victim and predator can take place whilst both are in their respective real world homes but sharing a private virtual space. Young (2005) profiles this kind of virtual opportunist as 'situational sex offenders' along



```
Perpetrator : HOW OLD RU
Victim      : 13 how old ru
Perpetrator : U SINGLE
Victim      : yeah
Victim      : i had a bf but we broke up when i move here
Perpetrator : OK U HAVE SEX AT 13
Victim      : u mean did i ever
Perpetrator : YEAH
Victim      : not like real sex
Victim      : did u ever do real sex
Perpetrator : SURE
Victim      : i didnt yet
```

Figure 1.1: A snippet of a child exploiting type chat.

with the ‘classical sex offenders’. Both these types of offenders are taking the advantages of the Internet to solicit and exploit children. This kind of solicitation or grooming by the use of an online medium for the purpose of exploiting a child may be referred to as the problem of ‘online child exploitation’. A broader explanation of this problem is provided in the next chapter (Chapter-2: Literature Review). This research is highly motivated by the responsibility to protect children from those online offenders.

Figure 1.1 shows a chat-snippet which gives a hint of the grotesque nature of child exploitation through online chatting. In this chat-snippet a perpetrator is grooming a 13 year old child by bringing up discussion about sex. The parents or guardians of a child will surely be concerned if they notice that an adult is exploiting the child with such luring, provocative and offensive language. However it is very difficult for parents, guardians and members of Law and Enforcement Agency (LEA) to watch over the children all the time to protect them from online paedophiles loitering over the vast space of the Internet. Moreover children in their adolescence and teenage years expect privacy when chatting online. This makes it more difficult for the guardians to protect them. To deter and prevent the problem of online child exploitation an automatic system is required that can identify the elements of child exploitation in chats. Such a system will be beneficial for the parents, guardians, members of LEA, and the society as a whole.

Most of the chatting programs like yahoo, windows live or GoogleTalk have the option of storing the chat-texts in log-archives. According to pjfi.org and Krone (2005) chat-logs have been used as evidence to establish in a court that a paedophile attempting to exploit children. Chat-texts are inherently informal in nature. Finding evidence of child exploitation (CE) in chats by analysing the informal texts can be an interesting new direction of the text processing field. The current research aims to develop a novel methodology that can automatically identify child exploitation in chats through the analysis of the content of the chat-logs using data-mining and machine learning techniques.

1.2 Research Problem

The current research aims to develop a reliable methodology that is capable of finding evidence of child exploitation (CE) in chat text through analysing its contents by using data mining and text processing techniques. Finding evidence of child exploitation in chat text is not trivial. The chat-text is conversational in nature and does not follow the rules of formal grammatical structure (Rosa and Ellen, 2009; Kucukyilmaz et al., 2008). It is difficult for existing natural language processing techniques to analyse the un-grammatical and erroneous chat-text. Most of the existing text processing techniques are based on lexical matching, finding semantic similarities and using specific knowledge-based systems. These techniques also require the texts to be grammatically parsed correctly. This is also difficult for those techniques to be applied on chat-text due to its un-structured and un-grammatical nature. In the literature review we will see that CE type chats follow a documented psychological behavioural deceptive communicative pattern. Instead of mere lexical or semantic analysis, a CE detection process also could benefit from addressing that psychological pattern. Under these circumstances a thorough investigation is required to find how chat texts can be analysed and then look for the evidence of CE in them.

Finding and developing suitable methodologies to analyse the content of chat-text and locating the evidence of CE are the main research problems addressed in this dissertation.

Research Questions

To investigate possible solutions for the above mentioned research problems the following prime research question needs to be addressed:

"How can reliable methodologies and computationally automatic techniques be developed for finding evidence of child exploitation (CE) in chat-logs by analysing the informal text of chat?"

The automation of the CE evidence finding process from chat text is not straightforward. It has been mentioned that chat texts are not like regular texts; they are not grammatically well structured, and are conversational rather than formal. Therefore answering the prime research question requires broader investigation and breaking the main research problem down into more focused research sub-problems. These sub-problems can be represented by the following research questions:

1. How do the traditional text classifiers behave in classifying chat-logs into Child Exploiting (CE) and non Child Exploiting (non-CE) classes?
2. How do the classifiers behave in classifying the participants of the chat into CE predator or CE victim?
3. How can the pattern of progression and profile of CE chats identified in the psychological literature be used to aid evidence detection?
4. How do we frame the problem of CE evidence detection into a manageable problem of Textual Entailment on chat-logs?

In the search for the answers of the above questions a research methodology is developed and, experiments and analysis are done sequentially throughout the course of this research.

1.3 Contributions

This research will be a bridge between two different IT fields; text processing and computer forensics. In this research project text processing and data mining techniques are applied to accomplish a forensic task. Using the methodology developed in this research, by processing the chat-texts the evidence of a crime would be detected to produce in the court of law.

There has been considerable amount of work done by researchers of different fields to protect children from the Internet offenders of child enticement. Law and Enforcement Agencies (LEA) have constructed necessary laws. Psychology and communication researchers also have done plenty of research about psychological and communicative issues of offenders and victims of the Internet child exploitation. But it is difficult to find much research in the IT field that concerns for the protection of those children, though the offence is being done by using a modern IT tool, the Internet. Through this research, contribution is made in the IT field in parallel with LEA and the field of psychology to protect children from those perpetrators.

Apart from the above this research intends to make the following specific contributions:

1. Utilization of a special psychometric and word information feature set in traditional text classifiers to capture the behavioural psychological signature which improves the effectiveness of text classifiers to categorize chat-logs into different types compared with Child Exploitation (CE) type.
2. Construction of a new CE Psychological term dictionary by mining the terms of CE chats associated with the CE behavioural psychological contextual stages. This dictionary would work as a lexical resource for a new "similarity measure for CE text" and a new "weighting measure for term importance in CE domain" to capture the documented CE psychological contexts in chat-texts.
3. Design and implementation of a new similarity measure for CE text fragments. The new similarity measure would compute the CE psychological contextual similarity between a pair of chat-posts.

4. Development of a new clustering method that would allow us to learn the behavioural pattern of a CE chat by accumulating the chat-posts into their CE psychological stages.
5. Construction of a new term weighting measure that would assist in finding term importance in chat posts in CE domain.
6. Construction of a new CE psychological domain vector space model that would work better than the common vector space model to accomplish the task of CE evidence finding.
7. Development of a soft entailment technique that would be applied on ungrammatical chat-texts for the task of CE evidence finding.

1.4 Organization of this Thesis

This thesis is organized into seven chapters as follows:

The current chapter gives an overall view of the research problem, its background and motivation. It also covers the objective of the research as the research questions and finally it mentions the contributions of the current research.

Chapter-2 provides a study on the related literature in social, psychological and legislative fields regarding online child exploitation. The construction of the chat-messages is also analysed. It also provides a review on selected text processing, data mining, and text entailment techniques which are to be useful to process the chat-text, analyse the contents, and locate the particular CE evidence.

Chapter-3 explains the construction of a new CE psychological dictionary focused on behavioural psychology of child exploitation. Using the new dictionary a new similarity measure called 'CEPsy similarity' is developed which captures CE psychological context similarity between a pair of chat-posts. The chapter also describes a new clustering approach for the child exploitation domain.

In Chapter-4 a new term weighting measure is constructed on CE chat corpus. Using this new measure a new CE psychological domain vector space model is constructed by transforming a term vector space model of the CE chat corpus. A new soft entailment method is designed by utilizing the CE psychological domain vector space model.

A three tier CE evidence detection model is designed in Chapter-5. The model incorporates a phase by phase approach to detect the evidence of child exploitation by analysing the content of chat-logs.

Chapter-6 describes the data-sets and the evaluation metrics used in the current research. That chapter also demonstrates the performance of the three tier CE evidence detection model through experiments, results and analyses for different stages of the model.

Finally, Chapter7 discusses the conclusions of this work. It reviews the research problems and the contributions of this research. Limitations and future directions of this research are provided. Some prospective applications of this research are also discussed.

Appendix A presents a chat-log from the chat data-set used in the experiments. An example of a hypothesis and its surrogates are provided in Appendix B. Appendix C presents a proof that if two chat posts P_a and P_b are equal in the measure of CEPsy similarity, that is, they have 100% CEPsy similarity in between themselves, then a third post P_c will have same CEPsy similarity with both P_a and P_b. Appendix D provides detailed computation of evaluation metrics for classification of the posts of a chatlog. Appendix E elaborates selected acronyms used in this dissertation. Appendix F lists the resources such as system, programming languages and software packages used in the experiments of this research.

Chapter 2

Literature Review

The research questions outlined in the previous chapter reveal that finding evidence from chat-texts requires knowledge of different fields. This includes: the knowledge of the psychological behaviour of the people behind the chat, legal aspects related to child protection, and a range of data mining and text processing techniques to analyse the chat-text.

Understanding the psychological patterns of behaviour and the progress of criminal behaviour would provide grounds for identifying and characterizing the chats. The beginning parts of this chapter will provide a brief review of the related literature in psychology, criminology and law focusing on the protection of children from online grooming.

To automate the digital forensic process of finding evidence of child exploitation, the methodology of this research incorporates data mining and text processing techniques. The different aspects of such techniques related to this current research are explained in the later parts of this chapter.

2.1 Social and Psychological Issues Surrounding Online Chatting

2.1.1 What is Online Chatting?

Online chat has become a popular and common form of communication for people of all ages. In the simplest form of chatting, two users use a common window where both of them can see what the other one is typing. Multiple user chat rooms are also available, where a number of people type in a common window. Contemporary chatting software like yahoo messenger, googletalk, windows live, and skype also provide voice and video communication facilities.

On one hand, online chatting provides opportunity for meeting people from different parts of the world, where one may not be able to visit in his life time. Using this tool one can communicate, learn, and acquire knowledge of diversified, multicultural people around the globe. On the other hand, for naive users it may pose different kinds of threats. Internet chatting has inherent characteristics of masquerading. A user's true identity is not obvious and there is the ability to impersonate someone else. This option has the advantage of fantasy and disadvantage of being a threat to others. Perpetrators can disguise themselves and elicit personal information from a user who is not alert.

Paedophiles are exploiting this concealing property of the Internet chatting as an opportunity to solicit children online, even sometimes by appearing to be another child of similar age. Generally, this kind of soliciting is known as online grooming. Online grooming is a part of online child sex exploitation.

2.1.2 The Problem of Online Child Exploitation

Throughout the history of human society there have been individuals having sexual fantasies or erotic attractions towards children (Choo, 2009). Some of those individuals may never enact upon these due to self-respect or social barriers; some may act upon it only when a safe opportunity is available and some may always find a way to do it. The Internet has opened a new realm for these second and third type of individuals to abuse children.

Research has shown that perpetrators of child sexual abuse exhibit specific psychological traits such as low self-esteem, interpersonal inadequacy (Fisher, Beech, and Browne, 1999; Panton, 1979; Ward, McCormack, and Hudson, 1997), a lack of empathy, a fear of intimacy, and the inability to form intimate relationships with adults (Marshall, Barbaree, and Fernandez, 1995; Ward, Hudson, and Marshall, 1996). Children are more susceptible to form bonds of intimate relationship and trust. So sex predators find them easy to prey upon.

Children do not have the cognitive maturity to understand the persuasive intent of the perpetrator. Children are still in the age of learning to communicate effectively, they are even less socially skilled than adults (Lamb and Brown 2006). The potential child victim has shortage of understanding skills regarding the perpetrator's actions. Thus, the perpetrator's actions are questionable to the adults, but may not be to the child.

It has been suggested that offenders typically target children with characteristics as the followings (Berliner 2002; Olson et al. 2007; Walsh and Wolak 2005) :

1. Low self-esteem or a lack of confidence – these children are easy to isolate emotionally or physically.
2. Emotionally insecure, needy or unsupported –for example children who are troubled or looking for parental substitutes.
3. Naive nature – children who easily engage with strangers in online but lack of understanding of how to protect themselves from 'dangerous' situations.
4. Adolescence – Teenager children who are more curious about sex are more vulnerable to become a victim.

Child sex offenders are finding it more convenient to operate online as modern children are more attracted to the Internet and social networking. Modern children are being called the 'digital generation'. They are also known as 'Generation Virtual' or 'Gen-V'. "The Gen-V is people from various demographic age groups who make social connections online – through virtual worlds, in video games, as bloggers, in social networks or through posting and reading user generated content at e-commerce sites" (Havenstein, 2007).

Regarding the usage of social networking, IT research company Gartner Inc. found in 2007, that since the launch of Facebook in February 2004, it is reportedly 'one of the top six most-trafficked Web sites, with 50 billion page views per month' (Valdes, 2007). This figure is increasing day by day. As of June 2014 Facebook statistics mentions that it has more than 829 million active users (Facebook-statistics, 2014). A large portion of these users are adolescent children who can be easy target of online paedophiles.

In another study carried out in the United States by Pew Internet and American Life Project, 55% of the 935 respondents (US youths aged between 12 and 17 years) were found to have used online social networking sites (Lenhart, 2007).

The Cox Communications in partnership with the National Center for Missing & Exploited Children (NCMEC) conducted a survey titled Teen Online & Wireless Safety Survey 2009. This shows the trend of teenagers' usage of the Internet technology. This survey was fielded among young people aged 13 to 18 years. It is supposed that the sixty to seventy-two percent of the teenagers who have instant messengers screen name or social networking profile could be the target of online predators.

Australian children are also vulnerable to online predators due to their keen interest in Internet use. The latest data regarding the Internet access and usage by children released in 2013 by the Australian Bureau of Statistics shows that almost all (96%-97.8%) adolescent children (9-14 years of age) have the access of the Internet either at home or elsewhere. 67.1% of adolescent children use the Internet at home for online social networking, 49.9% for emailing and even in such an early age 25.2% of the children use chat rooms. Another statistics from the same authority (released in Feb 2014) shows that when the children become teenagers, at 15-17 years of age 90% of them use social networking. If proper protection is not provided then Australian children and teenagers may become potential victims of online exploitation.

Kerlikowske and Wilson (2007) explain the reason that teenagers love online chatting and social networking. In the Internet in a hidden virtual environment, one can impersonate to be a super hero though in reality one is a dull guy. In that way one may gain popularity and a lot of friends online which is very difficult for one in the real life. Perpetrators utilise this opportunity by pretending as a cool teen or child to hunt on the vulnerable children by making friends with them.

Modern sexual predators like to use the new technology; they indeed also fall in the Gen-V class. The Internet has created an ideal criminogenic environment for them. As there is no coordinated and effective regulation, it provides abundant opportunities for highly motivated offenders.

According McNulty (2007), the fear of detection in the past kept many sexual offenders restricted from associating in the physical world. The cyberspace allows a good degree of anonymity which was previously unavailable. As a result, perpetrators have flocked to utilize the online communities for example: sharing images of child sexual abuse, and discussing their barbaric behaviour.

Seeking out the child victims has become much easier now for the offenders. They do not need to visit venues in the physical world, instead from the leisure of their home or Internet cafes by visiting online chatting rooms they can find out their preys.

Ropelato (2007) mentioned that 1 in 7 youths report being solicited for sex on the Internet. The recent UK cybercrime survey also reported an estimated 850,000 cases of unwanted online sexual approaches, during 2006. Those offenses were primarily messages of a sexual nature within Internet chat rooms. During the same period 238 offences of meeting a child following sexual grooming were recorded (Fafinski, 2007).

The development of the Internet has dramatically improved the ability to gather and share information. The Internet provides a wider avenue for both adults and children to come closer to each other. It facilitates the offenders to commit conventional sex abusing crime more easily. Interested offenders can find information of potential victims easily in the Internet from social networking sites. They can also share information concerning the vulnerabilities of victims with other offenders around the globe. It is unlikely that child sexual offenders will shy away from using new technologies to facilitate the process of grooming children for sexual abuse (Choo, 2009).

The Internet has opened up new possibilities for sex-offenders to groom, and abuse children. To strengthen protection for children from this particular form of predatory sexual behaviour we need to understand the pattern and progression of grooming devised by those perpetrators.

2.1.3 The Pattern of Child Exploitation

Grooming is a subset of online child sex exploitation. According to O'Connell (2003), grooming may or may not involve explicit conversations of a sexual nature; still grooming falls under the umbrella of cyber sexploitation, because the intention is to sexually abuse a child in the real world and as one of the points of contact occurs in cyberspace.

Regarding the definition of grooming, the anti-grooming legislation in UK, presented to Parliament by the Secretary of State for the Home Department in the November 2002 "Protecting the Public" White Paper refers to the following:

> "A course of conduct enacted by a suspected paedophile, which would give a reasonable person cause for concern that any meeting with a child arising from the conduct would be for unlawful purposes". (Cited in O'Connell 2003)

Howitt (1995) defined 'grooming' as: "the steps taken by paedophiles to 'entrap' their victims and is in some ways analogous to adult courtship" (Cited in Craven, Brown, and Gilchrist, 2006).

Schell, Martin, Hung and Rueda (2007) explain the term of child grooming as:

> "Child grooming is a term describing how a child sex abuser uses various techniques, including showing porn to children, to lower their defences and to get them to accept the sexual acts as 'normal' rather than 'abnormal' or 'abuse'."

Different researchers identified different number of phases or stages in the psychological behavioural communicative pattern of child exploitation. According to

Armagh and Battaglia (2006) the behaviour of offenders involved in online child exploitation cases usually develops in four stages (Cited in Choo, 2009):

1. Awareness –the offender becomes aware about their sexual preference for children. He then starts researching and gathering information through various ways, including: the Internet, printed and online articles, newscasts, pornographic websites, and chatting with other like-minded individuals online.
2. Fantasy – sexual fantasizing and stimulation are achieved through the materials and information gained from the earlier awareness-exploration stage. The fantasy eventually becomes more fixated with children. The offender then try to obtain child exploitation materials.
3. Stalking – the offender is escalated to grooming stage by loitering different physical and online venues where children are available.
4. Molestation – in this stage a meeting is setup with the intention of sexual contact with the child victim.

Rachel O'Connell (2003) identified the following six stages in cyber child exploitation:

1. Friendship-forming stage:
   In this stage the paedophile tries to know the child. He may ask for personal information and even a photograph for the identification of the child. He may do it to make sure that the child is in fact a child and matches his particular predilections. In this stage the adult may suggest moving from the public sphere of the chat room into a private chat room in which rather than the one-to-many facility of a public arena, an exclusive one-to-one conversation can be conducted.

2. Relationship-forming stage:
   This is an extension of friendship-forming stage. During this stage the adult tries to create an illusion to be the child's best friend. He may engage with the child in a prolonged discussion of the child's home, school, likes and dislikes.

3. Risk assessment stage:
   In this stage the perpetrator tries to assess the likelihood to be detected by the child's parents, guardians or others. He may ask about the location of the computer, if the child is alone, where are the parents.

4. Exclusivity stage:

   This stage typically follows the risk assessment stage. In this stage the adult psychologically separate the child from others and forms a secret relationship based on illusive mutual trust.

5. Sexual stage:

   Sexuality is introduced innocuously in the conversations in this stage, for example, by asking 'have you ever been kissed?' type questions. Patterns and progression varies in this stage according to the intention of the perpetrator. If he wants to keep the orchestrated frame of loving, caring friendship then the entrance in the sexual stage will be very gentle. Certainly the adult trains the child to come out of children's boundaries in the false sense of 'grown up'. He makes the child sexually ready to abuse in the next stage of fantasy enactment.

6. Fantasy enactment stage:

   In this stage the adult engages a child in enactment of sexual fantasy. This stage has much similarity with adult to adult cybersex related interactions. The adult may fluctuate between inviting and emotionally blackmailing a child into engaging in cybersex, which may involve descriptions of anything from mutual masturbation, oral sex or virtual penetrative sex. The ultimate goal of fantasy enactment is the achievement of sexual gratification.

Regarding the above mentioned stages O'Connell (2003) clarifies that not all users will progress through the stages in the conversations sequentially, that is the order and number of stages will vary person to person. Some adults will remain in one stage for longer periods than other adults and some will skip one or more stages entirely.

Olson, Daggs, Ellevold, and Rogers (2007) explain the grooming in their model of luring communication theory (LCT). The authors define five phases in their LCT model: 1. gaining access, 2. deceptive trust development, 3. grooming, 4. isolation, and 5. approach. From Figure 2.1 it can be understood that the development of trust is the core component of the grooming process. Sexual encounters in the physical world are also dependent on the offender's ability to cultivate trust. In the grooming process, the perpetrator develops a deceptive trust with the child victim, isolates him or her from others and gradually drags the child into the abusing process in the physical world.

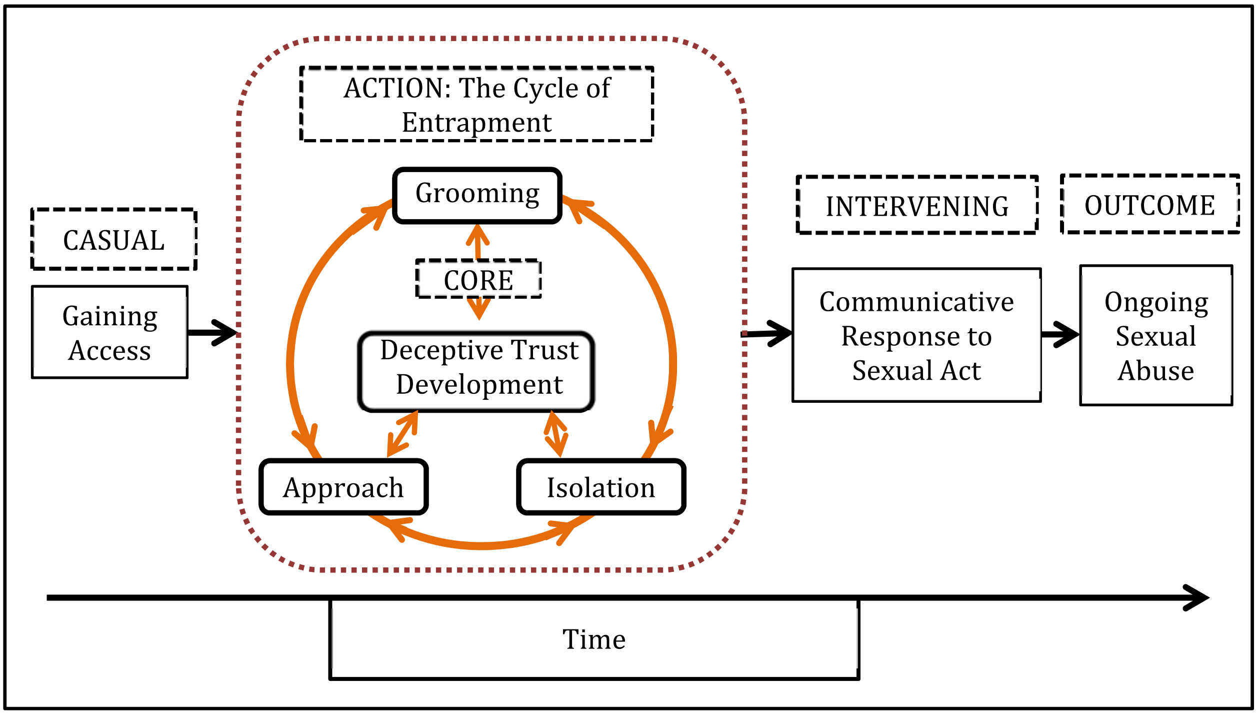

ACTION: The Cycle of Entrapment
Grooming
CORE
Deceptive Trust Development
Approach
Isolation
CASUAL
Gaining Access
INTERVENING
Communicative Response to Sexual Act
OUTCOME
Ongoing Sexual Abuse
Time

Figure 2.1: A model of luring communication theory.
(Source: Olson et al.(2007))

The LCT model of Olson et al. (2007) is expanded by Leatherman (2009) for online predation which contained nine phases (Cited in McGhee et al., 2011). Those phases include: 1. gaining access, 2. personal information, 3. Relationship, 4. activities, 5. compliments, 6. communicative desensitization, 7. reframing, 8. isolation and 9. approach. First time attempt to communicate with a child is considered as the 'gaining access' phase. It includes the greetings like 'hi', 'hello' at the beginning of the chat. The 'personal information' phase includes the exchange of information which is personal in type, for example name, age, hometown. In the 'relationship' stage the predator tries to form a relationship with the victim by discussing relationship about families and friends or even about their own mutual relationship. In the 'activity' stage discussions are found about non sexual general likes or dislikes. The 'compliment' stage contains language offering praise about appearance, activities or personalities. Vulgar language, discussion, innuendoes or vague references of sexual activities are considered as the stage of 'communicative desensitization'. 'Reframing' stage redefines the sexual behaviours in non-sexual terms. For example 'messing around', 'playing', 'learning' and 'practicing' used to refer sexual act. 'Isolation' stage includes questions about the physical location of the computer, victim's self, friend or family.

It also includes the discussion about lying to or concealing from parents or friends. Finally the predator tries to meet the victim physically in the 'approach' stage. This stage includes the arrangement of meeting, phone call, discussion of victim's location and time of meeting.

McGhee et al.(2011) viewed the Olson and Leatherman models as very complex for a chat conversation because of its short communication bursts. Some of the stages of Olson and Leatherman models do not fit within the context of chatting. Therefore McGhee et al. reduced the model and condensed into three broader phases of exploitation: 1. exchange of personal information, 2. grooming and 3. approach. McGhee et al. compared each phase of exploitation as an individual class. They considered each chat-post as one of the following three classes:

Class 200 – Exchange of personal information

Class 600 – Grooming

Class 900 – Approach

According to McGhee et al. (2011) the class 200 chat-posts exchange information of personal type. It includes questions about age, gender and location. Topics such as number of friends, previous or current boyfriends, and likes or dislikes are also discussed. The predator tries to collect as much personal information as possible about the victim. Chat-posts involving the use of sexual terminology are considered as class 600 or grooming type. The sexual innuendo may be explicit, for example asking about virginity, or using 'cum' in place of 'come', or implicit, for example "I can teach you to do that" used during a discussion about the sexual experience of the victim. The chat-posts which try to obtain the victim's phone number or address, arrange a meeting, or keep the relationship between the victim and predator a secret from parents or authorities are related to the type Approach (class 900). Apart from these three types McGhee et al. identified some chat-posts containing none of the classes. They simply keep the conversation going (e.g. yeah, lol) or appear to be truly innocent (e.g. moves in a game). These posts are considered as class 000.

Reviewing the literature, we adopted the four stage model of McGhee et al. (2011) in this current research with a slight modification. McGhee et al. considered some chat-posts as unclassified (000 class) due to innocent language. However, O'Connell (2003) defines the ‘Befriending’ stage where the predator does not use sexual provocative grooming language but uses innocent language to build up a friendship and trust of the victim. The idea of O’Connell for those posts to be ‘befriending’ seems more appropriate than the idea of those posts to be ‘innocent’. Therefore we consider those unclassified (000) kind of chat-posts as Befriending (BF) category. The modified four phases of psychological communicative pattern of CE chat used in this current research are (a) befriending (BF), (b) exchange of personal information or information exchange (IE), (c) grooming (GR), and (d) approach (AP).

Our literature review reveals that sexual abuse during childhood creates long-term problems for the victims. Many exhibit serious mental health problems as well as behaviour disorders and addictions. This occurs not only with children who experience offline sexual abuse, but also with the victims of online exploitation. If we do not act promptly to take protective measures now, there may be serious consequences for our children in the future.

2.2 Legal Aspects of Child Exploitation

2.2.1 Legislations against Child Exploitation

In recent years, along with other countries, Australia has introduced legislation to counter the online grooming or luring of children for sexual purposes. For example, on 28 November 2007, New South Wales amended its Crimes Act 1900 with the Crimes Amendment (Sexual Procurement or Grooming of Children) Bill 2007 to criminalise an adult procuring or grooming a child for unlawful sexual activity (Choo, 2009). The punishment for the child exploitation related offences in Australia ranges from a minimum of three years to a maximum of 25 years. The minimum of three years imprisonment is prescribed in the Northern Territory for an attempt to procure a child under 16 years. Most of the cases it is 10 to 12 years. Some perpetrator may receive 5 or 15 years of imprisonment. The amount of punishment depends on the severity of crime. The penalty also varies according to the age of the child victim. If the age of the child is less, then the punishment tends to be higher for a similar degree of the crime. For example, in NSW, different penalties are provided due to different age of the victim for the similar offence. Where the child is under the age of 10 years the penalty is: 25 years imprisonment; between the ages of 10 and 14 years: 15 years imprisonment; and between the ages of 14 and 16 years: 12 years imprisonment.

The legislation in the state of Victoria is somewhat similar. The Victorian Crimes Act 1958, sub division (8C) Sexual offences against children, the section 47 is to protect children from indecent act. According to this law a person who wilfully commit, or wilfully be in any way a party to the commission of, an indecent act with or in the presence of a child under the age of 16 to whom he or she is not married may receive a penalty of level 5 imprisonment which is 10 years maximum. If the sexual abuse persistently carries on then according to section 47A(4) the penalty is 25 years imprisonment. Recently some important changes have been made. A new 'Crimes Amendment (Protection of Children) Bill 2014' has been passed in the Parliament of Victoria in March 2014 (AustLII, 2014). According to the new amendment, in addition to the penalty of the perpetrator, a person in authority of the child may also be subjected to the charge of criminal offence due to his or her negligence. According to the newly inserted section 49C, failure by a person in authority to protect a child from sexual offence may receive a penalty of level-6 imprisonment which is 5 years maximum.

An important point worth mentioning here is that for building a case of online child exploitation, involvement of a real child is not mandatory. The perpetrators ill-motive to groom a child is important. For example the Queensland Criminal Code section 218B 'Grooming children under 16' states:

> "(1) Any adult who engages in any conduct in relation to a person under the age of 16 years, or a person the adult believes is under the age of 16 years, with intent to:
>
> (a) facilitate the procurement of the person to engage in a sexual act, either in Queensland or elsewhere; or
>
> (b) expose, without legitimate reason, the person to any indecent matter, either in Queensland or elsewhere;
>
> commits a crime. The maximum penalty is 5 years imprisonment."

Under such provision of law a perpetrator can be prosecuted if he tries to entice an undercover police officer or a trained volunteer who the perpetrator believes to be an under aged child.

All the jurisdictions mentioned above have more or less severe punishments for child exploitation. To convict a perpetrator with CE offence and impose the penalty the evidence has to be authentic. The following section put some light on the authenticity of chat as the evidence in lawsuit.

2.2.2Authenticity of Chat as Evidence

The Cambridge Dictionary (2014) defines evidence in law as " information that is given or objects that are shown in a court of law to help to prove if someone has committed a crime". The legal evidence is useful to establish or dismiss facts. Courts take the evidence and then evaluate whether a particular fact is proved or not. So the Court, by looking into the evidence produced (either Oral or Documentary) before it, may determine whether the facts are proved or presumed to be proved.

The chats, including the child exploitation (CE) type chats, are made up of texts typed by the participants; therefore, we consider it as digital forensic evidence in a text-document form. In Australia, the legal practices of producing documents as evidence though vary according to jurisdictions, however generally follow the Commonwealth Evidence Act and its admissibility requirements (NAA, 2014). According to the Commonwealth Evidence Act, a 'document' created and maintained in 'paper' or 'electronic' form can be admitted as evidence before federal courts. In a case of online child exploitation a chat-log can be admitted as digital forensic evidence in a 'document' form of either electronic or printed on a paper. The admissibility of a chat-log as evidence is a matter of discretion for the presiding judge and is subject to: compliance with the rules of admissibility, assessment of the quality of evidence, the interpretation, and the weight to be given to it. The current research analyses the content and context of a chat-log which can be used for appropriately interpreting and assessing the evidential quality of the chat and be placed before the presiding judge so that the judge can correctly decide the admissibility of the chat to be evidence of online child exploitation.

Some parts of this research's methodology use the Bayes' probabilistic theorem in Naïve Bayes classifiers to produce statistical evidence by analysing chat-contents. Using Bayes' theorem in law for establishing evidence is not new. In recent decades it has been an interdisciplinary study among the evidence scholars using knowledge of Law, Science, and Mathematics. The Oxford Journal 'Law, Probability and Risk' is a showcase of this multidisciplinary research (LPR.OxfordJournal, 2014). There have been a number of legal cases that involved important discussion, agreement, and disputes of probabilistic reasoning (BayesLegal, 2014). Therefore, using the Bayes' theorem, a statistical support can be provided for a chat-log to be an admissible forensic evidence of child exploitation in the court of law. The first part of the current research methodology deals with this idea.

Examples of convictions made by chat-logs increase its authenticity to be evidence in the court. Perverted-Justice.com (PJ) reports that until now (2014) there have been 587 convictions in USA made by using chat-logs as evidence, and the numbers are increasing with time. Some recent convictions include the following cases posted in the PJ website:

Robert E. Konieczko was arrested and prosecuted in Lake County, IL. He was charged with Indecent Solicitation of a Child, and Solicitation to Meet a Child. He ultimately accepted a plea agreement, and was sentenced to 24 months of probation, 120 hours of community service, registration as a sex offender and all that entails, and sex offender treatment. The offence was made in 2013. The case has a news headline in Chicago Tribune at the link http://articles.chicagotribune.com/2013-08-08/news/chi-bartlett-highland-park-soliciting-sex-charge-201308081_bartlett-man-highland-park-class-2-felony.

Christopher Richko pleaded guilty to one count of Indecent Solicitation to Commit Aggravated Criminal Sexual Abuse. He received 30 months probation, 100 hours of community service, and registration as a sex offender. The offence occurred in 2013. This also have a news headline at the link http://highlandpark.suntimes.com/crime/richko-HPN-10102013:article.

Daniel Eric Bowman was arrested and charged with two counts of Computer Child Exploitation and two counts of Obscene Contact with a Minor. He eventually accepted a plea agreement that gave him six months in the county jail, 20 years of probation, and registration as a sex offender. Based upon his chats, the Twiggs County Sheriff's Department arrested him from his home and finally he was convicted.

A report on the case of an U.S. army sergeant can be found in the following link: http://www.justice.gov/sites/default/files/psc/docs/Wunderler_42706.pdf. A chat-log was used as evidence in the United States District Court, Eastern District of Virginia. According to court documents, between July 29, 2005, and August 19, 2005, the accused engaged in numerous sexually oriented chat sessions on the Internet with a person he believed to be a fourteen year old girl. Eventually he arranged to meet the girl and on August 19, 2005, he drove to Herndon, Virginia, with the ill intention of sexual encounter with that minor girl. At the destination, he was met by a television reporter and eventually was arrested.

Brief overview of some example cases in Australia where conviction was made using chat-log as evidence are given below (cited in Choo, 2009):

A 25-year-old man of Queensland was convicted for grooming a 13-year-old girl. His primary communication was in a chat room and by sending emails. He invited her through emails to engage in sexual activity with him. Unfortunately, those emails were sent to an undercover police officer who was pretending to be the child in question. The defendant was convicted and sentenced to imprisonment for two-and-a-half years (R v Kennings (2004) QCA 162).

In another incident, an accused was sentenced to two years imprisonment for using online chat rooms to propose children to engage in sexual acts (Queensland Crime and Misconduct Commission 2006). This sentence was suspended after he served three months imprisonment, with a condition that he does not re-offend for a period of three years.

In Australian jurisdictions where no specific online child grooming legislation is available, the Commonwealth legislation can be used to prosecute offenders. An example of this is a conviction in Victorian County Court. A perpetrator was charged with the code s 474.26(1) of the Criminal Code Act 1995 (Cth): using a carriage service to transmit communications to a person under 16 years of age with the intention of procuring that person to engage in sexual activity. On 21 July 2006, the accused was sentenced to 24 months imprisonment, with an order that he be released after serving three months (Commonwealth Director of Public Prosecutions 2006).

These examples show that the chat-logs have authenticity to become evidence in the court of law to prosecute the offence of online child-exploitation.

This part of Chapter-2 has reviewed the psychological pattern and legislative issues regarding online child exploitation. The following parts will analyse the characteristics of chat message and review the text processing techniques related to the methodology used in this research to identify the CE psychological pattern and detect the CE evidence.

2.3 Analysis of Chat Messages

Generally there are two types of chats. One type is the client based peer to peer system; which is called Instant Messaging (IM) between two people. Some software provides the option to invite more people in the IM for group discussion. The other type of chat is a server based system, where different chat rooms are available. Many people from all over the world may join a chat-room to participate in the discussion. Example of widely used available chatting software are Yahoo Messenger, GoogleTalk, Skype, IRC (Internet Relay Chat), and WhatsApp. This list is only a subset; there are many more chatting systems available on the Internet. Some of the chatting system provide both the IM and chat-room facilities. A perpetrator looking for children may find a child in a chat-room, but it is more likely that he would take the child in a private IM for the clandestine heinous exploitation activity.

There is no effective central authority to monitor the chats and enforce good behaviour. In some cases (for example Yahoo chat rooms) there are chat-room moderators. However, it is not practicable to have human moderators for each of the thousands of chat-rooms online. It is also not practicable to have such authority as millions and billions of chats happen every day. Fortunately, almost all of the chatting software provide options to archive the texts of chat. Retrieving that archive a LEA (Law and Enforcement Agency) agent may investigate evidence of crime in it. However, chatting between two people, especially in the case of child exploitation, goes on and on for several days even months. Consequently the chat text archive becomes prohibitively long for manual processing. An automatic evidence detection system would benefit the LEA by making their task easier.

The texts in the chat possess some unique characteristics that distinguish them from other literary formal texts (Rosa and Ellen, 2009; Kucukyilmaz et al., 2008). A chat-log is constitutes by a series of posts from the users. A chat-post is a text fragment looks like a pseudo-sentence. Chat-users are supposed to type spontaneously and instantly. So the individual post is very brief, as short as a word. Frequently it is a single sentence or less. They are not grammatically correct, and this makes them more difficult to process by traditional sentence parsers. Chat-users are typing texts, but are actually trying to talk with each other through it. So the text is typed very quickly, frequently unedited, errors and abbreviations are more common. For example, "ASL" is a common chat abbreviation for Age, Sex and Location asked at the introduction stage. "P911" is a chatting code used by teenagers. It stands for "Parent Alert!" (TeenChatDecoder.com). These kinds of previously unseen abbreviations and erroneous texts are difficult to be handled by any currently available text processing techniques.

Chatting is a purely textual communication medium. So for transferring emotional feelings like happiness, sadness and anger, emoticons (emotion + icon = emoticon; a chat jargon) are widely used. These are different sequences of punctuation marks that display graphical representation of different emotional feelings. For example, ":-)" means "happy" and ":-(" represents "sad". Another way of emotion transfer is by emphasizing a word with repeating some specific characters. For example, "soryyyyyyyyyyyy". This kind of deliberate misspelling is also frequent in chat text. The emoticons and intentional misspelled words may contain valuable contextual information in a chat text. For example, in the grooming phase the perpetrator may reconstruct relation by an emphasized "soryyyyyyyyy" when the child felt threatening by any obtrusive language. Another example may be the emoticon for "hug (>:d<)"

and “kiss (:-*)” for a soft introduction of sexual stage. However, preserving such information makes traditional text processing methods (e.g., stemming and part of speech tagging) unsuitable for processing chat text (Kucukyilmaz et al. 2008).

The concern of the current research is child exploiting chats. This kind of chat involves two people. The perpetrator types the text to entice a child. Sexually explicit language, though not found in the beginning, may be introduced gradually in the text as the conversation progresses. Matching those words may show some preliminary detection of exploitation, yet this raises some confusions. If the perpetrator is an experienced groomer he may cleverly avoid sexually exploiting words. Instead he may use gentle and soft pressure on the child’s sexual boundaries as described in the previous section of psychological literature review. On the other hand a chat-log between two adults, who have sexual relationship, may also have sexually explicit languages in their intimate private chat sessions. Therefore matching only sexually explicit words does not solve the problem. A robust analysis of the entire chat text is required that may detect the particular child exploiting behavioural stages in the chat-log.

With all the above mentioned textual characteristics chat-text has some forensic characteristics as well. The chat-text is somewhat semi-structured. It contains the usernames, date and time stamps. Though the real identity of the perpetrator is hidden under the virtual identity, still the username would be one of the evidence if a connection to the accused is found. A forensic expert may look into the confiscated hard disk drive (HDD) of the accused for that particular username to find out the connection.

2.4 Text Classification Techniques for Chat-text

The detection of child exploitation in a chat requires robust analysis of the text in it. A text classifier (TC) can be used in the beginning stage for a probabilistic statistical analysis of whole chat-log to find shallow circumstantial evidence as to whether the chat-log is of a suspected CE type or not.

In the recent years, the chat-text analysis and chat-mining field have drawn a good attention of the text and language research community. To solve different problems in those areas different techniques evolved over the time. However all those techniques are not perfect in all situations. Our literature review suggests that the performance of most of the existing techniques are highly related to specific contexts. The context of our current research is particularly unique, therefore the existing techniques may have serious drawbacks for the current problem.

Research focusing on applying Text Classification (TC) techniques to the specific context of the current research problems is difficult to be found. Discussion on different applications of TC would provide an idea of its usefulness and help us to learn a way of adapting it in our research approach. Therefore we start with a brief overview of the TC related literatures in this section. Research related on specifically CE detection will be discussed in section 2.7 of this current chapter.

2.4.1 Background

Text classification (TC) techniques have been used for many years for different text and document processing tasks. Those tasks include: document indexing , document filtering, population of hierarchical catalogues of Web resources, automated metadata generation, word sense disambiguation, and in general any application requiring document organization or selective and adaptive document dispatching (Sebastiani, 2002). Contemporary implementation of TC includes fishing or spam mail detection (Gansterer and Pölz, 2009; Jezek and Hynek, 2007), authorship analysis (Zheng et al., 2006; Chaski 2005; Diederich et al., 2003; Tsuboi and Matsumoto, 2002), opinion detection (Osman, Yearwood, and Vamplew, 2010, 2009; Osman and Yearwood, 2007), blog opinion detection using sentiment lexicon (Zhang, Zhou and Wu, 2009), and emotion and expectation detection (Ivkovic and Ma, 2009; Osherenko, 2008).

Besides these applications in formal literary texts, in recent years TC has also been applied into the informal texts like chats and tweets[1]. Chat-post categorization (Rosa and Ellen, 2009), topic detection (Rosa and Ellen, 2009; Adams and Martell, 2008; Bengel et al. 2004; Wu et al. 2005), authorship prediction (Kucukyilmaz et al. 2008), discourse analysis (Forsyth and Martell 2007), and sentiment discovery (Bifet and Frank 2010) are some of such recent works. The foci of these works are different from the focus of our current research. Therefore it is very unlikely that any of these directions would be directly applied to solve the problem of the current research.

The work of Dinakar et al. (2011) is interesting. The authors used different text classifiers for detection of cyber bullying in YouTube comments of controversial videos. The authors manually divided the YouTube comments into the categories of bullying on 'sex', 'race' and 'intelligence'. 1500 instances were annotated for each group. 627, 841 and 809 instances were found to be positive for bullying on sexuality, race and culture, and intelligence respectively. The benign comments are categorised as 'neutral'. Varieties of features were used including: TFiDF (term frequency inverse document frequency), Part-of-speech tags, Ortony lexicon (Ortony, 1987) for negative effect, list of profane words, and topic specific unigrams and bigrams. The authors used Naïve Bayes (NB), Repeated Incremental Pruning to Produce Error Reduction (RIPPER), Decision Tree (DT), and Support Vector Machine (SVM) in a binary- and a multi- class task. The results of the experiments show that building label-specific binary classifiers are more effective than multiclass classifiers at detecting such sensitive bullying messages. In terms of accuracy, RIPPER was the best (80.2%), although the kappa values (0.598) was less compared to SVM (0.79). SVM's high kappa value suggests better reliability. Naïve Bayes classifiers (72%) perform better than DT (70%) in accuracy and in kappa in some cases. This work does not focus on child exploitation in chats however it focuses on a problem which we may refer as a "sister-problem" as detection of cyber bullying is concerned with the protection of children from being online victims. This work does not solve the problem of our current research however it uses classifiers on short text messages with special

[1] The tweets are the text-fragments in the social networking site Twitter. It looks like somewhat in between blogs and chats. The posts are very short in length, similar to chat-posts.

feature sets. The authors used a special cognitive feature ‘the negative affect of terms’ in TC for the detection of cyberbullying; we have used another special psychometric feature set in TC for classification of chat-logs as a preliminary process of detection of child exploitation. Our work and results have been reported in Miah, Yearwood and Kulkarni (2011) and included in Chapter 6 of this thesis.

Adams and Martell (2008) worked on topic detection and topic thread extraction in chat-logs. Each chat-post or line is treated as a document. The main approach was the typical term frequency-inverse document frequency (TFiDF)-based vector space model (VSM) in combination with other techniques. The authors used chat texts from Internet public chat rooms. The best-performing detectors, with an F-score of 0.6667, were the ones that employed time-distance penalization together with TFiDF. However, the authors conceded that more evaluation is needed across a more diverse data set to determine the consistency of this result. This article is interesting, as it addresses an issue that has some similarity with our work in the sense of linking together the chat-posts (chat-threads) with the same topic. However, the definition of ‘topic’ differs from our case. The authors define and confine the topics only on the content of the chat-posts, and attention to the topic of ‘child exploitation’ is not present.

Problem of sentiment analysis of Twitter and MySpace comments has some similarity with the problem of CE detection in a sense that they require to find out ‘aboutness’ of the concerned text by analysing and classifying a small chunk of ungrammatical text. The twitter analysis looks for polarities (positive, negative or neutral) and strength of sentimental feeling of the user; on the other hand the CE detection problem tries to find out whether the concerned chat-post is a threat to a child. Thelwall et al. (2010) developed SentiStrength algorithm for detection of sentiment strength in short text of MySpace comments. Based on the terms in LIWC (Pennebaker et al., 2003), Thelwall et al. handpicked a list of terms representing sentiment strength. The list contains 298 positive terms and 465 negative terms. Initially the strength of each term was assigned manually and later optimised using machine learning algorithms. Using the list as a look-up table SentiStrength analyse short messages and assign each message a score of positive and negative sentiment on a scale of 1 to 5; 1 being no sentiment and 5 being the highest positive or negative sentiment. In the experiments the authors used a collection of MySpace comments labelled with sentiment strength by human coders. The number of MySpace comments in the development-set was 2,600 and the test-set was 1,041. The authors compared the results of SentiStrength with the results of some of the classifiers in WEKA (Hall et al., 2009). To predict the positive sentiment, the SentiStrength achieved a 60.6% accuracy which is better than the accuracy 58.5% achieved by the set of WEKA-classifiers used by the authors. For negative sentiment detection, although the SentiStrength achieved a 72.8% accuracy, however it is less than the accuracy of 73.5% achieved by the set of WEKA-classifiers. SentiStrength has also been used by Thelwall et al. (2011) to analyse the possible relation between events and changes in the intensity of sentiments expressed in Twitter events. The main objective was to assess whether popular events are typically associated with increases in sentiment strength. Experiments performed by the authors used a collection of 34,770,790 tweets in English, downloaded over 29 days, with the selection of the 30 most important events that occurred during those days. In the result the authors claim that some popular events are normally associated with increases in negative sentiment strength in the comments in Twitter. However, some popular events have small average change in sentiment associated with it. Therefore, it does not seem likely that important issues can be identified by the intensity of sentiment, but rather may be identified by the volume of tweets posted on them. The focus of Thelwall et al. is different than the focus of our research. To the best of our knowledge we did not find any literature that associates sentiment-strength in identification of online child exploitation. Therefore we are not convinced that SentiStrength might be useful in addressing the CE detection problem. However the authors' use of LIWC list of terms in SentiStrength is interesting. LIWC provides psychological information of individual terms, and our literature review in section 2.1 suggests that CE follows psychological pattern; therefore we are convinced to use LIWC in our experiments. A detailed description about our usage of LIWC is provided later in this chapter. Thelwall et al. also used WEKA-classifiers in their experiments, we also used some of those classifiers. We will discuss those in our experiment chapter.

Lee et al. (2011) categorise Twitter-trends into a number of general topics such as sports, politics and technology by using text-based classification and network-based classification. The authors used a curated database of randomly selected 768 Twitter-trends over 18 classes of general topics. In text-based classification the authors used 'bag of words' approach. A trend definition together with all the tweets of that trend works as a document. Naïve Bayes, Naïve Bayes Multinomial, and SVM classification techniques are used. Among them Naive Bayes Multinomial classifier performed best with a 65% accuracy. In network-based classification method the authors used the number of common influential (important) users to identify the topic of a Twitter-trend. If the set of influential users of a trend t_a highly overlap to those of another trend t_b, then the topics of t_a and t_b are similar. To classify a given Twitter-trend, the number of common influential users between the given trend and its similar topics are used in different classification algorithms such as C5.0 decision tree learner, K-Nearest Neighbor, SVM, and Logistic Regression. Among those classifiers C5.0 decision tree learner outperforms others with a 70% accuracy. The child exploitation (CE) detection problem is different than the problem of trend classification or sentiment analysis in Twitter. The platform is also different. Twitter is a social networking platform, millions of users can be found; whereas in online child exploitation a perpetrator secludes a child victim into a private one-to-one chat session, therefore a network-based classification method is not helpful. However the text-based classification techniques seems a good starting point. We have used different text classifiers in the beginning part of our methodology which will be discussed later.

The above mentioned works do not focus on solving the problem of this research but suggest that the text classifiers are effective data mining tool and would be useful in the current research. From the previous section we have already known that chat-texts are different than formal texts, therefore the formal classification problem requires a reformulation to handle the chat-texts. This is explained in the following section.

2.4.2 Formulation of the Problem of Chat Classification

To understand the problem of detecting the indication of child exploitation (CE) from chat texts one needs to look at chats from the CE point of view. In this view, chats can be defined into the following three categories:

1. CE chat: These are Child Exploiting (CE) chats. An adult perpetrator is involved in this type of chat with a minor. The purpose of the perpetrator is to solicit the child and achieve sexual gratification. The exploitation may occur either online or a physical meeting is arranged for further abuse.

2. Near to CE chat: These chats are Sex Fantasy (SF) chats between two adults. Sexual gratification is one of the common motives in both the CE and the SF types of chats. Similar sexually explicit terms are present in both of them. They may also have similar progression style. As no minor child is involved, these chats are not CE. However both types have some similarity, so we consider SF chats as near to CE type.

3. Far from CE chat: Other general (GN) type of chats which do not have any similarity with CE type chats and easy to distinguish from them. For example chat between a client and an expert to solve a technical problem.

After defining the categories of chats from the CE point of view, the problem of predicting the type of a chat is similar to the text classification problem with careful consideration of the unique characteristics of chat. Adapting the definition of formal text categorization provided by Sebastiani (2002) and Manning et al.(2009) , chat categorization would be defined as the task of mapping the target chat-documents to the predefined classes through a classification function γ as below:

$$\gamma : \mathbb{D} \rightarrow \mathbb{C} \qquad \cdots \text{ Equation 2.1}$$

Where:

$\mathbb{D}$ is the chat-document space; the description of a chat-document is given as $d \in \mathbb{D}$.

$\mathbb{C}$ is the predefined set of classes $\mathbb{C} = \{c_1, c_2, \dots, c_m\}$. In a binary classification the class types ($\mathbb{C}$) includes CE type chats and Non-CE type chats. In the case of multi-class classification the suspected CE chats are one of the predefined multiple types.

The classification function γ learns through a supervised machine learning method using a training data set $\widehat{\mathbb{D}}$ of labelled chat-documents $\langle d, c \rangle$, where $\langle d, c \rangle \in \widehat{\mathbb{D}} \times \mathbb{C}$. After learning, the classification function assigns each new document to a class as $\gamma(d_i) = c_j$.

In the experiments of current research two different types of feature sets are used in the supervised machine learning process. The first type is the traditional term-based feature set where the vocabulary of the message collection in the chat-log constitutes the feature set. Each term corresponds to a feature. For the second type of feature set a new approach has been used in this research. Psychometric information associated with each term is used to select the feature set. Each chat-log file is considered as a document. By this formulation, the problem of chat classification is reduced to a standard text classification problem.

2.4.3 Classifiers for Chat-text Categorization

Contemporary researchers use different text classifiers to categorize text documents. These text classifiers can be applied to categorize chat-texts with the formulation explained in the previous section. According to the decision boundaries, the classifiers can be categorized into 'linear' and 'non-linear' types. The following descriptions of linear and non-linear classifiers are adapted from Manning et al. (2009).

2.4.3.1 Linear Vs Nonlinear Classifiers

Linear Classifiers:

Linear classifiers separate one class of objects from the other by finding out a decision hyperplane in between the two classes. In the case of a two dimensional situation, a linear classifier defines a straight line that can separate one class of objects from the other class. This is shown in Figure 2.2.

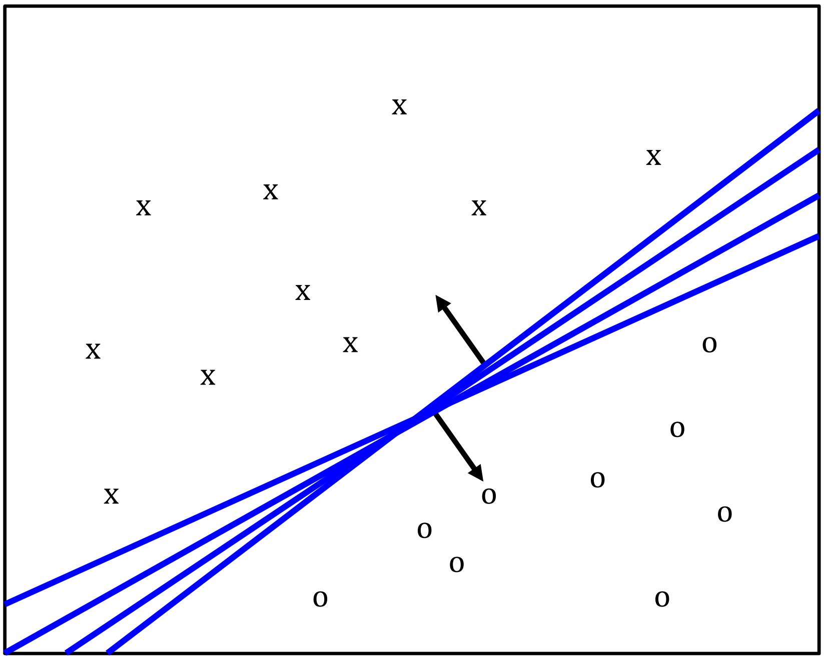


Figure 2.2: Linear classifier; There are infinite number of hyperplanes that separate two linearly separable classes.
(Source: Reproduced from Manning et al. (2009), p. 301)

Each of the lines shown in Figure 2.2 represents a linear classifier having the functional form of a straight line equation $ax + by = c$. The equation is rewritten in a more generalized form as $w_1x_1 + w_2x_2 = b$; where, the document objects are represented by the two-dimensional vector $(x_1, x_2)^T$, and the coefficients are represented by the parameter vector $(w_1, w_2)^T$ that defines (together with b) the decision boundary. The classification rule of a linear classifier is to assign a document to c if $w_1x_1 + w_2x_2 > b$ and to $\bar{c}$ if $w_1x_1 + w_2x_2 \leq b$ (Manning et al., 2009).

A number of classifiers fall into the linear category. These include (but are not limited to) Naïve Bayes (NB) classifier, Decision Tree classifier, Classification via Regression and Support Vector Machine (SVM).

Non-Linear Classifiers:

Non-linear classifiers can represent an arbitrarily complex decision boundary. An example of non-linear classification problem is shown in Figure 2.3. A linear separator

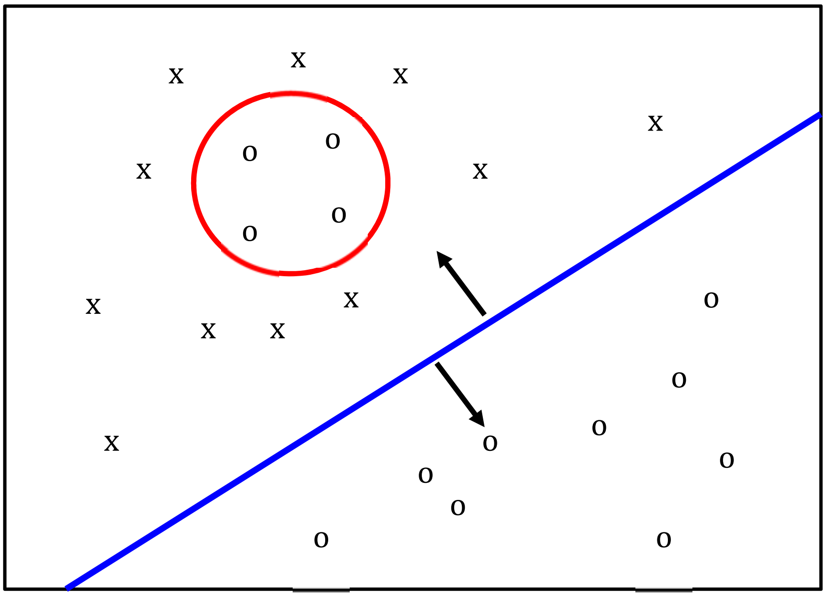


Figure 2.3: A non-linear classification problem.
(Source: Reproduced from Manning et al. (2009), p.305)

does not exists between the class distributions $P(d|c)$ and $P(d|\bar{c})$ due to the circular enclave in the left part of the graph. Therefore linear classifiers would misclassify the circular enclave. However, for this type of problem, the class distributions can be captured by using a nonlinear classifier if the training set is large enough. List of nonlinear classifiers includes kNN classifier, and nonlinear Support Vector Machine (SVM) classifier.

The Support Vector Machine (SVM) is a useful kernel based data classification technique. Basically it is a linear classifier. However, it can use linear models to implement nonlinear class boundaries (Witten et al. 2011). This is done by a 'kernel trick' that transform the input using a nonlinear mapping. In other words, the instance space is transformed into a new space. A straight line in the new space, with a nonlinear mapping, doesn't look straight in the original instance space. Thus a linear model constructed in the new space can represent a nonlinear decision boundary in the original space.

A kNN (k Nearest Neighbour) classifier determines the decision boundary locally, hence works as a non-linear classifier. It assigns each document to the majority class of its *k* closest neighbours in the vector space. The rationale of kNN classification is that, based on the contiguity hypothesis, it is expected that a test document *d* would have similar features to the training documents located in the local region surrounding d. Generally an odd number is used for the parameter k to ensure no tie exists. *k*=3 and *k*=5 are the most common choices, but test runs can be made to optimize based on training data. kNN has properties that are quite different from most other classification algorithms. Training a kNN classifier simply consists of determining '*k*' and pre-processing documents.

To categorize micro texts in military chats Rosa and Ellen (2009) found kNN providing 84.6% recall with 87.13% precision. Kucukyilmaz et al. (2008) also used kNN along with other TCs for predicting user and message attributes in chats. The user-attributes include age, name, gender and education of the user, and the message-attributes include domain and time of the day of the message. Each of the attributes has a number of classes. The authors used term- and style- based feature sets. With the term-based feature set kNN showed 25.1% to 100% accuracy for predicting different classes of chat messages. In the case of style-based feature set kNN had 12.4% to 98.3 % of accuracies. The authors have not explained what the circumstances of such a big range were.

If a problem is nonlinear and its class boundaries cannot be approximated well with linear hyper-planes, then nonlinear classifiers are often more accurate than linear classifiers. If a problem is linear, it is best to use a linear classifier. It has been empirically observed that linear classifiers with proper regularization are often sufficient for solving practical text categorization problems, with performance comparable or better than non-linear classifiers (Yang and Joachims, 2008). Furthermore, linear methods are generally computationally efficient, both at training as well as at classification. Considering these advantages the following linear classifiers are chosen to be used in this current research:

1. Naïve Bayes (NB) Classifier
2. Decision Tree Classifier
3. Classification via Regression

In the following sections we present brief descriptions of theoretical backgrounds of these classifiers. Contemporary uses of these classifiers are also provided.

2.4.4 Naïve Bayes (NB) Classifier

The Naïve Bayes (NB) Classifier is a widely used text classification method. This is because of its simplicity, speed, and effectiveness for text classification. This classification technique is based on the Bayes theorem. This technique is most effective when the dimensionality of the inputs is high that allows it to most often outperform other more sophisticated classification methods. A NB classifier is a simple probabilistic classification technique based on strong independence assumptions. The naive part refers to two assumptions that the classifier makes:

1. Positional independence: that the position of a term in a document has no bearing on its class.
2. Conditional independence: that the presence (or absence) of a particular feature of a class is independent to the presence (or absence) of any other feature. For example the probability of a term occurring in a given class and document is independent of the other terms in that same document.

These may not be completely correct, but making those assumptions makes things workable. NB is surprisingly accurate for text classification.

There are two types of NB classifier models available; binomial and multinomial. The binomial model just accounts for term's presence in a document. The multinomial model counts the number of times each term occurs in the document and uses that as a value.

2.4.4.1 Naïve Bayes Model

A Naïve Bayes model is built by training with a set of representative documents of predefined classes $\mathbb{C} = \{c_1, c_2, \ldots, c_m\}$.

The probability of a document d being in class c is computed as:

$$P(c|d) \propto P(c) \prod_{1 \leq k \leq n_d} P(t_k|c) \qquad \text{...Equation 2.2}$$

In Equation 2.2, $P(c)$ is the prior probability of any document occurring in class c. If a document's terms do not provide clear evidence for one class versus another, we choose the one that has a higher prior probability. $P(t_k|c)$ is the conditional probability of term t_k occurring in a document of class c. In the test set d represents a test chat-log whose category is to be predicted. $P(t_k|c)$ is interpreted as a measure of how much evidence t_k contributes that c is the correct class. The tokens in d are represented as $t_k \in \{t_1, t_2, \dots, t_{n_d}\}$ which are part of the vocabulary (from training set) used for classification and n_d is the number of such tokens in d.

In text classification, the goal is to find the 'best' class for the document. The best class in NB classification is the most likely or 'maximum a posteriori' (MAP) class c_{map}:

$$c_{map} = \underset{c \in \mathbb{C}}{\operatorname{argmax}} \hat{P}(c|d) = \underset{c \in \mathbb{C}}{\operatorname{argmax}} \hat{P}(c) \prod_{1 \leq k \leq n_d} \hat{P}(t_k|c) \qquad \text{...Equation 2.3}$$

$\hat{P}$ is written for P because the true values of the parameters $P(c)$ and $P(t_k|c)$ are not known, but estimated from the training set.

To estimate the parameters $\hat{P}(c)$ and $\hat{P}(t_k|c)$, Maximum Likelihood Estimate (MLE) is used. A Maximum Likelihood Estimate (MLE) would give estimation of the parameters $\hat{P}(c)$ and $\hat{P}(t_k|c)$:

$$\hat{P}(c) = \frac{N_c}{N} \qquad \text{...Equation 2.4}$$

and

$$\hat{P}(t_k|c) = \frac{T_{ct_k}}{\sum_{t_i \in V} T_{ct_i}} \qquad \text{...Equation 2.5}$$

where, N_c is the number of documents in class c and N is the total number of documents. T_{ct_k} is the number of occurrences of token t_k in training documents from class c. V is the vocabulary.

The problem with the MLE estimate is that it is zero for a term–class combination that did not occur in the training data. For example, consider a document d_x of class c_i. Document d_x contains many terms that gives very high value of $\hat{P}(t_k|c)$ and therefore should be predicted as of class c_i . However d_x also contains a term t_y for which $\hat{P}(t_y|c) = 0$ because t_y is not present in the vocabulary of class c_i in the training corpus; therefore the result of multiplication is zero, that is $\prod_{1 \le k \le n_d} \hat{P}(t_k|c) = 0$; consequently the probability for d_x to be of class c_i becomes zero, which is incorrect. To eliminate zeros, 'add-one' or 'Laplace smoothing' can be used, which simply adds one to each count:

$$\hat{P}(t_k|c) = \frac{T_{ct_k} + 1}{\sum_{t_i \epsilon V}(T_{ct_i} + 1)} = \frac{T_{ct_k} + 1}{(\sum_{t_i \epsilon V} T_{ct_i}) + B} \qquad \text{...Equation 2.6}$$

In Equation 2.6 $B = |V|$ is the number of distinct terms in the vocabulary.

Using the values of $\hat{P}(c)$ and $\hat{P}(t_k|c)$ from Equation 2.4 and Equation 2.6 in Equation 2.2 the probability $\hat{P}(c|d)$ can be estimated for a document d to be in a particular class (c). Using Equation 2.3 a classification prediction can be made for the document d.

2.4.4.2 NB Used in Chat-text classification

Rosa and Ellen (2009) used NB along with other TC for categorization of military chat texts. The recall of NB was 74.02% with 72.19% precision. In the experiment of Kucukyilmaz et al. (2008) for predicting user and message attribute NB showed a range of results for different domains. The accuracy rate of the NB classifier was from 39% to 91.8% for predicting different attributes like education, gender, identity and Internet connectivity in different domains and data sets. Forsyth and Martell (2007) also found different results with NB for different classes in their lexical and discourse analysis of online chat dialogs. For different types of discourse F-score of NB was 0.133 to 0.987. Most recently Bifet and Frank (2010) used NB classifier for sentiment analysis of Twitter posts. They found NB is giving 73.81% to 86.11% accuracies. The above mentioned research shows that the effectiveness of NB is reasonably high and it is used to handle diversified problems on text classification. The methodology of current research incorporates NB classifier in the beginning part with the hope that it will also effectively work in categorizing chat-logs into CE vs Non-CE .

2.4.5 Decision Tree Classifier

2.4.5.1 Decision Tree Model

A decision tree is made of nodes connected by arcs. Decisions are represented by leaf nodes. To reach a decision, that is to reach a leaf-node, one needs to traverse through a number of non-leaf nodes starting from the root node. Each non-leaf node of a decision tree represents an input attribute, and each arc corresponds to a possible value of that attribute. A path from the root node to a leaf node describes the input attributes which correspond to the expected value of the output attribute at the leaf node.

The Decision Tree classifier uses C4.5 algorithm (Quinlan, 1993; Wu et al., 2008). Given a set S of training data, C4.5 grows a tree using the divide-and-conquer algorithm. It uses the concept of information entropy (Shannon, 1948) for splitting a node. The training data set $S = \{s_1, s_2, \dots , s_n\}$ is a set of already classified samples. Each sample s_i consists of a p-dimensional vector $(x_{i,1}, x_{i,2}, \dots, x_{i,p})$, where the $x_{i,j}$ represent attributes or features of the sample, as well as the class in which s_i falls.

The steps of the general algorithm for building a decision tree are:

1. If all the samples in the training set belong to a one class simply create a leaf node for the decision tree saying to choose that class.
2. If none of the features provide any information gain create a decision tree with a single node with the class of the most frequent output class in the training set.
3. If instance of previously-unseen class encountered create a decision node higher up the tree using the expected value (the most frequent class in the training set).
4. If a single node cannot be produced according to the first three steps' criteria; find the normalized information gain from splitting the tree for each attribute A_i. Consider A_{imax} to be the attribute with the highest 'normalized information

gain'. Create a decision node that splits on A_{imax}. This also splits the training set S into sublists $s_1, s_2, \ldots, s_p$.

5. Recurse on the sublists obtained by splitting on A_{imax}, and add those nodes as children. Repeat this step until reaching the leaf-nodes.

The format of the splitting outcomes depend on the type of attribute used in a decision tree. The attributes can be: numeric or nominal. The splitting for a numeric attribute A can be determined by defining a threshold h and then splitting as $\{A \leq h, A > h\}$. The threshold h can be found by sorting the training data set S on the values of A and then choosing the split between successive values that maximizes the 'normalized information gain'. An attribute A with nominal discrete values has by default one outcome for each value. The values can optionally be grouped into two or more subsets with one outcome for each subset.

To avoid overfitting C4.5 uses pruning technique once a decision tree has been created. Pruning is carried out by traversing backward from the leaves to the root and removing branches that do not help by replacing them with leaf nodes or alternative branches. The pruning process is completed in one pass through the tree.

2.4.5.2 Use of decision tree in text and chat classification

Uğuz (2011) used a decision tree along with a k-nearest neighbour (kNN) on Reuters-21,578 and Classic3 datasets collection for text categorization. The experimental results achieved high categorization effectiveness. Ross et al. (2013) found decision tree classifier along with NB and SVM useful in text categorization of heart, lung, and blood studies in the database of genotypes and phenotypes (dbGap) utilizing n-grams and metadata features. It has already been mentioned in section 2.2.2.1 that Dinakar et al. (2011) used a text classifier for detection of cyberbullying by classification of youTube comments. The authors achieved an accuracy of 61% to 70% by a decision tree classifier. The problems in the above mentioned research is different than the problem of current research. However the above research shows that a decision tree classifier is useful and effective for varieties of text classification problems.

The work in McGhee et al. (2011) addresses a problem which has some similarity with our research. With some other classifiers the authors also used a decision tree (DT) classifier for classifying chat-text into the psychological stages and compare the results with the results of their system Chatcoder2. A brief discussion of the complete work will be provided in the section 2.7 of this chapter. We discuss only the results of the DT classifier in this section. Using a DT classifier the authors achieved an accuracy as high as 96.99% and as low as 51.8% for individual chat-logs. However the authors admitted that the results are misleading as the training and testing data were same. It has already been mentioned that chat-posts are usually short text fragments with a very few terms. A single term post is also very common. A particular chat-log may have a unique set of terms for each of the psychological stages. In that case a decision tree will be only one or two steps from the root with a number of leaves each containing those unique terms. As it is overfitting, this tree will work very well on that particular chat-log but will completely fail to work on other data. We suspect this happened in the work of McGhee et al. in the case of high effectiveness with decision tree (DT) classifier. In our preliminary investigation we found that text classifiers including DT behave unreliably on the short text level of chat-post. Using TC on the whole text of a chat-log may not suffer from this problem and would effectively differentiate between CE and Non-CE chats-logs. For differentiating and identifying the psychological stages, new methodologies are to be used which will be discussed progressively throughout this thesis.

2.4.6 Classification via Regression

The regression is a statistical technique for finding a mathematical expression to describe a set of data. This is a classical technique which existed before the invention of computers. Because of simplicity, stability and effectiveness even today regression techniques are being widely used and finding new scopes. It often provides a comprehensible interpretation of how the output changes due to the changes in the inputs. In situations with small numbers of training cases, and sparse data sometimes its prediction is better than nonlinear models. In this section first we briefly describe the mathematical model for linear regression and then explain how the model can be used in text classification task. The theory and equations of the regression technique in this section are revised from Hastie et al. (2009) and Witten et al. (2011).

2.4.6.1 Linear Regression Model

Let us consider an input vector $X = (X_1, X_2, \dots, X_P)$ which produces a real valued output Y. The following linear equation can be used to model a linear function $f(X)$ that can predict output Y using input X:

$$\begin{aligned} Y = f(X) &= \beta_0 + X_1\beta_1 + X_2\beta_2 + \cdots + X_P\beta_P \\ &= \beta_0 + \sum_{j=1}^{p} X_j\beta_j \end{aligned} \qquad \text{... Equation 2.7}$$

Where:

$Y = f(X)$; the output variable which depend on the value of input X.

$X = (X_1, X_2, \dots, X_P)$; the input vector, independent of the value of output Y.

P = Number of elements in input vector X. If X is considered as an input instance (sample) then P denotes the number of features or attributes of that instance.

β_j = parameter or coefficients of X_j .

The Equation 2.7 is the basic form of the linear regression model. In this equation the term β_0 is the intercept, also known as the bias in machine learning. This resembles the equation of a straight line *y = mx + C* shown in Figure 2.4. In the figure the scattered dots are individual samples and the straight line is the estimated linear regression line.

If we include the constant variable $x_0 \equiv 1$ in X, then β_0 can be included in the vector of coefficients β_j. With this rearrangement the linear model can be rewritten in vector form as an inner product as follows:

$$Y = X\beta \qquad \text{... Equation 2.8}$$

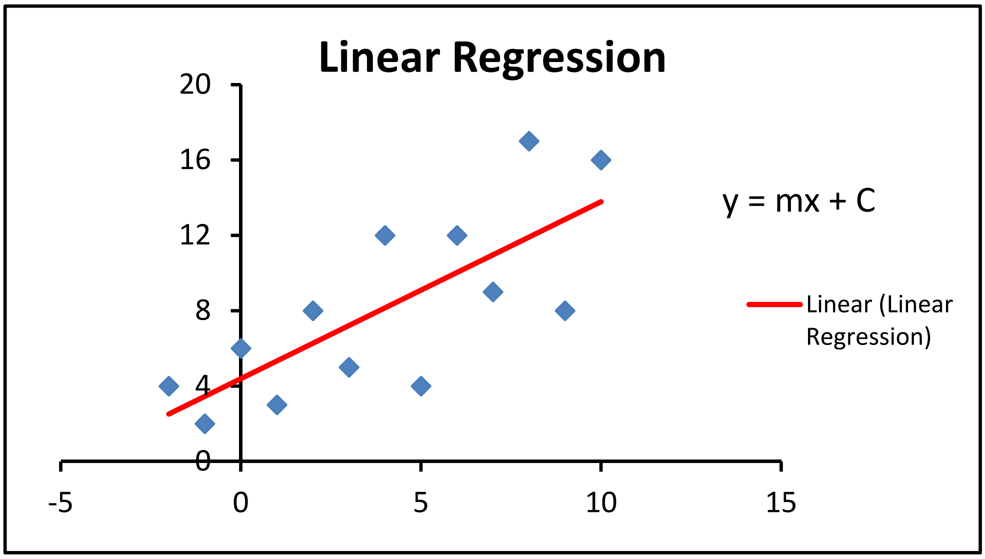


Figure 2.4: Linear regression as a straight line.

The difference between the predicted output $f(X)$ and the actual output Y will be minimum if the expected loss function $E\left(L(Y, f(X))\right)$ is minimized. The loss function is given by square loss:

$$L(Y, f(X)) = \left(Y - f(X)\right)^2 \qquad \text{... Equation 2.9}$$

The optimal predictor for output[1] $\hat{Y}$ is given by:

$$\hat{Y} = E(Y|X) = \hat{f}(X) = \underset{f(\mathrm{X})}{\operatorname{argmin}}\left(\left(Y - f(X)\right)^2\right) \qquad \text{... Equation 2.10}$$

Now the issue of finding the regression function $f(X)$ is converted to estimating optimal $\beta_j\,; j = 0, 1, ..., P$. The parameters β can be estimated from a training set containing N number of data $(x_1, y_1) \dots (x_N, y_N)$. Each $x_i = (x_{i1}, x_{i2}, \dots, x_{iP})$ is an input vector of feature measurements for the i-th case. For estimation of the coefficients (β) the least squares method can be used. In this method the residual sum of square is minimized.

[1] Y-Hat is used instead of Y because it is not actual but predicted or estimated.

The residual sum of squares of β is given by:

$$\begin{aligned} RSS(\beta) &= \sum_{i=1}^{N} \left(y_i - f(x_i)\right)^2 \\ &= \sum_{i=1}^{N} \left(y_i - \beta_0 - \sum_{j=1}^{p} x_{ij}\beta_j \right)^2 \end{aligned} \qquad \text{... Equation 2.11}$$

By minimizing the $RSS(\beta)$ in Equation 2.11 we can find the best linear fit to the data. We consider $\boldsymbol{X}$ to be the input matrix of dimension $N \times (P + 1)$ with each row as an input vector (with $x_0 \equiv 1$ in the first position), β be a $(P + 1)$-vector of coefficients of $\mathbf{X}$ and similarly $\mathbf{y}$ be the N-vector of outputs in the training set as shown below:

$$\begin{bmatrix} 1 & x_{1,1} & x_{1,2} & \cdots & x_{1,P} \\ 1 & x_{2,1} & x_{2,2} & \cdots & x_{2,P} \\ \cdots & \cdots & \cdots & \cdots & \cdots \\ 1 & x_{N,1} & x_{N,2} & \cdots & x_{N,P} \end{bmatrix} \begin{bmatrix} \beta_0 \\ \beta_1 \\ \cdots \\ \beta_P \end{bmatrix} = \begin{bmatrix} y_1 \\ y_2 \\ \cdots \\ y_N \end{bmatrix}$$

That is: $\mathbf{X}\beta = \mathbf{y}$

Then we can write the residual sum-of-squares as:

$$RSS(\beta) = (\mathbf{y} - \mathbf{X}\beta)^T(\mathbf{y} - \mathbf{X}\beta) \qquad \text{... Equation 2.12}$$

This gives:

$$\hat{\beta} = (\mathbf{X}^T\mathbf{X})^{-1}\mathbf{X}^T\mathbf{y} \qquad \text{... Equation 2.13}$$

$\hat{\beta}$ in Equation 2.13 gives the estimated value of the β .

The predicted values at an input vector x_i is given by:

$$\hat{f}(x_i) = x_i\hat{\beta}; \qquad \text{... Equation 2.14}$$

Where:
$x_i = (x_{i,0}, x_{i,1}, \cdots, x_{iP})$ and
$\hat{\beta} = (\hat{\beta}_0, \hat{\beta}_1, \cdots, \hat{\beta}_P)$

The fitted values with the training inputs are given by:

$$\hat{\mathbf{y}} = \hat{\beta}_0 + \sum_{j=1}^{p} X_j\hat{\beta}_j \qquad \text{... Equation 2.15}$$

Or,

$$\hat{\mathbf{y}} = \mathbf{X}\hat{\beta} = \mathbf{X}(\mathbf{X}^T\mathbf{X})^{-1}\mathbf{X}^T\mathbf{y} \qquad \text{... Equation 2.16}$$

where $\hat{y}_i = \hat{f}(x_i)$ is the estimated output for i-th instance.

2.4.6.2 Classification Task using Linear Regression

Regression method predicts a continuous value in the output, on the other hand the output class labels in a classification are discrete values. To use linear regression for classification, the outputs are coded with numeric values of 1's and 0's for each class; 1's are for training instances of the positive class (instances belong to the class) and 0's are for training instances of the negative class (instances do not belong to the class). Using the regression method, linear expressions are modeled for each of the available classes. For determining the class of an unknown test sample, the output value of each linear expression is calculated and the class is selected which has the largest value. For example: in a classification task there are K number of classes in a set of categories $C = \{c_1, c_2, \ldots, c_K\}$. The training input $X = \{x_1, x_2, \ldots, x_N\}$ has N number of instances. Each of the inputs x_n belong to one of the classes c_k. For an unknown input x the task is to find the class $G(x) \in C$. To apply linear regression method in this classification problem, the outputs are coded with indicators 0's and 1's and collected in a vector $Y = (Y_1, Y_2, \ldots, Y_k)$ corresponding to classes $\{c_1, c_2, \ldots, c_K\}$. If the class of the corresponding input matches in Y, that is, if $G(x_n) = c_k$ then $Y_k = 1$; otherwise $Y_k = 0$. N number of such vectors make an $N \times K$ indicator response matrix **Y.** This is shown in Figure 2.5.

$$\begin{bmatrix} 1 & x_{1,1} & x_{1,2} & \cdots & x_{1,P} \\ 1 & x_{2,1} & x_{2,2} & \cdots & x_{2,P} \\ \cdots & \cdots & \cdots & \cdots & \cdots \\ 1 & x_{N,1} & x_{N,2} & \cdots & x_{N,P} \end{bmatrix} \begin{bmatrix} \beta_{0,1} & \beta_{0,2} & \cdots & \beta_{0,k} \\ \beta_{1,1} & \beta_{1,2} & \cdots & \beta_{1,k} \\ & & \cdots & \\ \beta_{P,1} & \beta_{P,2} & \cdots & \beta_{P,k} \end{bmatrix} = \begin{bmatrix} \boxed{y_{1,1}} & y_{1,2} & \cdots & y_{1,k} \\ y_{2,1} & y_{2,2} & \cdots & y_{2,k} \\ & & \cdots & \\ y_{N,1} & y_{N,2} & \cdots & y_{N,k} \end{bmatrix}$$

Figure 2.5 : Input training instance **X**, coefficient $\boldsymbol{\beta}$, and output indicator matrix **Y**

Each row of the indicator matrix **Y** would have a single 1 because each input instance falls into only one class. We fit a linear regression model to each of the instances, and the fit is given by Equation 2.16 re-written as:

$$\begin{aligned}\hat{\mathbf{Y}} &= \mathbf{X}(\mathbf{X}^T\mathbf{X})^{-1}\mathbf{X}^T\mathbf{Y} \\ &= \mathbf{X}\hat{\mathbf{B}} \qquad \text{... Equation 2.17}\end{aligned}$$

Here, $\hat{\mathbf{B}} = (\mathbf{X}^T\mathbf{X})^{-1}\mathbf{X}^T\mathbf{Y}$ is a coefficient matrix of dimension $(P+1) \times K$ and we have a coefficient vector $\beta_{pk} = (\beta_{0k}, \beta_{1k}\ \beta_{2k}, \dots, \beta_{Pk})$ for each response column of **Y**. The training instance matrix **X** has $(P+1)$ columns corresponding to the P number of features in each input instance, and a leading column of 1's for the intercept.

A new test input x can be classified by the following steps:

1. Compute the fitted output $\hat{Y}_k = \hat{f}_k(x) = (1, x)\beta_{pk}$; $\hat{Y}_k$ is a vector with K components. In the input vector $(1, x)$, the 1 is for the leading column of 1's for the intercepts. For a single instance the input vector $(1, x)$, the coefficients β , and output y are encircled in the Figure 2.5 . To get a one of the K components of the indicator vector Y_k one column vector of coefficient matrix is used. To get all of the components (y_{nk}) of Y_k , all of the column vectors β_{pk} of the coefficient matrix **B** should be used.
2. Identify the largest component among $\hat{f}_k(x), k = 1, 2, \dots, K$; and classify accordingly:

$$\hat{G}(x) = \underset{c_k \in C}{\arg\max}\ \hat{f}_k(x) \qquad \text{... Equation 2.18}$$

$\hat{G}(x)$ in Equation 2.18 gives a discrete value which is one of the class labels from K number of classes in the category set C.

2.4.6.3 Use of Regression in Text Classification

The scope of the regression method is not confined to numerical statistical predictions; instead it is widened into new areas. Contemporary researchers use regression techniques for a wide range of text classification related tasks. Starting from classical text categorization the regression technique is found to be used for modern day analysis of sentiment, webpage and micro texts.

Al-Tahrawi (2014) used regression techniques to find the significance of low frequent terms in text classification (TC). A number of text classifiers are tested on the benchmark Reuters Data Set. One of the classifiers used the regression technique. The author compared the results of two different experiments; once with the low frequency terms and another without them. The results show superior performance of TC when the low frequent terms are used in classification.

The regression technique also finds its way in webpage analysis. Wawer et al. (2014) used regression for predicting webpage trustworthiness using linguistic features. The authors analysed the available features with regression models and did an optional feature selection according to a percentile of the highest F- scoring features. They found that the best performing regression models used only top 20th percentile of features.

Taddy (2013) used regression in his multinomial inverse regression model of text classification for sentiment detection. The author shows that logistic regression of phrase counts onto document annotations can be used to obtain low dimension document representations that are rich in sentiment information.

Baccianella et al.(2013) used the regression method incorporated into the support vector machine (SVM) family for ordinal text classification. The authors used ordinal regression and vector regression with SVM to accomplish the task. For feature selection the authors logically break down each training document of length *k* into *k* training 'micro-documents', each consisting of a single word occurrence and endowed with the same class information of the original training document. The results of the experiments show that the use of this strategy substantially improves the accuracy of ordinal text classification.

The goals of the above research are different than our goal. However, they show that the regression can be used effectively in text classification task.

2.4.7 Psycholinguistic Features for Chat-Text Classifiers

Along with simple term-based features in our experiments we also used psycholinguistic features for the classification task. The psycholinguistic information is obtained from the LIWC (Linguistic Inquiry and Word Count) system. The LIWC system counts the number of structural and psychologically significant words in the text. It is a text analysis application designed to provide an efficient and effective method for studying the various emotional, cognitive, and structural components present in the individuals verbal and written speech samples (Pennebaker et al. 2007). LIWC accepts texts or groups of texts and produces a feature vector consisting of 80 output variables that represent the document(s). The variables include four general descriptors (total word count, average number of words per sentence, the percentage of long (greater than six letters) words and the percentage of the words in the document captured), 22 standard linguistic dimensions (for example: the percentage of words that are pronouns, articles and auxiliary verbs) 32 word categories tapping psychological processes (for example: affect, cognition, biological process,), 7 personal concert categories (for example: work, home, leisure activities), 3 paralinguistic dimensions (assents, fillers, non-fluencies) and 12 punctuation categories (for example: periods, commas). The basis of the LIWC application are dictionaries consisting of almost 4,500 words and word stems. It also recognises and matches word stems. Each word or stem can be defined in several categories (for example the word 'cried' is in 5 categories: sadness, negative emotion, effect and past tense of a common verb). The standard dictionaries supplied with the LIWC application have been shown to capture 86% of words used in everyday speech and writing (Pennebaker et al., 2007). We will see in Chapter-6 that the psycholinguistic feature set improves the effectiveness of text classifiers for the classification of chats.

From the literature review on the Text Classifiers we can understand that the TCs are effective data mining tools that have been used for a wide range of text processing tasks. Along with the traditional text classification, the applications of the TCs include informal text processing such as blogs, webpages, YouTube comments. Therefore we expect that with proper training and good feature selection, existing TC techniques can be used to classify a whole chat-log as a ‘child exploiting type vs benign type’ and classifying the participants into ‘child victim vs adult perpetrator’. These classifications work as shallow statistical evidence. However, classifiers would not find any strong substantial evidence. Neither of the classifiers are capable of finding the psychological stages in the exploiting chats nor would they find out the evidence of exploitation, for example, an excerpt of chat-text that shows exploitation activity. Therefore starting with classifiers our methodology incorporates clustering the chat-posts and entailment of textual hypothesis to find out the evidence. Brief overviews of these two sectors are provided in the following sections.

2.5 Text Similarity Measures and Clustering the Chat-text

2.5.1 Background

McGhee et al. (2011) suggest that identifying and capturing context and incorporating a window of text may improve the analysis and CE detection process. The section 2.1.3 of this chapter mentions that the progress of child exploiting chats follow a psychological contextual pattern of BF, IE, GR, or AP stages. An automatic text clustering technique, that can handle the unique characteristics of chat-text, and capture those CE psychological contexts, may divide the whole chat-log into the blocks of chat-texts representing those psychological stages. These blocks of texts would be more useful in the analysis of chat contents and CE detection task than the single chat-posts. A clustering method is an unsupervised learning method. Using this method, if the posts of a chat-log can automatically be clustered into those four CE stages then it can be evidence that the chat-log is following the psychological pattern of a CE chat.

Similarities among the posts of a chat play an important role in differentiating as well as in clustering the posts into those psychological stages. Chat-posts belonging to the same stage are supposed to have higher similarities than the posts belonging to different stages. A measure that can tap this kind of similarity would assist to identify the stages in a child-exploiting (CE) chat. However, it has been mentioned before that the chat-text is conversational in nature and grammatically informal and unstructured. Each chat-post contains only a very few terms. Single-termed posts are also very frequent. Therefore, a chat-post may be considered as a pseudo-sentence. Term co-occurrence is very rare for chat-posts even in the same behavioural stage. Under these circumstances finding similarity among chat-posts is a challenge for traditional similarity measures.

To find the similarity of document-level texts, the vector space model is widely used in information retrieval because of its ability to adequately capture much of the content. Vector space measures such as cosine similarity are based on term (word) co-occurrence (Manning, Raghavan, & Schütze, 2009). They are very good at measuring the similarity between documents because many common terms are likely to co-occur in similar documents. However, while the assumption that similarity can be measured by term co-occurrence may be valid at the document level, the assumption does not hold for small-sized text fragments such as sentences, or informal pseudo-sentences like chat posts. Two sentences or chat-posts may be semantically or psychologically related to each other despite having few or no terms in common. In this kind of situation where term overlap is rare, latent semantic analysis may find some 'latent-similarity' among the texts by extracting the conceptual content of a body of text (Landauer, Foltz, & Laham, 1998). Therefore, instead of using the cosine similarity directly on document terms, if we transform the data space into the reduced latent semantic space and then apply the cosine similarity, the measurement of semantic similarity may improve. We computed the latent similarity, used it in clustering the chat-posts, and compared the result with our developed clustering approach.

To solve the problem of similarity measurement in small text like sentences or chat-posts, a number of 'sentence similarity' measures have recently been proposed (Li, McLean, Bandar, O'Shea, & Crockett, 2006; Mihalcea, Corley, & Strapparava, 2006). Rather than representing sentences in a common vector space, these measures define sentence similarity as some function of inter-sentence term-to-term similarities. Usually there are two ways to measure these similarities: a corpus-based measure or a knowledge-based measure. In a corpus-based measure the similarity is derived from distributional information from some corpora. Semantic information represented in external sources such as WordNet (Fellbaum, 1998) contributes to the similarities in a knowledge-based measure. Some of these measures are described in the next section. Inspired by these sentence similarity measures this research constructs a new technique that can tackle the current problem of finding similarity among pseudo-sentence-like chat-posts. The new similarity technique is explained in Chapter-3.

The proposed method has a subtle difference from the existing techniques. The existing sentence similarity measures look for semantic alikeness of the contents by comparing semantic definitions of the terms. The current challenge is not to find the semantic similarity of the content; instead, it requires assessing the psychological alikeness of the context. To find the psycholinguistic information we build a new dictionary focused on CE psychological contextual pattern, and based on CE chat corpora. This new dictionary can play the role of a background psycholinguistic knowledge-base to contribute the contextual psychological aspects by comparing the term contents of CE chats. Details of the dictionary-building process are also provided in Chapter-3. The next subsection provides a brief review on existing related works on sentence similarity measures.

2.5.2 Existing works on Sentence Similarity Measures

Li et al. (2006) suggested a hybrid method for measuring the content similarity of small text-like sentences. This method derives text similarity from semantic and syntactic information contained in the texts to be compared. Their method dynamically forms a joint word set using all the distinct words present only in the pairs of sentences. For each sentence, a raw semantic vector is derived with the assistance of the WordNet lexical database. Also, a word order vector is formed for each sentence, again using information from WordNet. Since each word in a sentence contributes differently to the meaning of the whole sentence, the significance of a word is weighted by using information content derived from a corpus. By combining the raw semantic vector with information content from the corpus, a semantic vector is obtained for each of the two sentences. Semantic similarity is computed based on the two semantic vectors. An order similarity is calculated using the two order vectors. Finally, the sentence similarity is derived by combining semantic similarity and order similarity.

Mihalcea et al. (2006) proposed another combined method for measuring the semantic similarity of sentences. The similarity is determined by utilizing the information drawn from the similarity of the component words. In their method they used two corpus-based measures and six knowledge-based measures of word semantic similarity. Corpus-based measures include PMI-IR (pointwise mutual information and information retrieval) and LSA (latent semantic analysis). Knowledge-based measures include Jiang and Conrath (1997), Leacock and Chodorow (1998), Lesk (1986), Lin (1998), Resnik (1995), and Wu and Palmer (1994). Mihalcea et al. (2006) combine the results to show how these measures can be used to derive a text-to-text similarity metric. They evaluate their method on a paraphrase recognition task. The main drawback of this method is that it computes the similarity of words from eight different methods, which is not computationally efficient.

These two hybrid measures (Li et al., 2006; Mihalcea et al., 2006) do not take into account the psychological importance of terms in a chat-post. To the best of our knowledge they have not yet been used to find similarity between two chat-posts. As chat-posts are not grammatical sentences, we need a measure that does not depend on grammatical sentence structures. In this context the measure proposed by Li et al. seems to partially meet the demands of the current problem. In the Li-measure each term of a sentence is considered individually and the related semantic information of that term is extracted from WordNet. Moreover, it does not use any sentence parser to parse the sentence to extract the grammatical structure. Therefore, we chose the Li measure as the basis of the new similarity measure. The current research aims to find the contextual psychological similarity of chat-posts. This requires psycholinguistic information on each term which WordNet does not provide. Therefore, we cannot use

WordNet for this purpose. The LIWC (Pennebaker et al., 2007) could be an alternative. It provides a number of psycholinguistic information. Our experiments show (Figure 3.4 and Figure 3.5 in Chapter-3) that LIWC does not convincingly improve the result, as the information from LIWC is very general in nature and not focused on CE chats. Therefore, we built a new dictionary focused on contextual psychological similarity measure for detecting the psychological stages of CE chats. We use this new dictionary in the new similarity measure; instead of using the WordNet lexical database, as used by Li et al., we used our own new dictionary. The construction of the new dictionary and the proposed similarity measure will be discussed in details in Chapter-3. Using the new similarity measure a new clusterer is designed to collect the chat posts together into the CE psychological contextual stages. The description of the new clusterer is also presented in that chapter. In the experiments we compared the results of the new clusterer with existing clusterers. In the next section we present the text clusterers used for those comparisons.

2.5.3 Text Clustering

According to mutual similarity and dissimilarity the posts of a child exploiting chat would intuitively be grouped into the documented psychological stages. As mentioned before, if the grouping can be done automatically without any human supervision then the chat-log would have evidence of following the CE pattern. To accomplish this the ‘text clustering’, which is an unsupervised machine learning approach, can be a good candidate. Figure 2.6 shows an example of the posts of a chat grouped into four clusters labelled as BF, IE, GR and AP according to the CE psychological stages. The figure is an example showing clearly visible groups only to explain the concept of clustering, practical clustering of chat-posts may vary.

Chats are not formal texts therefore require a formulation so that the existing clustering algorithms can be applied on them. The following section discusses on the formulation of the chat posts clustering.

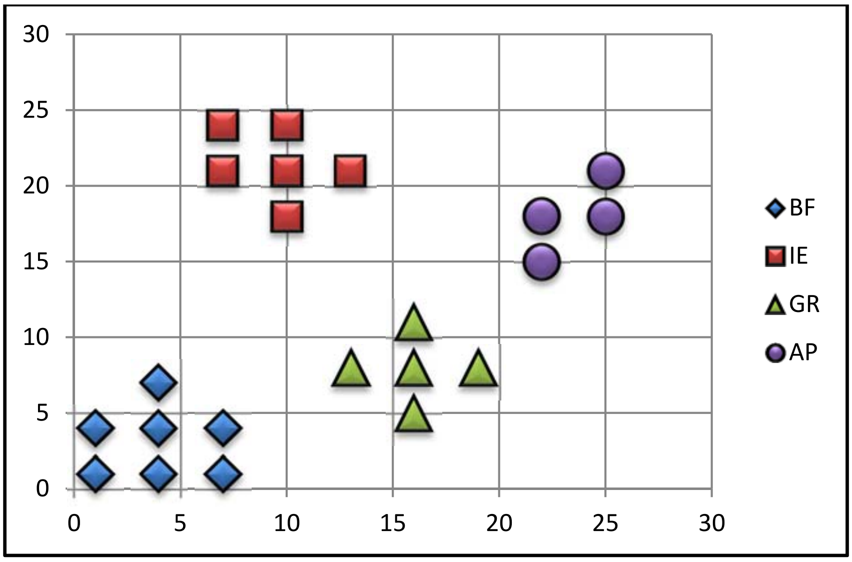


Figure 2.6: An example of a chat-posts data set with a clear cluster structure.

Formulation of the problem of clustering the chat posts:

The problem of clustering the set of posts of a chat is the process of grouping the posts into subsets or 'clusters' in such a manner that the posts within a group are similar among themselves but dissimilar among the posts of other groups. Adopting the clustering problem statement of Manning et al. (2009) we formulate the problem of chat post clustering as follows:

Let a chat log contains the set of posts $P = \{p_1, \ldots, p_N\}$. Those posts are to be grouped into K number of clusters. The problem of clustering the chat posts is to compute an assignment $\gamma : P \rightarrow \{1, 2, \ldots, K\}$ that minimizes (or, in some cases, maximizes) an 'objective function' that evaluates the quality of a clustering. The objective function is often defined in terms of similarity or distance between chat-posts. Using the Cosine similarity or the Euclidean distance in vector space the similarity or distance between a pair of posts is measured. In our newly proposed clustering approach we measured the similarity by using a new approach which will be discussed in Chapter-3.

The K-means (MacQueen, 1967) is a widely used flat clustering algorithm due to its simplicity and efficiency. The EM algorithm (Dempster, 1977) is a generalization of K-means and can be applied to a large variety of data representations and distributions. HAC (Ward Jr, 1963) algorithm is useful when a hierarchy of clustering is required.

Those three clustering algorithms are used to compare the results with the new clustering algorithm developed in this research. Followings sections provide brief discussions about those three clustering algorithms.

2.5.4 *K*-means Clustering

In *K*-means clustering at first *K* centroids, one for each cluster, are randomly chosen in the vector space model. Next each point in the vector space corresponding to each instance[1] is associated to the nearest centroid. When all instances have been assigned, the positions of the *K* centroids are recalculated in order to minimize the 'objective function'. This changes the positions of the centroids. A new binding is then done between the instance points and the nearest new centroids. These two steps: (i) assignment of instances to centroids and (ii) recalculation of centroids; are repeated until the centroids do not change the positions any more. The 'objective function' in *K*-means clustering is to minimize the squared error function or 'the 'residual sum of squares' (RSS). The RSS is given by (Manning et al, 2009):

$$\mathrm{RSS}_k = \sum_{x \in w_k} |\boldsymbol{x} - \boldsymbol{\mu}(w)|^2$$

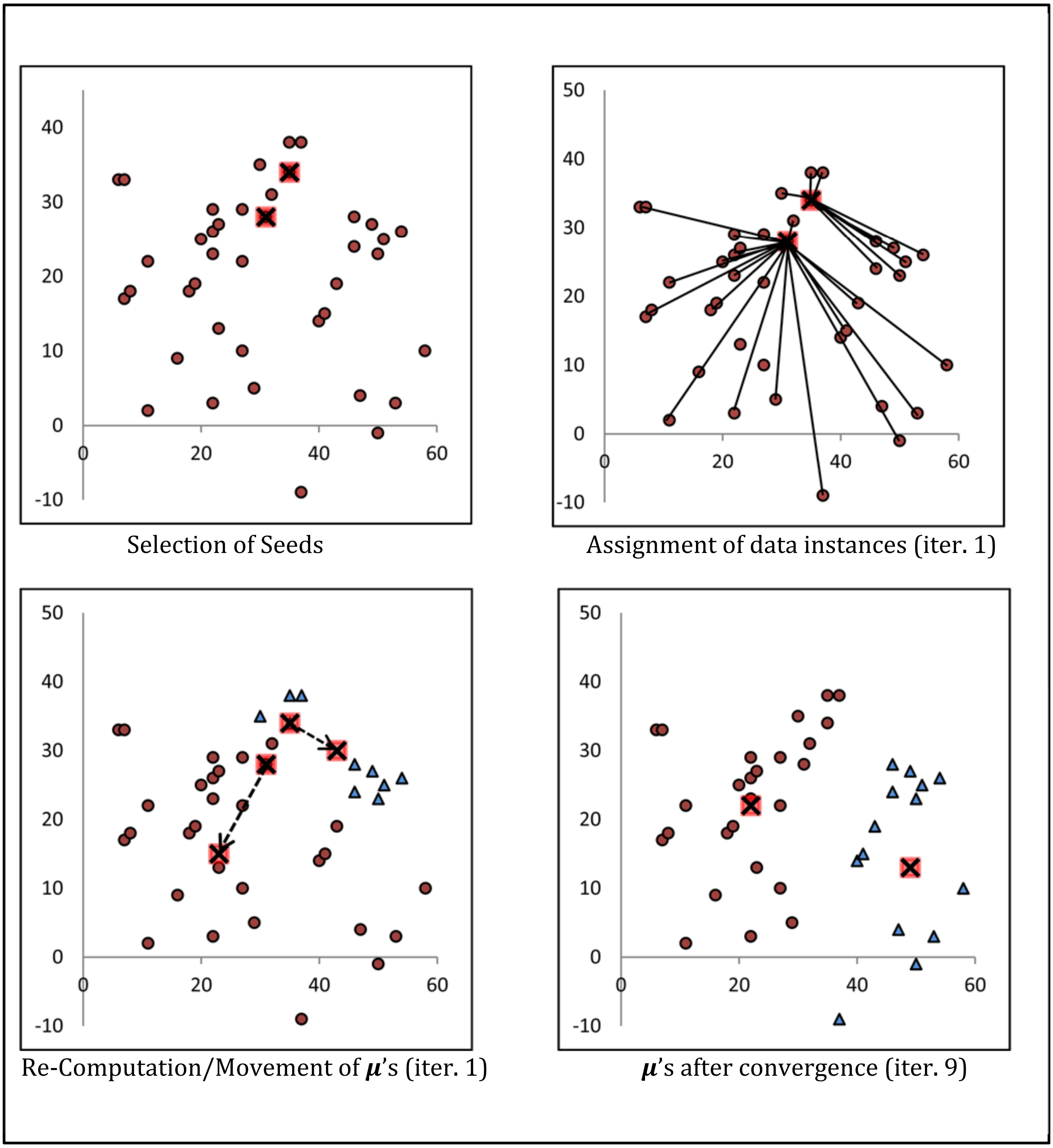
and,

$$\mathrm{RSS} = \sum_{k=1}^{K} \mathrm{RSS}_k \qquad \ldots \quad \text{Equation 2.19}$$

The centroid $\boldsymbol{\mu}$ in Equation 2.19 is the centre of a cluster w containing a group of instances $(X = \{x_1, \ldots, x_k\})$ as its members and is given by:

$$\boldsymbol{\mu}(w) = \frac{1}{|w|} \sum_{x \in w} \boldsymbol{x} \qquad \ldots \quad \text{Equation 2.20}$$

[1] Here an instance means a text-object. It is a unit of text which can be a document, a sentence or a chat-post. A chat-post is a unit text fragment of a chat-log therefore acts as a clustering instance in this research.



Selection of Seeds

Assignment of data instances (iter. 1)

Re-Computation/Movement of $\boldsymbol{\mu}$'s (iter. 1)

$\boldsymbol{\mu}$'s after convergence (iter. 9)

Figure 2.7: A K-means example for $K = 2$ in $\mathbb{R}^2$. The position of the two centroids ($\boldsymbol{\mu}$'s shown as crosses (×)) converges after nine iterations.
(Source: Reproduced from Manning et al. (2009), p. 362)

Figure 2.7 shows snapshots of iterations of the K-means algorithm for a set of data points in a two dimensional space. The number of target clusters is two (K=2), therefore two data points are randomly selected as seed. These two seeds work as the initial centroids of two clusters. The centroids are denoted by crosses (×) in the figures. In the first iteration all the data instances are assigned to the two clusters according to the closest centroids. Using the assigned data of each cluster the centroids are re-computed. In the next iteration the data instances are re-assigned to the new centroids. In this way the data in the example in Figure 2.7 takes nine iterations for convergence. The final clusters and their centroids are also shown in the figure.

K-means algorithm has been adopted to many problem domains. We used this algorithm to cluster the chat-posts according to predefined psychological stages and compared the results with a newly developed clusterer. The results are presented in Chapter-6.

2.5.5 Expectation Maximization Clustering

The algorithm behind the Expectation Maximization (EM) clustering finds the maximum likelihood estimates of parameters in a probabilistic model. It assumes that a background model generates the data and therefore attempts to estimate the background model from the training data. The estimated model then is used to cluster new data instances. The EM algorithm alternates between an expectation (E) step, and a maximization (M) step. In the E-step the data instances are reassigned into clusters according to current estimate of the model parameters. In the M-step the likelihood function is maximized by recomputing the model parameters. The parameters found in the M-step are then used to begin another E-step, and the process is repeated.

In this research the posts of a chat are the data instances to be clustered. Adopting the EM clustering concept and the related equations from Manning et al. (2009) we formulate the EM clustering to cluster the posts of a chat as follows:

Let us assume that the parameters of a probabilistic model are given by:

$\Theta = \{\Theta_1, \ldots, \Theta_K\}, \Theta_K = (\alpha_k, q_{1k}, \ldots, q_{Mk})$, and $q_{Mk} = P(\mathcal{U}_m = 1|w_k)$

Where, $P(\mathcal{U}_m = 1|w_k)$ denotes the probability that a chat-post belonging to cluster w_k contains a term t_m. The prior of cluster w_k is given by α_k , and it is defined as the initial probability that a chat-post *d*[1] is in w_k when no internal information about *d* is available.

The probabilistic mixture model then is given by:

$$P(d|\Theta) = \sum_{k=1}^{K} \alpha_k \left(\prod_{t_m \in d} q_{mk} \right) \left(\prod_{t_m \notin d} 1 - q_{mk} \right) \qquad \ldots \quad \text{Equation 2.21}$$

A chat-post instance *d* (a member of a cluster *k*) is being generated in this model by first picking a cluster *k* with probability α_k and then generating the terms of the chat-post according to the parameters q_{mk}.

Maximization (M) Step:

The conditional parameters q_{mk} and the priors α_k are computed in the M-step as follows:

$$q_{mk} = \frac{\sum_{n=1}^{N} r_{nk} I(t_m \in d_n)}{\sum_{n=1}^{N} r_{nk}}; \quad \alpha_k = \frac{\sum_{n=1}^{N} r_{nk}}{N} \qquad \ldots \quad \text{Equation 2.22}$$

Where: $I(t_m \in d_n) = 1$ if $t_m \in d_n$; otherwise $I(t_m \in d_n) = 0$. r_{nk} is the soft assignment of chat-post d_n to cluster *k* as computed in the preceding iteration. These are the maximum likelihood estimates to maximize the likelihood of the data given the model.

Expectation (E) Step:

After getting the current parameters q_{mk} and α_k from M-Step, the E-step estimates the soft assignment of chat-posts to clusters (r_{nk}) as:

[1] In this thesis most of the time we denote a chat-post as *P* or *p*. Here *P* is being used to denote probability. Therefore to avoid confusion we are using *d* instead of *p* to denote a chat-post.

$$r_{nk} = \frac{\alpha_k \left(\prod_{t_m \in d} q_{mk}\right)\left(\prod_{t_m \notin d} 1 - q_{mk}\right)}{\sum_{k=1}^{K} \alpha_k \left(\prod_{t_m \in d} q_{mk}\right)\left(\prod_{t_m \notin d} 1 - q_{mk}\right)} \quad \ldots \quad \text{Equation 2.23}$$

The numerator $\alpha_k \left(\prod_{t_m \in d} q_{mk}\right)\left(\prod_{t_m \notin d} 1 - q_{mk}\right)$ of Equation 2.23 is the assignment probability $P(d|w_k;\Theta)$ for each pair of chat-post and cluster once we have Θ. The denominator $\sum_{k=1}^{K} \alpha_k \left(\prod_{t_m \in d} q_{mk}\right)\left(\prod_{t_m \notin d} 1 - q_{mk}\right)$ is the probability $P(d|\Theta)$ in Equation 2.21. Therefore Equation 2.23 becomes:

$$r_{nk} = \frac{P(d|w_k;\Theta)}{P(d|\Theta)} \quad \ldots \quad \text{Equation 2.24}$$

This expectation step is computing the likelihood that w_k generated chat-post d_n. It resembles the Naive Bayes classification to get a probability distribution over clusters.

EM has a serious problem of getting stuck in local optima if the seeds are not chosen well. This is a general problem of EM algorithm itself that also occurs in applications other than clustering. Therefore, the initial assignment of data instance to clusters is often computed by a different algorithm.

2.5.6 Hierarchical Clustering

Hierarchical clustering outputs a hierarchy of clusters formed by the data instances. Hierarchical clustering can be of two types:

1. Agglomerative (bottom-up): This starts with each data instance being a single cluster. In each step smaller clusters are merged together to form bigger clusters. Eventually all data instances belong to the same cluster.
2. Divisive (top-down): This starts with all data instances belong to the same cluster. The bigger cluster is split into smaller clusters in each step. Eventually each node forms a cluster on its own.

The final mode in both agglomerative and divisive is of no use. Therefore the process is stopped at a desired level before arriving to the final mode.

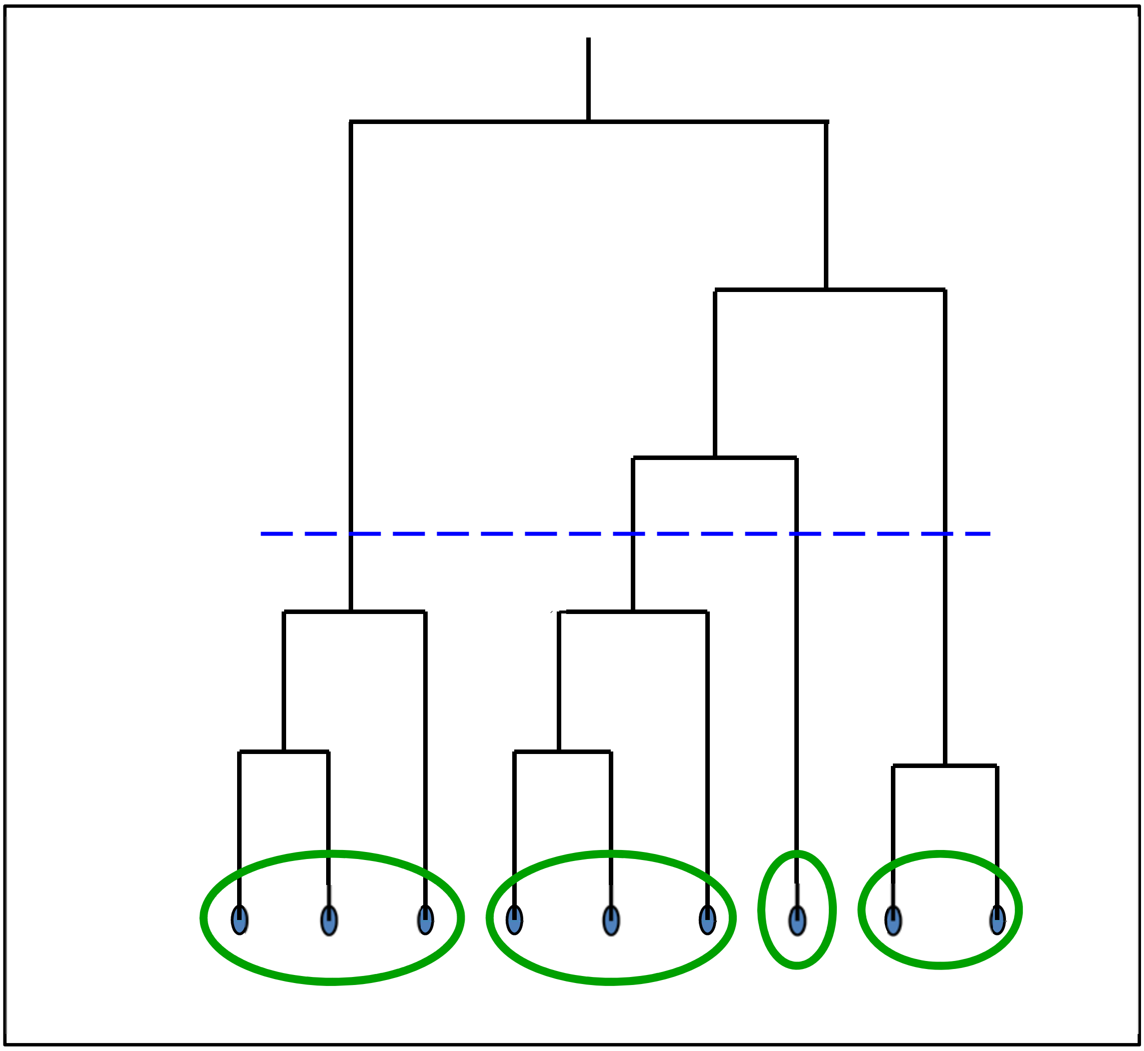


Figure 2.8: A dendrogram showing hierarchical agglomerative clustering (HAC).

Figure 2.8 visualizes a Hierarchical Agglomerative Clustering (HAC) clustering through a 'dendrogram'. A horizontal line represents each merge in the HAC algorithm. The vertical scale shows the similarity level at which clusters are merged. In a dendrogram the data instances are viewed as singleton clusters. Clustering is obtained by cutting the dendrogram at a desired level. At that level each connected component forms a cluster. For example in the Figure 2.8 there will be four clusters at the blue dotted line. The members of each of those four clusters are shown with green circles.

The following algorithm of HAC and different merging techniques are adopted from Manning et al. (2009) to fit in the clustering problem of the chat-posts.

Algorithm of HAC:

Consider each individual post in a chat as a singleton cluster. Compute a $N \times N$ similarity matrix C, where N is the number of posts (singleton clusters). Then iteratively merge the currently most similar clusters. In each iteration, merge the two clusters which have a highest similarity in between them and update the rows and columns of the merged cluster *i* in *C*. If any tie occurs break the tie with a deterministic method, for example, choose the merge that comes first when the similarities are equal between two pairs. Two lists are used to keep tracks of merged and unmerged clusters: a list-*A* keeps the merged list and another list-I keeps the clusters which are still available to be merged. A similarity function SIM(*i, m, j*) computes the similarity of cluster j with the merge of clusters i and m. It is simply a function of C[*j*][*i*] and C[*j*][m] depending on the merging techniques, for example, in a 'centroid-link' it is equal to the average of these two values. Different merging techniques are discussed in the following section.

Merging techniques in HAC:

There are four different similarity measures used as the merging technique in HAC algorithms. Those are: single-link, complete-link, group-average, and centroid similarity. The algorithms of HAC are also named according to these similarity methods. The merge criteria of these four variants of HAC are shown in Figure 2.9. In that figure an inter-similarity is a similarity between two chat-posts from different clusters.

The single-link HAC uses maximum similarity of pairs as below:

$$\text{SIM-SL}\left(w_i, w_j\right) = \max_{p_x \in w_i, p_y \in w_j} \text{sim}\left(p_x, p_y\right) \qquad \ldots \quad \text{Equation 2.25}$$

Where w's represent the clusters and p's represent the chat-posts.

After merging w_i, and w_j, the similarity of the resulting cluster to another cluster, w_k, is:

$$\text{SIM-SL}\left(\left(w_i \cup w_j\right), w_k\right) = \max\left(\text{sim}(w_i, w_k), \text{sim}\left(w_j, w_k\right)\right) \ldots \quad \text{Equation 2.26}$$

The single-link HAC can result in long and thin clusters due to chaining effect.

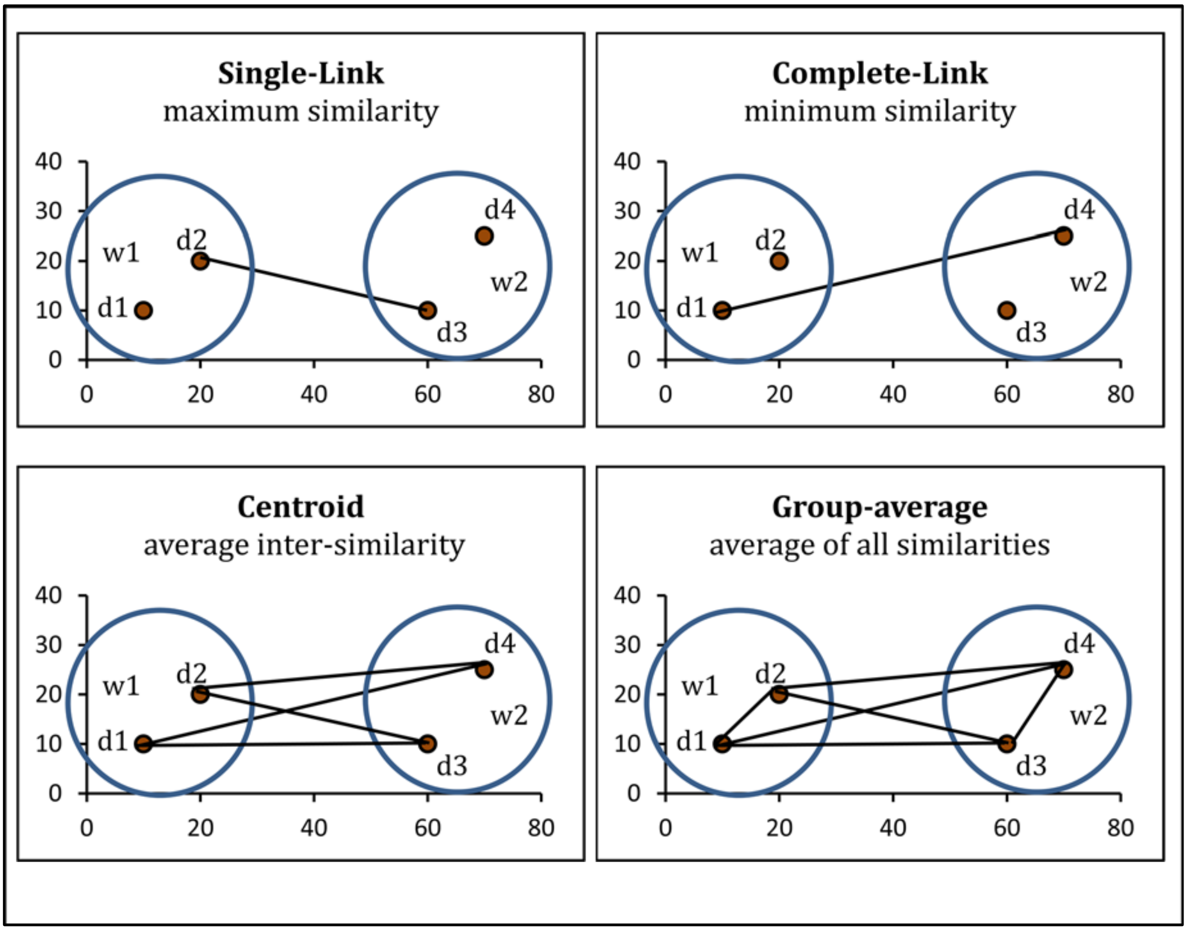


Figure 2.9: The different notions of cluster similarity used by the four HAC algorithms. (Source : Reproduced from Manning et al. (2009), p. 381)

The complete-link HAC uses minimum similarity of pairs as below:

$$\text{SIM-CL}\left(w_i, w_j\right) = \min_{p_x \in w_i, p_y \in w_j} \text{sim}\left(p_x, p_y\right) \qquad \dots \quad \text{Equation 2.27}$$

After merging w_i and w_j, the similarity of the resulting cluster to another cluster w_k is:

$$\text{SIM-CL}\left(\left(w_i \cup w_j\right), w_k\right) = \min\left(\text{sim}(w_i, w_k), \text{sim}\left(w_j, w_k\right)\right) \;\cdots\quad \text{Equation 2.28}$$

Complete-link clustering suffers from sensitivity to outliers. It pays too much attention to outliers sometimes end up in undesired structure of the cluster. A single data instance far from the centre can dramatically increase diameters of candidate merge clusters and completely change the final clustering.

Group Average Agglomerative Clustering (GAAC) use average similarity across all pairs within the merged cluster. The self-similarities are not included in the average. The similarity between two clusters in GAAC is given by SIM-GA as:

$$\begin{aligned}
&\text{SIM-GA}\left(w_i, w_j\right) \\
&= \frac{1}{\left(\left|w_i \cup w_j\right|\right)\left(\left|w_i \cup w_j\right| - 1\right)} \sum_{p_x \in \left(w_i \cup w_j\right)} \sum_{p_y \in \left(w_i \cup w_j\right) : p_x \neq p_y} \left(\vec{p}_x \cdot \vec{p}_y\right) \\
&= \frac{1}{\left(N_i + N_j\right)\left(N_i + N_j - 1\right)} \sum_{p_x \in \left(w_i \cup w_j\right)} \sum_{p_y \in \left(w_i \cup w_j\right) : p_x \neq p_y} \left(\vec{p}_x \cdot \vec{p}_y\right) \qquad \ldots \quad \text{Equation 2.29}
\end{aligned}$$

Where: $\vec{p}$ is the length-normalized vector of chat-post *p*, and, *N_i* and *N_j* are the number of chat-posts in clusters w_i and w_j respectively. $\left(\vec{p}_x \cdot \vec{p}_y\right)$ denotes the dot product between chat-post vectors $\vec{p}_x$ and $\vec{p}_y$.

Some preconditions of GAAC includes:

(i) chat-posts are represented as vectors,
(ii) those vectors are length normalized, so that self-similarities become 1.0, and
(iii) the similarity between vectors are computed with the measure of dot product.

The centroid similarity, in the 'centroid HAC', is defined as the similarity between the centroids of two clusters as below:

$$\begin{aligned}
\text{SIM-CENT}\left(w_i, w_j\right) &= \boldsymbol{\mu}(w_i) \cdot \boldsymbol{\mu}\left(w_j\right) \\
&= \left(\frac{1}{N_i} \sum_{p_x \in w_i} \boldsymbol{p}_x\right) \cdot \left(\frac{1}{N_j} \sum_{p_y \in w_j} \boldsymbol{p}_y\right) \\
&= \frac{1}{\left(N_i N_j\right)} \sum_{p_x \in w_i} \sum_{p_y \in w_j} \left(\boldsymbol{p}_x \cdot \boldsymbol{p}_y\right) \qquad \ldots \quad \text{Equation 2.30}
\end{aligned}$$

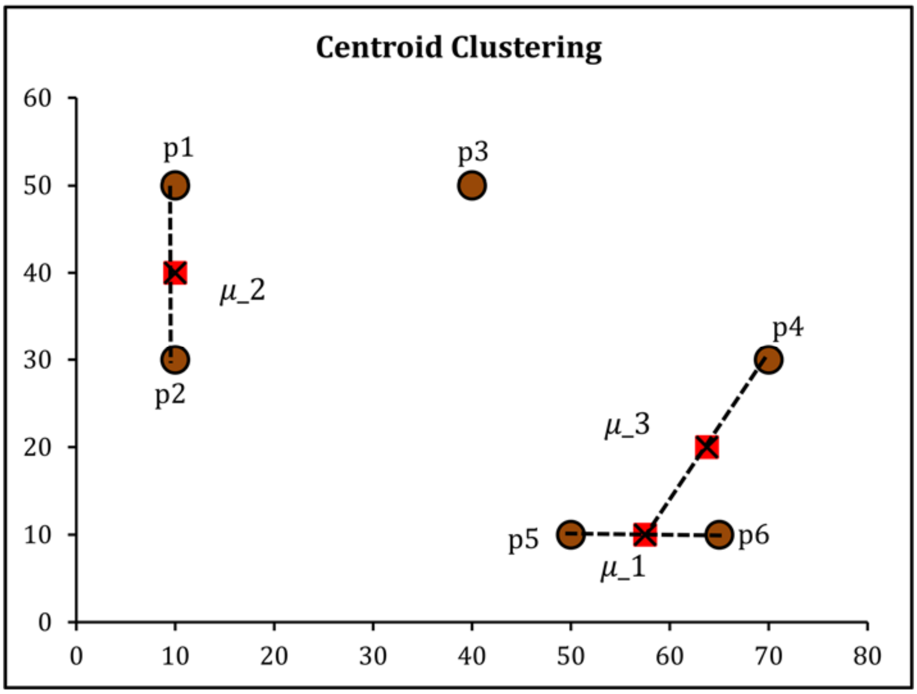


Figure 2.10: Three iterations of centroid clustering. Each iteration merges the two clusters whose centroids are closest.
(Source: Manning et al. (2009), p. 391.)

The convergence of a centroid clustering through the first three steps is shown in Figure 2.10. The first two iterations find highest centroid similarities in the pairs $\langle \boldsymbol{p}_5, \boldsymbol{p}_6 \rangle$ and $\langle \boldsymbol{p}_1, \boldsymbol{p}_2 \rangle$. Therefore they are grouped into clusters $\{\boldsymbol{p}_5, \boldsymbol{p}_6\}$ and $\{\boldsymbol{p}_1, \boldsymbol{p}_2\}$ with corresponding centroids $\boldsymbol{\mu}_1$ and $\boldsymbol{\mu}_2$. The third iteration produces the cluster {$\boldsymbol{p}_4$, $\boldsymbol{p}_5$, $\boldsymbol{p}_6$} because in this iteration the highest centroid similarity is found between $\boldsymbol{\mu}_1$ and $\boldsymbol{p}_4$. The new cluster {$\boldsymbol{p}_4$, $\boldsymbol{p}_5$, $\boldsymbol{p}_6$} has a centroid $\boldsymbol{\mu}_3$.

The difference between GAAC and centroid clustering is that GAAC considers all pairs of chat-posts in computing average pairwise similarity whereas centroid clustering excludes pairs from the same cluster. Single-link, complete-link and GAAC are monotonic HAC algorithms which means similarity is 'monotonically decreasing' from iteration to iteration. However, centroid HAC clustering is not 'monotonic'. In this method 'inversion' can occur, that is, similarity can 'increase' instead of 'decreasing' during sequence of two clustering steps (Manning et al., 2009, page 392). However, due to the conceptually simple similarity measure of two centroids, despite the shortcoming of non-monotonicity, centroid clustering is often used. Centroid and GAAC do not have problem of 'chaining' effect of single-link or 'sensitivity to outlier' of complete-link (Manning et al., 2009). When a pairwise matrix is available then the simpler centroid HAC is preferable over the relatively difficult GAAC. In our experiment the centroid HAC is used with a new measure of pair-wise similarity between chat-posts. More details about the adoption of centroid HAC in the current research is explained in Chapter-3.

2.6 Recognition of Textual Entailment (RTE) in Chat

2.6.1 Background

Detection of strong evidence in the suspected chat-logs requires establishing the presence of some specific propositions in it. Examples of these propositions may include: "an adult pursues child-grooming activities", "the suspect exchanged personal information with the victim", and "the suspect approached with the intention of physically meeting with the victim". It is obvious that exact match of these kinds of statements hardly would ever be found in the discrete, sparse, conversation style chat texts. Therefore instead of explicit lexical matching an illative entailment is required. This particular problem of inferential establishment in chat texts resembles with the problem of RTE (Recognition of Textual Entailment). The RTE task consists of developing a system that, given two text fragments, can determine whether the meaning of one text is entailed, that is can be inferred, from the other text. Since its inception, use of RTE is constantly growing in the Natural Language Processing (NLP) applications such as Question Answering (QA), Information Retrieval (IR), Information Extraction (IE), (multi-) document summarisation and even the task of recognizing paraphrases. It seems to work as a common framework for NLP applications to deal with semantic inference. Therefore we would like to investigate the applicability of the concept of RTE to address the task of CE detection in chats.

2.6.2 Definition of RTE

The problem of RTE is formally defined as the task of determining the entailment relationship between a pair of texts referred to as hypothesis (H) and text (T). The hypothesis (H) is a short, succinct piece of text and the text (T) is an elaborated text that may be a document or a part of it or even it can be a big sentence. Text (T) includes some words, the meaning of which may or may not entail the meaning of the hypothesis. If the meaning of H can be inferred from T, then the relationship is denoted by $T \vDash \mathrm{H}$ (read as T entails H). This definition is abridged from Ofoghi and Yearwood (2009, 2011), Akhmatova and Mollá (2006), de Salvo Braz et al. (2006), Glickman (2006), Wang and Zhang (2008), and Castillo (2010).

For example[1], say;

hypothesis (H) = " Rita picked up strength." and

text (T) = "Hurricane Rita was upgraded from a tropical storm as it threatened the south eastern United States, forcing an alert in southern Florida and scuttling plans to repopulate New Orleans after Hurricane Katrina turned it into a ghost city three weeks earlier."

Then the relation: $\mathrm{T} \vDash \mathrm{H}$ holds true.

The classification of the relationship between the hypothesis and the text can be either a 3-way classification or a 2-way classification task. The 3-way classes are:

[1] This example is cited in the task guidelines of 6th textual entailment challenge at TAC 2010 (Bentivogli et al., 2010).

Entailment: where $T \vDash H$.

Contradiction: where $T \vDash \neg H$.

Unknown: where $T \nvDash H$ there is not enough evidence available in the text to decide whether $T \vDash H$ or $T \vDash \neg H$.

In the 2-way classification method, the Contradiction and Unknown relations are unified into a single class called No Entailment $T \nvDash H$.

The entailment relation is directional because even if "T entails H", the reverse "H entails T" is much less certain (Dagan and Glickman (2004); Tatar et al. (2009)).

2.6.1 Different Sources of Entailment

There are three main sources of text entailment:

1. Syntactic Information
2. Semantic Information and
3. Logical Information

Syntactic Information:

Sentences with different syntactic structure may express the same information. For example:

1. The builders have finished building the new house.
2. A new house has been built.

In such cases the main source of entailment is syntactic information. The entailment is derived by syntactic transformation.

Semantic Information:

Actual meaning of the words may constitute the source of textual entailment. In such cases the semantic relations existing between words are more useful. For example, “The man saw a poodle” entails “the man saw a dog” because ‘dog’ and ‘poodle’ are semantically related.

Logical Information:

Entailment can be found by logical concept matching. If the logical concept of one sentence implies the logical concept of another sentence then the first entails the second. For example, from the sentence “Australian Prime Minister Tony Abbott has made a surprise visit to Afghanistan” one can entail that Tony Abbott is a prime minister of Australia and he visited Afghanistan. As people have common sense and can use world knowledge it would easily be deduced also that Tony Abbott exists, he is human, he is a resident of Australia. The source of these entailments would not come through syntactic or lexical analysis this time, though these two types of linguistic information might play an auxiliary role in the process. Instead, the source of these entailments is knowledge representation and reasoning using a knowledge base that holds necessary rules represented in lexical axioms. An example of one of the lexical axioms is:
prime-minister(X) :– human(X), resident(X) .
This example is a representation of a rule from a knowledge base (KB). This particular rule means X is a prime-minister if X is a human and resident; that is for X to be a prime minister he has to be a human and also has to be a resident. The KB may contain this kind of many other rules. The knowledge base would be complemented with a tool for reasoning that can work for the entailment task with this type of information.

In practice, textual entailment is the combination of syntax, semantics, and logic. This can be seen with the example “Tom cat chased and killed Micky mouse”. This sentence is a conjunction of two pieces of information, and therefore it entails ‘Tom cat killed Micky mouse’. A simple syntactic transformation allows the sentence to entail ‘Micky mouse was killed’. Furthermore, there is a cause-effect relation between kill and die, and therefore the sentence entails ‘Micky mouse died’.

2.6.2 Classification of Entailments

According to the methods and available tools the categories of textual entailment broadly include:

1. Lexico-syntactic entailment
2. Descriptive entailment
3. Knowledge-based entailment
4. Similarity based Approximate entailment

Lexico-Syntactic Entailment:

In this type, entailments are detected with the help of syntactic and lexical knowledge only. In other words, the hypothesis is just a lexico-syntactic variant of the text sentence. The only tools required to prove the entailment relation are those concerned with the extraction of syntactic structures of the text and hypothesis, plus a lexical database. Examples of this type of entailment are:

text: The guests dined upon roast beef.

hypothesis: The guests had dinner.

Descriptive Entailment:

Entailments of this group are characterized by the substitution of entire descriptions or definitions with a shorter expression. For example[1]:

text: Israeli Prime Minister Ariel Sharon threatened to dismiss Cabinet ministers who don't support his plan to withdraw from the Gaza Strip.

hypothesis: Israeli Prime Minister Ariel Sharon threatened to fire cabinet opponents of his Gaza withdrawal plan.

The following syntactic compression has been made in the above entailment example:

a. Generalization: Gaza Strip→Gaza

b. Nominalization: plan to withdraw→withdrawal plan

[1] This example is cited in Akhmatova and Mollá (2006)

c. Lexical substitution: to dismiss→to fire

d. Definition substitution: ministers who do not support X's plan→opponents.

The above definition substitution is difficult to detect automatically given that it would most likely not appear in standard lexical knowledge bases.

Knowledge-Based Entailment:

Entailments that would need some extra knowledge, possibly from some entailment database, are derived in this type. Lexical resources and syntax play an auxiliary role only. For example:

text: Eating lots of foods that are a good source of fibre may keep your blood glucose from rising too fast after you eat.

hypothesis: Fibre improves blood sugar control.

This type of entailment is much harder than the other two for the simple reason that currently there is no knowledge base containing all the common-sense knowledge required, and even if there were any it is not obvious how to find the required information among a sea of unrelated information (Akhmatova and Mollá 2006).

The above mentioned entailments require expensive language analysis tools and give somewhat 'strong' decision. Therefore those types of entailments can be categorized as 'strong entailment'. In contrast of those an 'approximate entailment' is recently proposed by Esteva et al.(2010, 2012). The following section puts some light on this.

Approximate Entailment:

Following the similarity-based reasoning (Ruspini, 1991) Esteva et al.(2010, 2012) proposed a mathematical model for 'approximate' entailment. The authors follow a quantitative approach and use fuzzy similarity relations. Consider a variety of possible situations represented by a set of propositions. Each proposition represents a set of situations, namely those situations in which it holds. It is then often the case that certain propositions are close to each other, whereas other propositions differ from each other to a large extent. In an 'approximate' entailment, instead of giving a hard yes-no answer the entailment is provided using a similarity relation. The basic idea of the similarity based approximate entailment is used in this current research. A detailed explanation is provided in Chapter-4.

2.6.3 Existing Works on RTE

We could not find any existing literature on RTE which mentions application into the informal chat-text. RTE is comparatively a new and ongoing research field even in the formal text processing area. In absence of RTE on chat-text a brief review is given in this section from some of the RTE methods worked on formal texts.

Stern and Dagan (2014) formulated the task of recognizing implied predicate-argument relationships and proposed it to solve the RTE problem. Fifteen different features including statistical discourse, local discourse, local candidate properties and predicate-argument relatedness are used for this purpose. The authors argued that the task of RTE would be easier in this method; it would become the task of only to verify that a predicate-argument relationship in the Hypothesis is implied from the given Text. The method was applied on RTE-6 data set. The F-score was achieved as 45.2% which, according to the author, was better than the median result in the RTE-6 challenge (36.14%). The structure of predicate-argument is associated mostly with (content) verbs and noun phrases (NPs). To find the POS tags (parts of speech tags of noun and verb) with each term of a text an efficient parser is required. As a chat-text does not follow a formal sentence structure, we will see in Chapter-4 that a parser does not work correctly on it. Therefore although the above method works good in formal text however is not applicable in the current research problem.

Using textual entailment and text similarity measures, Dhruva, Ferschkey and Gurevych (2014) attempt to solve open-domain multiple choice reading comprehension questions about short English narrative texts. Each answer option is scored with a combination of all evaluation metrics and ranked according to their overall score in order to determine the most likely correct answer. The performance of the proposed system is presented with c@1 measure (Penas and Rodrigo, 2011).

The best performance achieved was c@1 score of 0.375. The system attempted 56 questions out of 60. The answers were correct for 21 and incorrect for 35. The proposed system uses a number of expensive tools including: EXCITEMENT Open Platform (EOP) (Pado et al., 2013), Stanford Named Entity Recognizer (Finkel, Grenager and Manning, 2005), Stanford coreference resolver (Raghunathan et al., 2010) and Stanford parser (Klein and Manning, 2003). Applying these natural language processing (NLP) tools require the input text to be grammatically correct. However, the current research problem handles chat texts which are not grammatically sound, therefore possess difficulties for those NLP tools. More details about these difficulties are explained in section 4.2 of Chapter-4.

Ofoghi and Yearwood (2009, 2011) used linguistic and lexical resources for the recognition of textual entailment task. The authors adapted the atomic proposition technique of (Akhmatova and Mollá 2006) and augmented information from WordNet and FrameNet for better entailment. The authors tested their system on TAC-RTE datasets and achieved an average accuracy of 0.500 on RTE5 (2009) data set.

Castillo (2010) used a number of lexical features to accomplish the RTE task. The author used 32 lexical features which include widely known features such as Levenshtein distance, percentage of bigrams and trigrams, TFiDF measure, LCS (longest common substring), and Wordnet similarity. The achieved overall accuracy on RTE5 dataset was 0.6117.

A Description Logic based hierarchical knowledge representation, EFDL (Extended Feature Description Logic) was employed by de Salvo Braz et al. (2006) to infer the semantic entailment in texts. In their system they represent the surface level text, augmented with induced syntactic and semantic parses and word and phrase level abstractions. In the background a knowledge base (KB) is used. The KB consists of syntactic and semantic rewrite rules, written in EFDL. The overall system performance was 65.9% on RTE1 dataset.

The above mentioned works of Ofoghi and Yearwood (2009, 2011), Castillo (2010), and de Salvo Braz et al. (2006) use parsing a formal sentence. As already has been mentioned that a chat text is not formal and causes difficulties in parsing correctly, the methods are not applicable for this current problem of CE detection in chats.

RTE has been used for different types of language processing tasks like question answering (QA) (Dhruva, Ferschkey and Gurevych, 2014; Heilman and Smith 2010; de Salvo Braz et al. 2005), information extraction (IE), information retrieval (IR), and in summarization (Tatar, Mihis, and Lupsa 2008). However to the best of our knowledge any RTE research focusing on entailing any evidential hypothesis from a chat text is still missing. Grammatically unstructured, erroneous, and discrete properties of chat-data made it difficult to directly employ any existing RTE system for this purpose. The current research designs a system that is able to handle the problem of chat-data and accomplish the CE evidence detection task in an entailment setting.

One of the important questions that needs to be addressed is what kind of entailment will prove to be useful in application of the current research. Investigation of RTE literature reveals that different researchers used different types of techniques for the textual entailment. Due to the comparative simplicity lexico-syntactic entailments are widely used (Mehdad, Moschitti, & Zanzotto, 2010; Ofoghi & Yearwood, 2009, 2011). The other techniques of RTE includes machine learning and knowledge base approach (de Salvo Braz et al., 2005,2006; Heilman & Smith, 2010). However, all these approaches require formal structured grammatical sentences in the text. The sentences have to be parsed correctly with a sentence parser before it goes to the next level of synonym, hypernym, meronym similarity measure. Whatever the next step's methodology; tree edit(Lin, 2007), atomic proposition (Akhmatova and Mollá 2006), rewriting first order logic rules (Zanzotto, Pennacchiotti, & Moschitti, 2009); the very first step is to parse the sentence. The chat messages do not follow structured grammars, hence gives erroneous result in the parsing. Under these circumstances some new technique is required to entail the hypotheses without using a parser. The new soft entailment system developed in this current research attempts to satisfy this condition.

2.7 Existing Works on CE Detection

Research on cybercrimes using informal texts like chat and short-text media is comparatively new and still evolving in the field of text-processing. It is difficult to find much text-processing research about child protection from chat-exploitation. To the best of our knowledge we could not find any research that focuses exactly on the current research problem of finding evidence of CE in chat-logs. However we have found some recent interesting research having similarity with some parts of our research problem. Brief discussions on those works are given below.

Bogdanova, Rosso, and Solorio (2014) performed a task of binary text categorization to predict whether a chat text is a case of cyber-paedophilia or not. The work is very interesting as it has some similarity with the work in the first part of our research methodology which was previously published beforehand in Miah, Yearwood and Kulkarni (2011). The authors consider chats as of three kinds: 1. Paedophilia, 2. Cybersex, and 3. Common. The idea is same as our published idea that the chat texts are of three types: 1. Child Exploiting (CE), 2. Near CE or Sex Fantasy (SF), and 3. Far from CE or General (GN). Although this was published beforehand the authors failed to cite our paper. The task also has similarity with a part of our task; classification between CE vs SF and CE vs GN. The CE type data is collected from the same source: Perverted.Justice (PJ) website. The source of SF and GN type data is different. They collected Cybersex type chats from http://oocities.org/urgrl21f/; we collected from http://fugly.com/ and http://chatdump.com/ . As the general type chats they used NPS chats; we used chats from different open websites. The selected features are different than our work. In our experiments we did not separate predator's and victim's chats; all chats were pair of user dialogues ; the authors of this paper separated and used only predators text of CE type chats , all other chats are also of single user texts only. Bogdanova et al. (2014) used two different kinds of features: (a) Low-level features (baseline), and (b) High-level features. The baseline low-level features include: (a) Bag of words, (b) Word bigrams, (c) Word trigrams, (d) Character bigrams, and (e) Character trigrams. The high-level features include:

1. Sentiment markers from SentiWordNet (Baccianella et al., 2010): positive, negative words;

2. Emotional markers from WordNet-Affect (Strapparava and Valitutti, 2004; Strapparava and Mihalcea, 2007): anger, disgust, fear, joy, sadness, and surprise words.
3. Features borrowed from McGhee et al. (2011): approach, relationship, family, communicative desensitization, and information words.
4. Features helpful to detect neuroticism level from Argamon et al. (2009): personal pronouns, reflexive pronouns, obligation verbs.
5. Fixated discourse estimated by lexical chain (Morris and Hirst, 1991) constructed from WordNet semantic similarity of the term 'sex' measured by Leacock and Chodorow (1998), and, Resnik (1995).
6. Emoticons, and Imperative sentences.

Using these features in a Support Vector Machine (SVM) classifier the authors formulated two separate tasks of chat text classification:

1. PJ (Paedophilia) vs Cybersex, and
2. PJ (Paedophilia) vs NPS (Common chats).

With all the high level features the SVM classifier gives a mixed response. This achieved 94% accuracy for classifying 'Paedophilia vs Cybersex'. For classifying 'Paedophilia vs NPS' the features show a reverse effect; the classifier achieved 81% accuracy. This is opposite to the common sense idea that a child exploiting chats have more differences with general chats than cybersex chats. In our work published in Miah, Yearwood and Kulkarni (2011) in similar tasks a Naïve Bayes classifier with psychometric features achieved an accuracy of 90.2% for classifying 'CE vs SF', and 95.4% for classifying 'CE vs GN'; which comply with the common sense idea.

The character trigrams features worked much better for classifying 'Paedophilia vs NPS' in the work of Bogdanova et al. (2014), this achieved 97% accuracy; but was very low for classifying 'Paedophilia vs Cybersex' which achieved accuracy of only 64%. The authors' approach achieved the best accuracy of 97% for classifying 'Paedophilia vs Cybersex' in a setting of ablation test by combining features of emotional, fixated discourse and those from McGhee et al. (2011).

In the fixated discourse features the authors used lexical chain constructed from WordNet by calculating semantic similarity measure defined by Leacock and

Chodorow (1998), and, Resnik (1995). For this the authors used JavaWordNet Similarity library (Hope, 2008), which is a Java implementation of Perl Wordnet::Similarity (Pedersen et al., 2004). Our experience says that to obtain WordNet concepts POS tag (verb or noun) is required for each term. The authors did not mention anything about how did they parse the chat-posts to identify whether a term is a verb or a noun. Defining a term with different POS tags results in different synonym sets or concepts from the WordNet which gives different estimate of the lexical chain. The authors did not mention how they have overcome this problem.

The work of Bogdanova et al. (2014) has some similarity with part of our problem however it is not exactly the same; they did not do the task of classifying CE vs Non-CE . By 'Non-CE ' we mean a mixture of cybersex and common chats. This is a more natural way to filter out the CE chats from a mixture of chat data. The proposed low-level features by Bogdanova et al. achieved very poor accuracy for discriminating cybersex chats from CE chats and again the high-level features performed contrary to expectation on the common chat data, therefore we do not know how the features would act to differentiate the CE chats from the 'Non-CE' chats. Moreover the classification part of the research methodology was completed and previously published beforehand therefore we are not interested to use the features proposed by Bogdanova et al. at this time; an elaborated comparison can be an interesting work in the future.

An International Sexual Predator Identification Competition was organized at PAN in CLEF 2012 (http://pan.webis.de/; Inches and Crestani, 2012). Given a collection containing chat-logs involving two (or more) people the participants had to solve the following two kinds of problems:

Problem-1: Identify the predators among all users in the different conversations.
Problem-2: Identify the part (the lines) of the conversations which are the most distinctive of the predator behaviour.

The chat-corpus was built by collecting chat-texts from different websites including : Perverted-Justice.com, Omegle and IRCLog. According to one of the participants (Villatoro-Tello et al., 2012) the data set was as follows:

Training set:
Total of 66,928 different chat conversations
Where 97,690 different users are involved
Only 148 are tagged as sexual predators

Test set:
Total of 155,129 chat texts
Where 218,702 different users are involved and
Only 250 are tagged as sexual predators.

Conversations were not longer than 150 messages. Total predators across the whole data set were no more than 4%. Number of users in each conversation was not confined in a pair, as expected in a sex-predatory chat, instead to make it like a practical problem the number varied from a single user to two or more than two users (Inches and Crestani, 2012).

The system that achieved the highest performance in solving Problem-1 was submitted by Villatoro-Tello et al. (2012). Their system was based on lexical features and a two-step classification. The first step was pre-filtering by removing those conversations containing: (a) only one user, (b) less than 6 posts per-user and (c) long sequences of unrecognised characters (apparently images). This significantly reduced the number of chat conversations from 66,928 to 6,588, users from 97,690 to 11,038, and sexual predators from 148 to 136. The reduction ratio is 90% approximately for conversations and users, and, only 8% for predators. The second step was a classification task. The authors used bag of words representation employing either a boolean or a TFiDF weighting scheme with Neural Network (NN) and SVM classifiers. The system retrieved a total of 204 predators, among them 200 were relevant, achieving a precision (P) of 0.9804, recall (R) of 0.7874, F1 measure ($F_{\beta=1}$) of 0.8734 and F0.5 measure ($F_{\beta=0.5}$) of 0.9346. Unfortunately their solution of problem 2 was positioned 13 among 16 participants.

For the second problem (predatory conversation detection), no training data was available for the participants (Inches and Crestani, 2012). The participants wanted to solve the problem in an unsupervised manner. Later it was revealed that the PAN training chat-corpus contained 6478 conversations which were considered suspicious (of a perverted behavior). Popescu and Grozea (2012) (In: Forner et al., 2012) positioned top for solving Problem2. The authors simply retrieved all chat-texts written by all predators identified in their solution of Problem-1. Although the authors' system positioned seventh in solving Problem1 with 5-grams features in SVM and Random Forest classifiers; their solution for Problem-2 achieved very high recall which elevated their position to the topmost in task-2. Their system for task-2 retrieved 63,290 conversations among which 5,790 was relevant to predatory. The system achieved a precision(P) of 0.0915, recall (R) of 0.8938, F1 measure ($F_{\beta=1}$) of 0.1660, and F3 measure ($F_{\beta=3}$) of 0.4762.

Though both the tasks of PAN-12 appear to be similar with the current research problem however the sense of predator is different. The task in PAN defines predator in a broader and generalized sense of "a person or group that ruthlessly exploits others in a sexual and predatory manner", whereas our sense of predators show paedophilic behaviours and we are interested when they intend exploiting children only. We assume the task in problem-2 was nearer to a partial problem of our research of locating evidence, however the topmost participant did not suggest any new unsupervised method. Moreover the PAN12 data-set was released after we completed our classification task and moved on to the next step of our research. Due to time constrains we could not address and compare our approach with the approaches proposed by the PAN12 participants. A comparison with our approach can be interesting research in the future.

McGhee et al. (2011) also attempted to identify child exploitation in chats. The authors manually annotated 33 chat-logs collected from Perverted-Justice.com. They considered each post of chat as one of the generalized four categories: Class-200, Class-600, Class-900, and Class-000 corresponding to the psychological stages of IE, GR, AP, and BF as have been mentioned previously in Section 2.1.3. In order to distinguish between these four types of chat posts the authors used a rule-based system named ChatCoder2 and compared the result with different machine learning classification approaches including kNN, decision tree, and RIPPER (Repeated Incremental Pruning to Produce Error Reduction). The authors concluded that the machine-learning approach does not improve the result when compared with the rule-based ChatCoder2 system. Chat-Coder2 achieves an overall average accuracy of 68.11% for the 33 chat-logs. The accuracies for the individual classes for individual chat-log range from 0% to 95.89%. A comparison of our system with ChatCoder2 will be provided for the common test chat-logs in Chapter-6.

Pendar (2007) performed an analysis on CE chat-logs to distinguish predators' posts from victims' posts. The author used a set of 701 conversations obtained from the Perverted-Justice webpage (Perverted-Justice.com). The set of victims' posts were manually separated from the set of predators' posts. A two-class problem was formulated to automatically identify the predators' posts. For analysis the author used word unigrams, bigrams, and trigrams and processed them with the classification algorithms support vector machine (SVM) and k nearest neighbours (kNN). Several experiments were performed varying the number of features from 5,000 to 10,000. The best result was achieved with the f-measure of 0.943. The author concluded that 10,000 features were needed for satisfactory performance and the kNN algorithm with k equal to 30 provides the most effective classification when trigrams are used. The current research has a part in its methodology where a similar problem of Predator vs. Victim is addressed. In the experiment of current research a Naïve Bayes classifier, which is much simpler than a kNN classifier, is used and a better results is achieved with a less training data. The results and comparison will be provided in Chapter-6.

The focus of our current work is different from the work to date. The existing works are generally focused on classification either in the post level or in the chat document level, or, differentiating predators from victims. The goal of our current research is to find substantial evidence of CE activity inside the chat document. In the course of accomplishing the goal we started with text classifiers to collect statistical shallow evidence. After that deeper analyses of the chat-posts with novel techniques are conducted to find strong substantial evidence.

2.8 Chapter Summary

The review on social and psychological literature reveals that the psychological stages involved in child exploitation (CE) chats are an important indication of CE. The exploitation does not occur instantly. The perpetrator takes some time in grooming the child to prepare him or her to get ready to serve the adult's purpose. This grooming process follows some certain behavioural psychological stages or phases. Those phases in the online chat text provide us with the basis of the identification of child exploitation and locating the evidence in it.

An analysis indicates that an individual chat-post is very brief, as short as a word. Reviews on text classifiers (TC) show that they are effective when the classification object (documents) have a good number of terms. For classification of chat-posts the TCs sometimes suffer from overfitting or become unreliable because the number of terms in a chat-post is very low and single term posts are also very frequent. Therefore instead of applying TC on classification of chat-posts, it will be more appropriate to apply a TC on the whole chat-log to identify the suspected chat-logs out of a mixed chat data set. Regarding the features of the TCs, a brief discussion of the psycholinguistic features suggests that this may assist the TCs in this current problem.

Text classifiers neither provide particular evidence of CE nor do they detect the psychological stages in CE chats. For further analysis the CE chats are to be segmented into those psychological stages through an unsupervised clustering process. Reviews on clusterers indicate that the similarity (or distance) between pairs of chat-posts plays an important role in this process. From the literature of existing sentence similarity measure, we have learnt that they can be applied to the small-text chat-posts however they need some modification to overcome their limitations when applied to chat-text. For this purpose, based on the existing sentence similarity measure a new similarity measure and a new clustering method will be constructed in Chapter-3.

Chat-posts seldom overlap textual content even for similar psychological context. Recognition of Text Entailment (RTE) is an immerging technique to recognize an entailment relationship between two pieces of texts even though they do not contain common textual contents. We have learnt from the literature on RTE that they work fairly well on texts which are grammatically sound. So far we could not find any RTE research that process chat-texts. Because chat-texts are ungrammatical the existing RTE techniques would not work properly on chat-texts. An elaborated discussion on the limitations of existing RTE and a new approach to overcome them are provided in Chapter-4.

Our review of the existing works on CE detection reveals that text and language processing researchers are growing interest in this topic very recently. Hitherto a benchmark data set is not available; the researchers used varieties of text processing techniques on different data sets. Different data mining techniques including kNN, DT, RIPPER, SVM, NB, NN are used with different feature sets including terms, n-grams; sentiment, emotional and psychological markers; synonym and lexical chain using the WordNet. Some researchers also used a rule based approach. As a data-set of the CE type chats the chat-logs from the PervertedJustice.com (PJ) website are commonly used. Different researchers used the other types of chats from other different sources. Use of the CE type chats from the PJ website also have some variations: some researchers use them as a whole chat-logs, some as a conversation fragments and some as post level fragments. In our research in the beginning parts of classification task we use them as whole chat-logs and in later parts of further analysis we use them as post-level fragments. A detailed explanation of the data-sets of this research will be provided in Chapter-5.

Chapter 3

A Similarity Measure and a Clustering Approach for the Child Exploitation Domain

Child exploitation chats tend to follow a predefined pattern consisting of documented psychological communicative stages. This chapter introduces a novel similarity measure that is based on the psychological distance between a pair of chat-posts. A new clustering method is also described which collects similar posts together. We start this chapter with the construction of a CE chat and explanation of the need for a new similarity measure.

3.1 Structure of CE Chats and Need for a New Similarity Measure

The structure of a CE chat is (like most other chats) constructed by a series of posts. The posts are sentence-like text fragments or pseudo-sentences. They do not follow

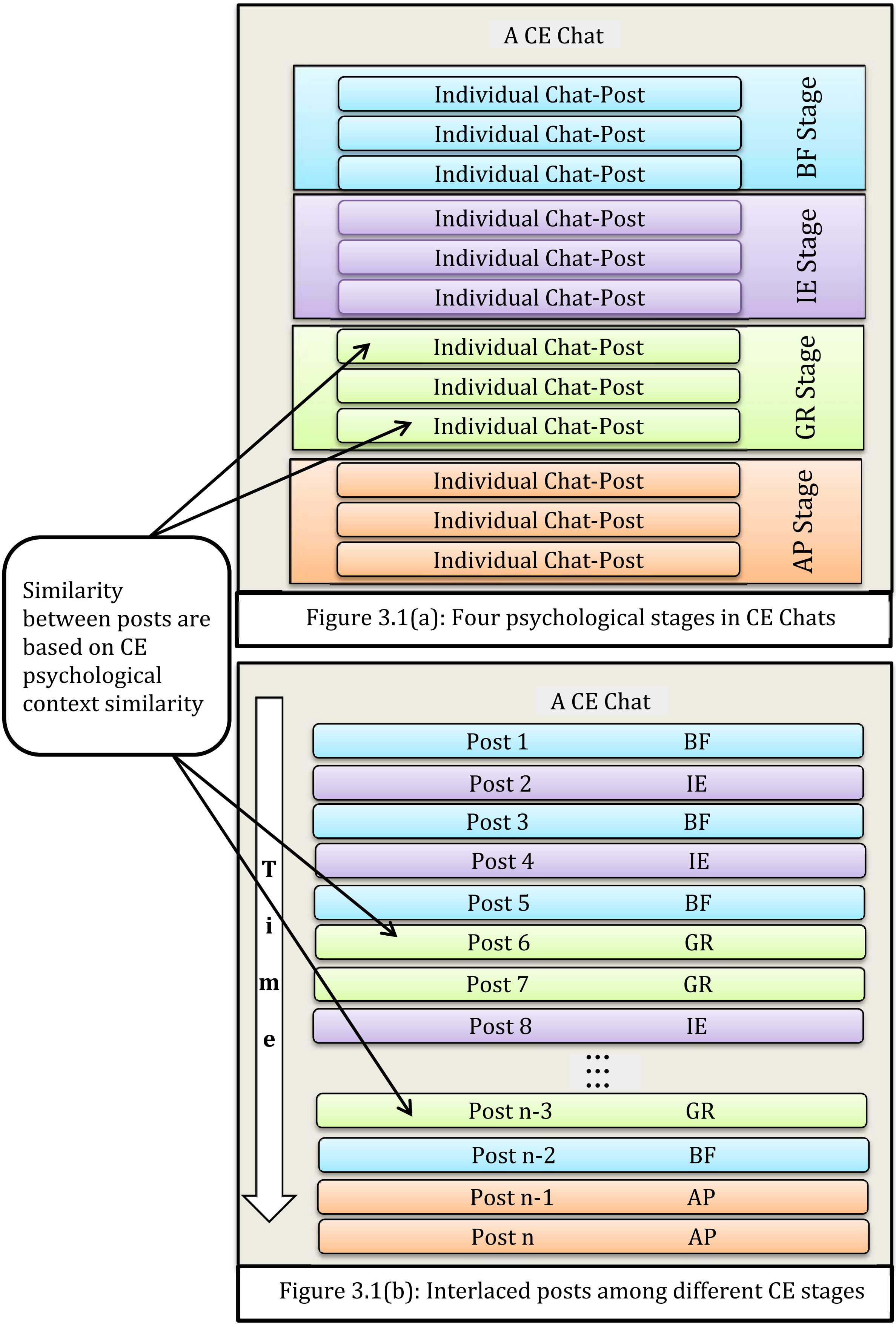


Figure 3.1(a): Four psychological stages in CE Chats

Figure 3.1(b): Interlaced posts among different CE stages

Figure 3.1: Psychological structure of a CE chat.

the structure of formal grammatical sentences. For example consider the chat posts "asl" and "h r u". Those two chat posts are very common and generally used in the beginning of a chat and both belong to the psychological befriending (BF) stage. Grammatical translation of these two posts may give the sentences "Please inform about your 'age', 'sex' and 'location' ", and "How are you".

Figure 3.1 shows the psychological structure of a CE chat. The thin bars represent the individual posts in the chat. Each post is a member of one of the four CE psychological stages Befriending (BF), Information Exchange (IE), Grooming (GR), and Approach (AP). The names of the stages are mentioned in the bars. The structural position of each post in Figure 3.1(a) is for example only, practically the posts of different stages are interlaced. An example of the interlaced posts is shown in Figure 3.1(b). The sequence of the posts generally follows the psychological behaviour of the perpetrators. A perpetrator has a natural tendency of first befriending the victim by discussing innocent things, then collecting personal information, then grooming the child to come out of the children's boundary , and if it is safe then approach for abuse. Therefore the posts also have a tendency of an overall order of BF, IE, GR and AP, however, blending the posts of different stages are also very common.

Identifying those CE psychological stages can assist in the process of detection of CE in the text of chat as they are good indicators for a chat-text being CE type. Similarities amongst the individual posts of a chat in the psychological context play an important role in differentiating as well as in identifying those stages. The chat posts rarely overlap any textual content, for example the two chat-posts, mentioned earlier, "asl" and "h r u" do not match in any textual content though they belong to the same CE psychological stage BF. For this reason, currently existing text similarity measures face difficulties in finding similarity among the chat-posts as the techniques are based on matching the characters, strings or synonym like textual contents rather than matching the psychological contexts. The documented CE psychological stages BF, GR, IE and AP are purely based on psychological behavioural contexts; not textual contents. Therefore capturing the similarity of CE psychological sense between two posts is more important in this case than finding the similarity of contents. In this situation a new similarity measure is required which is focused on CE psychological contextual sense of chat-text. We introduce a novel similarity method which we call

‘CEPsySimilarity’ measure that is capable of measuring the alikeness between two chat posts in the context of CE psychological contextual stages. The new measure depends upon the psychological contexts associated with the terms in a chat-post. For this a new ‘CE Psychological dictionary’ is constructed which contains the terms used by the perpetrators in CE chats and the CE psychological contexts associated with each term. The CEPsySimilarity measure uses the new dictionary in the background. To understand the new similarity measure one needs to understand the dictionary as well. In the following sections first we describe the construction of the CE Psychological dictionary then elaborate the techniques of the new similarity measure. After developing the similarity measure it is used to cluster the chat posts into the CE psychological phases using a novel clustering method. The construction of the new similarity measure and the results of the new clusterer are also published in our article in Miah, Yearwood and Kulkarni (2014).

3.2 CE Psychological Dictionary

To develop the proposed similarity measure we construct a dictionary to work as a background knowledge base. The dictionary is constructed by mining the terms of CE chat-posts according to their association to the psychological behavioural communication stages. We will call this dictionary ‘CE Psychological Dictionary’ or in short CEPsy dictionary.

The training chat data set (will be described in section 5.5.2.2) is used for the construction of the dictionary. All posts of one CE psychological contextual type from one predator are collected in one file and considered as a single document. Therefore, there are four different documents according to the BF, IE, GR, and AP; four different CE psychological contextual types of posts for each single predator. After separating all the posts of all the predators in their corresponding documents we calculate the document frequency (DF) for all terms in each type of post.

This DF gives us an idea of how a particular term is used by the predators. DF is actually the predators’ frequency, that is, the number of predators using a particular

Table 3.1: An excerpt of DF-table

Term	BF	IE	GR	AP	MeanDF (μ)
about	33	16	29	30	27
address	7	2	0	25	8.5
cute	25	0	8	1	8.5
dad	20	6	7	11	11
mom	20	9	6	19	13.5
mature	7	0	1	0	2
meet	13	8	3	30	13.5
kiss	7	1	25	7	10
dream	12	0	6	1	4.75
aybe	0	0	1	0	0.25
sex	3	1	35	12	12.75

term in a particular stage of exploitation. For example, if the term "cute" has DF of 25 in the BF category, and DF of eight in the GR category, then this means that 25 predators used this term for BF purposes and eight predators used it for GR purposes. We determine the DFs of all terms and collect them in a combined DF-table. An excerpt from the DF-table is shown in Table 3.1. The DF-table is used by the dictionary making algorithm to produce the final dictionary.

3.2.1 Criteria and Rationale behind the Dictionary

The following criteria and rationale are used to build the CEPsy dictionary:

Criterion 1. Criterion to exclude over-used terms:

> "A term is not included in the dictionary entry if its Mean DF $\equiv \mu > \frac{D}{2}$"
> where:
> Mean DF $\equiv \mu$ = average document frequency across the four categories
> D = Number of individual predators (that is number of chat documents in the training set)

In the training set of labelled data the number of individual predators is 48. For any particular term, the average document frequency (μ) above 24 means more than half of the predators used the term for chat-posts of all four of the category types.

Therefore, the term is very common and likely to be insignificant for differentiating among the categories. These terms should not be included in the dictionary. For example, the term 'about' in Table 3.1 has Mean DF μ = 27; which is above 24; so it is not considered for the final dictionary entry. An excerpt of the final dictionary is shown in Table 3.3.

Criterion 2. Criterion for under-used terms, typo, or mistypes:

> "A term is not included in the dictionary entry if its Max DF < M "
> where:
> Max DF = the Maximum Document Frequency across the categories
> M = a minimum threshold number

If a term is not used by a minimum M number of predators then the term is a typing error or an insignificant term. To determine a suitable value of M we varied its value from 0 to 10 and found the number of terms (N) to be included in the dictionary. This is shown in Table 3.2.

The target for the dictionary entry is to have more valid terms and less typos (terms with typing mistakes). If M is low then the number of terms in the dictionary entry (N) is high with a high number of typos. A manual analysis revealed that 4 is a suitable

Table 3.2: Number of terms (N) in the dictionary according to minimum number in MaxDF (M).

Minimum num in MaxDF (M)	No. of terms in Dictionary entry (N)
0	7996
1	7996
2	2580
3	1658
4	1234
5	982
6	827
7	707
8	611
9	530
10	478

value of M. It gives 1,234 terms in the dictionary entry which have mostly valid terms with very few typos. Therefore, for a term to be included in the dictionary entry, its maximum DF among the category types, has to be at least 4. For example the term “aybe” (in Table 3.1) has MaxDF = 1; which is less than 4; so it is an insignificant term and filtered out. Finally, Criterion 2 becomes: “A term is not included in the dictionary entry if its Max DF < 4.”

Criterion 3. Criterion for category discrimination:

3a. “A term is included in the dictionary entry if its Category DF = CaDF ≥ μ ” or
3b. “If CaDF < μ and μ > T then a term is included in the dictionary entry if its CaDF ≥ T ”

where:
CaDF = individual document frequency of a term in a category
μ = Mean DF = average document frequency across the four categories
T = A threshold number

A term requires a considerable DF for a particular category to be associated with it or to be an indicator of that category. Therefore, for a term to be defined as a particular category type, it is considered that the category DF has to be more than or equal to the average (mean) DF. This is described in Criterion 3a. For example, the term ‘dream’ has mean DF = 4.75. Any category which has DF more than 4.75 should be added into the type-list of ‘dream’. Therefore, the types of this term are BF and GR because both BF and GR have CaDF greater than 4.75. The term ‘dream’ and its category types are shown in the dictionary excerpt in Table 3.3.

Criterion 3b deals with situations where the mean DF is very high. In such situations a particular term may miss a particular category although its category DF (CaDF) is high but unfortunately lower than the mean DF. For example, consider the term ‘meet’ in Table 3.1. The Mean DF is 13.5. According to Criterion 3a the term ‘meet’ is categorized as only AP type (DF = 30 > 13.5). However, notice the CaDF of the category type BF. It is 13, which is a fairly high frequency but less than the mean frequency 13.5.

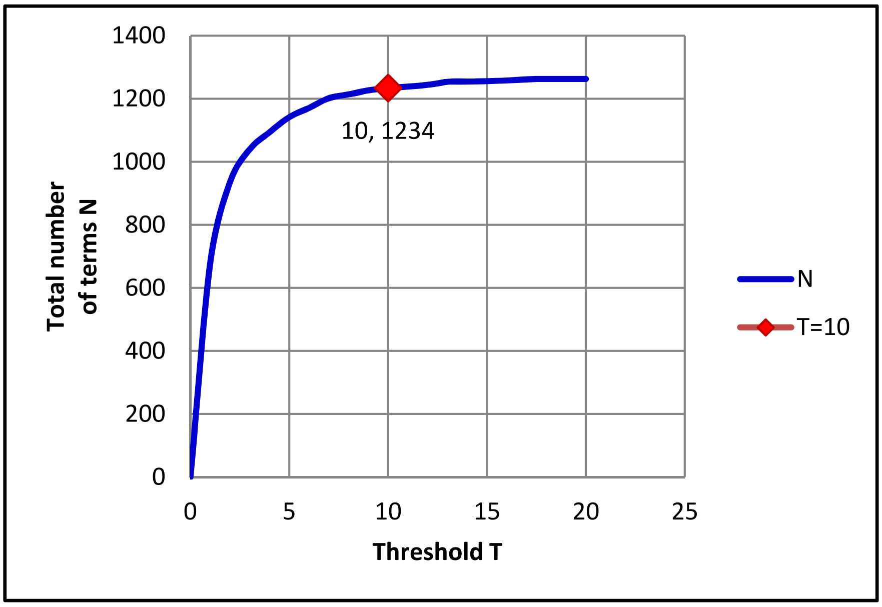


Figure 3.2: Total number of terms (*N*) vs threshold *T*.

As the document frequency of the type BF is fairly high, the term 'meet' should have BF in its category type-list along with AP. To deal with this kind of situation we define a threshold number T in Criterion 3b. If the mean DF is greater than the threshold number T then a category is allocated to a term if the CaDF of the term for that particular category is above the threshold T.

To determine the threshold number T, the following two conditions should be considered:

Condition 1: The total number of terms (N) in the CEPsy Dictionary should be as high as possible.

Condition 2: No term of the dictionary should miss any category for which it has fairly high document frequency.

Figure 3.2 shows the variation of the total number of terms (N) with the variation of threshold *T*. If the threshold number *T* is too low then most of the terms get all four categories in their category type-lists. According to Criterion 4, they are added in the stop-list, deleted from the dictionary entry, and eventually the total number of terms

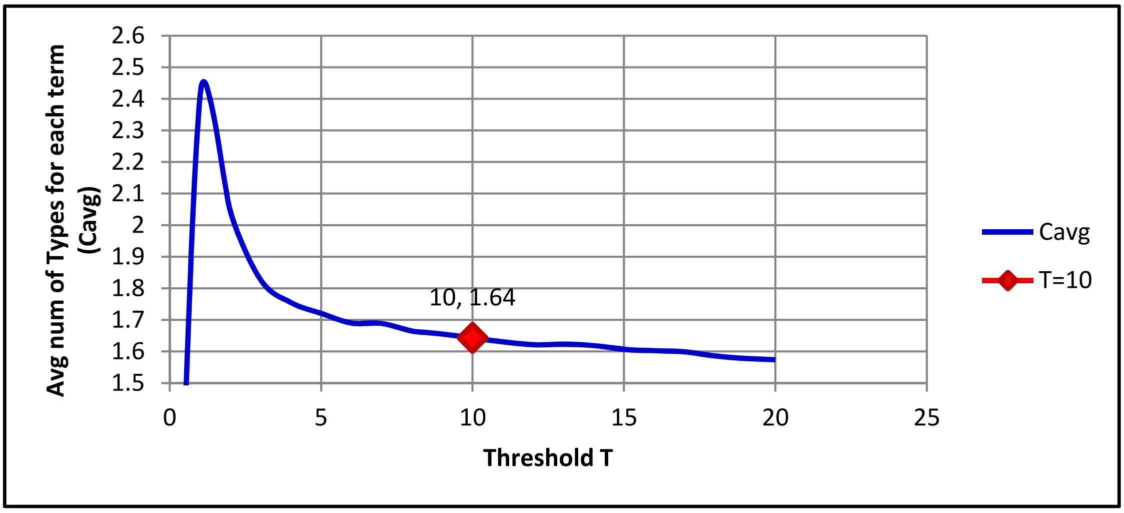


Figure 3.3: Average number of category types for each term (C_{avg}) vs threshold T.

in the dictionary is low. With the increase of T the value of N also increases. However, after a certain value of T the increase in N slows considerably. Moreover, if the threshold number T is too high, then a term will get less types (if not only one) in its type-list, which will make it difficult to find similarity among different terms. To trade-off between these two conditions we choose T = 10 and want to see how it affects the number of category types per term.

Figure 3.3 shows the average number of category types for each term (C_{avg}) against the threshold T. At T = 10 the value of C_{avg} is 1.64, which is an acceptable moderate value. To increase C_{avg} in Figure 3.3 if the value of T is decreased, then in Figure 3.2 the total number of terms N also decreases. Therefore, T is not decreased and we keep the value of T as 10. This gives a total number of terms (N) in the CEPsy Dictionary as 1,234. Finally, Criterion 3b becomes: "If CaDF < μ and μ > 10 then a term is included in the dictionary entry if CaDF ≥ 10."

Criterion 4. Criterion for stop-list:

"If a term gets all four categories in its category-list then it is added to a stop-list."

A term with all four category types should not be added into the dictionary entry. If all four categories are present then it will match up with all posts of all categories. This means that the term will not discriminate among the posts of different categories. Therefore, these terms are added to the stop-list and not included into the dictionary.

Table 3.3: An excerpt of CEPsy dictionary.

Term	→	Category
address	→	AP
cute	→	BF
dad	→	AP BF
mom	→	AP BF
mature	→	BF
meet	→	AP BF
kiss	→	GR
dream	→	BF GR
sex	→	AP GR

3.2.2 Building the Dictionary

The CEPsy dictionary is built automatically by using the DF-table along with the criteria. The dictionary is stored in a hash table. The terms of the CEPsy dictionary are the keys of the hash and the values of the hash hold the category type names of that term. In the first phase the terms are chosen for the dictionary entries. Criterion 1 and Criterion 2 are used in this phase. In the second phase the categories for each term are determined by Criterion 3a and Criterion 3b. In the third phase, using Criterion 4 the stop-list terms are deleted from the dictionary entry. Finally, the hash table is written in the output dictionary file. An excerpt from the CEPsy dictionary is shown in Table 3.3.

The newly built CEPsy Dictionary works in the background of the new CEPsySimilarity measure. The next subsection describes the construction of the new similarity method.

3.3 CE Psychological Similarity Measure

It has been mentioned in section 2.5.2 of Chapter-2 that the Li measure (Li et al., 2006) is designed to capture semantics at the sentence and short text level when used in conjunction with a semantic lexicon such as WordNet. Chat texts consist of short posts and we require a lexicon that captures the psychological sense of these short texts. We use a modified Li measure (Li et al., 2006) to calculate the similarity between a pair of chat-posts. As this similarity is measured according to posts' membership in the psychological behavioural communication stages in CE chats we call it the CE Psychological Similarity or in short CEPsySimilarity. The two posts to be compared are represented in a reduced vector space (Li et al., 2006), reduced from the common vector space used in the traditional information retrieval (IR). The common vector space represents all words of all sentences in a corpus. In the case of current research it is comparable with all terms in all chat-posts in a set of chat-logs. In the reduced vector space (RVS) the dimension *n* is reduced to the number of distinct terms in the union of the terms of the two posts. The terms which are not in the CEPsy dictionary are not included in the RVS. To measure the similarity between the pair of posts, at first two vectors $\mathbf{V_1}$ and $\mathbf{V_2}$ are constructed. These vectors represent posts P_1 and P_2 in the reduced space. Then the similarity between P_1 and P_2 is defined as the Cosine similarity (Manning, Raghavan, & Schütze, 2009) between $\mathbf{V_1}$ and $\mathbf{V_2}$.
Formally:

$$\text{CEPsySim}\ (P_1,P_2) = \text{CosSim}\ (\mathbf{V_1,V_2}) \qquad \text{... Equation 3.1}$$

The elements of $\mathbf{V_i}$ are determined as follows. Let v_{ij} be the j^{th} element of $\mathbf{V_i}$, and let t_j be the term corresponding to dimension j in the reduced vector space. There are two cases to consider, depending on whether t_j appears in P_i:

Case 1: If t_j appears in Pi, set v_{ij} equal to 1.

Case 2: If t_j does not appear in Pi, calculate a term to term similarity (CEPsyDictSim) score between t_j and each term in Pi, and set v_{ij} to the highest of these similarity scores. That is:

$$v_{ij} = \underset{x\,\in\{P_i\}}{\text{argmax}}\,\text{CEPsyDictSim}\big(t_j, x\big) \qquad \text{... Equation 3.2}$$

The CEPsyDictSim is measured by using the Jaccard coefficient (Rijsbergen, 1979) with the help of the constructed CEPsy dictionary as follows:

The CEPsyDictSim between terms t_a and t_b is:

$$\text{CEPsyDictSim}(\,t_a\,, t_b) = \frac{|A\ \cap\ B|}{|A\ \cup\ B|} \qquad \text{... Equation 3.3}$$

where A and B are the sets of corresponding entries in CEPsy Dictionary for terms t_a and t_b . For example if dictionary entries for t_a and t_b are as follows:

t_a → AP, BF and
t_b → AP, GR

then term to term CEPsyDictSim is:

$$\text{CEPsyDictSim}(t_a, t_b) = \frac{|\{\text{AP}, \text{BF}\}\ \cap\ \{\text{AP}, \text{GR}\}|}{|\{\text{AP}, \text{BF}\}\ \cup\ \{\text{AP}, \text{GR}\}|} \qquad \text{... Equation 3.4}$$

Example:

Consider the following two chat-posts:

P_1 = can we meet
P_2 = what is ur address

Both of the above posts are 'approach' type predator posts. Therefore they should have high similarity between them though there is no term overlap. A traditional similarity measure like cosine similarity will give zero similarity as there are no matching terms between the pair of posts.

In the proposed similarity measure we determine the reduced vector space as follows:

RVS (P_1, P_2) = [address, can, is, meet, ur, we, what]

The CEPsy Dictionary entries of the terms of RVS is in Table 3.4. We see that only two terms of RVS (P_1, P_2) are present in the CEPsy Dictionary. The other terms of P_1 and P_2 are actually not in the dictionary and so are not included in the RVS. Therefore:

RVS (P_1, P_2) = [address, meet]

Now the reduced spaced vectors **V₁** and **V₂** corresponding to posts P₁ and P₂ are:

$$\mathbf{V_1} = [x, 1]$$
$$\mathbf{V_2} = [1, y]$$

Table 3.4: CEPsy dictionary entries for the terms in RVS (P₁, P₂).

Term	Categories
address →	AP
meet →	AP BF

The x's and y's are calculated according to 'Case2' the corresponding term to term CEPsyDictSim that we are going to explain next. The 1's represents 'Case1' where the corresponding term in RVS is present in the related post. For example the first term 'address' of RVS is present in P₂. So the first value in the vector **V₂** is 1. However it is not present in P₁, so the first value in **V₁** is x.

This particular example has only two terms in the RVS. Therefore the term to term CEPsyDictSim x and y are same, because:

x= CEPsyDictSim (address, meet) and

y= CEPsyDictSim (meet, address)

are equal.

From the dictionary entries (Table 3.4) we find:

$$\begin{aligned}\text{CEPsyDictSim(address, meet)} &= \frac{|\{\text{AP}\} \cap \{\text{AP, BF}\}|}{|\{\text{AP}\} \cup \{\text{AP, BF}\}|} \\ &= \frac{1}{2} \\ &= 0.5\end{aligned}$$

Putting this value in **v₁** and **v₂** :

V₁ = [0.5, 1]

V₂ = [1, 0.5]

Length normalized **V₁** and **V₂**:

v₁ = [0.4472, 0.8944]

v₂ = [0.8944, 0.4472]

Now, from Equation 3.1 we get:

$$\begin{aligned}\mathrm{CEPsySim}(\mathrm{P}_1, \mathrm{P}_2) &= \mathrm{CosSim}(\mathbf{v_1}, \mathbf{v_2}) \\ &= (0.4472 \times 0.8944) + (0.8944 \times 0.4472) \\ &= 0.8\end{aligned}$$

Therefore using the new CEPsySimilarity measure the similarity between the posts P_1 and P_2 becomes 0.8 instead of 0 (zero) calculated by traditional similarity metrics. This is expected as both of the posts are of the same ‘Approach’ category type.

The example demonstrates that the similarity between a pair of chat-posts associated with the same category type is increased by the newly constructed similarity measure. To understand the improved strength of CEPsySimilarity a comparison with other similarity methods is provided in the following section.

Discriminating Power of the CEPsySimilarity Measure:

Using our new similarity measure the similarity among posts of the CE chats is measured. We generate pairwise similarity matrices for the posts of all chat-logs (one matrix for each transcript). These pairwise similarity matrices are used in the experiments for clustering. However it is difficult to manually understand the improvement in the pairwise similarity matrix if the number of posts is very high. Therefore to capture the improvement, we started with a small number of posts. Beginning with 12 posts, 3 posts from each of the 4 types; we gradually increased the number and finally randomly selected 40 posts, 10 posts from each of the 4 types BF, IE, GR, and AP. These 40 posts along with their type names are shown in Table 3.5.

Using the 40 posts of Table 3.5 as data set we measured pairwise similarity using the following methods:

1. CEPsySimilarity measure (CEPsySim).
2. Reduced vector space sentence similarity measure with LIWC dictionary (LiwcSim).
3. Reduced vector space sentence similarity measure using cosine similarity (CosSim).

Table 3.5: Randomly selected 40 posts.

Sl. No.	Chat-Post	Post-Type
1	I m 27 male from elliott county	IE
2	ur asl pls	IE
3	what area u r in?	IE
4	I m from south area	IE
5	I m 5 ft 7	IE
6	how tall r u?	IE
7	do you have a pic	IE
8	mine is on my profile if you wanna see it	IE
9	do you have any more pix	IE
10	when is your bday	IE
11	how r u :)	BF
12	dats awesome :)	BF
13	im actually kinda shy	BF
14	so u chat here a lot	BF
15	busy with college classes	BF
16	is your homework done	BF
17	i think it is dumb	BF
18	i would like to be your friend	BF
19	yeah sounds great	BF
20	its expensive these days	BF
21	do u mind if i want to know about ur body?	GR
22	are you a virgin	GR
23	whats ur bra size	GR
24	do u like dirty chats	GR
25	did you kissed your ex bf	GR
26	do u masterbate	GR
27	do u have any naked one	GR
28	do u shave	GR
29	i can teach you alot about sex if you want	GR
30	did he touch your nipples	GR
31	what is ur address	AP
32	where is mom and dad	AP
33	she could have me put in jail	AP
34	for me - it would be 25-30 yrs in jail	AP
35	how can I be sure ur are not a cop or playa?	AP
36	can we meet	AP
37	u should gimme directions	AP
38	if she leaves at 2, call me at 130 so i can be there at 230	AP
39	I'll mapquest it, what is zip code	AP
40	we go to a hotel and then make out	AP

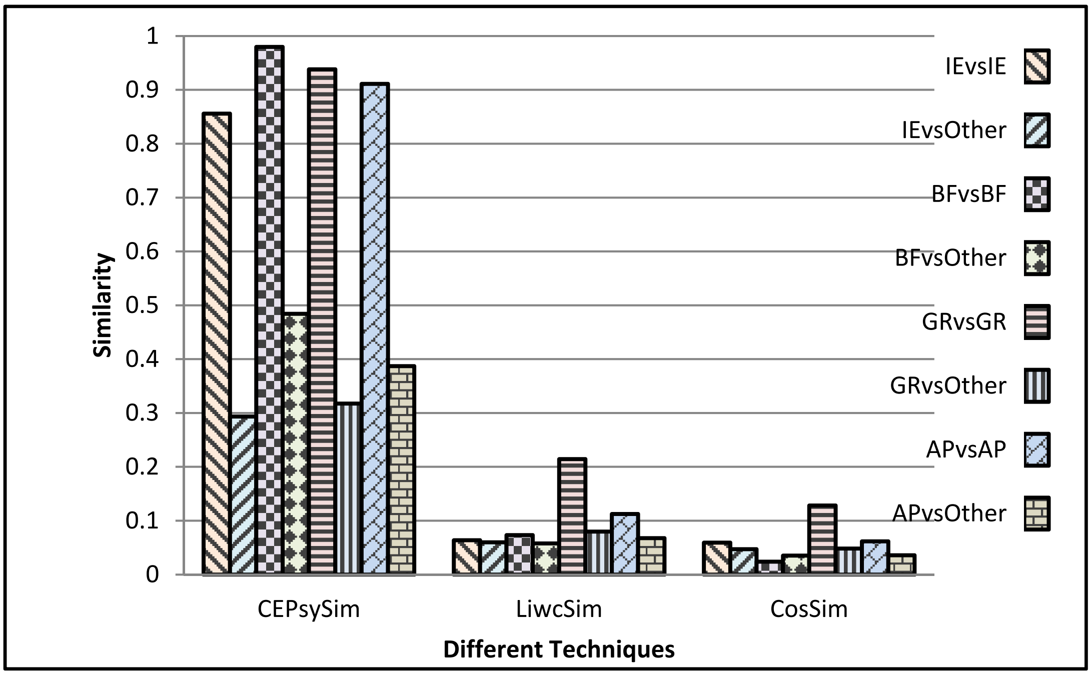


Figure 3.4: Comparison of average pairwise similarity among the chat-posts. IEvsIE means average pairwise similarity among the posts belonging to IE category (intra-category similarity); IEvsOther means average pairwise similarity among the posts belonging to IE category and other categories (cross-category similarity); the other notations express similar meaning.

The results are presented using bar charts in Figure 3.4 for the experiments to find similarity among the posts of CE chats. The bar-charts are comparing the average pairwise similarity among the posts. They show the intra-category similarity and cross-category (inter-category) similarity for each category in each of the techniques used in the experiment. The bars in CEPsySim group represent the average pairwise similarity measured using the new CEPsySimilarity method. The LiwcSim and CosSim group of bars depict the similarity measured by the corresponding techniques. The intra-category similarity and cross-category similarity are shown side by side.

For example, the IEvsIE bar expresses the intra-category similarity of IE category. It is the pairwise average similarity among the posts belong to only IE category. The IEvsOther bar describes the cross-category pairwise similarity among the posts of IE category and the posts of other 3 categories. In a similar manner the other bars represent their corresponding intra-category or cross-category similarity according to the name. Ideally the intra-category similarity should be 1, because a pair of posts belonging to the same category should have 100% similarity. On the other hand the cross-category similarity should be ideally 0, as two posts from two different categories should not have any similarity between them. From Figure 3.4 we see that the intra-category similarity for LiwcSim and CosSim is very low in the range of 0.02 to 0.2. Using CEPsySimilarity measure this is improved significantly up to the range of 0.86 to 0.98.

The strength of the CEPsySimilarity method to distinguish among the posts can be better appreciated by defining the discriminating power (δ). We define it as the difference between the similarities of intra-category and cross-category posts. That is:

$$\delta \equiv \text{intra category similarity} - \text{cross category similarity} \qquad \text{... Equation 3.5}$$

The ideal similarity between 2 posts of a same type (intra-category similarity) is 1, and the ideal similarity between 2 posts of different types (cross-category similarity) is 0; therefore, in the ideal case the value of δ is given by $\delta = 1 - 0 = 1$. A method having the value of δ close to 1 is better than a method having the value close to 0. From Figure 3.5 we see that in both LiwcSim and CosSim methods the values of δ are very low and close to 0. As a worst case scenario, the CosSim method is having a negative value for the BF type posts. That means the CosSim method is completely

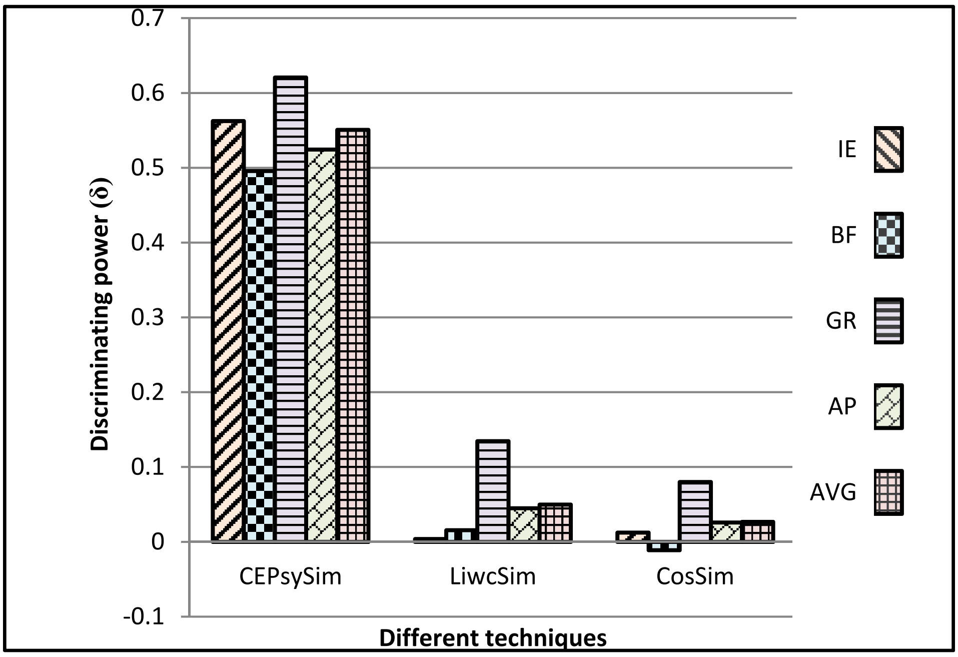


Figure 3.5: Discriminating power (δ) of different similarity techniques.

misjudging the BF type posts. It is giving higher similarity values for cross-category posts (for example: a pair of posts where one of them is BF type and other one is of another type) than the similarity values for intra-category posts (for example: a pair of posts where both of them are BF type). Using the CEPsySimilarity method the values of δ are becoming very high up to the range of 0.86 to 0.98 which is near to 1. Therefore we can say that the CEPsySimilarity method is more effectively distinguishing among the intra-category and cross-category posts.

After constructing the new CEPsySimilarity and finding its improved discriminating ability for within category texts against between category texts we use it for psychological contextual clustering of the chat-posts. The clustering technique is described in the next section.

3.4 PsyHAC Clustering

Most of the currently available research on CE detection uses a single chat-post as an instance. McGhee et al. (2011) suggest that the effectiveness of CE detection may be improved by (a) identifying and capturing context and (b) incorporating a window of text instead of using a single post as an instance. The CEPsySimilarity measure, developed in the previous section, can be used to capture the psychological behavioural communicative contexts of each chat-post and cluster the posts to collect those contextual elements into blocks of texts corresponding to the BF, IE, GR, and AP stages. These blocks of texts would be more useful in the CE detection task than the single chat-posts. Moreover if the posts of a chat-log can be automatically clustered into those four groups by an unsupervised machine learning method of clustering, then it can be evidence that the chat-log is following the CE psychological pattern. For this purpose a novel clustering method is designed. The algorithm of the new clusterer is presented in the following section.

PsyHAC Clustering Algorithm:

The newly designed clustering algorithm is based on the Hierarchical Agglomerative Clustering (Manning, Raghavan, & Schütze, 2009). The Hierarchical Agglomerative Clustering (HAC) algorithm starts by making a pairwise similarity matrix of the clustering objects. According to the amount of pairwise similarity the objects are hierarchically merged to form the required number of clusters. In the traditional HAC the pairwise similarity among the objects are measured by conventional similarity measures such as Cosine similarity or based on metrics like Euclidian distance or Manhattan distance. These conventional measures alone are not suitable to measure the contextual psychological likeness of CE chat-posts. Therefore we use the new CEPsySimilarity method to measure the pairwise psychological similarities among the objects[1]. As this modified HAC uses the CEPsySimilarity measure along with the CEPsy dictionary and is being used to cluster chat-posts into their corresponding psychological stages we call it PsyHAC (Psychological Hierarchical Agglomerative Clusterer). In the linking or merging phase of this PsyHAC the centroid measure is used.

The PsyHAC clustering algorithm proceeds in the following steps:

1. Compute an $N \times N$ pairwise similarity (pwSim) matrix using the CEPsySimilarity measure. N is the number of posts in the chat-log. For example, if the number of posts in a chat-log is 100 then the size of the pwSim matrix is 100 rows × 100 columns = 10,000 items.
2. Consider each chat-post as a singleton cluster, so there will be $N_{current}$ = N active clusters in the beginning. Store this in an active clusters list (A). For the previous example, the size of A in the beginning is 100.
3. Find a pair of clusters (say i and j) with maximum similarity excluding self-similarity and inactive clusters. Inactive clusters (I) is a list of those clusters which have been already merged with some other active clusters. So they cannot be merged again. In the beginning the size of Inactive clusters (I) is zero.
4. Merge the pair i and j into k using the centroids of i and j. The measure of centroid is mentioned in the next section.

[1] chat-posts act as clustering objects in this work

5. Append pair i and j to inactive cluster list (I).
6. Update active cluster list (A) by deleting i , j and appending k. So the new number of active clusters will be $N_{Active} = N_{Current} - 1$. That is, for the previously mentioned example, after the first iteration $N_{Active} = 100 - 1 = 99$. Set $N_{Current}$ with the new N_{Active} .
7. Recompute the new $N_{Current} \times N_{Current}$ pwSim matrix using the centroid similarity measure. For the previously mentioned example, the size of pwSim matrix after first iteration is 99 × 99 = 9801.
8. Repeat step 3 to step 7 until $N_{current} = N_{Expected}$ (Expected number of clusters). In this current research $N_{Expected} = 4$ as there are four category types of posts in a CE chat-log. Therefore the repetition will go until the condition $N_{current} = 4$ is satisfied.

Measure of Centroid Similarity:

To merge two clusters (in step 4 in the PsyHAC algorithm) we use the 'centroid similarity measure' explained in section 2.5.6 in Chapter2. For convenience here we rewrite the formula of the measure of centroid similarity (Equation 2.30). For two clusters w_i and w_j the centroid similarity SIM-CENT (w_i, w_j) is given by:

$$\begin{aligned}\text{SIM-CENT}(w_i, w_j) &= \boldsymbol{\mu}(w_i) \cdot \boldsymbol{\mu}(w_j) \\ &= \left(\frac{1}{N_i} \sum_{\boldsymbol{P_m} \in w_i} \boldsymbol{p_m}\right) \cdot \left(\frac{1}{N_j} \sum_{P_n \in w_j} \boldsymbol{p_n}\right) \qquad \text{... Equation 3.6} \\ &= \frac{1}{N_i N_j} \sum_{\boldsymbol{P_m} \in w_i} \sum_{\boldsymbol{P_n} \in w_j} \boldsymbol{p_m} \cdot \boldsymbol{p_n}\end{aligned}$$

Where:

w_i = Cluster i; w_j = Cluster j

N_i = Number of documents in cluster $\mathrm{w_i}$

N_j = Number of documents in cluster $\mathrm{w_j}$

$\boldsymbol{P_m}$ = Post belong to cluster $\mathrm{w_i}$

$\boldsymbol{P_n}$ = Post belong to cluster $\mathrm{w_j}$

$\boldsymbol{\mu}(w_i)$= Centroid of cluster $\mathrm{w_i}$

$\boldsymbol{\mu}(w_j)$ = Centroid of cluster $\mathrm{w_j}$

$\boldsymbol{p_m}$ = Length normalized vector of post $\boldsymbol{P_m}$

$\boldsymbol{p_n}$ = Length normalized vector of post $\boldsymbol{P_n}$

Complexity Analysis of PsyHAC Algorithm:

In step 1, to compute an N × N pairwise similarity matrix the algorithm executes a loop for N × N times. Therefore the time cost is C1 (N × N); where we assume that C1 is the cost of each iteration of the loop and N is the number of iterations. From step 6 and step 7 we can see that the algorithm recomputes the N × N matrix for N−1 number of times. Therefore the time cost for those two steps is:

$$C_6(N-1)\ C_1(N \times N) = C_{16}(N^3) - C_{16}(N^2)$$; where C1 × C6 = C16 , a constant.

This gives the time complexity of $O(N^3)$ for those steps. The other steps of the algorithm cost ≤ $O(N^2)$.

Therefore the time complexity of the algorithm eventually becomes $O(N^3)$.

3.5 Chapter Summary

This chapter introduced the theoretical aspects of the proposed CEPsy Similarity measure. It also describes the construction of the CEPsy Dictionary which works in the background of the new similarity measure. The dictionary is built by mining the terms associated with the CE psychological contextual stages of chat-texts. For the CEPsy Similarity measure between two chat-posts, first a reduced vector space is formed with all the distinct words present only in the pair of the corresponding chat-posts. Then for each chat-post a reduced vector is derived from the reduced vector space with the assistance of the CEPsy Dictionary. Finally, the cosine similarity between the reduced vectors of the chat-posts is computed as the CEPsy Similarity measure. The strength of the similarity and the discriminating power is improved by the new measure in the CE psychological contextual domain. A comparison of the discriminating power with other methods is also provided here.

Without any supervision if the posts of a chat-log automatically organise themselves into the four CE psychological stages then that will be an evidence that the chat-log is following the behavioural pattern of a CE chat. This can be investigated by using an unsupervised machine learning method of clustering. The algorithm of a new clustering method PsyHAC is also explained in this chapter. Using the CEPsy Similarity measure a pairwise similarity matrix is computed for all the posts of a chat-log. At the beginning each of the posts works as a singleton cluster. The clusters are progressively merged according to their highest centroid similarity until four clusters remain corresponding to four CE psychological stages BF, IE, GR and AP. For testing the effectiveness of the new clusterer experiments are carried out and results and analysis will be presented in Chapter-6.

Chapter 4

Soft Entailment for the Child Exploitation Context

Finding CE evidence in the text fragments of a chat is not trivial. The term content of a chat-post does not match with the term content of an evidential statement. This makes it difficult for existing techniques to find the evidence. To solve this problem a new approach is required that can entail the CE evidential contexts by analysing the content of chat-text fragments. This chapter introduces a new soft entailment system for that purpose. Before describing the new approach a brief discussion is provided about why an entailment technique is required for CE evidence finding and how the proposed entailment system differs from the existing systems.

4.1 Why Entailment in CE detection?

In forensic science, the famous Locard's Exchange Principle states "Every contact leaves a trace". (cited in: Horswell, 2004; Walls, 1968; James *et al.*, 1980; Eltzeroth and Elzerman,1981). This principle can be taken to describe the complicated relationship between the perpetrator's chat message and evidence. A CE chat message certainly contains the traces of CE evidence. To detect those traces of evidence

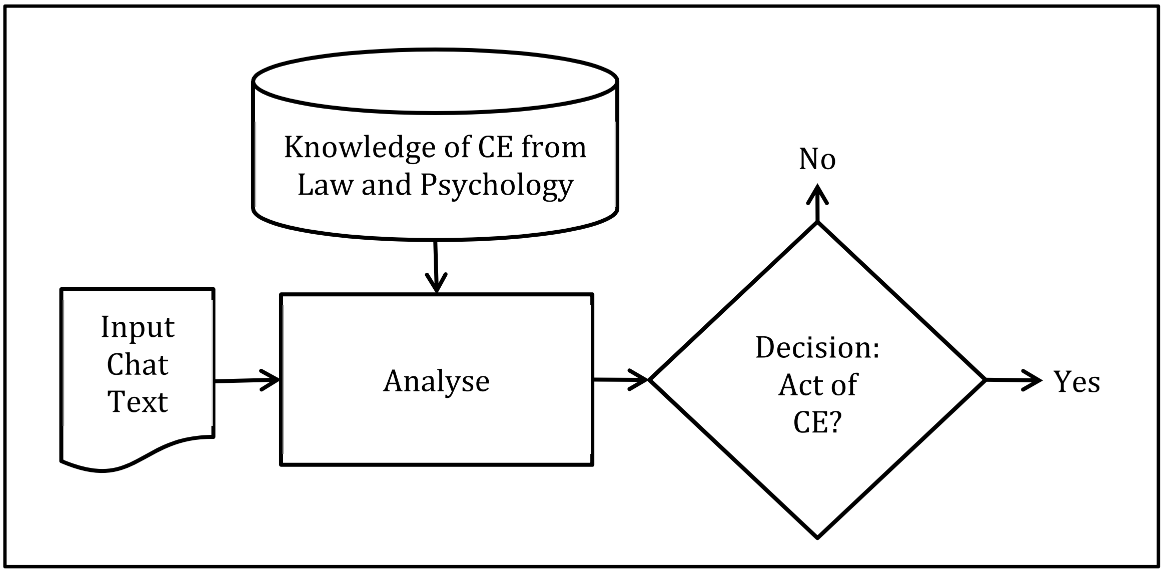


Figure 4.1: Procedure of a system to decide whether a chat contains the act of Child Exploitation.

from the chat-log we need to locate the text excerpts that would constitute evidence of certain criminal activities of child-exploitation defined by the experts of psychology and law.

Figure 4.1 shows an overall procedure to decide on a chat whether it provides evidence of CE act. It requires background forensic knowledge of CE from different fields including law and psychology to make a correct decision. For example the NSW Crimes Act 1900 defines the CE offence as "procuring or grooming a child under 16 years for unlawful sexual activity". The terms in this definition work as an example of the content of a CE evidential statement. The exact matching of these words would not be found in the chat because the perpetrators use completely different words to groom a child. For example an adult is asking a child through chat-text: "are you a virgin". Any concerned person will be alarmed by this act and understand that this could be a part of child grooming process though the text fragment does not have any match with the above legislative definition of child grooming. It has been discussed in the literature review chapter that the perpetrators tend to follow the behavioural phase by phase grooming process. Some of the grooming phases even contain innocent words by which the offender pretends to become an innocent friend of the child. Under these circumstances it is very unlikely that the evidence would be found with currently existing information extraction systems based on matching traditional textual features like strings, words or synonyms. As the chat messages are fragmented the textual features of CE evidence are seldom matched with the traces of chat-texts left behind by the predator leading to difficulties to find any evidence. It has been previously discussed in Chapter-2 that the predators follow a psycho-communicative pattern in CE chats. Therefore investigating the psychological communicative traces in the suspected chat may lead to a successful CE evidence detection. For this purpose a system is required that can relate the perpetrator's chat message to the evidential propositions without depending on matching only the textual features. A textual entailment (de Salvo Braz et al. 2006; Esteva et al. 2010) system that does not depend only on traditional text-matching can be a desirable system in this case.

It has already been mentioned that a Recognition of Textual Entailment (RTE) system entails a predefined 'hypothesis' (H) by a 'text' (T). The 'hypothesis' (H) is a piece of text and the 'text' (T) is another piece of text. Despite the fact that the two different pieces of texts (H and T) may not share any common lexical terms, an effective RTE system would be able to find an entailment relationship between them. Based on this idea of "finding a relationship between two apparently lexically disjoint texts" we are proposing a new "Soft Entailment" approach specifically for CE evidence detection. The following section puts some light on the similarity and dissimilarity between the existing RTE systems and the proposed new approach.

4.2 Hard vs Soft Entailment

Formal RTE systems require robust linguistic analysis techniques for the entailment task. Those techniques include efficient parser, named entity recognizer (NER), a semantic thesaurus and sometimes expensive logical knowledge-bases. We assume the output entailment of those systems are logically strong, therefore we call the existing formal RTE systems 'strong' or 'hard' RTE. On the other hand our approach does not depend upon expensive linguistic analysis systems. It uses similarity based inference of CE psychological contexts. In contrast with the strong textual entailments it rather provides an approximation of entailment for CE psychological contexts; hence the term "Soft Entailment" is associated with our approach.

The inputs of the traditional hard RTE system are formal texts. Those texts are usually compiled by professionals and contain grammatically sound sentences. Grammatical soundness is the key criteria for the correct outputs by the linguistic analysers and eventually for the RTE system. As the chat-posts are highly ungrammatical, a hard RTE system will face considerable difficulties at different levels of linguistic analysis and eventually fail to provide a correct entailment result. The first step in a hard RTE is to parse the sentence. In that very first step existing parsers find it difficult to correctly parse an ungrammatical chat-post. In our preliminary experiments, some chat messages have been tested in the link grammar parser (LGP; Sleator et al., 2004). Sometimes LGP returns the whole fragment with no parsing information as a chat-post is not a grammatical sentence and sometimes gives partial results that may not be useful for entailment. The parsing problem of a chat-post by another good parser is shown in Figure 4.2.

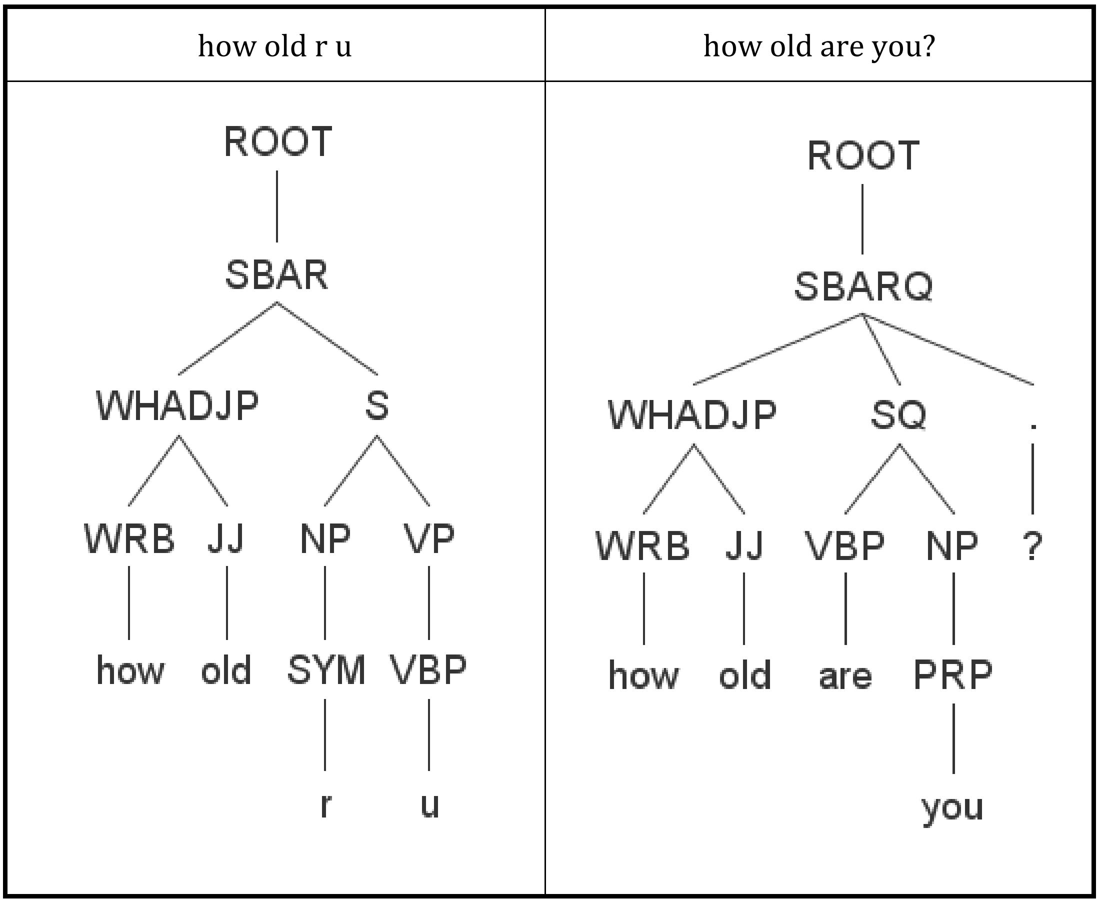


Figure 4.2: Parse tree output from Stanford parser for two strings with the same meaning.

Figure 4.2 shows the parse-trees for two strings with the same meaning. The parse-trees are produced by the Stanford Parser (Klein and Manning, 2003) using the probabilistic context free grammar (PCFG). One of the two strings is a grammatical sentence "how old are you?" and the other one is its chat-post version "how old r u". From the Figure 4.2 we see that even a good parser which is using a strong parsing module PCFG, fails to correctly parse a very common chat-post. When the chat post is written correctly in a grammatical form, the same parser provides a correct parse tree.

The other linguistic analysis tools like named entity recognizer (NER) and semantic thesaurus depend upon correct parts of speech information of the input terms. With the flaws in the output of the parser for the terms of chat-posts a semantic database like WordNet (Fellbaum, 1998) or a named entity recognizer like Stanford NER (Finkel, Grenager and Manning, 2005) will also fail to provide correct output. A knowledge base that encompasses all information associated with terms in chat-posts is currently unavailable and will be expensive and time consuming to build one. For these reasons a traditional 'Hard' RTE is currently unable to handle the ungrammatical chat-posts. Moreover the focus of existing hard RTE is to entail the meaning of two texts by matching the semantic or logical content, whereas, the main goal in the current research is to capture the evidential traces of psychological CE context in chats. Consequently a hard RTE will miss the point.

Our approach does not depend upon correct parsing or formal linguistic analysis; it can handle the informal chat-posts. In addition it is focused on detection of CE context. It uses the CEPsy Dictionary (discussed in Chapter-3) which provides CE psychological contextual information associated with each term in a chat-post regardless of its grammatical significance. This makes the system flexible and not tied on the strict grammatical structures of formal sentences and gives it the capability to handle unstructured chat-posts.

Following section explains the logics for the approach of soft entailment.

4.3 Logics for Soft Entailment

Our approach of 'Soft' entailment uses similarity-based inference. In logical settings similarity based reasoning has a good number of implementations. For example, Ruspini (1991) used similarity-based reasoning to define semantics for fuzzy sets based on fuzzy similarity relations. Dubois et al. (1997), Esteva et al. (1997), and Godo et al. (2008) also employed this kind of reasoning from a logical perspective. Based on the 'Similarity-based inference' a fairly new entailment logic named 'Approximate Entailment' is proposed by Esteva et al. (2012).

The 'Approximate Entailment' (Esteva et al., 2012) is a generalized mathematical theoretical proposition. Though we did not start our work with the theory of 'Approximate Entailment', however after completing our experiments we found that it has some similarity with our work in the outer level of overall logical and mathematical framework. Nevertheless it requires further modification in the inner level to fit in the domain of CE Psychological chat-text. Our approach of 'Soft' entailment incorporates the following logic:

Let W be a set of possible worlds. For the current problem of CE detection W represents the sets of propositions for the four CE psychological phases BF, IE, GR, and AP. All the chat-posts of the perpetrators are also included as individual propositions in W. The propositions are modelled classically by sets of possible worlds, that is, by subsets of W (capital W). Let $A \subseteq W$ model a proposition; for example A represents a proposition constituted by a single or a group of chat-posts of a perpetrator. We write $w \vDash A$ to express that the text content represented by A holds in the world w (small w), that is, $w \in W$. Let us consider another proposition, for example "A perpetrator is involved in child-grooming activity " belonging to the psychological phase GR. We denote the new proposition by $B \subseteq W$. If B also holds in the world w, for the classical implication we would write:

For any $w \in W$, $w \vDash A$ implies, $w \vDash B$. ... Equation 4.1

This is the classical entailment relationship, therefore we write $A \vDash B$ or "*A entails B*" for Equation 4.1.

The idea of 'soft' entailment is to formalise this implicational relationship to hold only approximately. We write $w \vDash^{\theta} B$, where[1] $\theta \in [0,1]$, to express that w is not necessarily a world in which B holds, but there is a world v such that $v \vDash B$ and the similarity between w and v is denoted as $S(w, v) \geq \theta$. At this level Esteva et al. (2012) did not provide any explanation of how the similarity would be measured between the two worlds. We propose for the CE Psychological domain the similarity S can be measured using CEPsy Similarity. Then, we say that A implies B to the degree θ if:

For any $w \in W$, $w \vDash A$ implies, $w \vDash^{\theta} B$ . ... Equation 4.2

That is, for any world in which A holds is CE Psychologically (CEPsy) similar with the amount of θ to a world in which B holds, for Equation 4.2 , we write $A \vDash^{\theta} B$ or "*A softly entails B*".

To explain the above mentioned logics for soft entailment the following example can be used.

Consider A to be a piece of chat-text, for example "are you a virgin". A world w holds this piece of chat-text. That is $w \vDash A$. Let us assume B is another piece of chat-text, say "the perpetrator involves in child-grooming". It is an evidential proposition. There exists a world v that holds the proposition B. The world v also includes all the chat-posts which have been accepted and labelled as grooming evidence. Obviously the proposition B does not hold in the world w as there is no lexical overlap between "are you a virgin" and "the perpetrator involves in child-grooming". That is $w \nvDash B$. However there exists a CEPsy similarity $S(w, v)$ between the two worlds w and v. To express a similarity threshold we use the notation θ, with a value between 0 and 1. If $S(w, v) \geq \theta$ then we say that B softly holds in w with the proximity of θ or $w \vDash^{\theta} B$. Therefore finally from Equation 4.2 we write $A \vDash^{\theta} B$ or "*A softly entails B*".

[1] Theta (θ) is not binary but a value ranges from zero to one.

4.4 Our Approach to Soft Entailment

A new "CE Psychological Domain Vector Space Model" is constructed as the core component of our approach to soft entailment. The main purpose of this approach is to locate CE evidence in chats. We will call this approach the "Recognition of CE Entailment (RCE)". This section describes the construction of the "CE Psychological Domain Vector Space Model" followed by the explanation of the procedure of RCE.

4.4.1 Construction of the CE Psychological Domain Vector Space Model

The conventional Term Vector Space Model (Salton, Wong and Yang, 1975; Manning, 2009) heavily depends on term overlapping. It is very unlikely that texts in different chat-posts share common terms because term overlapping is very rare in chat-posts. Therefore a conventional Term Vector Space Model (TVSM) is seldom useful in processing the posts of chat-logs. Utilizing the CEPsy Dictionary the Term Vector Space Model (TVSM) can be transferred into a dimensionally reduced CEPsy Domain

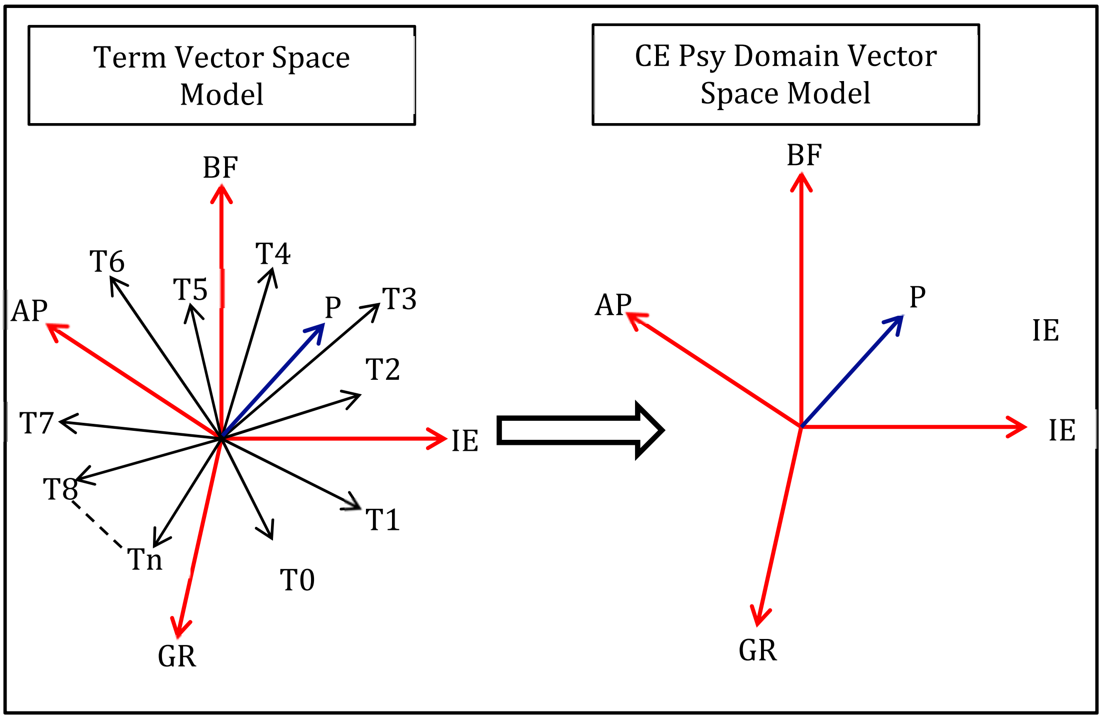


Figure 4.3: Term vector space model to CE Psychological domain vector space model. P is a test chat-post in the vector space.

Vector Space Model (CEPDVSM) as shown in Figure 4.3. Each dimension of the CEPDVSM represents one of the four CE Psychological contexts BF, IE, GR or AP. For the transformation of TVSM to CEPDVSM we need a new term weighting measure for finding term importance in the CE domain. The procedure of the construction of a new term weighting measure is described below.

4.4.1.1 Construction of a New Term Weighting Measure

The number of predators using a particular term in a particular CE psychological category is proportional to the importance of that term to represent that category. On the other hand the more a term appears across the categories the less strength it will have to discriminate among the categories, that is the importance is proportional to the inverse of the category frequency. Therefore expressing the predators' frequency as PF and the inverse of category frequency as iCF and multiplying these two notions we get the expression $PFiCF$. This would give a new term weighting measure for finding term importance in the CE domain.

Using the training data set (the data set will be explained in section 5.5.2.2 of Chapter-5) we compute the new term weighting measure for each and every term in the term vector space model. The computation proceeds in the following steps:

Step 1: Using the training-set construct the CEPsy Dictionary. The methodology to construct the CEPsy Dictionary is described in Chapter-3.

Step 2: Determine the Predators Frequency (PF) for each type of post for each term and construct a term PF vectors. For example consider the term 'meet'. Its term PFs across the four CE psychological categories are computed as below:

Term ->	BF	IE	GR	AP
meet ->	13	8	3	30

Step 3: Make Crossed PF (CPF) term-vectors by crossing the PF term-vectors with the CEPsy Dictionary in the following way:

In the CEPsy Dictionary each term is associated with some particular psychological types. For a term the PFs of the types other than the types defined in the CEPsy Dictionary are usually very low and increase noise if taken into consideration. Therefore, in the CrossedPF-term-vectors for a particular term we keep only those PF values from the PF term-vectors whose type matches with the types defined in the CEPsy Dictionary, the PFs of the absent types are made zero. For example the term ‘meet’ would have:

PF-term-vector:

Term ->	BF	IE	GR	AP
meet ->	13	8	3	30

CEPsy Dictionary entry:

meet -> BF AP

CrossedPF-Term-vector:

Term ->	BF	IE	GR	AP
meet ->	13	0	0	30

Step 4: To compute the Inverse Category Frequency (iCF) we consider the following hypothesis:

The Category Frequency is defined as the number of categories for which the PF values are non-zero in the entry of CrossedPF term-vector; iCF is the weight of importance of a term. If a term has all 4 non-zero categories, that is, if its CF is 4, then, it actually does not have any power to differentiate among the categories. Therefore the weight of its importance (iCF) should be the lowest (zero). On the other hand if CF is 1 then iCF should be the highest (that is 1). Figure 4.4 shows how the iCF can be computed against the CF.

From the straight line graph in Figure 4.4 we get:

$$\mathrm{iCF} = mx + c = -\left(\frac{x}{3}\right) + \frac{4}{3} \qquad \text{... Equation 4.3}$$

where:

x = CF = number of categories with a non-zero value for a particular term in its entry in the Crossed PF term-vectors.

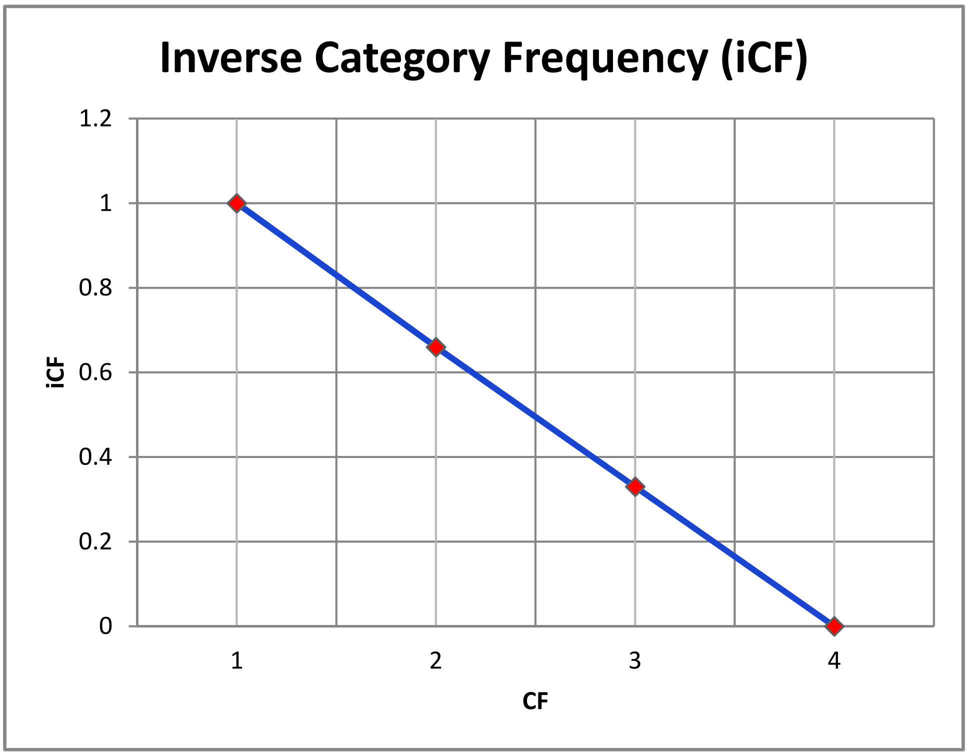


Figure 4.4: Determining inverse category frequency (iCF).

Step 5: Multiply each PF values in the CrossedPF term-vector with the corresponding iCF value for each term. This will give the expected expression of the new term weighting measure PFiCF crossed with the CE Psychological dictionary. For convenience we call this measure as "Crossed Predator Frequency inverse Category Frequency" (CPFiCF). For the previously mentioned term 'meet' the CPFiCF is computed as below:

Crossed-PF-Term-vector:
Term -> BF IE GR AP
meet -> 13 0 0 30

For the term 'meet' the number of categories with a non-zero entry is 2. Therefore:

$$
\begin{aligned}
\text{iCF} &= -\left(\frac{x}{3}\right) + \frac{4}{3} \\
&= -\left(\frac{2}{3}\right) + \frac{4}{3} \\
&= 0.66
\end{aligned}
$$

Multiplying the iCF we get the CrossedPFiCF term-vector as:

Term ->	BF	IE	GR	AP
meet ->	8.6	0	0	19.8

Comparison between CPFiCF and classical TFiDF:

The idea of Crossed PFiCF (CPFiCF) has some similarity to the classical idea of TFiDF in the sense of implementation. The TF is directly proportional to the ability of a term to indicate the class of a document and the DF is inversely proportional to the importance of the term. In the same way, increase of PF increases the probability of a term to be in a category, whereas increase of CF decreases the probability for that term to be an important one. Apart from this similarity the CPFiCF and the TFiDF have the following subtle differences:

1. In the current case, we consider a pseudo-sentence like chat-post as an instance (or as a "document" in the terminology of classical TFiDF). Therefore TF (Term Frequency) is the number of appearance of a term in a chat-post. However PF is not the same as TF, instead it is the number of distinct predators using a term in a particular psychological category. More generally speaking, PF is the 'number of authors' using a term in their writings of a particular type of documents.

2. The classical DF (Document Frequency) in the current problem would be the number of chat-posts (documents) containing a particular term. However, CF represents a different meaning. It is the number of psychological categories where a particular term is used irrespective of the number of chat-posts it appears in. That is, if speaking in the TFiDF terminology, CF is the number of 'types' of documents in which a particular term appears, not the number of 'documents'.

3. After computing the PFiCF the Crossed PFiCF (CPFiCF) is obtained by doing AND operation with the CEPsy dictionary. In the classical TFiDF nothing is crossed with it.

4.4.1.2 Transforming the Term Vector Space Model to the CE Psychological Domain Vector Space Model

The CrossedPFiCF measure computes the values projected by any term of the Term Vector Space Model (TVSM) on the Psychological Domain Vector Space Model (CEPDVSM) as shown in Figure 4.3. Therefore by computing CrossedPFiCF we are actually transforming the high dimensional TVSM into reduced dimensional CEPDVSM. Placing a chat-post in the new CE Psychological Domain Vector Space now we can compute its vector components for each of the CE psychological contexts. The computation procedure is described in the next section.

4.4.1.3 Computing Contextual CEPsy Vector Components

The following examples show how contextual CEPsy vector components are computed using CEPDVSM. Consider two chat-posts P1 and P2 to be from two different CE contextual dimensions. The post P1 is from the AP context and the post P2 is from the GR context. If the contextual CEPsy vector components are correctly computed by using CEPDVSM then the context vector relevant to a post should be the highest among all the context vectors computed for that particular post. That is, for the post P1 the contextual CEPsy vector component of AP context should be the highest among all the context vectors computed for P1; and for P2 this should be highest for the GR context. An excerpt of the table corresponding to the CE Psy Domain Vector Space Model (CEPDVSM) is shown in Table 4.1.

Each term in the table is a vector having elements (values) from the four dimensions of the CEPDVSM. Each dimension of the CEPDVSM represents each of the four CEPsy contexts of BF, IE, GR and AP. The table is used for the computation of the context vectors of posts P1 and P2. The columns BF, IE, GR and AP of the table represent the corresponding dimensions in the vector model shown in Figure 4.3.

Table 4.1: Excerpt of the table corresponding to the CE Psy Domain Vector Space Model (CEPDVSM)

Term	BF	IE	GR	AP
:(	4.62	0	0.33	0.99
:)	0	0	0	0
:->	1.32	0	0	0.66
a	0	0	0	0
able	2.97	0	0.66	3.3
about	0	0	0	0
above	1.98	0	1.98	0
abovt	0	0	0	1
address	2.31	0.66	0	8.25
is	0	0	0	0
meet	8.6	0	0	19.8
r	0	0	0	0
u	0	0	0	0
ur	0	0	0	0
virgin	0	0	18	0
what	0	0	0	0

Example1: P1 = what is ur address

Using CEPDVSM excerpt in Table 4.1 the vector components of P1 for CE contexts are computed as below:

1. vector component in the IE dimension = sqrt($0^2 + 0^2 + \ldots + (0.66)^2$) = 0.66
2. vector component in the GR dimension = sqrt($0^2 + 0^2 + \ldots + 0^2$) = 0
3. vector component in the AP dimension = sqrt($0^2 + 0^2 + \ldots + (8.25)^2$) = 8.25

Example2: P2 = r u a virgin

Using CEPDVSM excerpt in Table 4.1 the vector components of P2 for CE contexts are computed as below:

1. vector component in the IE dimension is = sqrt($0^2 + 0^2 + \ldots + 0^2$) = 0
2. vector component in the GR dimension is = sqrt($0^2 + 0^2 + \ldots + (18)^2$) = 18
3. vector component in the AP dimension is = sqrt($0^2 + 0^2 + \ldots + 0^2$) = 0

From the values of computed vector components of P1 and P2 it can be seen that the vector component for P1 is highest in the AP dimension and for P2 it is highest in the GR dimension which are expected as the relevant CE contexts of the posts.

4.4.1.4 Use of the CE Psychological Domain Vector Space Model

Using the model of CEPDVSM we have developed a soft entailment approach called RCE that entails the psychological aspects of child exploiting chat-texts. The soft entailment approach is described in the following sections. Apart from that the CEPDVSM can also be used to classify the chat-posts. Ranking the posts within a particular category can also be done by the vector components obtained from CEPDVSM.

4.4.2 Recognition of CE Entailment (RCE)

The main target of our approach of soft entailment is to find the entailment relationship between a suspected piece of chat-text and the evidential hypotheses. By the evidential hypotheses we mean the defining statements of CE act provided by the legal system or by the psychological researchers. Some examples of the evidential hypotheses are shown in the list of propositions in Figure 4.5. The first three propositions are derived from the CE psychological contextual phases defined by psychological and communicative researchers. The last two propositions are derived

1. “A perpetrator is involved in child-grooming activity ”. (GR type)
2. “Valuable personal information has been exchanged between the perpetrator and the victim”. (IE type)
3. “An approach has been made to physically meet the victim”. (AP type)
4. “The acts of the suspected adult raises concern to a reasonable person who cares for the child”. (Legal proposition 1)
5. “An adult is procuring or grooming a child under 16 years for unlawful sexual activity”. (Legal proposition 2)

Figure 4.5: A list of examples of the evidential hypotheses.

from the legal definitions of child grooming. The psychological and legal definitions of CE act have been mentioned in Chapter-2.

It is very unlikely that the current existing technology would allow building such a hard RTE system using the chat-post texts that can textually entail the hypotheses listed in Figure 4.5. Therefore a new soft entailment system is required. As have been mentioned before, we call our approach of soft entailment as the "Recognition of CE Entailment (RCE)". The following section describes the overall algorithm of the RCE approach.

Algorithm of the RCE approach:

Consider a chat-post "r u a virgin" to be the target test text T. For T to be a CE evidence it requires to entail any of the evidential hypotheses listed in Figure 4.5. Consider H to be the evidential hypothesis "A perpetrator is involved in child-grooming activity"; enlisted at serial number 1 in the list of Figure 4.5. The hypothesis (H) is to be entailed by the text (T). According to the definitions of CE psychological phases mentioned in Chapter-2 one can understand that this particular hypothesis H comply with the grooming (GR) context. That is, H is a member of the GR world. Now if a system can determine that T is also a member of the world GR or a world similar to GR then according to the soft entailment logic in section 4.3 an entailment relationship can be established between T and H. To accomplish this task a set of example texts is required to teach the system about the world of hypothesis H. These example texts would work as surrogates of H. Using the CEPDVSM the set of surrogates is chosen. The CEPsy contextual similarity is measured between the target text T and each member of the set of surrogated hypothesis. If the average amount of CEPsy contextual similarity of the set is above a pre-defined threshold then we consider that "T entails H". Detailed step by step procedure of this approach is explained below.

The following major steps are used for the process of the RCE approach:

1. Make a set of surrogates corresponding to a CE evidential hypothesis.
2. For each and every post of a suspected chat-log find soft entailment using the surrogates of the hypothesis. A suspected chat-post is considered to be CE evidence if it softly entails any of the CE evidential hypotheses.

Details of each step are explained next.

Making Surrogates of Evidential Hypothesis:

Using the psychological definitions human annotators labelled the chat-posts of the data-set. Each of the chat-posts is labelled as one of the CE psychological contextual (BF, IE, GR, or AP) types. Therefore those annotated chat-posts can be used as the surrogated texts to represent the hypotheses constituted by the legal or psychological definitions. The surrogated texts work as the training examples for the soft entailment approach. This is similar to the training of a child in the 'language-game' in 'Philosophical Investigations' of Wittgenstein (1958). To understand a language-object a child is usually trained with discrete limited words in a primitive language. Similarly we train our system with terms of limited chat posts; this may not an appropriate training to understand and use the whole English language but may work only for the narrowly circumscribed region of CE chat-language.

Figure 4.6. shows the algorithm to make surrogates of hypotheses. The inputs are a set of all types of chat posts (P) and a set of hypotheses (H). Each of the hypotheses in the set H belongs to one of the psychological context of IE, GR and AP. For each hypothesis (h) the algorithm collects posts from P to work as a set of surrogates representing h. First, it collects all chat-posts related to the psychological context of h into a list A. For example, if h is the evidential proposition "a perpetrator is involved in child-grooming activity " then it collects all chat posts labelled with the related context of Grooming (GR) activities.

```
Input: P and H;
       P = A set of all types of chat posts.
       p ∈ P
       H = A set of hypotheses belonging to Psychological
           definitions of IE, GR and AP
       h ∈ H

Output: S
       S = A set of chat posts as surrogates of a hypothesis h

MakeSurrogatesOfHypothesis(h):

FOR all h in H:
    //Collect all posts related to base hypothesis h
    FOR all p in P:
        IF p is annotated as h:
            A.Append(p); //A is a list of chat-posts
        End_IF;
    End_FOR;

    //Filter duplicates and find distinct posts
    B.Append(A[0]);     // Put the first element of A
                        // into a new list B
    FOR i = 1 to A.length:
        a = A[i];
        FOR all b in B:
            IF CEPsySim(a,b) ≠ 1:
                B.Append(a);
            End_IF;
        End_FOR;
    End_FOR;

    // Compute h context vector component using
    // CEPDVSM for each post and put in a hash G
    FOR all b in B:
        G.key   = vector component with context h
                  in CEPDVSM for s;
        G.value = b; //G is a hash with key->value pairs
        K.Append(G.key); // K is the list of keys
    End_FOR;

    //Collect topmost 100 posts
    K.Sort.Descending; //sort the keys in descending order
    For i = 0 to 99:
        p = G{K[i]};
        S.Append(p);
    End_FOR;

    Return S; //S is the list of surrogates for h
End_FOR;
```

Figure 4.6: Algorithm to make surrogates of hypothesis

Some collected posts in A may have equal contextual similarity value though they do not share any common term. Therefore the algorithm uses CEPsySim measure to find and filter those duplicate chat-posts. A chat-post a is considered duplicate of another chat-post b if :

$$\text{CEPsySim}(a, b) = 1 \qquad \ldots \qquad \text{Equation 4.4}$$

Filtering out the duplicates only the distinct chat-posts are collected in a list B. At this stage one may think that selecting post a as a surrogate and not post b may have different effect because a test post p_t may have different similarity with those two posts a and b. This may intuitively be true; but not in the case of currently implemented CEPsy similarity measure; post p_t will have same CEPsy similarity with both of the posts a and b if there is 100% CEPsy similarity in between themselves. A proof of this has been provided in Appendix C. The similarity between two posts does not depend on the base hypothesis h, therefore one may argue why the filtering is not done as a pre-processing step. The reason is to reduce overhead in the filtering process. In the current step it works on the set of chat-posts in A which is a subset of the set of chat-posts in P. Filtering as a pre-processing step would require working on the set of chat-posts in P which potentially would have much bigger overhead.

After getting the distinct posts in list B the algorithm uses CEPDVSM to compute the 'context vector component' of the target context for each of the distinct chat-posts according to the hypothesis. CEPDVSM has dimensions of four different contexts: BF, IE, GR, AP each corresponds to different hypotheses. The posts and their context vector components are stored in a hash G as pairs of keys and values. Using the values of context vector components the chat-posts are ranked in the order of largest to smallest by sorting the hash G by value. The top 100 chat-posts are then taken into the list S as the preliminary set of surrogates for the base hypothesis h.

Entailment of CE Evidence:

The following steps are used for the entailment of CE evidence:

1. First we set a threshold θ representing the cut-off entailment value. The entailment value is computed by average CEPsy similarity between a suspected chat-post and a set of surrogated-texts representing a CE

evidential hypothesis. The θ is comparable with the similarity value explained in section 4.3 of this chapter. If the average entailment value for a text is above the threshold θ then it is a YES entailment; that is, the text entails the hypothesis; otherwise it is a NO entailment. To determine the optimal value of θ we varied the value from 0.70 to 0.95 with an increment of 0.05 in each step and plotted in graphs in Figure 4.7(a, b, c, d and e). We used the range 0.7 to 0.95 because below 0.7 almost all entailment results become YES, and above 0.95 almost all entailment results become NO.

2. To find out the optimal number of surrogates in the hypothesis set we varied the number of posts from 10 to 100, increasing 10 in each step. For each value of θ the recall, accuracy, precision, F1 measure and F2 measure are plotted against the numbers of hypothesis-surrogates (λ) in the graphs in Figure 4.7.
3. From the graphs in Figure 4.7 we can see that the highest value of precision (75.5%) and accuracy (81.8%) is obtained with θ = 0.95, and λ = 50 and 10. However with those values of the parameters (θ = 0.95) and (λ = 50 and 10) the recall is very low as 21.4% and 38.2% respectively. The evidence finding system is concerned about finding as many of the evidential posts as possible out of the chat-logs even though some innocent posts being caught. Therefore in this particular case recall (R) is more important than precision(P) and accuracy (A). With almost all values of λ , the graph with θ = 0.7 shows good values of recall, however its precision and accuracy graphs are inferior than the graphs produced by some other values of θ. The recall graph produced by θ = 0.8 is not as high as the graph produced by θ = 0.7 but is very much comparable. Moreover the precision and accuracy graphs produced by θ = 0.8 is better than others. Therefore on a balance θ = 0.8 can be an acceptable threshold value. The highest recall 90.5% is found with θ = 0.7 and λ = 30. However with these values of θ and λ the precision and accuracy are as low as 30.1% and 51% respectively. A small sacrifice in the value of recall improves the value of precision and accuracy. If θ is taken as 0.8 instead of 0.7 and λ = 60 instead of 30 then the corresponding values of precision and accuracy are increased by 8.1% and 16% with a price of 2.3% drop of recall. With those parameter-values (θ = 0.8 and λ = 60) recall, precision and accuracy become 88.7%, 38.2% and 67% respectively. Therefore the pair of values 0.8 and 60 is a good candidate to be the acceptable values for the parameter pair θ and λ in the entailment system. The graph of F1 and F2 measures in Figure 4.7(d and e) also supports that 60 hypothesis-surrogates (λ = 60) and entailment threshold θ = 0.8 gives the best acceptable results of F1 = 51.6% and F2 = 67.4%.

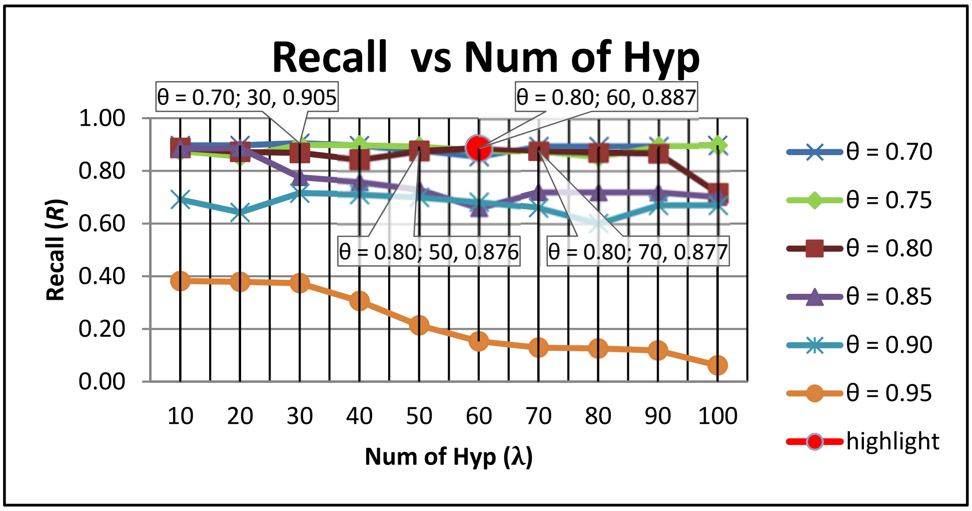


Figure 4.7(a)

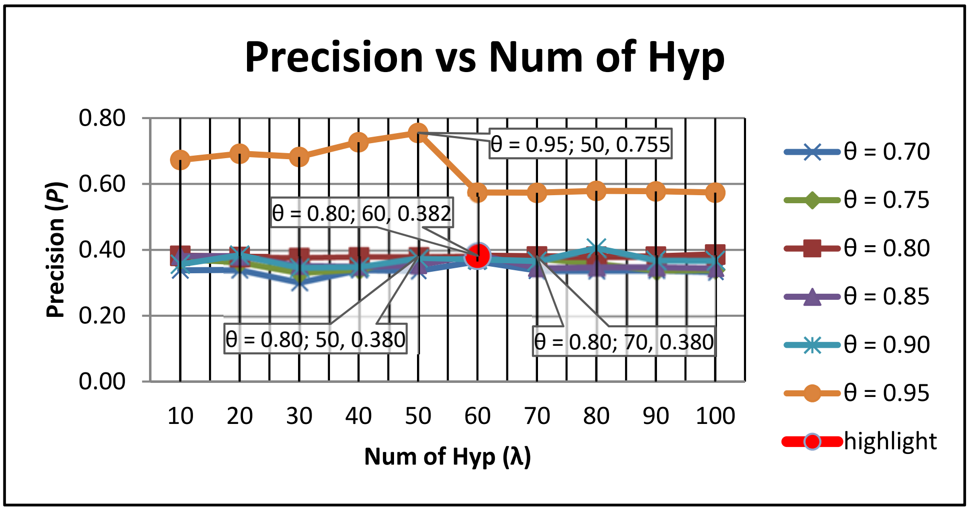


Figure 4.7(b)

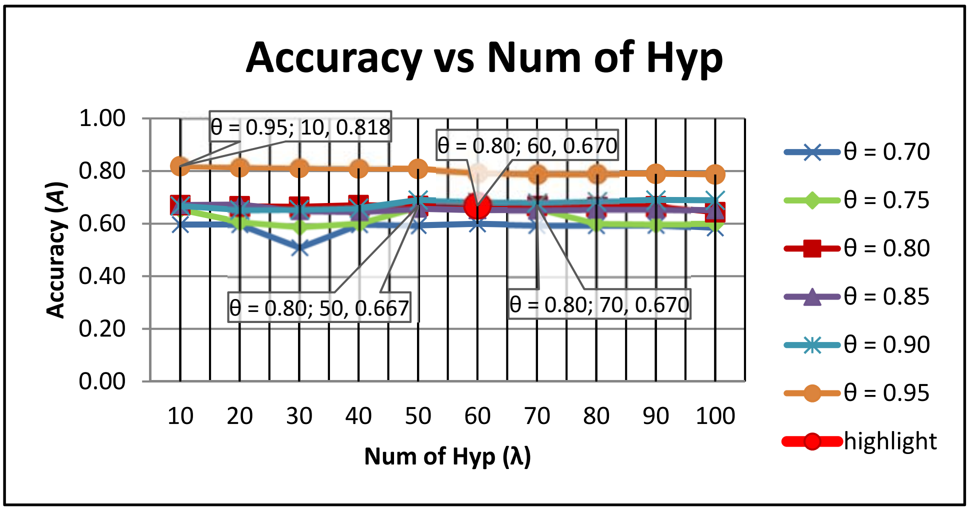


Figure 4.7 (c)

Figure 4.7: Determining number of hypotheses-surrogates and entailment-threshold.

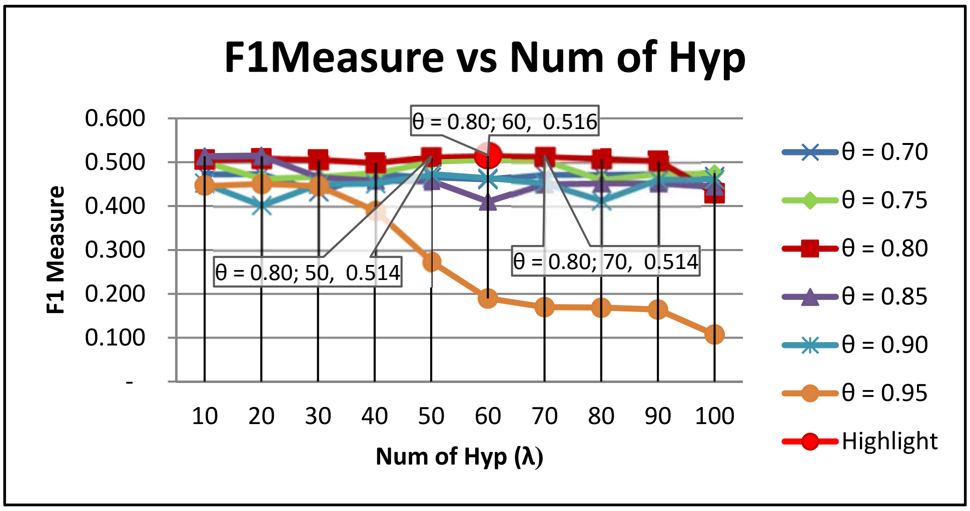


Figure 4.7 (d)

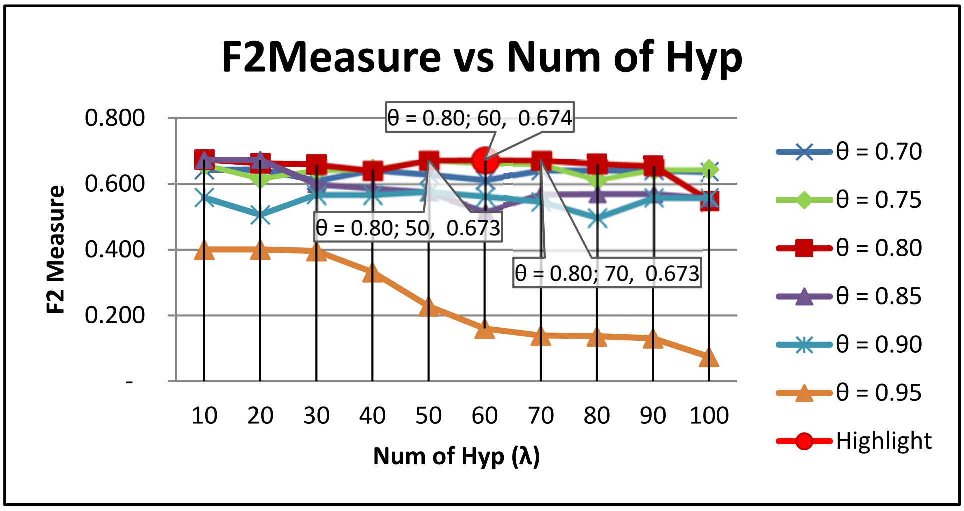


Figure 4.7 (e)

Figure 4.7 (continued..): Determining number of hypotheses-surrogates and entailment-threshold.

Therefore, λ = 60 is used as the number of chat-posts to make a surrogated-text set to represent an evidential CE psychological contextual hypothesis, and θ = 0.80 is used as the cut-off value of average entailment. The parameters λ = 60 and θ = 0.80 are determined by using training-set only. The determination process is completely unseen by the test-set.

4. The entailment engine is tuned with these parameters (λ = 60 and θ = 0.80) and then applied on the test-set. If entailment value for any chat-post is above θ then it is a YES entailment otherwise it is a NO entailment. The suspected posts which entail any of the CE evidential hypotheses are located as evidence.

The YES-NO entailment results are written into the YN entailment hash. This YN hash is used to compute the evaluation metrics. The evaluation procedures are discussed in Chapter-5.

4.4.3 Limitations of RCE

Due to the limitation of available surrogated texts (labelled data set) at this time the first three CE psychological (IE, GR, and AP) types of propositions mentioned in the evidential hypotheses list in Figure 4.5 can be directly entailed by the RCE approach. The last two legal type evidential propositions of that list cannot directly be entailed at this time as surrogated texts cannot be found for them. However, if the three types of hypotheses (IE, GR and AP) are entailed in a particular chat-log then it provides the evidence that the perpetrator communicated with the victim, exchanged personal information, conducted grooming activity and eventually approached for a physical meeting. When this evidence is found it will certainly raise concern to a reasonable person who cares for the child. With this evidence it can also be comprehended that the adult is procuring or grooming a child for unlawful sexual activity. In this way indirectly the legal evidential hypotheses can also been entailed.

4.5 Chapter Summary

The theoretical aspects of our approach to new soft entailment are discussed in this chapter. Logics behind this technique are also explained. Our approach does not depend on expensive linguistic tools like the existing formal RTE systems. A new CE psychological domain vector space model (CEPDVSM) works as the core of our approach of entailment. The CEPDVSM is derived from the term vector space model (TVSM) by reducing the high dimensionality of TVSM to four dimensions of the CE psychological contexts. To accomplish this a new term weighting measure called “crossed predator frequency inverse category frequency (CPFiCF)” is constructed and utilized. To locate the CE evidence in chat-logs a new procedure called "Recognition of CE Entailment (RCE)" is introduced. In this procedure, using the CEPDVSM, a number of chat-posts are selected from the training chat-logs as a set of surrogated-texts representing a CE evidential hypothesis. If the average CEPsy Similarity between a suspected chat-post and the set of surrogated-texts is more than a predefined threshold then it is considered that the text of the suspected chat-post softly entails the CE evidential hypothesis. The suspected posts which entail any of the CE evidential hypotheses are put forward as CE evidence.

Chapter 5

The Design of a Three Tier Model

This chapter discusses the methodology for addressing the research problems identified in Chapter-1 on finding evidence of child exploitation out of chat-logs. A three tier CE detection model (CEDM) have been developed throughout the current research. The design and architectural framework of CEDM have been discussed in details in this chapter.

5.1 Overall Approach

In the course of locating evidence our methodology starts with an outer-level analysis before delving into the inner-level for extracting specific evidential content and finally producing them. It incorporates a three tier CE Evidence Detection Model (CEDM). Figure 5.1 shows the diagram of the overall system architecture of the proposed model. It consists of the following two broad phases:

1. Shallow Evidence Analysis and
2. Particular Evidence Detection

By the term 'shallow evidence' we would like to refer to the evidential artefacts that can be inferred through statistical analysis of the original chat-texts. Those evidential artefacts are not the chat-texts themselves or part of it but information

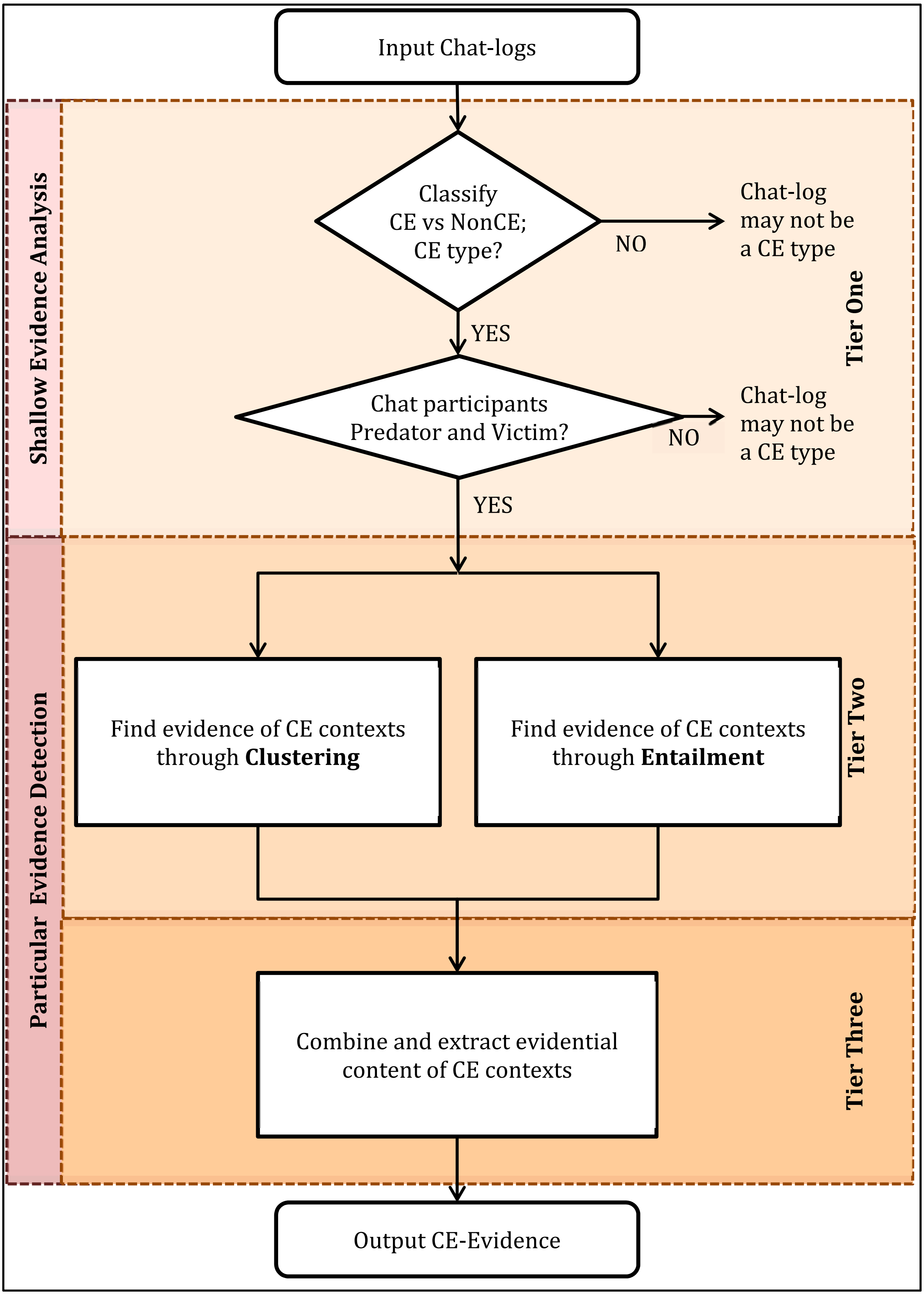


Figure 5.1: Overall system architecture of three tier CE evidence detection model.

about it. This kind of evidence is comparable with 'circumstantial evidence' in the legal terminology. For example, using statistical analysis to determine whether the suspected chat-log falls into CE category and whether the chat is between a CE predator and a CE victim. The 'particular evidence' comprises the evidential artefacts consisting of chat-texts that are part of the suspected chat-log. Those parts of the chat-log that potentially entail the CE propositions defined by the legal system or by the CE researchers.

Within the two broad phases the 'three tier evidence detection' is composed of the three main stages in the module. These are:

1. Tier one: Statistical analysis of overall context through 'Classification'
2. Tier two: Analysis of contents to find specific CE contexts through 'Clustering' and 'Entailment'
3. Tier three: Accumulating and producing CE evidence

The first tier is in the shallow evidence analysis phase. The last two tiers are in the particular evidence detection phase. Design and implementation of these three tiers is the main scope of this current research. This chapter explains the design stage and the next chapter covers the implementation stage.

5.2 Tier One: Shallow Evidence Analysis

In the shallow evidence analysis phase the module classifies the chat-logs as CE or not. The 'Tier One' of the three tier module lies in this phase. No specific evidence is detected in this phase. The suspected offensive CE chat-logs are then analysed further by the next steps of particular evidence detection phase. The shallow evidence analysis phase consists of the following two sub-phases:

1. CE vs. Non-CE Classification Phase: Classifying a chat-log as CE vs. Non-CE
2. Predator vs. Victim Classification Phase: Classifying the participants of a chat into Predator vs. Victim

5.2.1 Methodology for the CE vs. Non-CE Classification Phase

In this current phase of ‘tier one’ text classifiers are used to categorize the available chat-logs into offensive (CE) and benign (Non-CE) chats as in Figure 5.2. This sorts out the suspected CE chat-logs for further analysis. For a suspected chat-log the predicted classification decision of being CE type can work as a shallow or circumstantial evidence.

As a feature set we introduce psychometric information associated with each term. Motivation of using this special feature set is the assumption that as psychological behaviour is closely related to child exploitation, underlying psychological information associated with the terms may help the text classifiers for better

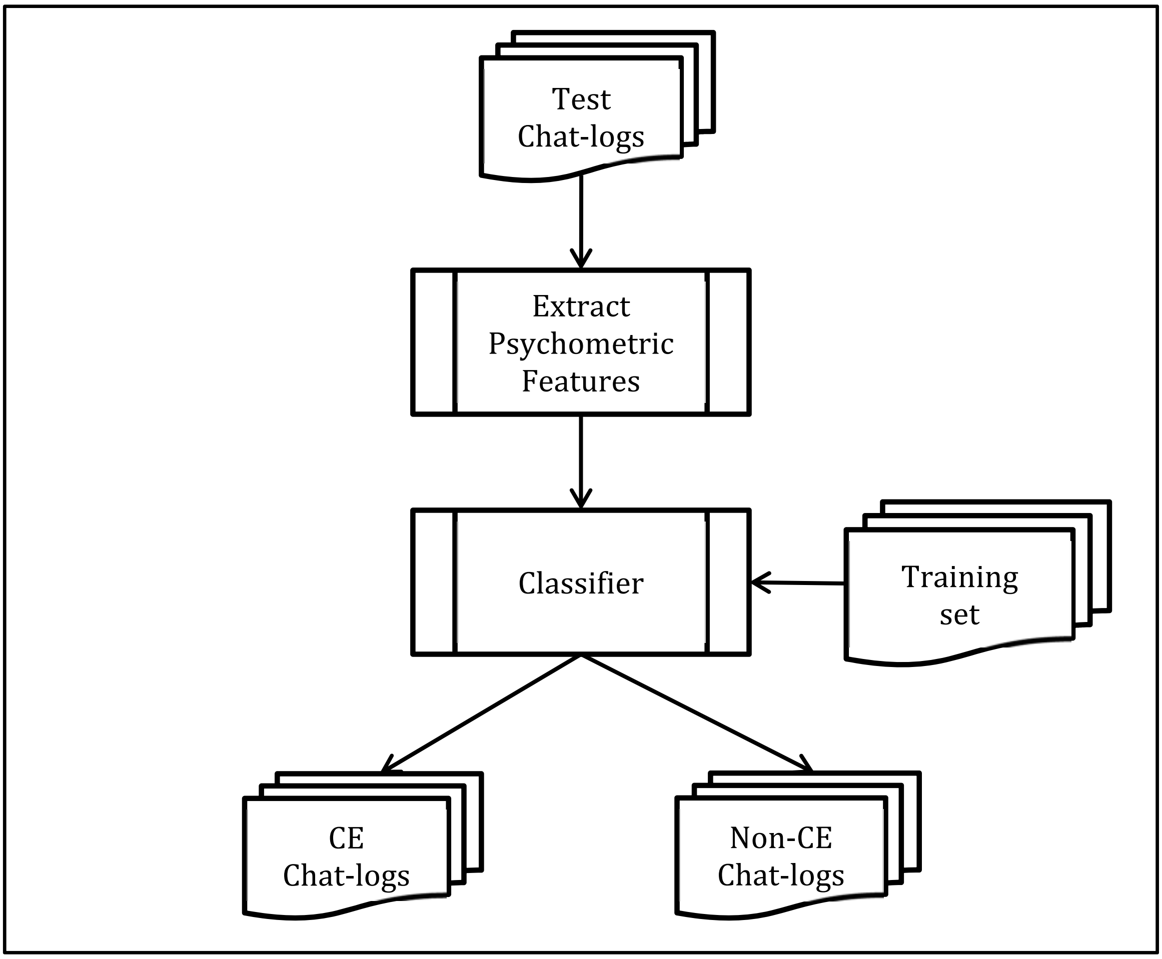


Figure 5.2: Classification of chat-logs into CE and Non-CE

prediction than mere term based feature set. To extract the psychometric information of terms of chat-logs the 'Linguistic Inquiry and Word Count' (LIWC; Pennebaker et al., 2007) is used.

Figure 5.2 shows the basic procedure followed in the classification task. A training set of chat-logs with psychometric features is used to train the classifier. Using a separate test set the effectiveness of classifiers is observed. The psychometric features are extracted from the test set and fed into the classifiers to predict whether the test chat-log is CE or Non-CE . The offensive CE chats are taken forward for further analysis.

This stage of the approach works as the first filter to catch a potential CE chat. If a chat-log passes this stage as a Non-CE chat then it is considered as benign and no further analysis is done. Therefore the classifier needs to be very suspicious; so that no CE chat-log can pass through it as a Non-CE chat-log even though some benign chat-logs are mistakenly predicted as CE chat-logs. In this case classifiers with higher recall should be preferable than classifiers having higher precision but lower recall.

5.2.2 Methodology for the Predator vs. Victim Classification Phase

In this phase classifiers are again used to find out if the participants of the suspected chat match the profile of a CE predator and a victim. A perpetrator may meet a child in a common public chat-room where many other chat users are present. Before starting to groom, the perpetrator tend to take the child victim to a secluded environment of 'Instant Messaging'. Therefore, for a CE chat it is more usual to have only two participants: an adult predator and a child victim. This implies that, a chat that has one participant with the CE predatory behaviour and one participant with CE victim's profile is more likely to be CE than the chats having more than two participants or than the chats having two participants but none of the participants resemble predators. Therefore it is one of the important CE evidential artefacts that

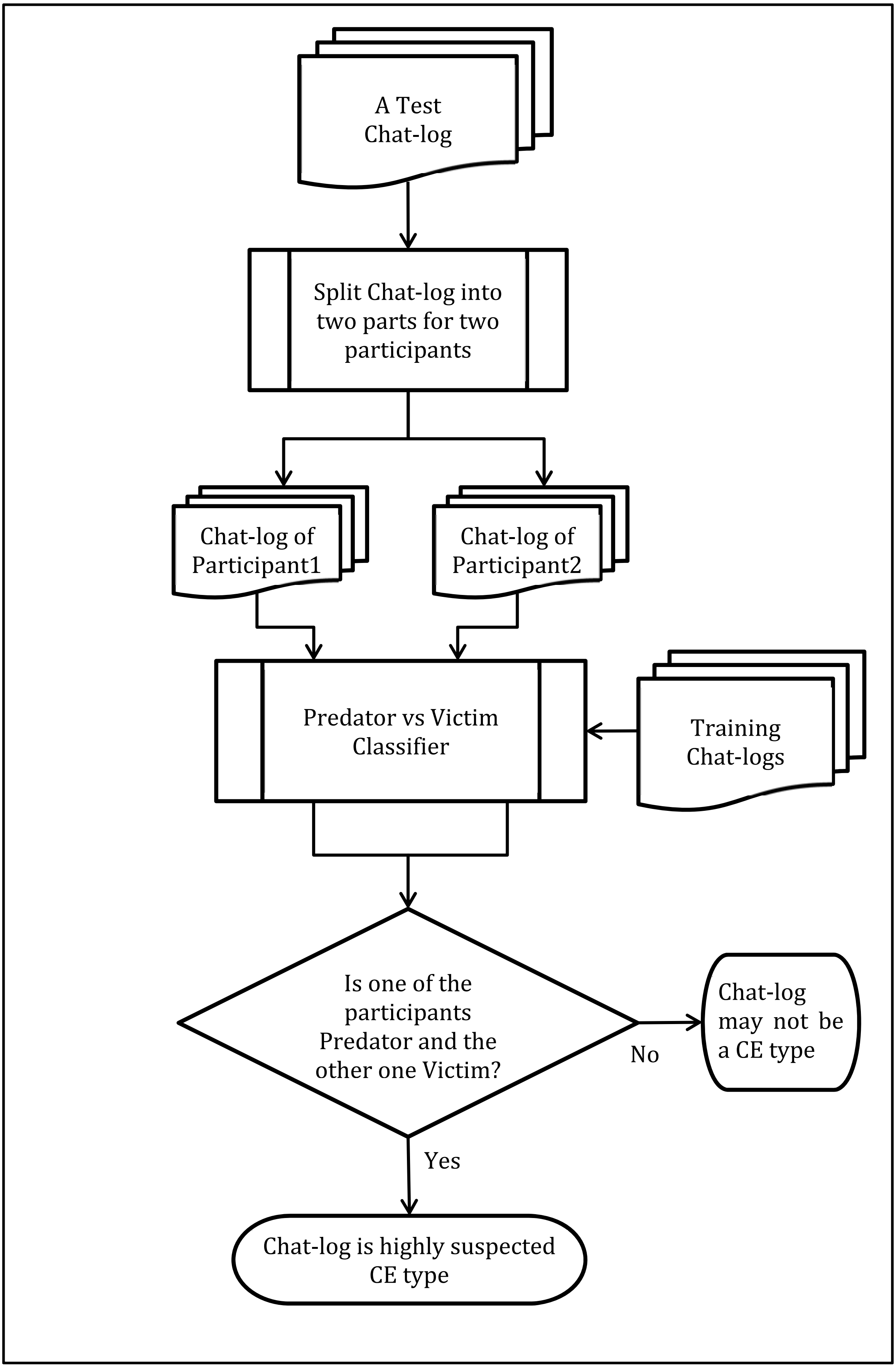

A Test
Chat-log
Split Chat-log into two parts for two participants
Chat-log of Participant1
Chat-log of Participant2
Predator vs Victim Classifier
Training Chat-logs
Is one of the participants Predator and the other one Victim?
No
Chat-log may not be a CE type
Yes
Chat-log is highly suspected CE type

Figure 5.3: Method for Predator vs Victim classification task

a CE predator and a CE victim are involved in a CE chat. Though this is not concrete evidence however it can be important supporting evidence.

Figure 5.3 shows the method used in identifying the participants of a chat as predator and victim. Pendar (2007) performed a classification task to identify predator vs. victim in a CE chat. A similar classification task is incorporated as a part in this phase of the 'tier one' of the module. Using a text classifier the participants of the suspected chat are categorized and checked as to whether they appear to be a CE predator and a CE victim. If this is found then it is important in building an evidential case. The vocabulary of a child victim is different in comparison with the vocabulary of an adult perpetrator. In a CE chat the predator and the victim may discuss the same topic, however they may use different sets of terms as the aim of the two sides of the conversations are different; the predator's aim is to groom the child and the child's aim is to enjoy chatting with an online friend. This may suffice for a text classifier to detect the difference of the linguistic term-sets used. Therefore in this current phase the text classifiers use simple term based features. Texts from the training chat-logs are split into two groups of 'CE predator' and 'CE victim' according to the chat-posts of each participant. A text classifier is trained on this term based data-set to learn the distribution of the terms used by a CE predator and by a CE victim. The test chat-text is also split into parts according to the participants. The classifier's learning function is then used on each participant's part of the test chat-text to identify as to whether a CE predator and a CE victim are involved.

From tier one, when the two shallow evidential artefacts are found in a chat then it is considered to be a highly suspected CE chat. However, the text-classifiers work with a bag-of-words in a probabilistic approach. They do not attempt to extract the exact excerpts from the chat-text; they only classify the whole chats into the predefined categories as learned from the training. Locating exact excerpts of CE evidential chat-fragments requires further analysis of the chat-log. The highly suspected CE chat-logs obtained from tier one are taken to the next tier (Tier Two) for further analysis.

5.3 Tier Two: Particular Evidence Detection

From the literature review chapter we already know that the perpetrators follow certain psychological communicative behavioural stages in the course of child exploitation. The content of CE chat-text keeps the traces of evidence of the pattern of those psychological stages. A robust analysis on the contents of the suspected chat is required to find out those traces of particular substantial evidence. This is done in 'tier two' by incorporating two new text processing techniques developed in Chapter-3 and Chapter-4. This tier includes two parallel phases:

1. Clustering phase and
2. Entailment phase

5.3.1 Methodology for the Clustering Phase

It has already been mentioned that there are four CE psychological stages in a CE chat according to the psycho-behavioural pattern of a CE predator. If the predator's posts of a suspected chat-log are automatically clustered into four groups corresponding to the four CE stages without any supervision then this will provide supporting evidence that the chat is of a CE type. An unsupervised machine learning method clusterer, that analyses the psychological contexts of the predator's posts of a chat-log and effectively arranges them into the four groups, can be used to trace the CE evidence. The evidence will be stronger if the chat-posts grouped by the clusterer show associations with the offensive CE psychological stages.

Figure 5.4 shows the procedural framework of the clustering phase. For each chat-log the predator's posts are considered as individual object units and separated from the rest of the chat-logs for further analysis. Using these units as individual instances our newly developed PsyHAC clusterer (explained in Chapter-3) is used to cluster the predator's posts of the test set into the predefined CE psychological behavioural stages.

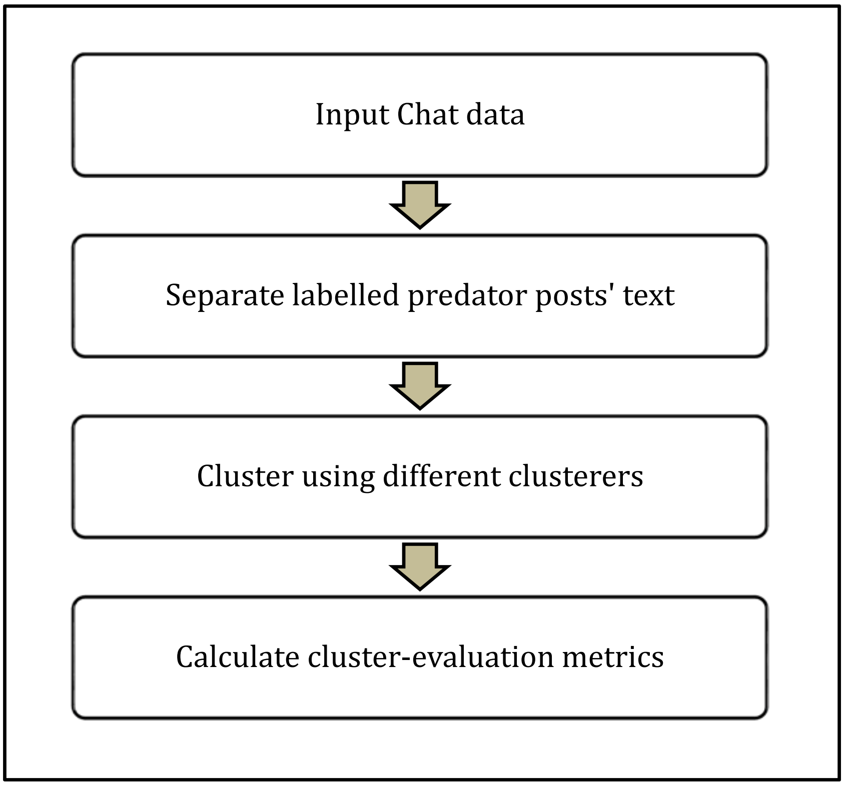

Input Chat data
Separate labelled predator posts' text
Cluster using different clusterers
Calculate cluster-evaluation metrics

Figure 5.4: Procedural framework for clustering-phase.

The results of our PsyHAC clusterer are compared with four other clusterers. Three of the clusterers use conventional similarity measures on the full vector space; they are *K*-means (KM), Expectation Maximization (EM), and Hierarchical Agglomerative Clusterer (HAC). Procedures of these three clusterers are described in Chapter-2. The other clusterer uses Latent Semantic Analysis (LSA) with HAC; we call this LSA-HAC. Procedure of LSA-HAC is provided below:

Procedure of LSA-HAC:

For clustering with LSA-HAC the chat text data is transformed into the reduced latent semantic space. The pairwise latent similarity among the chat-posts are computed by measuring the cosine similarities of the data instances in the latent space. Using the pairwise similarities the chat-posts are clustered with a hierarchical agglomerative clusterer. The latent semantic analysis decomposes the term matrix of the chat-posts into singular values. Instead of using all the available singular values, using a highly ranked subset reduces the unwanted dimensions and gives a

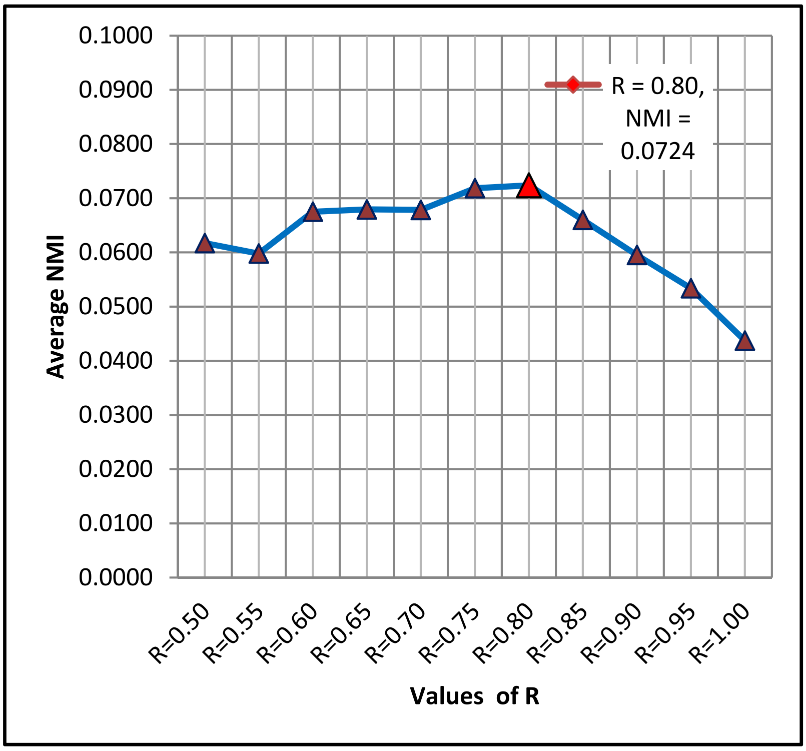


Figure 5.5: Average NMI values vs values of R in the training set.

higher latent similarity (Landauer, Foltz, & Laham, 1998) between the pairs of chat-posts, hence a better clustering is expected. However, our experiment shows that it does not happen in all cases. The reason is, it also produces higher similarity between some pairs of posts which are not of the same type. In our experiments, the subset of the singular values is expressed as a percentage and denoted by R. The training set is used to determine the optimal value of R that gives the maximum average NMI value. This optimal value of R is used to find the NMI result of the test set. In the training set the value of R is varied from 0.50 to 1.0 (50% to 100% SVD) in increments of 0.05. Figure 5.5 shows the graph of the NMI value vs the value of R for the training set. From the graph in Figure 5.5 we can see that R = 0.80 gives the best average NMI value on the training set. Therefore we used R = 0.80 for the test-set to find the latent semantic similarity and then used it for clustering and evaluation.

Normalized mutual information (NMI) is computed as a cluster-evaluation metric. Results of the different clusterers are compared and presented in Chapter-6.

Association of Clusters to CE Evidence:

A clusterer uses unsupervised learning to accumulate chat-posts into four different groups according to the four CE psychological stages. It does not tell us which group is associated with which CE stage; as a supervised learning method like a classifier would do. For the chat-posts to be of particular evidential artefacts their association with the CE psychological stages is required to be found. In this regards after clustering, association of each group of posts with their corresponding CE stages is obtained in the following manner:

Consider that a clusterer has grouped the chat-posts into the clusters $w_k \in \{w_1, \dots, w_K\}$, and the evidential CE psychological stages are $c_m \in \{c_1, \dots, c_M\}$. In this current research the number of clusters (K) is equal to the number of CE stages (M) which is four. Each cluster w_k needs to be associated with the corresponding correct CE stage c_m in such a way so that the highest overall accuracy is obtained. We make all possible combinations of the associations of w_k with c_m. Examples of the combinations are:

(a) $w_0c_0, w_1c_1, \dots, w_{\mathrm{k}-1}c_{m-1}, w_kc_m$;

(b) $w_0c_1, w_1c_2, \dots, w_{\mathrm{k}-1}c_m, w_kc_0$;

… and so on up to:

(c)$w_0c_{\mathrm{m}}, w_1c_0, \dots, w_{\mathrm{k}-1}c_{m-2}, w_kc_{\mathrm{m}-1}$;

where: each $\mathrm{w_kc_m}$ pair indicates a mutual association.

For each combination the overall accuracy is measured as:

$A = \frac{n}{N}$;

where:

n = sum of the chat-posts correctly associated with each of the CE stages

N = Total number of chat-posts

The combination which gives the highest accuracy is taken and the chat-posts are associated with the evidential CE stages according to it. The result is stored in a hash in a format of 'key → value' pairs as:

'$p_i \rightarrow c_m$';

where: p_i is a key of the hash represented by a chat-post and c_m is the value given by one of the evidential CE psychological stages BF, IE, GR, and AP associated with that chat-post. This hash would work as an evidential artefact. The hash is used for computing evaluation metrics and also in the evidence producing stage.

5.3.2 Methodology for the Entailment Phase

The legal or psychological propositions of CE that fulfill the legislative requirement for evidence of child exploitation can be used as evidential hypotheses. Breaking the whole chat-text into fragments of posts and then entailing the hypotheses by those chat-fragments would be useful to prove the evidence that the chat contains criminal activity of child exploitation. As mentioned in Chapter-4 examples of some evidential hypotheses include the following propositions:

1. "A perpetrator is involved in child-grooming activity "; (GR type)
2. "Valuable personal information has been exchanged between the perpetrator and the victim"; (IE type)
3. "An approach has been made to physically meet the victim"; (AP type)

These propositions are abridged from the definitions provided by psychological and communication researchers. Using the same definitions human annotators labelled the chat-post data set (this will be discussed in more details in section 5.5.2.2 of this chapter). Therefore the labelled chat-posts can be used as surrogated texts to represent these evidential hypotheses. Using the RCE soft entailment method (explained in Chapter-4) these evidential hypotheses are entailed by chat-post texts. For each and every post of a chat-log a 'YES' or a 'NO' result of the entailment is obtained for a particular evidential CE hypothesis. This is shown in Figure 5.6. Explanations of the notations in the figure are:

$\mathrm{T_i} \in \mathbb{C} =$ Chat-posts in chat $\mathbb{C}$; $\mathrm{i} = \{1, \ldots, N\}$; $N =$ Number of posts in chat $\mathbb{C}$.

$x \in \{\mathrm{IE, GR, AP}\} =$ CE Psychological Stages corresponding to CE evidence.

$\mathrm{RCE_x} =$ RCE system with text-surrogates representing hypotheses of CE Psychological Stage of x.

$\mathrm{E_{xi}} =$ Yes or No $=$ Entailment result of $\mathrm{T_i}$ for CE stage x.

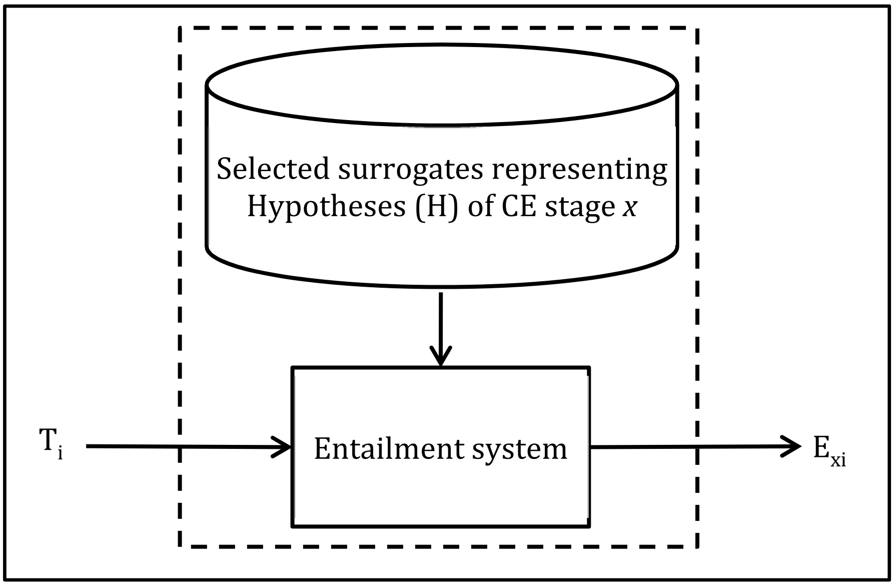

Selected surrogates representing Hypotheses (H) of CE stage *x*
Entailment system
T_i
E_{xi}

Figure 5.6: Entailment by RCE.

The entailment result E_{xi} indicates whether the text T_i of a chat-post is an evidential artefact of the CE stage x. The results of entailment for all the posts of a chat-log are stored in a hash in the 'key → value' format as '$T_i \rightarrow E_{xi}$'. We call it a YN-Entailment hash. This hash is used in the next stage of evidence producing. It is also used to evaluate the entailment system.

5.4 Tier Three: Evidence Producing

There are two steps in this tier:

1. Combining by accumulating and
2. Extracting

To produce evidence of CE from the text of a chat the results from the clustering phase and entailment phase of tier two are accumulated as combined evidence. Some of the evidence may be detected by the clustering phase which cannot be detected by the entailment phase and vice versa. Therefore accumulating the evidence from both clustering and entailment phase can improve the evidence detection process.

We explain the evidence producing procedure with the following example:
Consider locating the evidential chat-posts which entail the hypothesis:

“A perpetrator is involved in child-grooming activity ”.

This proposition is evidence of grooming type activity. Using $\mathrm{RCE}_{\mathrm{grooming}}$ we need to locate the chat-posts from a suspected chat-log of the test-set which entail the hypothesis of this proposition. A chat-post complying with this proposition will result a ‘YES’ entailment in the entailment hash ($T_i \rightarrow E_{xi}$) obtained by the $\mathrm{RCE}_{\mathrm{grooming}}$ system. If the post does not comply then it will result a ‘NO’ entailment in that hash. The entailment phase produces a YES-NO hash for entailing this hypothesis by all the chat-fragments (chat-posts) in a chat-log. Consider it to be an ‘YN Entailment hash’.

The evidence association part of the clustering phase gives another hash containing $p_i \rightarrow c_m$; where p_i is the chat-post and c_m is the associated CE stage BF, IE GR, or AP as evidence. This hash is required to be transformed into an YN hash for combining with ‘YN Entailment hash’. The transformation is as follows:

Change $p_i \rightarrow c_m$ into $p_i \rightarrow$ YES; where: $c_m = \mathrm{GR}$, otherwise $p_i \rightarrow$ NO;

The hypothesis of the current example is related to the GR stage therefore c_m is replaced by YES for GR for the corresponding ‘key → value’ pairs in the hash. Consider it to be an ‘YN Transformed hash’.

The keys T_i and p_i of both the hashes ‘YN Entailment hash’ and ‘YN Transformed hash’ are the same because both of them contain text of the same chat-post. We use an OR logic to combine the YES-NO values of the ‘YN Entailment hash’ and ‘YN Transformed hash’ into a new ‘YN Combined hash’. The OR logic works as follows:

Y OR Y = Y; Y OR N = Y; N OR N = N;
Where: Y is for ‘YES’ and N is for ‘NO’.

The chat-posts having values of ‘Y OR N’ appear to be weaker evidence than the chat-posts having values of ‘Y OR Y’. However, the main aim at this time is to produce evidence of CE. To obtain a good recall an excerpt should not be discarded that has been pointed as evidence of CE by one part of the CEDM approach. Therefore, all the posts which have been marked as evidence (Y) by any of the two phases, clustering phase or entailment phase, are considered as evidence of CE and are accumulated to produce as combined evidence.

The ‘YN Combined hash’ is used for evaluation and extraction. The posts representing the ‘YES’ value in the hash are extracted and produced as the evidence of the CE criminal activity associated with a particular CE stage.

5.5 Chapter Summary

This chapter presents the detailed design of a three tier CE evidence detection model (CEDM). The approach developed in this chapter addresses the research questions mentioned in the introduction chapter. The CEDM model analyses the chat text using text processing and data mining techniques, and automatically locate the evidence of child exploitation (CE) in the chat text. The automation of CE evidence finding process is accomplished in three broader phases or ‘tiers’ in the CEDM approach. The first tier attempts to find out whether the traditional text classifiers effectively classify chat-logs into Child Exploiting (CE) and non Child Exploiting (Non-CE) classes. To embellish the feature set psychometric information is introduced in this phase. It is expected that with the new feature set the classifiers will effectively classify chats into CE vs. non-CE. Text classifiers are again used with term-based features for identifying the participants of the chat as to whether a CE predator and a CE victim are involved. The classifiers in ‘tier one’ of the CEDM neither provide any particular evidence nor do they extract any excerpt of text which can be evidential artefacts. The results of the classifiers work as shallow circumstantial supporting evidence.

The successive tiers of the CEDM approach perform deeper analysis of the chat text through a new clustering technique PsyHAC and a new soft entailment technique RCE for locating particular evidence. The pattern of progression and profile of CE chats identified in the psychological literature is extensively used to solve the problem of specific evidence finding. In the RCE technique we frame the CE evidence finding problem into a manageable problem of RTE (Recognition of Textual Entailment) on chat-logs.

The next chapter presents the collection of data, evaluation metrics, experiments, results and analysis based on the research methodology discussed in this chapter.

Chapter 6

Experiments and Results

This chapter discusses the collection of data in this research. The evaluation methods are also explained. The experimental procedures and analyses of the results are provided for each part. The experimental setup follows the stages of the three tier module explained in the previous chapter.

6.1 Collection of Data

6.1.1 Difficulties in Finding a Formal Benchmark

To evaluate a formal text processing system a formal collection of data, sometimes referred as benchmarks, is used. Traditional benchmarks include collections of Reuters and TREC. In recent time (2009) Microsoft in association with Yahoo Inc released another collection set called LETOR (Learning to Rank for Information Retrieval). LETOR also include some collection from TREC. Among these benchmarks TREC collections include a variety of data which includes genomics data, video data, and even internet blogs. Nevertheless the field of text processing and information technology is still growing and continuously spreading into new applications. It is happening so rapidly that researchers are raising questions like "Is the field mature enough to talk about benchmarking?" (Dekhtyar & Hayes, 2006). Until the recent time most of the tasks about text processing and information retrieval were on formal texts. This may be a reason that all these benchmarks have collection of documents that contain mostly formal texts (with some exceptions like gnomics and blogs data), whereas the chat-text is not that formal. System evaluation usually lags behind system development and implementation. In the initial phase, the priority is model design and system development. Without available models or systems, it is impossible to conduct system evaluation (Zhang, 2008). It is only recently that researchers are paying attention on informal texts from chat like conversational communication media. This may be another reason that a robust, well-designed time tested, and, eventually well-established and accepted benchmark is still unavailable in this particular domain.

In search of a chat-text collection, a chat-corpus has been found in the Naval Postgraduate School (NPS), but it contains multi-user chat posts collected from open public chat-rooms (Adams & Martell, 2008). Though a perpetrator may first contact a child in an open chat-room but he tends not to do the exploiting act in the public area. It is most likely that he will take the child into a private instant messaging (IM) session. A private IM is more secured, risk free and easy for future contact and grooming. A corpus built with posts from public chat-rooms may not be useful to find online child exploitation. Therefore the NPS chat-corpus does not meet the need of the current research. In such a situation without a formal benchmark, the current research looked for an alternative collection that may serve its purpose of testing and evaluation.

6.1.2 Data Used in this Research

Due to the sexually explicit nature of the child exploiting chats, and the surrounding legal and ethical issues, it is difficult to find such data in authenticated academically available research databases. Fortunately, a number of such chat-logs have been found in the Perverted-Justice.com (PJ) website at http://www.perverted-justice.com/. The Perverted-Justice.com is a part of Perverted Justice Foundation

Incorporated (pjfi.org). This organization worked with Law and Enforcement Agency (LEA) in a covert operation to catch the online paedophiles. The chat-logs produced in the process of that covert operation was recorded and published online. The chat-logs contain chat-text between trained users posing as a child and perpetrators trying to procure children over the Internet for exploitation. The perpetrators involved in those chats are prosecuted according to US law. The chat-texts were used as evidence and the perpetrators were convicted. In absence of chats between a real child and a paedophile these chat-texts work as a benchmark because they contain evidence of child exploitation and the evidence have been established in a court of law. The chat-texts are open for all in the World Wide Web. Permission through email from the administrator of the website has been received to use those chat-logs for the purpose of current research. For classification experiments other kinds of chats, which are not of child exploitation nature, are also needed. Those chats have been collected from various websites that are also open for all. These kinds of benign chats are completely anonymous and general in nature.

To construct the data set of current research 708 chat-logs have been collected. Within these chat-logs 516 are of Child Exploiting (CE) type chats collected from Perverted-Justice.Com website. The number of total[1] words found in the chat text files is 9,567,152. The average document size is 15,258 words per chat-log. The number of distinct[2] words in the chat-logs is 79,711 including noises. After deleting non alphanumeric noises (e.g. ***** , @[]#$$$) the figure becomes about 69,000. Still this figure contains some other types of noises like miss-spelling (e.g. hallow, haloo, hallooo are considered as different words). Ignoring those noises this figure (69,000) can be considered as the vocabulary size of these chat-logs. If those noises are removed then the vocabulary size will be reduced further.

English language has about a quarter of a million (250,000) distinct words in its vocabulary (Oxford Dictionary, 2014). Formal text documents may contain low frequency, divergent and wide range of words. To capture the features from the wide

[1] The total number of words has been determined by using Total Assistant software.

[2] The number of distinct words has been determined by using Weka 'String to Word Vector' filter.

range of vocabulary a text classifier needs a large number of documents in the training data set to classify formal text documents. By contrast, the range in the vocabulary for chat text is much narrower than the formal texts as chat texts are conversational and informal in nature. The participants tend to use common high frequency easy words. Therefore a small number of documents in the training set are acceptable for the text classifiers to classify chat text documents.

PAN 12 sexual predator identification task data also contains chat-texts. It mainly used the chat-logs from Perverted-Justice.com (PJ) website and edited them into collection of conversations. As it was from the same source of PJ website and as the experiments of our classification tier were already completed when the data was released (in May 2012); we did not use PAN data in this research. However it can be a future interest how our classification method works on those data.

McGhee et al. (2011) published labelled chat-texts in chatcoder.com. That data-set is also open for researchers to use. We have collected those data and used in 'tier two' and 'tier three' in our experiments. The chat-logs, collected from PJ website and various other websites, are modified to fit the requirements of the 'tier one' of our experiments. More details of data used in each tier of the proposed system are discussed in the following subsections.

6.1.2.1 Data Set for Tier One

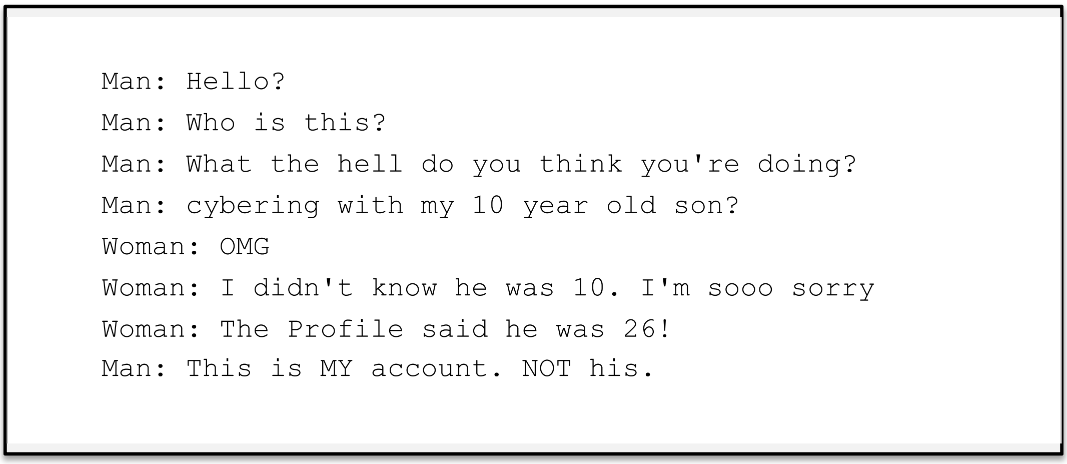
Data set for 'CE vs Non CE' classification Phase:

In 'tier one', the classification phase of the proposed evidence detection approach, we used the chat-logs collected from the Perverted-Justice.Com website at http://www.perverted-justice.com/. As have been mentioned before those chats work as a benchmark because they contain evidence of child exploitation and the evidence has been established in the court of Law. It has been mentioned in section 2.4.2 of Chapter-2 that from the CE point of view chats can be considered of the types:

(a) Child Exploiting (CE), (b) Near to CE or Sex Fantasy (SF), and (c) Far from CE or General (GN). For the classification experiments apart from CE, different other kinds of chats are also needed which include sex fantasy (SF) type chats between two consenting adults and very general (GN) type chats which do not have any sexual content.

Websites like http://www.fugly.com and http://chatdump.com have a collection of anonymous chats. The chats were provided by volunteers making fun with people online. Some of the chats can be considered as SF type. This type of chat contains elements of sex fantasy. However as the main purpose was only to make fun, in some part of the chat one of the user behave weirdly to make fun out of the already built sex fantasy. For example, after a considerable time of chatting and starting up a romantic relationship, a user appears to be a different person (though he is not) and turns the conversation into a different direction other than sex fantasy. An example excerpt of a turning point is in Figure 6.1. The direction changing parts were edited (mostly deleted) to keep the chat as SF. The chat-logs were combined in a collection with some other type of chat-logs for analysing by a researcher of psychology to verify the SF types. The researcher of psychology identified 73 of the chats in the collection as SF types. To increase the number of chats in SF type, some of the SF chats are randomly crossed with each other. Finally 85 SF type chats are used in the experiments.

Finally the data set consists of text of a number of chat-log files for the 'CE vs Non-CE ' classification task. The logs include child exploiting offensive CE chat-logs, general



```
Man: Hello?
Man: Who is this?
Man: What the hell do you think you're doing?
Man: cybering with my 10 year old son?
Woman: OMG
Woman: I didn't know he was 10. I'm sooo sorry
Woman: The Profile said he was 26!
Man: This is MY account. NOT his.
```

Figure 6.1: Example of direction changing portion of a SF chat.

non offensive (GN) chat-logs and sex fantasy type SF chat-logs. Total number of chat-logs is 392. Among the 392 instances 200 are CE chat-logs, 85 are SF type and 107 are GN type chat-logs. That is, 200 CE chats and 192 Non-CE chats. Instead of taking all 516 CE chats collected from Perverted-Justice.Com, a set of 200 is randomly chosen to make a balance between the CE and Non-CE chats.

Data set for 'Predator vs Victim' classification Phase:

Pendar (2007) has summarized the possible types of chat interactions between a predator and others as below:

1. Predator vs Victim (Victim is underage)
2. Predator vs pseudo-Victim (Volunteer posing as a child)
3. Predator vs Law enforcement officer posing as a child

For the task of 'Predator vs Victim' classification the above mentioned type-1 data would have been most appropriate. However obtaining that type of data is very difficult. Getting access to type-3 data is also problematic as law enforcement agencies do not want to give away their secured data. In such a situation the freely available type-2 data at Perverted-Justice.Com (PJ) can be an alternative. It has been mentioned that the predators of PJ chat-logs are convicted in the court of law; their predatory behaviour is already established. Therefore the part of the chats written by them is a data-set representing real adult CE predators.

However, there may be concern about the other part of the chat written by the adult volunteers posing as children; as the chat-texts are not written by real children how appropriate this would be to represent child-victims of CE. Our reasons behind using this data are: firstly, no data of real child-victims of CE can be found at this time; secondly, the volunteers were trained to act like real child-victims (McGhee, 2011 and PJ website), thirdly, the predators possessing human intelligence believed that they are grooming a real child not an adult posing as a child. Therefore we assume that the chat-texts contain the features of a data-set that can represent victims of CE. Moreover, the main aim of this task is to find evidence against predators; not against victims. It has been mentioned in section 2.2.1 of Chapter-2 that, according to legislative provisions, involvement of a real child is not mandatory to build a case against a CE predator. In the current classification task, finding the involvement of a CE predator is more important than finding the involvement of a real-child. Therefore, by the term 'victim' we would like to represent 'opposite of a predator' who can be a real-child-victim or can be a person the predator believes to be a child. By this formation of the task we expect the data-set can be acceptable.

489 chat-logs out of the 516 CE chat-logs collected from Perverted-Justice.Com (PJ) are to be used for our 'Predator vs Victim' classification task. The other PJ chats cannot be used due to errors in the files. Each chat-log is divided into two: one including the chat lines produced by the predator and one including the chat lines produced by the victim. Total number of individual logs becomes 978 out of which 489 are predators' log and 489 are victims' log.

6.1.2.2 Data Set for Tier Two and Three

The chat-log data set used in 'tier one' do not contain contextual annotation in its content, therefore is not suitable for the experiments in 'tier two' and 'tier three' of this research. In these tiers we need chat-logs in which each of the posts has been labelled with CE psychological contexts. Such a data set is downloaded from chatcoder.com (McGhee et al., 2011). The authors collected the original text chat-logs from the PervertedJustice.com (PJ) website. As have been mentioned before, the text chat-logs contain conversations between convicted predators and trained volunteers posing as children. The posts were extracted from all chat-logs for all screen names. The timeframes represented by the chat-logs varied from a few hours to several months. The chat-logs ranged in length from 83 lines to 12,704 lines (McGhee et al., 2011). Two trained analysts were appointed by McGhee et al. (2011) to manually annotate the text of predators' posts of the chat-logs into the CE psychological contextual stages defined by the psychological and communication researchers. Only the predators' posts have been labelled as the goal is to learn the CE profile of the perpetrator. This data set fulfils our research requirements as we would like to trace the evidence of child exploiting act left by the perpetrator in their texts. Therefore at this time the victims' response is not necessary however this may be a subject of future research.

The downloaded data from chatcoder.com is in xml form. It includes 56 chat-logs though McGhee et al. (2011) mentioned about 50 chats in their paper. It seems that they managed to label 6 more chats at the time of uploading to the chatcoder.com website. Due to file formatting problem the authors used only 33 chat-logs out of the total dataset. We checked all the chat-logs and made some changes to make all of them work. Each of the chat-logs contains chat-posts of predators as well as victims. A thorough inspection reveals that some of the chat-logs contain multiple predator-victim pairs. But our interest is to analyse each predator-victim pair so we separated conversations of each predator-victim pair into separate chat-logs. Eventually we get 60 different chat-logs. We keep 20% of the chat-logs as a test data set. The remaining 80% (48 out of 60) are used as the training set. In the chat-logs, only the predators' posts are labelled. We separated predators' posts from victims' post. The predators' posts were labelled with numbers 200, 600, and 900. Unclassified lines of predators' posts were assumed to have a classification of 000. The authors (McGhee et al., 2011) mentioned that the numbers (200, 600, 900) are not ordinal but they represent nominal values for IE, GR and AP categories. To reduce the confusion and ease of understanding the numbers are changed into their corresponding nominal acronym representations, that is IE for 200, GR for 600 and AP for 900.

The unlabelled (000) kind of chat-posts are labelled as BF category. Some of the chat-logs had XML-well-formed-ness error due to special characters like "<<" or "&". It took a bit of time to manually find them out and make corrections. Table 6.1 presents the information summary of posts count for all 60 chat-logs.

The pie chart in Figure 6.2 shows the distribution of predators' posts over the four categories. The BF type posts outnumber all other types of posts. It is obvious that the perpetrators take quite a bit of time to make friendship and to build a deceptive trust before the exploitation; hence the number of BF type posts is much more than the number of any other type of posts.

Table 6.1 : Information summary of posts count.

Items	Number of Posts
Total No of Posts	77,608*
Total No of Victims' Posts	38,577
Total No of Predators' Posts	37,726
Total No of BF category (000) Predators' Posts of all chat-logs	26,452
Total No of IE category (200) Predators' Posts of all chat-logs	1,983
Total No of GR category (600) Predators' Posts of all chat-logs	4,902
Total No of AP category (900) Predators' Posts of all chat-logs	4,389
Maximum number of predator-posts in any single chat-log	3,524
Minimum number of predator-posts in any single chat-log	30

*In some of the chat-logs in the Perverted-Justice.Com website the volunteer has written some comments in some lines. Those comments are also tagged as 'post' in the xml data of McGhee et al. (2011). Therefore our program counts them as posts though they are neither victims' nor predators' posts. However the program has counted specifically the victims' posts and predators' posts in a separate manner. Therefore if we add the number of the victims' posts and predators' posts together (76,303) it becomes less than the total number of all posts (77,608).

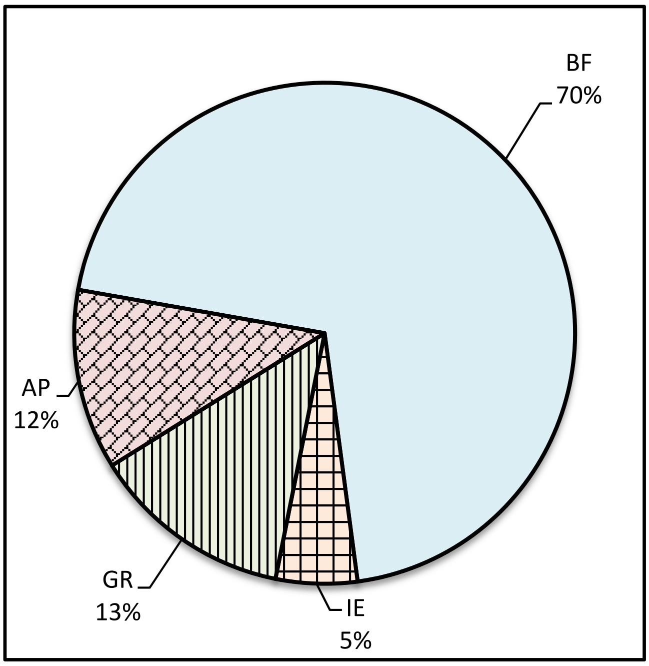


Figure 6.2: Distribution of predators' posts over the four categories of BF, IE, GR and AP.

6.2 Evaluation of the CEDM System

The performance of the three tier CE Evidence Detection Model (CEDM) is evaluated using standard metrics for text processing systems. Qualitative evaluation or effectiveness of the system is measured by using classic information retrieval (IR) notations including precision (P; also denoted by π), recall (R; also denoted by ρ), accuracy (A) and F-measure (F). To find out these measures confusion matrices are used along with False positive (FP) and False negative (FN). Explanations of these notations are given in the following subsections. Before that we discuss about training, test and development sets.

To measure the efficiency (computational performance) the whole set of chat-text data is divided into two sub-sets;

1. Training and Development set (D-set)
2. Test set (T-set).

Apart from being a training set, there is a part of D-set which is set aside and used for testing while developing the system. Finally when the system has been developed, all parameters have been set and the method is fully specified, the system runs on the test set (T-set). Because no information about the test set (T-set) is used in developing the system, the result should be indicative of actual performance in practice.

Cross-validation is a standard way of measuring the error rate of a learning scheme on a particular dataset. In this technique the available data is split into *n* approximately equal partitions or folds; each in turn is used for testing and the remainder is used for training. In this way every instance is used for training and testing. This mitigates any bias caused by a particular sample chosen for training or testing. We have used 10-fold cross validation in the classification experiments in this research.

6.2.1 Evaluation Metrics for Chat-Text Classifiers

The evaluation of text-classifiers (TC) incorporated in our system is done as like the evaluation of information-retrieval (IR) systems. It is typically conducted experimentally, rather than analytically. The evaluation of a TC usually measures its performance; for TC the performance is its effectiveness rather than its efficiency. Effectiveness refers to its ability to take the right classification decision.

Classification effectiveness is usually measured in terms of the classic IR notions of accuracy (A), precision (P), recall (R), and F-measure (F) adapted to the case of TC.

Accuracy (A):

Accuracy is the measure of how a classifier can correctly identify the class of the test instances. It is the proportion of the total number of predictions which are correct. For a classifier the accuracy (A) is given by:

$$A = \frac{Number\ of\ correctly\ predicted\ instances}{Total\ number\ of\ instances}$$

Precision (P):

Precision is viewed as the 'degree of soundness' of the classifier. It is measured as the proportion of the predicted instances which are correct. For a particular class C_i precision (P) is given by:

$$P = \frac{Number\ of\ correctly\ predicted\ instances\ as\ class\ C_i}{Total\ Number\ of\ Instances\ \ predicted\ as\ class\ C_i}$$

Recall (R):

Recall is viewed as the 'degree of completeness' of a classifier. It is measured as the proportion of the relevant instances which are correctly predicted. For a particular class C_i recall (R) is given by:

$$R = \frac{Number\ of\ correctly\ predicted\ instances\ as\ class\ C_i}{Total\ Number\ of\ Instances\ in\ class\ C_i}$$

Precision and recall correspondingly give a measure of 'perfection' and 'coverage' of a system. Therefore a system that is very strict in getting high perfection by increasing precision will invariably become poor in coverage with low recall. The vice versa is also true. To balance between these two (P and R) a combined measure called F-measure is used. F-measure will be discussed later in this section.

The evaluation metrics (A, P, R, and F) can be computed form 'Confusion Matrix' and 'Contingency Table' which are used to display the predictions of a classifier.

Confusion Matrix:

To visualize the prediction of classifiers a 'confusion matrix' is used. Examples of confusion matrices are shown in Table 6.2 and Table 6.3. Each row of the confusion matrix represents the instances in an actual class, while each column represents the instances in a predicted class. One benefit of a confusion matrix is that it is easy to see if the system is confusing two classes (that is commonly mislabelling one as another).

Table 6.2: A confusion matrix for a single-class classifier.

Class a		**Predicted**	
		YES	NO
Actual	YES	8	5
	NO	6	7

Table 6.3: A confusion matrix for a multi-class classifier.

		Predicted		
		Class a	Class b	Class c
Actual	Class a	8	5	0
	Class b	6	7	2
	Class c	0	1	9

The sum of a row in the confusion matrix provides the 'actual' number of instances in a class; the sum of a column provides the number of instances 'predicted' by a classifier. In the example confusion matrix in Table 6.3, actually there are 8+5+0 = 13 instances in 'class a', 15 instances in 'class b' and 10 instances in 'class c'. Out of the actual 13 instances of class a, the classifier predicted 8 as class a, and 5 as class b. For 15 instances of class b, it misclassified 2 as class c and 6 as class a. We can see from the matrix that the classifier system has trouble distinguishing between class a and class b, but can make the distinction between class c and other class types pretty well.

An 'overall accuracy' of a classifier can be computed using its confusion matrix. The numerator in the formula of accuracy, that is the 'number of correctly predicted instances', is the sum of the numbers in the diagonal cells of the confusion matrix. Therefore the overall accuracy of a classifier can be computed by :

$$A_{Overall} = \frac{Sum\ of\ the\ numbers\ in\ the\ diagonal\ cells\ of\ the\ confusion\ matrix}{Sum\ of\ the\ numbers\ in\ all\ cells\ of\ the\ confusion\ matrix}$$

It should be noted that, in a multiclass classifier, the overall accuracy of the classifier is computed from the confusion matrix, however, to compute individual class accuracies a contingency table should be used. Also, the precision and recall are first computed individually per class from the contingency table, then to get an overall precision and recall of the classifier the macro- and micro- averaging is used. These will be discussed next.

Contingency Table and Performance Measures:

To compute the performance measures like precision (P) and recall (R), the confusion matrix is converted to a 'contingency table'. The contingency table is made for each

Table 6.4: The contingency table for category c_i.

Category c_i		**Classifier Predicted**	
		YES	NO
Actual class	YES	TP_i	FN_i
	NO	FP_i	TN_i

individual class. It looks same as the confusion matrix for a single- or two- class classifier. For a multiclass confusion matrix each individual class has its own contingency table.

Consider a multi-class classification task has a set of classes C. When classifying test instances under an individual class $c_i \in C$ the followings can happen:

1. A number of test instances which are actually 'not *c*ᵢ' type but incorrectly classified under c_i type; this is denoted as FP_i (False Positives or errors of commission).

2. A number of test instances which are actually 'not c_i' type and correctly classified under 'not c_i' type; this is denoted as TN_i (true negatives).

3. A number of test instances which are actually c_i type and correctly classified under c_i type; this is denoted as TP_i (true positives).

4. A number of test instances which are actually c_i type but incorrectly classified under 'not c_i' type; this is denoted as FN_i (false negatives or errors of omission).

A contingency table displays the above situations. Such a table for c_i on a given test set is shown in Table 6.4. Estimates of accuracy (A_i), precision (P_i) and recall (R_i) for individual class c_i can be obtained from the contingency table as:

$$A_i = \frac{TP_i + TN_i}{TP_i + TN_i + FP_i + FN_i} \qquad \text{... Equation 6.1}$$

$$P_i = \frac{TP_i}{TPi + FP_i} \qquad \text{... Equation 6.2}$$

$$R_i = \frac{TP_i}{TPi + FN_i} \qquad \text{... Equation 6.3}$$

F-measures can also be computed from precision and recall as below:

$$F_1 = \frac{2PR}{P + R} \qquad \text{... Equation 6.4}$$

$$F_2 = \frac{5PR}{4P + R} \qquad \text{... Equation 6.5}$$

Formulas in Equation 5.3 and Equation 5.4 are derived from the following formula:

$$F_\beta = \frac{(\beta^2 + 1)PR}{\beta^2 P + R} \qquad \text{... Equation 6.6}$$

Where:

$\beta = \frac{R}{P}$ measures the comparative importance between recall vs. precision.

When recall and precision have the same importance then $\beta = 1$.

When recall has double importance than precision then $\beta = 2$.

Putting these values of β in Equation 6.6 Equation 6.4 and Equation 6.5 are derived.

An alternative measures of classification effectiveness, found in the ML (Machine Learning) literature is *error* (E) calculated as:

$$E = 1 - A = \frac{FP + FN}{TP + TN + FP + FN} \qquad \text{... Equation 6.7}$$

When a data set is unbalanced, that is when the number of samples in different classes greatly varies, the error rate (or accuracy) of a classifier does not show the true performance. This can easily be understood by an example; if there are 90 samples from class A and only 10 samples from class B, the classifier can easily be biased towards class A. If the classifier classifies all the samples as class A, the accuracy is 90%. This is not a good indication of the classifier's true performance. The classifier has a 100% recognition rate for class A but a 0% recognition rate for class B.

Accuracy (A) and error (E) are not widely used in text categorization (Sebastiani, 2002; Yang, 1999). In general, criteria different from effectiveness (P, R, and F) are seldom used in classifier evaluation.

Micro-Averaging and Macro-Averaging:

The 'individual performance' measures of each classes of a multi-class classifier are discussed in the previous sub-section. For obtaining estimates of 'overall performance' measure for all categories set in a multi-class classifier two different methods may be adopted; 'micro-averaging' and 'macro-averaging'.

In micro-averaging method, the performance measures (P and R) are obtained from a micro-averaged pooled table. The pooled table is created by averaging all individual contingency table of each category into one pooled contingency table. After that the micro-averaged precision (P_{micro}) and recall (R_{micro}) are computed from that new pooled contingency table using Equation 6.2 and Equation 6.3.

In macro-averaging method, for each category the local performance measures of precision and recall are computed first. Using those results of the different categories a global result is achieved by averaging over them.

These two methods (macro- and micro- averaged) may give different results in different circumstances, for example in a situation where some categories have a few positive training instances and some categories have many. Macro-averaging treats them equally, whereas micro-averaging result is influenced by the classifiers with high number of positive training instances. According to Manning et al. (2009) "Micro-averaged results are a measure of effectiveness on the large classes in a test collection. To get a sense of effectiveness on small classes, one should compute macro-averaged results".

6.2.2 Evaluation Metrics for Chat-Post Clusterer

In order to evaluate the quality of the clustering of chat-posts into the psychological phases we use standard evaluation metrics. Typical goal of clustering is to attain high intra-cluster similarity and low inter-cluster similarity, that is documents within a cluster are similar and documents from different clusters are dissimilar. This is an internal criterion for the quality of a clustering. But good scores on an internal criterion do not necessarily become good effectiveness in an application (Manning et al. 2009, pp356). Moreover it is expensive to achieve. Alternatively an external criterion can be used for cluster evaluation. A set of predefined classes can be used as a gold standard. Then it is computed how well the clustering matches the gold standard classes. Cluster evaluation may be either unsupervised or supervised. In the unsupervised evaluation no external information is used. In the supervised case external criterion is used to measure the goodness of the clustering. This section introduces some widely used clustering evaluation metrics.

Most of the unsupervised evaluation measures are only applicable to clusters represented using prototypes. Two exceptions are the Partition Coefficient (PC) (Bezdek, 1974) and the closely related Partition Entropy Coefficient (PE) (Bezdek, 1975; cited and used in Skabar and Abdalgader, 2011).

Partition Entropy Coefficient:

The Partition Entropy Coefficient (*PE*) is defined as:

$$PE = -\frac{1}{N}\sum_{i=1}^{N}\sum_{j=1}^{|W|}(u_{ij}\log_{\mathrm{a}} u_{ij}) \qquad \text{... Equation 6.8}$$

Where:
$W = \{w_1, w_2, \ldots\}$ is the set of clusters
N is the number of document objects
u_{ij} is the membership of instance *i* to cluster *j*.

The value of this index u_{ij} ranges from 0 to $\log_{\mathrm{a}}|W|$. The closer the value is to 0, the crisper the clustering is. The highest value is obtained when all of the u_{ij}s are equal.

Widely used four supervised clustering evaluation metrics are (a) Purity, (b) NMI, (c) Rand Index, and (d) F-Measure. To evaluate the chat-post clusterers we adopt these metrics from Manning et al. (2009).

For the followings the notations W , C and N is defined as:

$W = \{w_1, w_2, \ldots, w_k\}$ is the set of clusters,

$C = \{c_1, c_2, \ldots, c_j\}$ is the set of classes, and

N = Number of chat-post objects.

Purity:

Purity is a simple and transparent evaluation measure of a clusterer. It is comparable with the measure of accuracy in information retrieval.

Computationally:

$$\text{purity}(W, C) = \frac{1}{N}\sum_{k} \max_{j} |w_k \cap c_j| \qquad \text{... Equation 6.9}$$

Where most of the members of cluster w_k belong to the class c_j .

The problem of purity is its tendency to be high with a large number of clusters. It reaches an optimum value of 1 when each chat-post is in a singleton cluster. Thus, the quality of the clustering is compromised when the number of clusters are decreased. The measure of Normalized Mutual Information (NMI) minimizes this shortcoming by using normalization with entropy.

NMI:

Normalized Mutual Information (NMI) can be information-theoretically interpreted. Computationally NMI is defined as:

$$NMI(W, C) = \frac{I(W; C)}{[H(W) + H(C)]/2} \qquad \text{... Equation 6.10}$$

Where I is mutual information and H is entropy.

The Mutual Information (*I*) is defined as:

$$I(W;C) = \sum_{k}\sum_{j} P(w_k \cap c_j)\, log \frac{P(w_k \cap c_j)}{P(w_k)P(c_j)} \qquad \text{... Equation 6.11}$$

Where:

$P(w_k)$ is the probability of a chat-post being in cluster w_k,

$P(c_j)$ is the probability of a chat-post being in class c_j, and

$P(w_k \cap c_j)$ is the probability of the intersection of w_k and c_j (that is, cluster w_k be assigned the label as class c_j).

The estimate of each of the probabilities is the corresponding relative frequency. Thus, for maximum likelihood estimates of the probabilities the Equation 6.11 becomes:

$$I(W;C) \ = \sum_{k}\sum_{j} \frac{|w_k \cap c_j|}{N} \log \frac{N\,|w_k \cap c_j|}{|w_k|\,|c_j|} \qquad \text{... Equation 6.12}$$

The Entropy H is defined as:

$$H(W) = -\sum_{k} P(w_k) \log P(w_k) \qquad \text{... Equation 6.13}$$

Again, based on maximum likelihood estimates of the probabilities Equation 6.13 becomes:

$$H(W) = -\sum_{k} \frac{|w_k|}{N} \log \frac{|w_k|}{N} \qquad \text{... Equation 6.14}$$

The value of NMI ranges between 0 and 1. Because NMI is normalized, it is more reliable than other metrics and can be used upon a different number of clusters.

The measure of purity and NMI are based on statistics. Rand Index and F-measure are based on a combinatorial approach which considers each possible pair of chat-post objects. The target of a good clusterer is to assign a pair of chat-posts to the same cluster if and only if they are similar. Hence each pair can fall into one of the following four groups:

1. If both chat-posts belong to the same class and same cluster then the pair is a true positive (TP)
2. If both chat-posts belong to the same cluster but different classes then the pair is a false positive (FP)
3. If both chat-posts belong to the same class but different clusters then the pair is a false negative (FN);
4. If both chat-posts belong to different classes and different clusters, then the pair is a true negative (TN).

Using the TP, FP, FN and TN the Rand Index (RI) and F-measure are defined as follows:

Rand Index (RI):

The Rand index (RI) (Rand, 1971) measures the percentage of decisions that are correct. That is, it is simply accuracy and calculated as:

$$\mathrm{RI} = \frac{\mathrm{TP} + \mathrm{TN}}{\mathrm{TP} + \mathrm{FP} + \mathrm{FN} + \mathrm{TN}} \qquad \dots \quad \text{Equation 6.15}$$

The Rand index penalizes both false positive and false negative decisions during clustering.

F-measure:

The F-measure is an evaluation measure commonly used in the IR literature. It is also used as an evaluation metric for measuring the performance of clustering. F-measure is defined as the harmonic mean of Precision (P) and Recall(R). For clustering, these are computed using the combinatorial approach and finding the TP, FP, FN and TN as mentioned above. The formula for Precision (P), Recall(R) and F-measure (using the notations of the TP, FP, FN and TN) are provided in the previous section of 'Evaluation Metrics for Chat-Text Classifiers'.

Clustering Evaluation Metric used in this Research:

From the pie chart of Figure 6.2 it can be seen that the data set for clustering in the current research is highly unbalanced (70% BF, 5% IE, 13% GR, and 12% AP). A clusterer putting all the instances in one cluster (corresponding to BF Class) will achieve a very high purity. Therefore the purity measure would not necessarily correctly characterize the effectiveness of different clusterers; it suffers from the problem of biasing if the data is unbalanced. Rand Index is analogous to accuracy which also suffer from bias problem. F-measure is more usual in classification evaluation. The NMI measure is based on information theory. It is normalised so does not suffer from the problem of biasing. Therefore NMI is the best reliable evaluation measure for the current problem. We have used NMI to report the results of clustering experiments in this research.

6.2.3 Evaluation Metrics for Entailment and Evidence Producing

Evaluation Metrics for the Entailment tier and the Evidence Producing tier are the same as information retrieval evaluation metrics explained in the 'Evaluation Metrics for Chat-Text Classifiers' section. That is, performance of those phases are evaluated by computing precision (P), recall (R), F-measure (F) and accuracy(A). The formulas for these notions are provided in the previous section.

After describing the collection of data and evaluation metrics the experiments and results for each phase are provided in the following sections.

6.3 CE vs. Non-CE Classification Experiments

These experiments and results are also presented in a published conference paper Miah, Yearwood and Kulkarni (2011).

Objective:

The main objective of the experiments in this phase is to address the first question of the research sub-problems mentioned in section 1.2 of Chapter-1. We would like to investigate if the text classifiers are capable of distinguishing CE type chat-logs from different other types of chat-logs in a mixed data-set. This will work as shallow evidence. The chat-logs which are predicted as CE by the text classifiers will be highly suspected and will be taken forward for further analysis.

Data preparation and Pre-processing:

In the experiments the data set consists of text of a number of chat-log files. The logs include child exploiting offensive CE chat-logs and non-offensive Non-CE chat-logs. The total number of instances is 392. Among the 392 instances there are 200 CE and 192 Non-CE chat-logs. To make a balance between the CE and Non-CE chats, a set of 200 CE chats is randomly chosen from the collection of 516 CE chats from Perverted-Justice.com. The Non-CE portion of the data set consists of 107 general non offensive (GN) chat-logs and 85 sex fantasy type (SF) chat-logs. More details of the data set is discussed in previous section.

The chat-log files were pre-processed by cleansing and feature selection. In cleansing stage the usernames are removed. Then the text is converted into string vectors. Two types of features are selected for two sets of experiments. In one set of experiments the term based features are used. The other set of experiments use psychometric categorical information as features in the classifiers. These features are produced from the text of chat-logs by using the LIWC dictionary.

Method:

Three binary classifiers are used in the classification experiments of CE vs. Non-CE chat-logs. These are Naïve Bayes (NB), Decision Tree (DT), and Classification via Regression (CvR) classifiers. In each case stratified 10-fold cross-validation is used. For each fold there were 39 chat-logs in the test set and the remaining 353 chat-logs in the training set. The test set for each fold is randomly selected keeping the class proportion the same as the class proportion of the entire data set. Each and every chat-log is in the test-set in one of the 10-folds. The evaluation results of a classifier are averaged over the 10-folds.

To further investigate the classifiers ability to discriminate between different classes other classification set ups are also used. This includes a multi-class classification CE vs. SF vs. GN and two class classifications CE vs. SF and CE vs. GN. Stratified 10-fold cross-validation is also used in each of these classification set-ups. The size of each fold in the CE vs. SF vs. GN classification is: 39 chat-logs in the test set and 353 chat-logs in the training set, in CE vs. SF classification is: 30 in the test set and 255 in the training set, and, in CE vs. GN classification is: 31 in the test set and 276 in the training set.

The results are presented by confusion matrices and evaluated with the standard metrics of precision, recall and F-measure explained in Chapter-5. An analysis of the results is given in the following section.

Results and Analysis:

Table 6.5 shows the confusion matrices of the results from different classifiers. A number of experiments have been conducted with different combinations of the available chat data set. The combination of the data set is indicated in the corresponding Table. The Tables in the left side column show the confusion matrices of experiments with term-based feature set; the tables in the right side column are for experiments with feature set based on psychometric and word categorical information from LIWC. For example, the Table 6.5.1 corresponds to the results of the Experiment Set-1 for multi-class classification of CE vs. GN vs. SF.

Table 6.5: Confusion matrices for different classification experiments.

Experiments with Term-based feature set	Experiments with Psychometric and Word Category feature set

Table 6.5.1: Confusion Matrices for Experiment Set-1: CE vs. GN vs. SF

Total Number of Instances 392;
CE = 200, GN = 107, SF = 85

NB			DT			CvR			
CE	GN	SF	CE	GN	SF	CE	GN	SF	
168	28	4	181	7	12	188	2	10	CE
4	103	0	10	77	20	5	91	11	GN
2	57	26	13	22	50	5	17	63	SF

Table 6.5.2: Confusion Matrices for Experiment Set-2: CE vs. GN vs. SF

Total Number of Instances 392;
CE = 200, GN = 107, SF = 85

NB			DT			CvR			
CE	GN	SF	CE	GN	SF	CE	GN	SF	
187	7	6	174	12	14	189	10	1	CE
3	95	9	17	77	13	11	86	10	GN
14	13	58	11	12	62	20	13	52	SF

Table 6.5.3: Confusion Matrices for Experiment Set-3: CE vs. Non-CE

Total Number of Instances 392;
CE = 200, Non-CE = 192

NB		DT		CvR		
CE	Non-CE	CE	Non-CE	CE	Non-CE	
154	46	183	17	178	22	CE
10	182	19	173	17	175	Non-CE

Table 6.5.4: Confusion Matrices for Experiment Set-4: CE vs. Non-CE

Total Number of Instances 392;
CE = 200, Non-CE = 192

NB		DT		CvR		
CE	Non-CE	CE	Non-CE	CE	Non-CE	
188	12	170	30	182	18	CE
22	170	20	172	37	155	Non-CE

Table 6.5.5: Confusion Matrices for Experiment Set-5: CE vs. SF

Total Number of Instances 285;
CE = 200, SF = 85

NB		DT		CvR		
CE	SF	CE	SF	CE	SF	
179	21	179	21	188	12	CE
3	82	18	67	12	73	SF

Table 6.5.6: Confusion Matrices for Experiment Set-6: CE vs. SF

Total Number of Instances 285;
CE = 200, SF = 85

NB		DT		CvR		
CE	SF	CE	SF	CE	SF	
190	10	176	24	185	15	CE
18	67	17	68	24	61	SF

Table 6.5.7: Confusion Matrices for Experiment Set-7: CE vs. GN

Total Number of Instances 307;
CE = 200, GN = 107

NB		DT		CvR		
CE	GN	CE	GN	CE	GN	
171	29	186	14	185	15	CE
4	103	11	96	15	92	GN

Table 6.5.8: Confusion Matrices for Experiment Set-8: CE vs. GN

Total Number of Instances 307;
CE = 200, GN = 107

NB		DT		CvR		
CE	GN	CE	GN	CE	GN	
190	10	192	8	188	12	CE
4	103	14	93	21	86	GN

Experiment Set-1 uses 392 instances of chat-logs, where 200 are of CE type, 107 are of GN type and 85 are of SF type. The Table 6.5.1 shows the confusion matrices of the results from Naïve Bayes (NB), Decision Tree (DT), and Classification via Regression (CvR) classifiers in their respective columns. In the confusion matrices the rows specify true class and columns show the prediction of the classifier. Experiment Set-1 does not use psychometric information. It uses term-based feature set. On the other hand, Experiment Set-2 uses psychometric and word categorical information as the feature set with the same chat dataset and accomplish the same multi-class classification task as of Experiment Set-1. The results of Experiment Set-2 are in Table 6.5.2.

From the results it can be seen that psychometric and categorical information improves the performance of some classifiers. The results of Naïve Bayes (NB) classifier are in their corresponding columns in Table 6.5.1 and Table 6.5.2.

In those tables the correctly detected chats for the CE types are increased by 11.31% (from 168 to 187). Moreover incorrect classification of the CE type chats are decreased by 59.38% (from 32 to 13). Similar improvements are found in all results with NB classifiers using psychometric information. Results of Classification via regression (CvR) classifier is also improved in some cases (CvR columns of Table 6.5.2, Table 6.5.4 and Table 6.5.8) when psychometric information feature set is used. In those cases it is detecting more CE chats, however at the same time it is predicting more chats as CE which are actually not CE. For the Decision Tree (DT) classifiers, psychometric information does not make any improvement.

Comparing the results of multiclass classification with binary classification (Table 6.5.2 and Table 6.5.4) it is found that the effectiveness of the classifiers is almost the same in regards of correctly predicting CE chats. For example, NB classifier correctly detects CE chats 187 times in multiclass classification and 188 times in binary classification. Regarding the false negative case the figure is also very low and almost equal, 13 and 12. This indicates the consistency of the reliability of NB classifier with psychometric features to catch the CE chats regardless of the multi- or binary-classification settings.

The results of Experiment Set-5 and 6 (Table 6.5.5 and Table 6.5.6) and Experiment Set-7 and 8 (Table 6.5.7 and Table 6.5.8) shows that classifiers find more difficulties to distinguish CE vs. SF chats than to distinguish CE vs. GN chats. For example, the result of NB using LIWC (NB of Table 6.5.6 and Table 6.5.8) shows that, incorrectly classified instances in CE vs. SF is 9.8% ((10+18)/285 = 0.098) which is much higher than 4.5% ((10+4)/307 = 0.045) in CE vs GN. Results of other classifiers also support this idea. This supports our idea that SF chats are nearer to CE chats than GN chats as mentioned in section 2.4.2 of Chapter-2.

The aim of current research is to detect CE chats. Therefore the classifier should not spare any suspected chat-log. It has to be very strict in catching CE chats even if it makes some misjudgement about some other non CE chats. That means the classifier can be flexible in Type-I error (False positive) but should minimize Type-II error (False negative) as much as possible. Considering this, we try to find out the classifier which is performing best among the three classifiers. In multiclass classifications, in the case of term-based feature set (Table 6.5.1) CvR is detecting the highest number of CE chats. It is predicting 188 chats as CE whereas prediction by NB is 168 and prediction by DT is 181. When psychometric information are used (Table 6.5.2) both NB and CvR are competing with each other. Both of them are detecting almost the same number of CE chats (187 and 189). The number of false negative is also about the same (13 and 11).

For the three tier CE Evidence Detection Model (CEDM) a classifier is required which is capable of effectively catching the CE chats out of the Non-CE chats. In binary classification in Table 6.5.3 and Table 6.5.4, NB with psychometric information (Table 6.5.4), is performing the best. It is detecting 188 CE chats out of 200; CvR (Table 6.5.4) is catching 182, and DT (Table 6.5.3) is catching 183.

The recall and accuracy for CE vs. Non-CE classification are shown in the column charts in Figure 6.3. The charts show that NB with psychometric and word category feature outweighs other classifiers. With those features NB achieves the recall as high as 94.0% which is 3% better than the nearest result of CvR. The accuracy of NB is also as high as 91.3% which is 5.4% better than the result of CvR which achieved 85.9%.

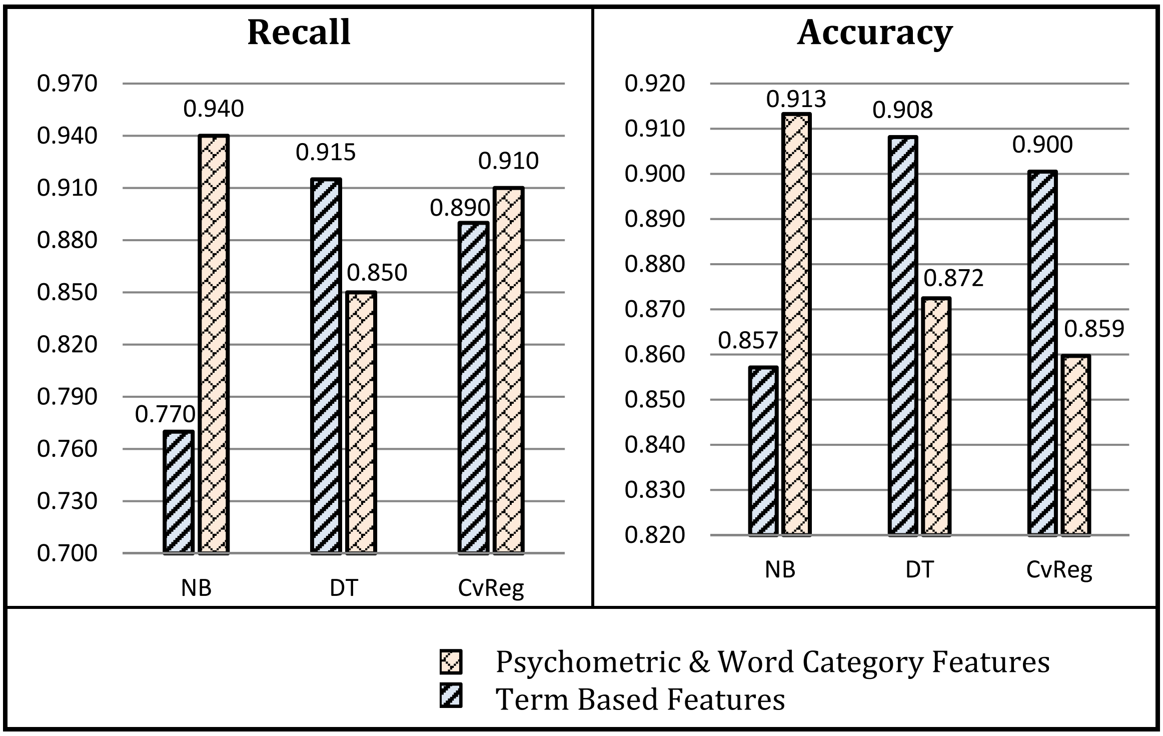


Figure 6.3: Recall and Accuracy of different classifiers for CE vs. Non-CE Classification.

Table 6.6: Summary of result metrics for CE vs. Non-CE classification.

	With Term Based Features			With Psychological & Word Category Features		
	NB	J48DT	CvReg	NB	J48DT	CvReg
Recall	0.770	0.915	0.890	**0.940**	0.850	0.910
Precision	**0.939**	0.905	0.912	0.900	0.894	0.831
Accuracy	0.857	0.908	0.900	**0.913**	0.872	0.859
F-Measure	0.846	0.910	0.901	**0.920**	0.871	0.868

Table 6.6 summarises the results metrics for CE vs. Non-CE classification. This table shows that NB with psychometric and word category features is performing best in F-measure (92.0%) along with accuracy (91.3%) and recall (94.0%). The only metric where it is performing a little lower is the precision, however is still staying within 90% and comparable with the other classifiers. Therefore a NB with psychometric and word category features would be the best choice for the CE vs. Non-CE classification phase in the 'tier one' of CEDM system.

6.4 Predator vs. Victim Classification Experiments

Objective:

The main objective of the experiments in this phase is to address the second question of the research sub-problems mentioned in section 1.2 of Chapter-1. In these experiments we would like to investigate as to whether the participants can be categorized into a CE predator and a victim of CE. This can be an important evidential artefact as the act of child exploitation involves an adult CE perpetrator and a victim of CE. By the term 'victim' we would like to represent 'opposite of a predator' who can be a real-child-victim or can be a person the predator believes to be a child he is preying on.

Data preparation and Pre-processing:

Using the chat-data set from Perverted-Justice.com, Pendar (2007) conducted a classification task of 'predator vs. victim' which is similar to the current experiment. The author mentioned that he used 701 chat-logs, however there were no more than 516 chat-logs available in the website at the time of data collection of this current research in 2011. It is assumed that the author may have collected the extra chat-logs directly from the people behind the website or from their offline database. It has been mentioned in section 6.1.2.1 of Chapter-5 that 516 chat-logs have been collected for this current research from the Perverted-Justice.com website. From that collection 489 chat-logs are used in this current experiment of CE Predator vs. Victim classification. Due to formatting and some unknown errors some other chat-logs could not be used. All the predators in those 489 chat-logs were adults and convicted for child exploitation charges. Therefore the chat-texts produced by them can represent a data-set of CE predators. The victims were trained volunteers posing as children. They used such linguistic terminologies that all the predators believed that each of them is grooming a child. As the main aim of this task is to find evidence against the predators therefore, in the absence of real child-victims the chat-texts produced by the trained volunteers can be acceptable as a data-set for the 'non-predator' or 'victims' part of the classification task.

All the chat-posts of the 489 chat-logs are divided into data groups of adult-predators and child-victims. After tokenizing there were a total of 56,061 tokens in the data-set. The user names were deleted. No other cleansing was done as that may also delete the 'emoticons' which are important features in chat-text. We used unigrams, bigrams and trigrams from the training data as features. The minimum term frequency per class was 2. In total, 21,966 unigrams, 204,123 bigrams, and 366,048 trigrams were extracted.

Method:

Naïve Bayes (NB), Decision Tree (DT), Classification via Regression (CvR) and k-Nearest Neighbour (kNN) classifiers are used on the above mentioned data set for the classification of Predator vs. Victim in this current experiment. For kNN classifier we varied the number of neighbours (k) from 5 to 30 increasing 5 in each step. We used the stratified 10-fold cross validation in the experiment. The size of each fold was : 49 chat-logs in the test set and 440 chat-logs in the training set. The following section presents the results of the classification experiment.

Results and Analysis:

The results of our experiment with kNN classifiers are presented in Table 6.7. The table shows the F-measures achieved by using different number of neighbours (k-values) with different n-gram feature sets. From the table we can see that the overall best result produced by kNN classifier is 0.931 which is achieved with unigrams and 30 neighbours (k = 30). The individual best results for bigrams and trigrams are 0.732 and 0.691 with k-values of 25 and 15 respectively.

Table 6.7: kNN Results for Predator vs. Victim classification.

Number of Neighbours (k)	Unigram	Bigram	Trigram
5	0.916	0.618	0.647
10	0.907	0.637	0.546
15	0.907	0.573	**0.691**
20	0.918	0.728	0.599
25	0.919	**0.732**	0.607
30	**0.931**	0.708	0.541

Table 6.8: Results from Different Classifiers for Predator vs. Victim classification.

Classifier	Unigram	Bigram	Trigram
Naïve Bayes	**0.960**	**0.965**	**0.976**
Decision Tree	0.870	0.886	0.853
Classification via Regression	0.897	0.893	0.828
kNN (Our Expt)	0.931	0.737	0.692
kNN (Pendar's Expt)	0.854	0.779	0.943

Table 6.8 presents the results of our experiments with different classifiers. The left-most column shows the classifiers' names, the right three columns show the results of the corresponding classifier using different feature sets of unigrams, bigrams and trigrams. The kNN results shown in this table are the individual best results of unigrams, bigrams and trigrams. From Table 6.8 it can clearly be seen that Naïve Bayes (NB) is performing the best among the classifiers. NB achieved the overall highest F-measure of 0.976 by using trigram features. Using the other feature-sets of unigrams and bigrams the results of NB are also highest in each case.

Table 6.8 also shows the kNN results of Pendar's experiment collected from the article Pendar (2007). The results of kNN classifiers are slightly different in our experiment than the results of the experiment conducted by Pendar. The reasons may be due to different data and experiment settings. Pendar used 701 PJ chat-logs whereas we used a subset 489 of those chat-logs. We did not use any stop-list; Pendar used his own stop-list at the time of feature selection. He also used feature reduction by using the average odds ratios and finally chosen 5000, 7,500 and 10,000 features only. On the other hand we did not use any feature reduction scheme, instead we have used all the available features. We wanted to observe the classifiers' behaviour with all the features. As the results of our experiment do not significantly degraded, instead improved, than the results of Pendar's experiment we preferred not to take the extra burden of feature reduction scheme.

The best result achieved by Pendar (2007) was F-measure of 0.943 with kNN classifier with k = 30 neighbours. As mentioned before, the best F-measure in our experiment is 0.976 by Naïve Bayes with trigrams. This result outperforms the result of Pendar by 3.3%. Therefore a NB classifier with trigram features would be preferable in the 'Predator vs Victim' classification phase.

6.5 Clustering Experiments

Objective:

The clustering experiments aim to address the third question of the research sub-problems mentioned in Chapter-1. We would like to investigate whether a clustering method, without any supervision, organizes the posts of a chat into the pattern of CE profile identified in the psychological literature. By clustering the posts into the CE psychological stages the CE profile can be learned. This can be another evidential artefact in the CE detection process. This can also assist locating particular CE evidences by establishing association of the clusters to the CE stages.

Data preparation and Pre-processing:

The set of 60 chat-logs described in Section 5.5.2.2 of Chapter-5 is used as data-set in this experiment. It has already been mentioned that the posts of the chat-logs are labelled by human analysts with the four CE psychological stages of BF, IE, GR and AP. Out of the 60 chat-logs 12 (20%) are kept as a test data set. The remaining 80% (48 out of 60) are used as the training set. It has been mentioned before that the training set is used to construct the CE Psychological dictionary. The test set is kept aside so that the dictionary building process does not know anything about the test set. For testing the effectiveness of the clusterers the test data set is used in the clustering experiments.

Method:

In these experiments we would like to learn the pattern of CE profile of the perpetrators not the victims. Therefore the chat-posts written by the perpetrators are to be analysed by clusterers. This assists to trace the evidence of CE act left by the perpetrators. Different clusterers may provide different results in learning the pattern. To investigate and compare the effectiveness of the clusterers, 5 different clustering algorithms including: PsyHAC, LSA-HAC, Traditional-HAC, *K*-means, and EM (Expectation-Maximization) are used to organize the posts of the chat-logs into four clusters resembling the four stages of CE psychological context.

For the PsyHAC clustering, the CEPsy Similarity measure is used for computing the pairwise similarity between posts. Using centroid-measure the chat-posts are merged together to form clusters. Details of the CEPsy Similarity and the PsyHAC Clustering Algorithm is provided in Chapter-3.

The LSA-HAC uses hierarchical agglomerative clustering algorithm. For measuring the similarity between a pair of chat-post-objects the Latent Semantic Analysis (LSA) is used. The procedure of LSA-HAC clusterer is described in Chapter-5 Section 5.3.1.

K-means, EM and Traditional-HAC clustering algorithms use conventional measures to compute the similarity (or distance) between a pair of chat-post objects. Procedures of these clusterers are provided in Chapter-2.

For evaluation of the clusterer the Normalized Mutual Information (NMI) is used. The NMI metric is explained in Chapter-5 Section 5.6.2.

After clustering, the association of clusters (w_i) to CE evidence classes (c_j) are obtained by finding the suitable combination for mutual association of w_i and c_j. The combination is chosen from all the combinations of $w_i c_j$ for which the overall accuracy is highest. The detailed procedure is described in section 5.3.1 of Chapter-5. The result of this association is similar to the result of a classification task. Therefore evaluation of this part of the experiments uses Precision (*P*), Recall (*R*), and Accuracy (*A*).

The results for 'clustering' and the 'CE evidence association' both are described in the following sections.

Results and Analysis of Clustering:

Without any supervision if the posts of a chat-log automatically be organized by the clusterers into the predefined CE psychological contextual types BF, IE, GR and AP then it will be evidence that the chat is following the CE profile. It has been mentioned in Chapter-5 that the Normalized Mutual Information (NMI) would be better than the other metrics to compare the effectiveness of clusterers. Table 6.9 represents the NMI results produced by the test set for comparing the clustering results by PsyHAC clusterer with other clusterers. The NMI results in that table can be considered as an estimate of how a chat-log is following the CE profile. All the chat-logs in the test set were from CE type chats therefore all of them supposed to follow the CE profile. A clusterer having highest NMI value is best capturing this information. From the Table 6.9[1] we can see that the PsyHAC clusterer is having the highest

[1] Due to limitation of space only first 8 characters are used as chat-log names.

Table 6.9: Comparison of NMI results of different clusterers for the test-set.

Sl No	Chat-log Name	PsyHAC	LSA-HAC	Traditional-HAC	K-means	EM
0	40Posts	81.78%	26.60%	12.00%	16.36%	16.00%
1	armysgt1	33.09%	21.42%	16.50%	17.56%	20.47%
2	arthinic	29.90%	3.71%	3.72%	1.75%	2.88%
3	fighting	15.16%	4.56%	14.39%	14.39%	18.96%
4	flxnonya	27.84%	11.26%	10.76%	19.84%	14.40%
5	icepirat	25.02%	7.75%	6.12%	7.85%	4.80%
6	italianl	44.48%	18.44%	22.48%	22.74%	30.29%
7	jleno9	22.09%	5.81%	8.65%	7.80%	11.63%
8	jon_rave	31.10%	13.60%	11.45%	14.42%	12.70%
9	manofdar	36.56%	22.89%	20.00%	11.00%	16.00%
10	sebastia	18.97%	23.45%	2.30%	10.10%	7.20%
11	thedude4	14.40%	13.16%	2.55%	7.52%	3.57%
12	user1945	20.57%	13.72%	13.37%	7.89%	9.54%
Average (without 40Posts)		26.60%	13.31%	11.02%	11.90%	12.70%

average NMI value hence would be the best choice in capturing the CE profile. The Table 6.9 also presents results from the clusterers for 40Posts chat data set. As mentioned before that the 40Posts contains equal number of chat posts from each of the four categories BF, IE, GR, and AP; hence works as a balanced chat posts set. Though 40Posts data is a synthesized chat log, not a real chat log, it has been included here to see how the new clusterer behaves with a balanced chat posts set. Form the table it can be seen that the new clusterer may work very good with a balanced chat posts whereas the other traditional clusterers may not. The average calculated in Table 6.9 does not include the results from 40Posts.

Table 6.10 provides the statistical measures showing pairwise comparisons of the other clusterers with the PsyHAC clusterer. A one-tail t-test is done with $\alpha = 0.005$ (99% confidence level) considering the null hypothesis h_0 as "PsyHAC does not make any significant improvement". The critical t-value for this test is 2.818. Table 6.10 shows that the t statistics values for all cases are more than the critical t-value.

Table 6.10: Pairwise statistical comparison of different clusterers with PsyHAC clusterer.

	PsyHAC vs LSA-HAC	PsyHAC vs Traditional-HAC	PsyHAC vs K-means	PsyHAC vs EM
Mean Difference	0.1328	0.1557	0.1469	0.1390
t-Statistics	4.0250	4.8156	4.6974	3.9946
p Value	0.0003	0.0000	0.0001	0.0003
Effect Size (*d*)	5.9453	7.1131	6.9385	5.9004

Moreover, the very low *p*-values (much less than α) in Table 6.10 also make it clear that the probability of h_0 to be valid is extremely low. Therefore we discard the null hypothesis (h_0) and conclude that PsyHAC shows 'significant' improvement over the other clusterers. Cohen's effect size (*d*) is also calculated and shown in Table 6.10. Usually Cohen's benchmark for effect size are $d = 0.2$ for 'small', $d = 0.5$ for 'medium', $d = 0.8$ for 'large' (Ellis, 2010, p. 41). The values of effect size in Table 6.10 are much greater than 0.8, therefore PsyHAC gives comparatively 'large' improvement in all the cases.

From the mean differences in Table 6.10 we get a quantitative measure of improvement. The PsyHAC clusterer has achieved a 'quantitative' improvement on an average of 13.28% compared to LSA, 15.57% compared to EM, 14.69% compared to K-Means and 13.9% compared to traditional HAC. Therefore it can be concluded that, to understand CE profile through clustering the chat posts into the CE psychological stages, the new PsyHAC significantly improves the effectiveness. This improvement is upto 15.57% compared to the traditional clusterers.

Results and Analysis of Association of Clusters to CE Evidence:

After association of Clusters (w_i) to CE evidential classes of BF, IE, GR and AP the result metrics (*A*, *P* and *R*) are computed using the formulas explained in the evaluation metric section of Chapter-5 and presented in Table 6.11. The results show that this approach achieved an overall accuracy of 55.8% averaged among all the chat-logs of

Table 6.11: Results of association of clusters to CE evidence.

A = Acc = Accuracy; *P*= Precision; *R* = Recall;

	Chat-log Name	Overall Acc		BF	IE	GR	AP	Avg_{BIGA}	Avg_{IGA}
1	armysgt1	0.604	*A*	0.667	0.917	0.854	0.771	0.802	0.847
			P	0.629	1.000	0.333	0.200	0.541	0.511
			R	0.880	0.556	0.167	0.125	0.432	0.282
2	arthinic	0.775	*A*	0.786	0.905	0.905	0.953	0.887	0.921
			P	0.836	0.196	0.880	0.200	0.528	0.425
			R	0.869	0.409	0.646	0.105	0.507	0.387
3	fighting	0.835	*A*	0.835	0.983	0.932	0.920	0.917	0.945
			P	0.831	1.000	1.000	1.000	0.958	1.000
			R	1.000	0.143	0.172	0.035	0.337	0.117
4	flxnonya	0.517	*A*	0.707	0.862	0.828	0.638	0.759	0.776
			P	0.565	0.000	0.778	0.400	0.436	0.393
			R	0.650	0.000	0.467	0.625	0.435	0.364
5	icepirat	0.633	*A*	0.676	0.936	0.899	0.755	0.817	0.863
			P	0.843	0.000	0.333	0.367	0.386	0.233
			R	0.674	0.000	0.462	0.733	0.467	0.398
6	italianl	0.576	*A*	0.758	0.697	0.818	0.879	0.788	0.798
			P	0.625	0.571	0.444	1.000	0.660	0.672
			R	0.833	0.364	0.800	0.200	0.549	0.455
7	jleno9	0.534	*A*	0.534	0.890	0.767	0.877	0.767	0.845
			P	0.447	1.000	0.667	1.000	0.778	0.889
			R	0.724	0.111	0.640	0.100	0.394	0.284
8	jon_rave	0.617	*A*	0.635	0.800	0.878	0.922	0.809	0.867
			P	0.691	0.111	0.474	0.833	0.527	0.473
			R	0.767	0.063	0.692	0.385	0.477	0.380
9	manofdar	0.500	*A*	0.500	0.838	0.825	0.838	0.750	0.833
			P	0.483	0.833	1.000	0.000	0.579	0.611
			R	0.737	0.294	0.333	0.000	0.341	0.209
10	sebastia	0.529	*A*	0.660	0.842	0.687	0.869	0.765	0.799
			P	0.694	0.078	0.619	0.412	0.451	0.370
			R	0.444	0.250	0.599	0.757	0.512	0.535
11	thedude4	0.563	*A*	0.702	0.849	0.653	0.922	0.782	0.808
			P	0.692	0.667	0.500	0.000	0.465	0.389
			R	0.698	0.150	0.577	0.000	0.356	0.242
12	user1945	0.567	*A*	0.701	0.897	0.722	0.814	0.784	0.811
			P	0.674	0.000	0.546	0.400	0.405	0.315
			R	0.689	0.000	0.600	0.400	0.422	0.333
Average of 12 Chat-logs		0.558	*A*	0.628	0.801	0.751	0.781	0.740	0.778
			P	0.616	0.420	0.583	0.447	0.516	0.483
			R	0.690	0.180	0.473	0.267	0.402	0.307
			F1	0.6508	0.2519	0.5223	0.3339	0.4523	0.3751

the test-data set. The maximum overall accuracy is 83.5% and a minimum is 50%. The individual accuracies for each CE evidential psychological stages BF, IE, GR and AP averaged among all the test-data set show that the accuracy for the IE type is highest 80.1%, The individual average accuracy for the other types are 62.8% for BF, 75.1% for GR and 78.1% for AP type. Macro-averaged individual accuracy across BF, IE, GR and AP (Avg_{BIGA}) is 74.0% and across IE, GR and AP (Avg_{IGA}) is 77.8%. This means that this approach more accurately predicts the critical CE stages of IE, GR and AP than the apparent innocent stage of BF.

The approach of associating Clusters (w_i) to CE evidential classes is similar to categorizing each post of a test chat-log to one of the CE psychological stages of BF, IE, GR and AP. After this formation we can compare the results of this approach with the results of ChatCoder 2 of McGhee et al. (2011). Table 6.12 provides the accuracies achieved by our approach compared with ChatCoder 2. There are 6 common chat-logs among the 33 chat-logs used by the ChatCoder 2 and the 12 chat-logs used as the test-data set in our experiments. We compute the accuracy and compare among the common chat-logs. The formula of individual accuracy used here is different than the individual accuracy formula used for the results in Table 6.11. Because the results in Table 6.12 are used to compare with the results of McGhee et al. (2011), to have a fair comparison, the individual accuracy formula (Equation 6.16) follows the formula of accuracy used in McGhee et al. Therefore the values in the individual accuracy columns are different in Table 6.11 and Table 6.12.

In Table 6.12 the accuracy of PsyHAC for a CE psychological stage *i* is measured as:

$$A_i = \frac{n_i}{N_i} \times 100\% \qquad \text{... Equation 6.16}$$

where $i \in \{\text{BF}, \text{IE}, \text{GR}, \text{AP}\}$.

The N_i represents the total number of posts originally labelled as a class of the psychological stage *i* in the chat-log. After clustering by PsyHAC each of the clusters is considered as one of the four psychological stages according to the comparative maximum presence of posts of a type that gives the highest overall accuracy. In the accuracy formula (A_i) the term n_i represents the number of posts of a CE psychological stage *i* found in a cluster considered as the same CE psychological type.

Table 6.12: Comparison of accuracies between PsyHAC and ChatCoder2.
PH = PsyHAC; CC2 = Chatcoder2

Sl No	Chat-log	System	BF	IE	GR	AP	**Avg$_{\mathbf{BIGA}}$**	**Avg$_{\mathbf{IGA}}$**
1	fighting	PH	100.0%	16.7%	19.2%	3.6%	34.9%	13.2%
		CC2	86.4%	16.7%	36.7%	35.9%	43.9%	29.8%
2	icepirat	PH	42.1%	0.0%	60.0%	95.7%	49.4%	51.9%
		CC2	83.0%	20.0%	69.2%	33.3%	51.4%	40.9%
3	italianl	PH	60.0%	66.7%	100.0%	20.0%	61.7%	62.2%
		CC2	93.3%	0.0%	50.0%	25.0%	42.1%	25.0%
4	jon_rave	PH	56.4%	71.4%	11.1%	75.0%	53.5%	52.5%
		CC2	95.9%	0.0%	31.3%	53.9%	45.3%	28.4%
5	sebastia	PH	29.3%	57.1%	68.4%	82.4%	59.3%	69.3%
		CC2	83.8%	12.5%	41.5%	32.4%	42.6%	28.8%
6	user1945	PH	41.7%	0.0%	64.3%	46.2%	38.0%	36.8%
		CC2	79.7%	0.0%	66.7%	53.9%	50.1%	40.2%
	Average	PH	54.9%	35.3%	53.8%	53.8%	49.5%	47.7%
		CC2	87.0%	8.2%	49.2%	39.1%	45.9%	32.2%

In Table 6.12 the average accuracies are calculates as bellow:

Average accuracy across BF, IE, GR, and AP:

$$\mathrm{Avg}_{\mathrm{BIGA}} = \frac{1}{4} \sum_{\mathrm{i} \in \{\mathrm{BF,IE,GR,AP}\}} A_i \qquad \text{...Equation 6.17}$$

Average accuracy across IE, GR, and AP:

$$\mathrm{Avg}_{\mathrm{IGA}} = \frac{1}{3} \sum_{\mathrm{i} \in \{\mathrm{IE,GR,AP}\}} A_i \qquad \text{... Equation 6.18}$$

From Table 6.12 it can be seen that the average accuracy across BF, IE, GR, and AP ($\mathrm{Avg}_{\mathrm{BIGA}}$) of PsyHAC is 3.6% above the average accuracy of Chatcoder2. The accuracies of the BF type posts are high in Chatcoder2, which increases its overall accuracy. However, the BF type chat-posts are quite passive in their level of indicativeness of child exploitation and contribute mainly to keep the conversation going. As Mcghee et al. considered the BF type posts to be the innocent texts, the other three types are more critical for indication of child exploitation. A system is potentially more useful if it focuses attention on the more critical psychological stages (IE, GR and AP). Table 6.12 shows that the average accuracy of PsyHAC among IE, GR and AP ($\mathrm{Avg}_{\mathrm{IGA}}$) is 15.5% higher than Chatcoder2.

6.6 Entailment Experiments

Objective:

The entailment experiments address the fourth question of the research sub-problems mentioned in Chapter1. The aim is to investigate whether the CE (Child Exploiting) evidence detection problem can be framed into a manageable problem of Textual Entailment on chat-logs. Detection of CE evidence in chat text requires locating the texts which prove the evidential propositions predefined by law and psychological literature. For this purpose a soft entailment approach is developed in Chapter-4. Throughout the current set of experiments our objective is to compute how effectively our soft entailment approach locates the particular evidence from a suspected chat-log.

Data preparation and Pre-processing:

As used in the clustering experiments, the set of 60 chat-logs described in Section 5.5.2.2 of Chapter-5 is also used as data-set in this set of experiments. Out of the 60 chat-logs 12 (20%) are kept as a test data set. The remaining 80% (48 out of 60) are used as the training set. Details about the data-set are explained in Chapter-5. As have been mentioned the chat-posts are annotated by human analysts into the CE evidential psychological contexts of BF, IE, GR and AP using the definitions provided by psychological literatures.

Method:

Our approach for recognition of CE entailment (RCE) is explained in section 4.4.2 of Chapter-4. Using that approach a set of chat-posts is selected from the training chat-logs as surrogates for a hypothesis (H). The hypothesis (H) corresponds to a particular CE evidence context defined by the psychological literature. From the test chat-log an entailment relationship is estimated for the text (T) of each post. If T has an average similarity with the surrogates above a predefined threshold then it is estimated that T entails H. The similarity is computed by the CEPsy Similarity measure explained in section-3.3 of Chapter-3. When T entails H it is represented by YES and otherwise it is represented by NO. As a result for each chat-log of the test data-set individual YES-NO (YN) entailment hashes are produced for each of the critical CE psychological contexts IE, GR and AP. The other CE psychological context BF tends to be innocent and not evidentially critical for indicating the presence of CE; therefore is not included in this experiment. Evaluation metrics, explained in Chapter-5, are computed from the YN hashes. Results are presented with the evaluation metrics of accuracy, precision, recall, F1 measure and F2 measure in the following section.

Results and Analysis:

Table 6.13 , Table 6.14 and Table 6.15 show the results of the entailment experiments. Each of the table presents the evaluation metrics of accuracy, precision, recall, F1 measure and F2 measure for the entailment contexts of IE, GR and AP for all the 12 chat-logs of the test data-set.

Table 6.13 shows that the average recall for GR contextual evidence is 88.7% with a minimum of 62.5% to a maximum of 100%. The accuracy is a minimum of 45.6% to a maximum of 76.3% with an average of 67%. This means that our approach of soft entailment (RCE) can capture on average 88.7% of the total GR type evidence present in a chat-log, and sometimes it can capture all of them. It also finds some other posts as GR evidence though they are originally not GR posts. However on average 67% of time this can correctly find a post whether it is a GR evidence or not. Precision for entailing GR evidence is on average 38.2% with a maximum of 58.8%.

Table 6.13: Results of GR evidence entailment.

	Chat-log Name	Accuracy	Precision	Recall	F1Measure	F2Measure
1	armysgt1	0.646	0.261	1.000	0.414	0.638
2	arthinic	0.456	0.278	0.965	0.432	0.646
3	fighting	0.624	0.163	0.862	0.274	0.464
4	flxnonya	0.724	0.484	1.000	0.652	0.824
5	icepirat	0.702	0.169	0.846	0.282	0.470
6	italianl	0.727	0.357	1.000	0.526	0.735
7	jleno9	0.671	0.510	1.000	0.676	0.839
8	jon_rave	0.652	0.227	0.625	0.333	0.463
9	manofdar	0.663	0.417	0.714	0.527	0.625
10	sebastia	0.666	0.560	0.836	0.671	0.761
11	thedude4	0.743	0.588	0.895	0.710	0.810
12	user1945	0.763	0.575	0.900	0.702	0.809
	Average	0.670	0.382	0.887	0.516	0.674

Table 6.14: Results of AP evidence entailment.

	Chat-log Name	Accuracy	Precision	Recall	F1Measure	F2Measure
1	armysgt1	0.542	0.250	0.875	0.389	0.583
2	arthinic	0.263	0.040	0.842	0.076	0.167
3	fighting	0.558	0.134	0.793	0.229	0.399
4	flxnonya	0.500	0.314	0.688	0.431	0.556
5	icepirat	0.633	0.271	0.767	0.400	0.561
6	italianl	0.636	0.267	0.800	0.400	0.571
7	jleno9	0.411	0.177	0.900	0.295	0.495
8	jon_rave	0.591	0.196	0.846	0.319	0.509
9	manofdar	0.613	0.091	0.750	0.162	0.306
10	sebastia	0.423	0.132	0.865	0.229	0.409
11	thedude4	0.514	0.008	1.000	0.017	0.040
12	user1945	0.557	0.241	0.867	0.377	0.570
	Average	0.520	0.177	0.833	0.277	0.431

Table 6.15: Results of IE evidence entailment.

	Chat-log Name	Accuracy	Precision	Recall	F1Measure	F2Measure
1	armysgt1	0.542	0.217	0.556	0.313	0.424
2	arthinic	0.280	0.034	0.591	0.064	0.138
3	fighting	0.550	0.031	0.714	0.060	0.132
4	flxnonya	0.448	0.069	0.286	0.111	0.175
5	icepirat	0.628	0.046	0.300	0.079	0.142
6	italianl	0.515	0.308	0.364	0.333	0.351
7	jleno9	0.274	0.042	0.222	0.070	0.119
8	jon_rave	0.574	0.091	0.308	0.140	0.208
9	manofdar	0.463	0.067	0.118	0.085	0.102
10	sebastia	0.412	0.019	0.250	0.035	0.073
11	thedude4	0.514	0.174	0.525	0.261	0.374
12	user1945	0.495	0.044	0.286	0.076	0.135
	Average	0.475	0.095	0.377	0.136	0.198

For GR evidence entailment results in Table 6.13 the average F1 and F2 measures are 0.516 and 0.674 respectively. The minimum and maximum of F1 measure are 0.274 and 0.710, and for F2 measure those are 0.463 and 0.839.

Table 6.14 shows the entailment result for the AP context. The recall for AP context is on average 83.3% with a minimum of 68.8% and a maximum of 100%. The average accuracy is 52% with a minimum and maximum of 26.3% and 63.6%. The maximum F2 measure is 0.583 with an average of 0.431.

Effectiveness of our soft entailment approach for entailing IE contextual evidence is lower than the GR and AP type evidence entailment. Table 6.15 shows that the average recall, accuracy and F2 measure for IE contextual evidence are 37.7%, 47.5%, and 19.8%. The RCE system achieved maximum of those measures as 71.4%, 62.8%, and 42.4% for IE contextual evidence entailment.

The averages of each performance measure for entailing each of the contexts are presented and compared in the column chart in Figure 6.4. From the chart it can be seen that effectiveness of our approach is best for GR evidence entailment. The recall of GR is 5.42% better than AP and 51.04% better than IE evidence entailment. All the

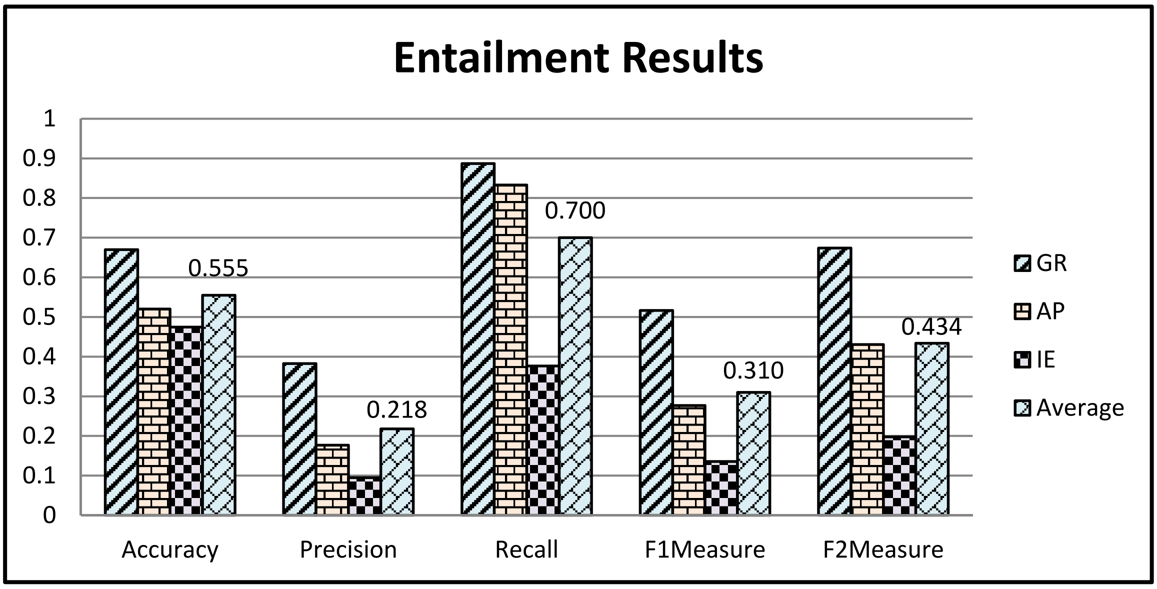


Figure 6.4: Comparison of evidence entailment among GR, AP and IE context.

other measures are also superior for GR evidence entailment. Among the performance measures we can see that our RCE approach is performing best in the measure of recall. The average recall is 70% among GR, AP, and IE evidence detection. After recall the second best effectiveness is in accuracy having an average of 55.5%. F2 measure is achieved as 0.434 and F1 measure is 0.310. Precision is lower than the other measures and achieved as 21.8%.

The importance of which one of the precision and recall to be higher depend on the circumstance. According to Manning et al. (2009) "various professional searchers such as paralegals and intelligence analysts are very concerned with trying to get as high recall as possible, and will tolerate fairly low precision results in order to get it." Similarly in the current case capturing more evidence is important than emphasizing on the correctness. Therefore a system which has a better recall is more desirable than a system which has a better precision. The RCE system in this entailment experiment has a better recall (70%) than a precision (21.8%). When recall has more importance in a system F2 measure is better than F1 measure. The RCE approach has better F2 measure (0.434) than F1 measure (0.310). As a conclusion we can say that the RCE approach in this experiment tend to be a desirable effective soft entailment system for CE evidence detection with a reasonable level of accuracy.

6.7 Experiments and Results of Combined phase

Objective:

The main goal in this set of experiments is to combine the CE evidence located through the 'Clustering' and the 'Entailment' phases and investigate if combining improves the effectiveness.

Method:

The procedures described in section 5.4 of Chapter-5 are used for this experiment. It has been mentioned in that section that the clustering phase may detect some evidence which the entailment phase would not detect; on the other hand the entailment phase may detect some evidence which the clustering phase would not detect. Accumulating the evidence from both the phases would improve the evidence detection system. Therefore to accumulate the evidence recognized by the two phases the two YES-NO hashes of a chat-log produced by the two phases are combined in this current experiment. Consider the YES-NO hashes produced by the entailment experiments and clustering experiments to be respectively an 'YN Entailment hash' and an 'YN Class hash'. Using an OR logic the two hashes are combined into an 'YN Combined hash'. Details of the combining procedure is explained in section 5.4 of Chapter-5. Evaluation metrics (A, P and R) are computed from the 'YN Combined hash' and presented in the following section.

Results and Analysis:

Table 6.16, Table 6.17, and Table 6.18 presents the accuracy, precision and recall achieved by the combined approach for all the chat-logs in the test data-set in the process of GR, AP and IE contextual evidence detection.

Table 6.16: Result of combined-phase for GR evidence.

Chat-log Name	Accuracy (A)	Precision (P)	Recall (R)
armysgt1	0.646	0.261	1.000
arthinic	0.456	0.278	0.965
fighting	0.624	0.163	0.862
flxnonya	0.724	0.484	1.000
icepirat	0.697	0.167	0.846
italianl	0.727	0.357	1.000
jleno9	0.671	0.510	1.000
jon_rave	0.591	0.196	0.625
manofdar	0.675	0.432	0.762
sebastia	0.660	0.555	0.836
thedude4	0.718	0.560	0.919
user1945	0.753	0.563	0.900

Table 6.17: Result of combined-phase for AP Evidence.

Chat-log Name	Accuracy (A)	Precision (P)	Recall (R)
armysgt1	0.479	0.226	0.875
arthinic	0.263	0.040	0.842
fighting	0.561	0.139	0.828
flxnonya	0.517	0.333	0.750
icepirat	0.628	0.273	0.800
italianl	0.667	0.313	1.000
jleno9	0.425	0.192	1.000
jon_rave	0.565	0.197	0.923
manofdar	0.513	0.073	0.750
sebastia	0.409	0.132	0.892
thedude4	0.453	0.007	1.000
user1945	0.557	0.241	0.867

Table 6.18: Result of combined-phase for IE Evidence.

Chat-log Name	Accuracy (A)	Precision (P)	Recall (R)
armysgt1	0.583	0.280	0.778
arthinic	0.284	0.044	0.773
fighting	0.553	0.037	0.857
flxnonya	0.431	0.067	0.286
icepirat	0.617	0.044	0.300
italianl	0.546	0.375	0.546
jleno9	0.288	0.061	0.333
jon_rave	0.583	0.111	0.385
manofdar	0.513	0.177	0.353
sebastia	0.396	0.023	0.313
thedude4	0.531	0.209	0.675
user1945	0.464	0.041	0.286

Table 6.19: Macro-averaged results of combined-phase.

Evidence Context	Accuracy	Precision	Recall	F1Measure	F2Measure
GR	0.662	0.377	0.893	0.513	0.701
AP	0.503	0.180	0.877	0.299	0.495
IE	0.482	0.122	0.490	0.196	0.306

Table 6.20: Micro-averaged results of combined-phase.

Evidence Context	Accuracy	Precision	Recall	F1Measure	F2Measure
GR	0.619	0.369	0.890	0.522	0.694
AP	0.449	0.120	0.856	0.210	0.383
IE	0.449	0.073	0.530	0.129	0.236

Table 6.19, and Table 6.20 shows the macro-averaged and micro-averaged results. These tables also include F1 and F2 measures. We already have come to know that in detecting GR contextual evidence the entailment system in tier two has better effectiveness than detecting AP and IE evidence. From the averaged results in Table 6.19, and Table 6.20 we can see that the superiority of effectiveness for GR evidence detection is still remaining in the combined phase of tier three. For example, the recall is 89.3% in GR evidence detection whereas for AP and IE it is 87.7% and 49% respectively. For the other measures the effectiveness for GR is also better.

Comparisons between the combined phase and entailment phase:

Comparisons between the combined phase and entailment phase for the GR, AP and IE contextual evidence detection are shown in the corresponding column charts of Figure 6.5, Figure 6.6 and Figure 6.7. From those charts we can see that the combined approach has mixed response in improvement of effectiveness in comparison with the RCE approach alone. For detecting the GR evidence the combined approach achieved a very low improvement in recall (0.6%) and F2 measure (2.7%). However effectiveness is decreased for accuracy (from 67% to 66.2%) and precision (38.2% to 37.7%). The effectiveness of the combined approach is slightly better in detection of AP evidence. Recall, precision and F2 measure all improved as 4.4%, 0.3% and 6.4% respectively; though accuracy dropped 1.7%.

For detecting the IE evidence the combined approach made a good improvement over the RCE approach. The effectiveness is improved in all measures. The best improvement it made is for recall. It achieved 49% recall which was 37.7% for RCE; that is the combined approach achieved an improvement of 11.3% in this case. The other improvements are 2.7%, 0.7%, 6% and 10.8% for precision, accuracy, F1 measure and F2 measure respectively.

Figure 6.8 shows the comparison between the combined phase and entailment phase with the results averaged among GR, AP and IE evidence detection. The combined phase made improvements in recall, precision and F1 and F2 measures. The respective averaged improvements are 5.3%, 0.9%, 2.6% and 6.7%.

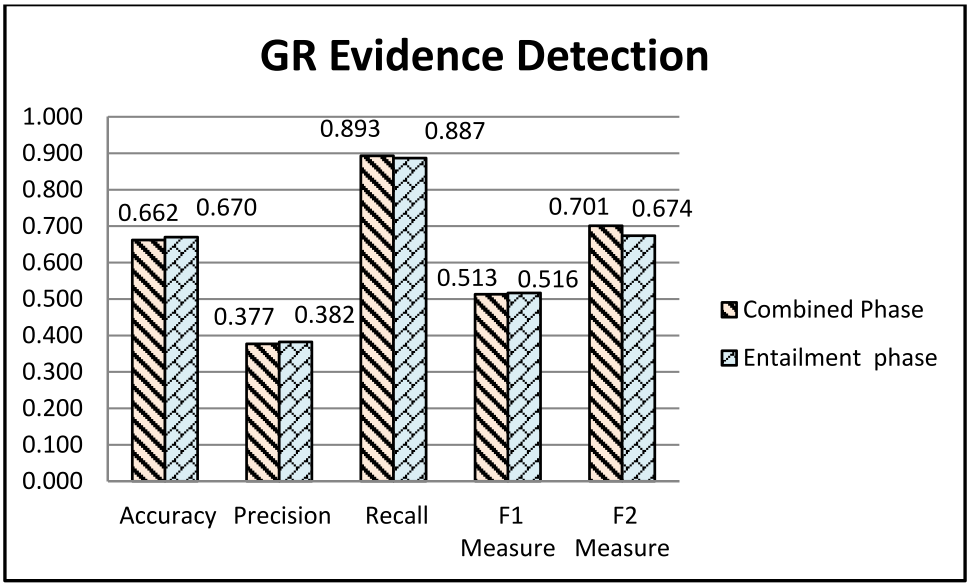


Figure 6.5: Comparison between the combined-phase and entailment-phase for the GR contextual evidence detection.

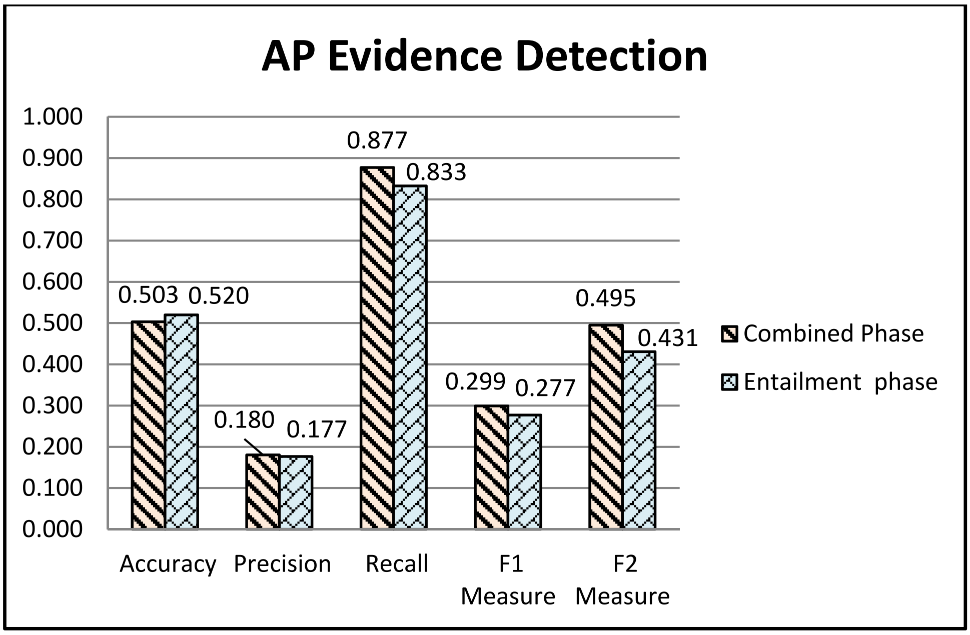


Figure 6.6: Comparison between the combined-phase and entailment-phase for the AP contextual evidence detection.

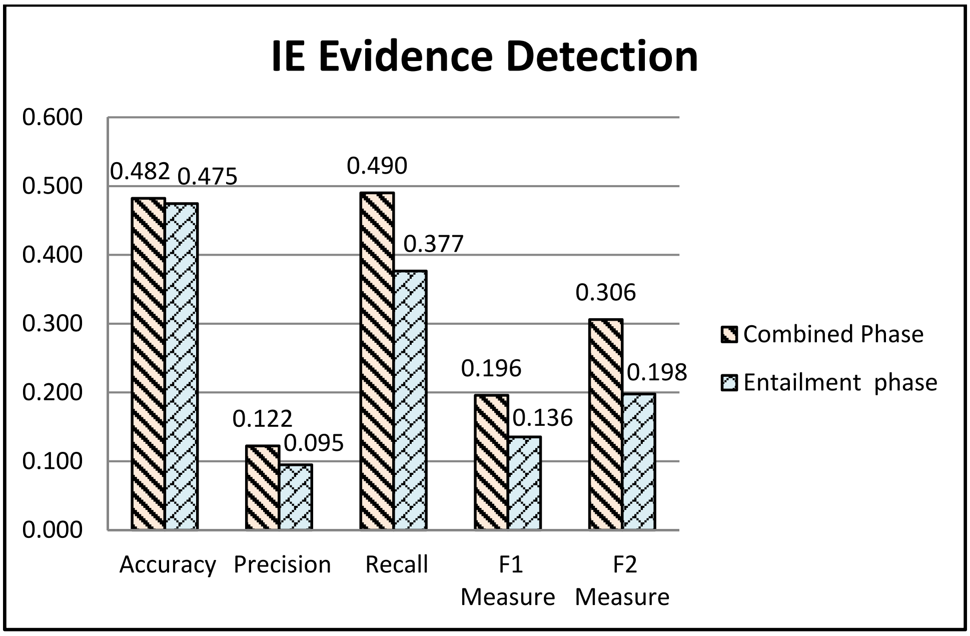


Figure 6.7: Comparison between the combined-phase and entailment-phase for the IE contextual evidence detection.

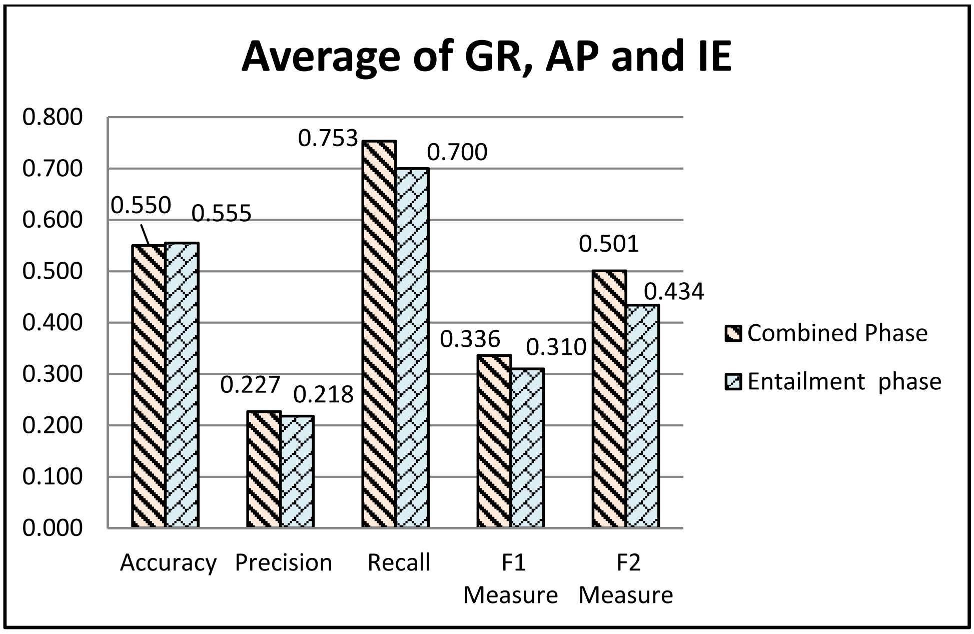


Figure 6.8: Comparison between the combined-phase and entailment-phase for the results averaged among GR, AP and IE contextual evidence detection.

The accuracy in the combined approach is dropped in a small quantity: from 55.5% of the RCE approach to 55%. This means that to increase the recall while it is detecting more correct evidence it is also detecting some incorrect posts as evidence. As have been mentioned earlier that detecting as much evidence as possible is more important than missing the evidence for the sake of correctness, a small reduction of accuracy is acceptable in this case. In conclusion we can say that the combined approach achieved improved effectiveness over the RCE approach alone.

The evidence detection approach achieved a high average recall of 75.3% with a fair average accuracy of 55%. The average F2 measure is also fair as 50.1% . Therefore the approach tends to be a good approach for the detection of CE in chat-logs.

Table 6.21: Statistical Comparison between Combined and Entailment Phases [Accuracy , Precision and Recall are averaged over IE, GR and AP for each phase]

	Accuracy	Precision	Recall
Mean Difference	-0.0057	0.0086	0.0548
t Critical one-tail	1.7959	1.7959	1.7958
t-Statistics	-1.4893	2.1017	5.2763
Alpha (α)	0.0500	0.0500	0.0500
P(T<=t) one-tail	0.0823	0.0297	0.0001

Table 6.21 presents the statistical comparison of the entailment and combined phases. A one-tail student *t*-test is done to compare the results of the two phases. Averages of accuracy, precision and recall are computed over IE, GR and AP results for each of the phases and then compared. We consider the base hypothesis “Combined phase does not make any significant improvement over Entailment phase”. From Table 6.21 we can see that for precision and recall the *t*-statistics is greater than the critical value of *t* . Also the *p*-value is less than the α-value. Therefore the base hypothesis can be rejected, that is the combined phase made a significant improvement for precision and recall. The results in accuracy is however were not improved by the combined phase according to the values of *t* and *p*. As mentioned before that in an evidence detection system it is more important to capture as much evidence as possible than to be strict on the correctness at first. For current situation a system having good recall is desirable, therefore improvement in recall is also desirable.

Using the three tier evidence detection approach we extracted evidence from the chat-logs. An example of a CE chat-log with the extracted predator's posts as evidence is shown in Appendix A. In that chat-log there are total of 47 predator's posts out of which human annotators labelled 24 predator's posts as of innocent BF contextual type. The other 23 predator's posts are of CE evidential GR, AP or IE contextual types. Out of the 23 the three tier CE Evidence Detection Model (CEDM) approach detected 21 predator's posts as CE evidence.

6.8 Chapter Summary

This chapter presents the experiments carried out for investigating the performance of our approach of the three tier CE evidence detection model (CEDM) for finding evidence of child exploitation acts in chat-logs. The experiments for each phase of the CEDM and the analysis of the results are explained.

In 'tier one' of the CEDM approach, experiments are carried out to find a suitable classifier that can effectively classify 'CE vs Non-CE' chats out of a mixed chat-data set. In this task a classifier has to be very strict in catching CE chats even if it predicts some non-CE chats as CE. It should not allow any suspected chat-log to pass through it as benign. That means the classifier can be flexible in Type-I error (False positive) but should minimize Type-II error (False negative) as much as possible. Considering this, a Naïve Bayes classifier with a feature set of psychometric and word information would be the best. The experimental results compared with different classifiers shows that the NB achieved a best recall of 90% with a best accuracy of 91.3%. Using psychometric and word information for CE detection in chat-logs is a new idea. From the results of the experiments it seems that the psychometric and word information enriched the feature set which improved the performance of the classifiers to predict CE type chat-logs more effectively than a mere term based feature-set.

The 'Predator vs. Victim' classification experiments aim to find out whether a CE predator and a CE victim are involved in the chat-log. Again Naïve Bayes classifier performed the best with recall and accuracy both as 95.8%.

The results of the experiments in the first tier of CEDM approach supports our idea that the classifiers can effectively provide statistical evidence as a shallow evidence of a chat to be of a suspected CE chat. They do not provide any exact excerpt of the evidence of CE act.

After the shallow evidence analysis, the CEDM approach carries out a content and context analysis in the second tier through the clustering phase and the entailment phase. In the clustering phase effectiveness of the newly developed PsyHAC clusterer is investigated and compared with other traditional clusterers. Results are presented with a reliable effectiveness measure of normalized mutual information (NMI). The PsyHAC achieved an average NMI of 30.84%. The results show that the PsyHAC can be used to learn the CE pattern of the predators' profile. The effectiveness of the learning is improved over existing clusterers as high as 19.7% in the NMI metric. The PsyHAC clustering approach is based on the CEPsy Dictionary and the CEPsySimilarity measure developed in Chapter-3. The results of the clustering experiment show that the new dictionary and the new similarity measure improved the effectiveness of the new clustering approach to such an extent that it has outperformed existing clustering methods for this current particular task.

Experiments are also carried out to find the association of the clusters produced by the PsyHAC with the CE context of each post in a chat-log. The result after this association is similar to the result of classifying each post into the CE psychological contextual categories. The result of the association experiment achieved a maximum accuracy of 83.5%, with an overall accuracy of 55.8% averaged among all the chat-logs of the test-set. The individual accuracy averaged among the CE psychological contexts BF, IE, GR and AP (Avg_{BIGA}) is 74% and the same measure averaged among IE, GR and AP (Avg_{IGA}) is 77.8%. A comparison with Chatcoder2 (McGhee et al. 2011), for the common chat-logs in the test-set, is also provided here. Our approach made a maximum improvement of 15.5% in the individual accuracy averaged among IE, GR and AP (Avg_{IGA}).

The entailment phase investigates the effectiveness of the newly developed soft entailment approach called Recognition of CE Entailment (RCE). The maximum average recall achieved by the RCE approach is 88.7% with accuracy of 67%. The maximum macro-averaged F2 measure is 67.4%. The result shows that the task of locating the CE evidence can be formed as a manageable task of textual entailment.

To improve the effectiveness of the evidence detection approach, in the third tier of CEDM the evidence detected by the PsyHAC and the evidence detected by the RCE approach are accumulated in the combined phase. The maximum macro-averaged recall achieved in the combined approach is 89.3%. This is 5.8% higher over the recall of the task of 'association of clusters with CE evidence' and 0.6% higher over RCE approach. The F2 measure in the combined approach is also good. The maximum macro-averaged F2 measure achieved by the combined approach is 70.11%.

The experimental results and analysis of this chapter shows that the evidence of CE can automatically be located to a reasonable level of accuracy using the CEDM approach developed throughout this research. The approach achieved a recall as high as 89.3% and an F2-measure as high as 70.11%.

Chapter 7

Conclusion

7.1 Overview

This chapter presents the major findings and the contributions made by the work presented in this dissertation. It also describes the future extensions of this research. The aim of this research was to develop computational text-processing techniques for finding evidence of Child Exploitation (CE) in chat-logs. This was motivated by the belief that successful utilization of the documented CE psychological stages would assist in capturing the interrelationships between pairs of chat-post-level ungrammatical informal text fragments. This would further widen the scope of successful application of text mining techniques like classification, clustering, and, linguistic tasks like text entailment to the CE evidence finding problems. To address this goal the following prime research question was posed:
"How can reliable methodologies and computationally automatic techniques be developed for finding evidence of child exploitation (CE) in chat-logs by analysing the informal text of chat?"

In order to answer the prime research question the main research problem was broken down into research sub-problems represented by the following research questions:

1. How do the traditional text classifiers behave in classifying chat-logs into Child Exploiting (CE) and non Child Exploiting (non-CE) classes?
2. How do the classifiers behave in classifying the participants of the chat into CE predator or CE victim?
3. How can the pattern of progression and profile of CE chats identified in the psychological literature be used to aid evidence detection?
4. How do we frame the problem of CE evidence detection into a manageable problem of Textual Entailment on chat-logs?

These research questions were progressively answered during the process of designing, developing and implementing the approaches described in Chapter-3 through to Chapter-6. While we believe that the research makes a significant contribution to the body of knowledge in the corresponding text-processing areas, the field continues to evolve rapidly, and new problems and challenges continue to emerge. The following section summarises the key contributions of the current work and then the later sections identify some promising directions for future work.

7.2 Research Contributions

This thesis makes a number of contributions. The main contribution is that an investigation is accomplished to understand the suitability of employing the standard data-mining and text processing techniques for the digital forensic task of finding evidence of CE in chat by analysing its informal ungrammatical text contents. Thereby a three tier CE Evidence Detection Model (CEDM) has been developed that incorporates a multi-level methodology with the data-mining techniques to recognize the documented phases of exploitation that constitute the CE-pattern. The model also incorporates the idea of textual entailment and has developed a unique soft entailment method for locating particular evidence. The novelty of this approach is that it is focused on the CE psychological contexts and has developed new techniques to capture them. It also has modified some of the existing techniques to fit in the environment of chat-text which has considerable difference in comparison with formal text. The major contributions of this thesis are recapitulated as follows:

1. Utilization of a special psychometric feature set in traditional text classifiers:

In Chapter-5 and 6 in the methodology and in the experiments we have investigated how do the traditional text classifiers behave in classifying chat-logs into Child Exploiting (CE) and non Child Exploiting (non-CE) classes. To accomplish this we have utilized the psychometric and categorical information (Pennebaker et al., 2007) as a special feature set in the text classifiers to effectively predict whether a chat-log is of the suspected child exploitation (CE) type or not. To the best of our knowledge before us no one else used the psychometric feature set in traditional text classifiers for categorization of chats into CE vs Non-CE . It seems that the chat dataset is enriched by the psychometric and categorical information. The new feature set significantly improves the performance of Naïve Bayes (NB) classifiers to predict CE type chats. In some cases it also improves the performance of Classification via Regression (CvR) classifier. This has previously been reported in Miah, Yearwood, and Kulkarni (2011).

2. Construction of a new CE Psychological term dictionary:

Chapter-3 introduces the construction of a new CE Psychological term dictionary (CEPsy dictionary) by mining the terms of CE chat corpus associated with the CE behavioural psychological contextual stages. The terms in the new CEPsy dictionary are good discriminators for the behavioural stages and can be used for categorizing the chat-posts into those stages. The new CEPsy dictionary is more effective than existing dictionaries such as LIWC (Pennebaker et al., 2007) for assessing similarity between behavioural stages in CE chats. Also it is more effective than LSA (Landauer et al., 1998) at finding contextual similarity among the posts of chat-text. This new dictionary works as a lexical resource for a new "similarity measure for CE chat-texts" and a new "weighting measure for term importance in CE domain" which are some of our other contributions.

3. Design of a new similarity measure for CE chat-text fragments:

We have developed a previously unseen similarity measure (called CEPsy similarity measure) that finds the CE psychological context similarity between a pair of chat-posts. The new CEPsy similarity measure is explained in Chapter-3. The new measure is based on the short-text sentence-similarity measure (Li et al., 2006). The CEPsy dictionary is used as a background lexical support for the new similarity measure. The new similarity measure takes advantage of the discriminating power of the terms in the CEPsy dictionary and improves the inter-post similarity for differentiating psychological stages in CE offensive chats. The new measure has achieved a good similarity value between a pair of posts belonging to the same psychological stage even though the pair does not share any common terms.

4. Development of a new clustering method:

We have developed a new clustering method (PsyHAC) based on hierarchical agglomerative clustering algorithm. The new PsyHAC clustering method is explained also in Chapter-3. In this method using the CEPsy similarity, the chat-posts are merged together by their cluster-centroids to ultimately collect them together into the clusters corresponding to the CE psychological stages. These clusters of CE stages give a behavioural pattern of child exploitation in the chat. Our clustering experiment results show that the new PsyHAC clusterer is useful for clustering the posts into their corresponding psychological categories. Compared to other clusterers used in the experiments, the PsyHAC clusterer makes a significant improvement in clustering the child-exploiting type predators' posts.

Construction of the CE psychological dictionary and the new similarity measure along with the new clustering method have been previously reported in our article in Miah, Yearwood, and Kulkarni (2014).

5. Construction of a new term weighting measure for finding term importance in CE domain:

Chapter-4 introduces a new term weighting measure for finding term importance in CE domain. A term is a good indicator of a particular CE psychological stage if a good number of perpetrators use the term in that particular stage. That is the importance of a term is proportional to the predators' frequency (*PF*). On the other hand the discriminating power of a term is reduced if the number of categories it appears is increased that is the importance is proportional to the inverse of the category frequency (iCF). Multiplying these two and crossing with the CE Psychological dictionary we get the new term weighting measure *CPFiCF*. This measure expresses the term importance in the CE domain for each of the CE psychological stages BF, IE, GR and AP.

6. Construction of a Domain Vector Space Model for CE domain:

Chapter-4 also explains a new vector space model associated with the CE psychological domain. The new CE Psychological Domain Vector Space Model (CEPDVSM) is constructed from a term vector space model (TVSM) by transforming the TVSM into the CE domain. For this purpose the new *CPFiCF* measure is used. The new vector space model is useful to find the CE context vectors of a chat-post. Placing a chat-post on the new CE Psychological Domain Vector Space and computing its vector components gives the CE psychological contexts of that chat-post. These contexts of a chat-post can be used to rank them. This can also be used for designing a new soft entailment technique.

7. Development of Soft Entailment Technique for CE evidence finding:

A new 'soft' entailment technique is developed also in Chapter-4. The new technique is useful in locating particular CE evidence in chat. The texts in formal documents are grammatically sound and descriptive in nature, whereas the texts in chats are not grammatical but are conversational and discrete in nature. Most of the existing traditional text-entailment techniques require huge linguistic and knowledge-based systems which are capable to analyse grammatical sentences in the formal text. Those strong grammatical expensive approaches are not suitable in the environment of ungrammatical chat-text. Therefore we have developed a soft entailment approach which does not need a huge grammatical and knowledge-based system. The new soft entailment approach uses the new CE Psychological Domain Vector Space Model to compute the CE contexts in a chat-post. Using the CE contextual matching a set of suitable surrogated texts from the training chat-logs are selected which represents the CE evidential hypothesis. A particular chat-post of the suspected test chat-log is entailing a CE evidential hypothesis if the CE context of that chat-post has a high similarity with the surrogates of the CE evidential hypothesis. The chat-posts which entail the evidential hypothesis are extracted and produced as evidence.

The methodology developed in this research shows an effective approach to process the informal text of chats. This opens an extended avenue in the research area of data mining and text processing field.

7.3 Limitations and Future Directions

This thesis has been concerned with developing computational text mining techniques that can be used for finding evidence in CE chats. The methodology of this research currently analyse the text contents only. However, meta-data can also be important for digital forensic evidence. Incorporating the analysis of the meta-data can be a future task.

A psychometric feature set has been used in text classifiers to predict the suspected CE chats out of other types of chats in the classification part of the three tier CE Evidence Detection Model (CEDM) of this research. The psychometric feature set significantly improved the performance of two text classifiers: Naïve Bayes (NB) and Classification via Regression (CvR). However it is interesting that while it is improving the performance of two classifiers, the same enriched dataset does not improve the performance of Decision Tree (DT) classifier. It can be a future scope to look at the profile of CE chats and investigate the interesting behaviour of different classifiers. The psychometric feature set was produced by using a psychological and word count dictionary which is generic in nature and not focused on the behavioural psychology of child exploitation. A dictionary which is focused on the CE behavioural psychology such as the CE Psychological Dictionary (constructed in this research) may further improve the performance of the text classifiers. Although we have used this new dictionary in other parts of the developed module but it has not yet been used with the text classifiers in the classification module. This can be one of the interesting future tasks to see how the new dictionary works with text classifiers.

The 'Predator vs Victim' task used only the PJ chat corpus. The victims' part of that corpus is from trained volunteers posing as children; not from real children. It would be interesting to obtain a third-party chat-corpus which contains chats between real child and adult. A classifier can be built using that new corpus as the training set and evaluated on the PJ chat-logs to investigate whether the participants can be categorized into child vs adult. If such a classifier can be built then it can be used to find evidence against the predator.

The newly constructed CEPsy similarity measure uses the CE Psychological dictionary which is based on terms only; currently, it has no phrase-matching capability. A dictionary with the phrase-matching facility may further improve the similarity measure and eventually give better results. However, this would require further consideration of the nature of chat text. Moreover, we have used only Li-measure (Li et al., 2006) of short-text sentence similarity measure as the basis for the construction of the new similarity measure. There are some other short-text semantic similarity measures such as Mihalcea et al. (2006). A future endeavour can be to investigate the other measures to use as the basis of constructing a new CE similarity measure.

For developing the new PsyHAC clusterer we have used CEPsy similarity measure in HAC algorithm only. The CEPsy similarity can also be used in other clustering algorithms such as K-means and EM. These can be future interesting tasks.

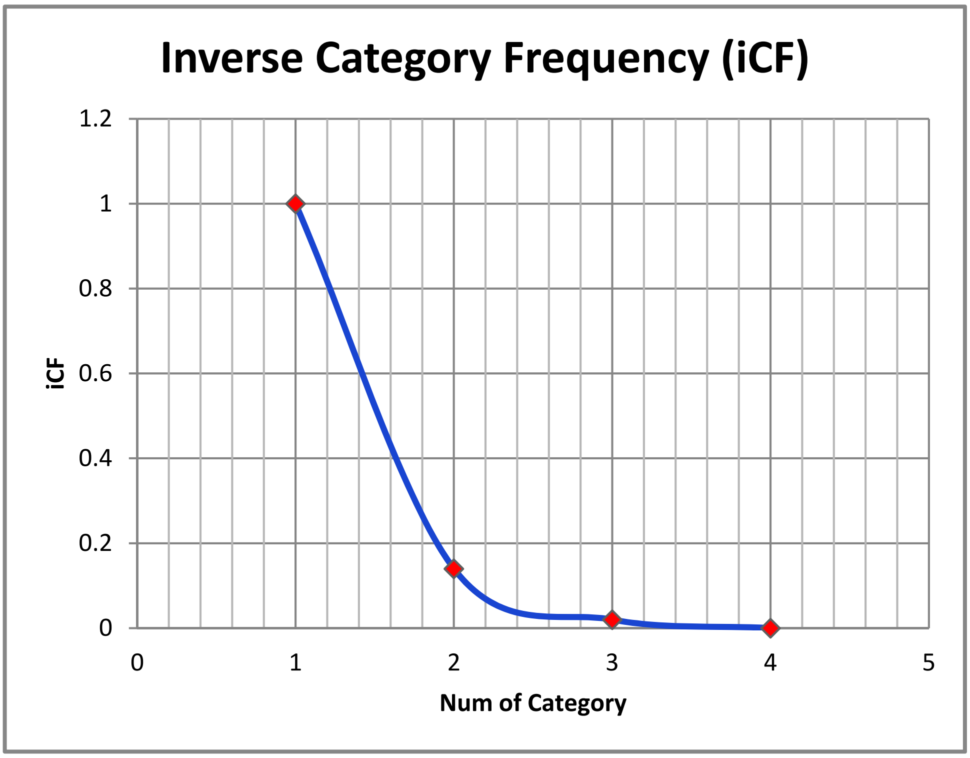


Figure 7.1: Exponential change of inverse category frequency (iCF).

In the newly constructed term weighting measure of Crossed Predator Frequency inverse Category Frequency (CPFiCF) the iCF has been computed using a straight line function $y = -\left(\frac{x}{3}\right) + \frac{4}{3}$ (Refer to Figure 4.4 in Chapter-4). Instead of a straight line function the change of iCF may also follow other functions. For example, it may follow an exponential decay function $y = e^{(-2x+2)}$ (Figure 7.1). We have used straight line function in this current research. The other functions can be a future endeavour.

In the CEPsy similarity measure we have used Jaccard's coefficient where a ratio of intersection and union of two sets is computed. The members of those two sets come from a coordinate representations with 1s and 0s according to the presence and absence of terms in chat-posts. That leads us towards the proof in Appendix C: "If two posts P_a and P_b are equal in CEPsy similarity a third post P_c will have the same CEPsy similarity to both P_a and P_b". That proof gives the idea of the filtering scheme used in making surrogates of posts in the proposed entailment system. However it is intuitive that accepting one of the candidates (P_a or P_b) as a surrogate may have a different effect in detecting a suspected chat-post ($P_{c)}$. This may be a limitation of the current similarity measure due to the use of Jaccard coefficient. A different coefficient may have different result. There may also be some other way of getting a better filtering. Those can be subjects for further research.

It has been mentioned in Chapter-4 that due to the limitations of available annotated chat data set some of the legal type evidential propositions cannot directly be entailed by the newly developed soft entailment approach at this time as surrogated texts cannot be found for them. It can be an interesting future research to get a chat data set which contains chat-texts annotated for all types of evidential propositions, and evaluate the new soft entailment approach on that.

The developed system of the current research has been applied only on chat-messages to identify online child exploitation. Online harassment and bullying are a kind of sister problem. The developed system with a little modification may also be applied to identify harassment and bullying in online chats and in micro-blogging platforms such as Twitter and MySpace. A thorough analysis and implications of applying the current system on other platforms can be an interesting future research.

7.4 Prospective Applications

It is anticipated that the outcome of the current research would have different practical applications. The methodology in this research would assist to develop a dedicated forensic tool for the law and enforcement agency to automatically and efficiently detect the child exploiting chat-logs in the confiscated storage device of an accused. Using this tool, specific indications of child exploitation in the chat-logs can be detected and particular evidence can be located and produced in a court of law. Manual identification of the evidence is a tedious and time consuming work, as one may have to read hundreds or thousands of pages of chat-texts from different chat-logs. Thus it is prone to error due to exhaustion. Moreover manual process may lead to a biased decision.

Another implementation of the methodology in this research can be a parent-filter. It is very difficult and impractical for the parents to watch over their children all the time. When they are grown up to the adolescence, they want privacy, especially at the time of the Internet chatting. At this time, it becomes more difficult for the parents to save the children. They do not know with whom the child is chatting; whether the person on the other side is safe or not. The methodology in the current research would assist to build a system that takes care for the safety of the children. It will work in the background and automatically analyse all the chat-texts. If any CE threat comes up for a child, it will notify the parents to be alert. This will help the parents to protect the children without violating their privacy.

Other areas that this research could be adapted would be for detection of CE across computer networks including the Internet. A network-based detection system will require three major components at the router level: IP packet interception, chat message decoding and CE detection. For IP packet interception, many state-of-the-art tools are available. After decoding to chat-text, the proposed methodology can be applied to detect as to whether the chat text contain any CE element in it or not. Different social networking sites facilitating chatting (like Facebook, Twitter) might be able to adapt this method. It would also be applied in mobile phone chatting. With all the above mentioned prospective applications the current research can have a good positive impact on the society.

Appendix

Appendix A

A Perpetrator's Posts of a CE chat-log

The following table presents the predator's posts of a CE chat-log taken from the test data set. Researchers suggest that posts of the BF context tend to be innocent; therefore in the evidence detection experiments BF posts are considered as not an evidence of child exploitation.

In this chat-log there are 23 posts annotated by human as evidence of CE evidential context of GR, AP and IE. Among those 23 posts the CEDM approach detected 21 posts as evidence. The two evidential posts missed by the approach are post no. 32 and post no. 48.

Sl. No	Chat-posts in the log	Evidence annotated by human	Evidence annotated by CEDM approach		
			GR Contextual Evidence	AP Contextual Evidence	IE Contextual Evidence
1	yeah	BF	NO	YES	NO
2	sure	BF	YES	YES	YES
3	ok	BF	NO	NO	NO
4	not sure how	BF	YES	YES	YES

Sl. No	Chat-posts in the log	Evidence annotated by human	Evidence annotated by CEDM approach		
			GR Contextual Evidence	AP Contextual Evidence	IE Contextual Evidence
5	thanks	BF	NO	YES	NO
6	sure	BF	YES	YES	YES
7	oh	BF	YES	YES	YES
8	ok	BF	NO	NO	NO
9	oh	BF	YES	YES	YES
10	i can buy u some mins	BF	NO	YES	NO
11	yes	BF	YES	YES	YES
12	im serious	BF	NO	YES	NO
13	50	BF	NO	NO	NO
14	ok	BF	NO	NO	NO
15	ok	BF	NO	NO	NO
16	ok	BF	NO	NO	NO
17	yes	BF	YES	YES	YES
18	uc	BF	NO	NO	NO
19	its ok	BF	YES	YES	YES
20	ok	BF	NO	NO	NO
21	yes	BF	YES	YES	YES
22	ok	BF	NO	NO	NO
23	star 67	BF	NO	YES	NO
24	no u wont	BF	NO	YES	NO
25	ok	BF	NO	NO	NO
26	im dennis us army soldier from cincinnati	IE	NO	YES	NO
27	how old ru	IE	YES	YES	YES
28	u single	IE	NO	NO	YES
29	u have any pics	IE	YES	YES	YES
30	yeah on profile can u send to my email	IE	NO	NO	YES
31	u send yours	IE	YES	YES	YES
32	can i get your number	IE	NO	NO	NO
33	no whats your number	IE	NO	NO	YES
34	name	IE	NO	NO	YES
35	ok u have sex at 13	GR	YES	YES	NO
36	nice u have any naked ones	GR	YES	YES	YES
37	were u taking your cloths off for people	GR	YES	YES	YES
38	oh so u didnt get dirty on cam	GR	YES	YES	YES
39	so u have big tits	GR	YES	YES	YES

Sl. No	Chat-posts in the log	Evidence annotated by human	Evidence annotated by CEDM approach		
			GR Contextual Evidence	AP Contextual Evidence	IE Contextual Evidence
40	u shave	GR	YES	NO	NO
41	i will pick up some condoms if u wanna have fun i will buy u a 50 dollar card	AP	YES	YES	YES
42	or just a blow job	AP	YES	YES	YES
43	u want to meet	AP	NO	YES	NO
44	where she work at	AP	YES	YES	YES
45	which one	AP	NO	YES	NO
46	where u wanna meet	AP	YES	YES	YES
47	50 dollars for 15 mins worth head where u wanna meet	AP	YES	YES	YES
48	ok where at	AP	NO	NO	NO

Appendix B

Example of Hypothesis and Surrogates

Following is an example of Information Exchange (IE) type hypothesis:

"Valuable personal information has been exchanged between the perpetrator and the victim".

The 100 surrogates in our RCE approach in ranked order are:

1	that last one is where i would like to cuddle with you and kiss your lips
2	for my day i got thumb smashed and than tonight i was going to use a friends auger and on the way here it started to buckel so had to call him and have him come help block it up so i could get it back to his place and now i will have to go help him fix it
3	what do u do for fun at ur age besides sex
4	we dated a while but she wanted someone with kids that were young so she would have someone when she was older
5	i went to mcdonalds to eat then when i got bac to my trk just to get ready to chat with ya he calld and i askd him where he was and he said that he was in troutdale oregon
6	they both can be fun but than i am use to being alone where i camped before so i guess we will have to see if there are a lot of people there
7	do u ever give ur number out to guys u talk to online ?
8	just want to say i do have a couple of more pics but i am a very nice guy .
9	is he older
10	hey hun its me robert i just wanted to say hi an that i miss chattin wit ya hope to talk to u soon bye 4 now robert
11	nothing, i had a choice of sitting at airport or rent a car and come back home so i can tlak to u, but i guess u r busy so anyway..let me know when u can tlak
12	you are too and i still think you look older than you are
13	i like the pic with the pink tank top or is that a bra
14	not much but i did look at your profile and seen a pretty young lady
15	i don't blame you you have a right to your friends the same as he has a right to his friends
16	so if a boy was to ask you out what would you say to him
17	look's older
18	i like vegas most of all i go to la i like montana area dallas is nice lubbic tx i awsum been to san fransico been to miami fl i really love the gulf coast are of mississippi
19	i like most anything, as long as it sounds good. friends o my boys think ikewl cause i like music like they do
20	got any of your full body?
21	so those two pics are the only ones you have right? lol

22	u said u live down south
23	m kinda a outsider kind of a guy
24	i know, i do lots of volunteer work on weekends, and one thing started doing is medical camp in big cities
25	i thinking about taking a shower
26	what size ring do you ware
27	whats the size of ur family
28	its ok...been doing this type of work for over 20 yrs
29	lying her listening to the movie cars and thinking of you is sarah the one in the pic with you.
30	how do u and ur dad get along? is he a cool guy?
31	southern comfort and hot damn has a cinimma taste
32	you like fat guys
33	my boys also, his is the 5th, this weds.
34	my youngest sons bday is weds, so were are going out for his bday, but i should be on hopefully sometime b4 8 weds. night
35	wow...very sorry,,,,how old were ya
36	so where "by columbus" are you? i have been down that way....my dad was stationed at ft benning
37	oh not real big and as long as they take care of them selfs but i do like long haire
38	wish i knew ur real last name
39	i guess that means north cali?thats as close as it gets?lol
40	i could be going that way soon
41	want to go tomarrow night if i get every thing done
42	no tx wish i was near you
43	k, let me send it..give me a min ok
44	i would make you like my little girlfriend ok
45	been divorced for a long time...havent found right girl...wasnt really looking
46	naw i dont david--sory man-- but iwill send when i get new ones-- wut bout you
47	ahh im 5'7" dark hair and eyes...
48	anyways..im latino, brwn hair/eyes
49	and your name is 15-yr-old-girl?
50	what kind of guy u like?
51	a girl that is not afaired to try new things
52	m in mojavie ca i reloaded in the l.a area and on my way to reno nevada area 4 one stop at a wal-mart distrubution center have to be thier by 4pm sat only have 385 miles to go for that one
53	cool, my boys had it off for parent teacher conferences, but then cause of all the snow the rest seemed to have gotten closed
54	i got one on my profile or i can open pic share and show you
55	i'll give u when i go home
56	did u have a guy in fl
57	are u guy'sclose
58	what type of guy do you like
59	so when r u sending me more pics? btw 3 u sent which one most recent?
60	don't your mom teach you to cook

61	kinda between dodgers and hollywood
62	we build grain legs for farmers and mills
63	ne way to send pic
64	well what's your fav thing?
65	have you ever been drunk
66	had to take my son to the hospital that make two there now
67	i make plenty of money
68	hi sweetie, i am work sooooooooo sleepy but i'll stay invisible and in and out from mtgs so if u log in just buzz me, if i am around i'll buzz back!! lol luv ya
69	i am nice to all my friends and i know what you are going throught and i don't think it is fair
70	hmm, kinda south west of jakson
71	i'm at my brothers house on his computer, motorhome and car are in the shop,pick them up tuesday
72	and i teach kids! lol on sun ages 6 -16 on human values
73	so im just cuurious..can u say the city ur from yet?
74	i will give you that number ok
75	so what have you been doing since you got home
76	do u have anymore pics other then those two?
77	i use to live in tenn. too
78	well i like living in outskirt of the city so u get best of both
79	by the way....my name is jim
80	but lately havent had a girl like that for quite sometime
81	have you been watching it today
82	hi midnight. nice profile. how are you tonight? where in ga?
83	who's this
84	been real sad all day
85	how long ago was your pic taken?
86	play sports, hang with friends, party occasionally, watch movies, etc....
87	do u always get up early?
88	do you hav a lap top or home comp
89	no i have been staying with my mom cuz she lives at home alone
90	i want to be more than friends and what that means is i want to see if we may be compatible to be partners for possibly the rest of our lives
91	check this out: http://profile.myspace.com/index.cfm?fuseaction=user.viewprofile&friendid=110385953
92	what do you like talking about
93	why you close the share pics
94	ok i'll get on when i get home which will be like 1 am
95	not all the time i do have to stop and sleep and do other stuff like wash clothes, do paper work, fuel up then look for you on line now
96	going to wash clothes ,call me,tim
97	been in jail
98	went skiing during the day yesterday and went partying last night
99	late summer, we play softball
100	yea i went to school for cooking i got culinary art degree and business degree too

Appendix C

Proof of "If CEPsySim(P_a, P_b) = 1, then CEPsySim(P_c, P_a) = CEPsySim(P_c, P_b)"

We have to proof that "if two posts P_a and P_b are equal in CEPsy similarity a third post P_c will have the same CEPsy similarity to both P_a and P_b", or
"If CEPsySim(P_a, P_b) = 1, then CEPsySim(P_c, P_a) = CEPsySim(P_c, P_b)".

The CE Psychological Similarity measure (CEPsySim) between two chat posts P_a and P_b can be rewritten from Equation 3.1 of Section 3.3 as:

$$\text{CEPsySim}\ (P_a, P_b) = \text{CosSim}\ (\mathbf{V_a}, \mathbf{V_b}) \qquad \text{... Equation C.1}$$

The vectors $\mathbf{V_a}$ and $\mathbf{V_b}$ in Equation C.1 are constructed using the Reduced Vector Spaces (RVS) corresponding to P_a and P_b. The RVS contains only those terms of P_a and P_b which are present in the CEPsy Dictionary. In the traditional cosine similarity measure document vectors contain only 1's and 0's corresponding to the presence and absence of vector space terms. Here the vectors $\mathbf{V_a}$ and $\mathbf{V_b}$ also contains 1's in the places where the RVS term is present in corresponding posts (P_a and P_b). However, if the RVS term is not present then a value is used instead of 0. That value is computed by CEPsyDictSim from Equation 3.3 rewritten here as Equation C.2. A detailed explanation with an example is provided in Section 3.3.

CEPsyDictSim is given by:

$$\text{CEPsyDictSim}\ (t_a, t_b) = \frac{|A \cap B|}{|A \cup B|} \qquad \text{... Equation C.2}$$

Where A and B are the sets of category entries in CEPsy Dictionary for terms t_a and t_b of posts P_a and P_b.

For any two posts P_a and P_b to have 100% CEPsy similarity their RVS terms $t_a \in \mathbf{V_a}$ and $t_b \in \mathbf{V_b}$ need to be same or CEPsyDictSim (t_a, t_b) needs to be equal to 1.

If the terms t_a and t_b are same then $\mathbf{V_a} = \mathbf{V_b}$.

Therefore, CEPsySim(P_c, P_a) = CosSim ($\mathbf{V_c}$, $\mathbf{V_a}$)

= CosSim ($\mathbf{V_c}$, $\mathbf{V_b}$)

= CEPsySim($\mathbf{P_c}$, $\mathbf{P_b}$) ... Equation C.3

If the terms t_a and t_b are not same then for CEPsySim(P_a, P_b) to be equal to 1 the CEPsyDictSim between t_a and t_b needs to be 1 , that is :

CEPsyDictSim (t_a, t_b) = 1 ... Equation C.4

From C.2 we get:

CEPsyDictSim (t_a, t_b) = $\frac{|A \cap B|}{|A \cup B|}$ = 1 ... Equation C.5

$\Rightarrow \quad |A \cap B| = |A \cup B|$ ... Equation C.6

From Equation C.6 we get the following logical deductions:

$\forall x \in (A \cap B) \;\Rightarrow\; x \in A \;\wedge\; x \in B$ ... Equation C.7

$\forall x \in (A \cup B) \;\Rightarrow\; x \in A \;\vee\; x \in B$ ... Equation C.8

From Equation C.7 we get $x \in A$. Applying this to Equation C.8 may appear that $x \notin B$, but that cannot be valid because Equation C.7 implies that $x \in B$. Putting $x \in B$ in C.8 does not nullify $x \in A$. Because the equations (C.7 and C.8) are equal (from Equation C.6), $x \in A$ and also $x \in B$ both are true; this implies that the sets A and B are equal, that is: $A = B$, and therefore $\mathbf{V_a} = \mathbf{V_b}$.

Now from Equation C.3 again we can show that:

CEPsySim(P_c, P_a) = CEPsySim(P_c, P_b)

This proves that , "if two posts P_a and P_b are equal in CEPsy similarity a third post P_c will have the same CEPsy similarity to both P_a and P_b".

Appendix D

Detailed Computation of Evaluation Metrics for Classification of the posts of a Chatlog

Computations in this appendix are to clarify the evaluation metrics presented in Table 6.11. Below is the multiclass Confusion Matrix resulted from the multiclass classification by PsyHAC clustering of the posts of chat-log named ‘armysgt1’. The task was to classify the perpetrator’s posts of a CE chat-log into BF, IE, GR and AP types.

Multiclass Confusion Matrix:

Predicted					
BF	*IE*	*GR*	*AP*		
22	0	0	3	*BF*	Actual
2	5	2	0	*IE*	
4	0	1	1	*GR*	
7	0	0	1	*AP*	

Computation For Overall Accuracy *A* :

Accuracy *A* is the proportion of the total number of predictions that were correct.

Accuracy *A* is given by:

Accuracy = $A = \frac{Number\ of\ correctly\ predicted\ instances}{Total\ Number\ of\ Instances}$

In a multi-class confusion matrix the number of correctly predicted instances are the numbers on the diagonal cells (yellow highlighted). Therefore overall accuracy is given by:

$$A_{Overall} = \frac{22+5+1+1}{22+3+2+5+2+4+1+1+7+1}$$

$$= \frac{29}{48}$$

$$= 0.604$$

For an individual class *C_i* the contingency table is given by:

C_i	*Not C_i / Other*	<--Predicted
TP	FN	*C_i*
FP	TN	*Not C_i / Other*

Where :

TP = True positive = Number of instances from class *C_i* predicted as *C_i*

FP = False positive = Number of instances from *Not C_i / Other* classes predicted as *C_i*

FN = False Negative = Number of instances from class *C_i* predicted as *Not C_i / Other*

TN = True Negative = Number of instances from *Not C_i / Other* classes predicted as *Not C_i / Other*

Evaluation Metrics for an individual class *Ci*:

Accuracy (A_{C_i}) =

$$= \frac{\begin{array}{c} Number\ of\ correctly\ predicted\ instances\ for\ Class\ C_i\ (TP)\ and \\ Number\ of\ correctly\ predicted\ instances\ for\ Not\ C_i\ Class\ \ i.e.\ Other\ classess\ (TN) \end{array}}{Total\ Number\ of\ Instances}$$

$$= \frac{TP+TN}{TP+TN+FP+FN}$$

Precision $= P_{C_i} = \frac{TP}{(TP+FP)}$

Recall $= R_{C_i} = \frac{TP}{(TP+FN)}$

Computation for individual class *BF* :

For ease of understanding the actual and predicted instances of *BF* class are yellow highlighted in the multiclass confusion matrix:

BF	*IE*	*GR*	*AP*	<--Predicted
22	0	0	3	*BF*
2	5	2	0	*IE*
4	0	1	1	*GR*
7	0	0	1	*AP*

For *BF*:

TP$_{\mathrm{BF}}$ = 22, FP$_{\mathrm{BF}}$ = 2+4+7 = 13, FN$_{\mathrm{BF}}$ = 0+0+3 = 3,

TN$_{\mathrm{BF}}$ = All other than yellow highlighted = 5+2+0 +0+1+1 +0+0+1 = 10;

BF Contingengy Table:

BF	*Not BF/ Other*	<--Predicted
22	3	*BF*
13	10	*Not BF / Other*

Evaluation Metrics for *BF* class:

Accuracy $= A_{BF} = \frac{(TP+TN)}{(TP+TN+FP+FN)} = \frac{22+10}{22+10+13+3} = \frac{32}{48} = 0.667$

Precision $= P_{BF} = \frac{TP}{(TP+FP)} = \frac{22}{22+13} = \frac{22}{35} = 0.629$

Recall $= R_{BF} = \frac{TP}{(TP+FN)} = \frac{22}{22+3} = \frac{22}{25} = 0.880$

Computation for individual class *IE* :

BF	IE	GR	AP	<--Predicted
22	0	0	3	BF
2	5	2	0	IE
4	0	1	1	GR
7	0	0	1	AP

For IE:

$TP_{IE} = 5$, $FP_{IE} = 0$, $FN_{IE} = 2+2+0 = 4$,

TN_{IE} = All other than yellow highlighted = 22+0+3 +4+1+1 +7+0+1 = 39;

IE Contingengy Table:

IE	*Not IE/ Other*	<--Predicted
5	4	*IE*
0	39	*Not IE/ Other*

Evaluation Metrics for IE class:

$$\text{Accuracy } = A_{IE} = \frac{(TP+TN)}{(TP+TN+FP+FN)} = \frac{5+39}{5+39+0+4} = \frac{44}{48} = 0.917$$

$$\text{Precision } = P_{IE} = \frac{TP}{(TP+FP)} = \frac{5}{5+0} = \frac{5}{5} = 1.0$$

$$\text{Recall } = R_{IE} = \frac{TP}{(TP+FN)} = \frac{5}{5+4} = \frac{5}{9} = 0.556$$

Computation for individual class *GR* :

BF	*IE*	*GR*	*AP*	<--Predicted
22	0	0	3	*BF*
2	5	2	0	*IE*
4	0	1	1	*GR*
7	0	0	1	*AP*

For GR:

TP_{GR} = 1, FP_{GR} = 0+2+0 = 2, FN_{GR} = 4+0+1 = 5,

TN_{GR} = All other than yellow highlighted = 22+0+3 +2+5+0 +7+0+1 = 40;

GR Contingegy Table:

GR	*Not GR/ Other*	<--Predicted
1	5	*GR*
2	40	*Not GR/ Other*

Evaluation Metrics for GR class:

$$\text{Accuracy } = A_{GR} = \frac{(TP+TN)}{(TP+TN+FP+FN)} = \frac{1+40}{1+40+2+5} = \frac{41}{48} = 0.854$$

$$\text{Precision } = P_{GR} = \frac{TP}{(TP+FP)} = \frac{1}{1+2} = \frac{1}{3} = 0.333$$

$$\text{Recall } = R_{GR} = \frac{TP}{(TP+FN)} = \frac{1}{1+5} = \frac{1}{6} = 0.167$$

Computation for individual class AP :

BF	*IE*	*GR*	*AP*	<--Predicted
22	0	0	3	*BF*
2	5	2	0	*IE*
4	0	1	1	*GR*
7	0	0	1	*AP*

For AP:

TP_{AP} = 1, FP_{AP} = 3+0+1 = 4, FN_{AP} = 7+0+0 = 7,

TN_{AP} = All other than yellow highlighted = 22+0+0 +2+5+2 +4+0+1 = 36;

AP Contingengy Table:

AP	*Not AP/ Other*	<--Predicted
1	7	*AP*
4	36	*Not AP/ Other*

Evaluation Metrics for *AP* class:

Accuracy = $A_{AP} = \frac{(TP+TN)}{(TP+TN+FP+FN)} = \frac{1+36}{1+36+4+7} = \frac{37}{48} = 0.771$

Precision = $P_{AP} = \frac{TP}{(TP+FP)} = \frac{1}{1+4} = \frac{1}{5} = 0.2$

Recall = $R_{AP} = \frac{TP}{(TP+FN)} = \frac{1}{1+7} = \frac{1}{8} = 0.125$

Appendix E

List of Selected Acronyms

ABS	=	Australian Bureau of Statistics
AP	=	Approach
AustLII	=	Australasian Legal Information Institute
BF	=	Befriending
CE	=	Child Exploiting or Child Exploitation
CEDM	=	Child-exploitation Evidence Detection Model
CEPDVSM	=	Child Exploiting Psychological Domain Vector Space Model
CEPsy	=	Child Exploiting Psychological
CEPsyDict	=	Child Exploiting Psychological Dictionary
CEPsyDictSim	=	Child Exploiting Psychological Dictionary Similarity
CEPsySim	=	Child Exploiting Psychological Similarity
CPFiCF	=	Crossed Predator Frequency Inverse Category Frequency
Cth	=	Commonwealth
CvR	=	Classification via Regression
DF	=	Document Frequency
DT	=	Decision Tree
DVSM	=	Domain Vector Space Model
EFDL	=	Extended Feature Description Logic
EM	=	Expectation Maximization
GAAC	=	Group Average Agglomerative Clustering
Gen-V	=	Generation Virtual
GN	=	General
GR	=	Grooming
HAC	=	Hierarchical Agglomerative Clustering
HDD	=	Hard Disk Drive
iCF	=	Inverse Category Frequency
IDE	=	Integrated Development Environment
IE	=	Information Exchange

IM	=	Instant Messaging
IR	=	Information Retrieval
IT	=	Information Technology
KB	=	Knowledge Base
kNN	=	k Nearest Neighbours
KM	=	K-means
LCT	=	Luring Communication Theory
LEA	=	Law and Enforcement Agency
LIWC	=	Linguistic Inquiry and Word Count
LSA	=	Latent Semantic Analysis
ML	=	Machine Learning
NAA	=	National Archives of Australia
NB	=	Naïve Bayes
NCMEC	=	National Center for Missing & Exploited Children
NER	=	Named Entity Recognizer
NLP	=	Natural Language Processing
NMI	=	Normalized Mutual Information
NN	=	Neural Networks
PCFG	=	Probabilistic Context Free Grammar
PF	=	Predator Frequency
PJ	=	Perverted Justice
PJFI	=	Perverted Justice Foundation Incorporated
PsyHAC	=	Psychological Hierarchical Agglomerative Clustering
POS	=	Parts Of Speech
QA	=	Question Answering
RCE	=	Recognition of CE Entailment
RIPPER	=	Repeated Incremental Pruning to Produce Error Reduction
RTE	=	Recognition of Textual Entailment
RVS	=	Reduced Vector Space
SF	=	Sex Fantasy
SVM	=	Support Vector Machine
TC	=	Text Classifier
TFiDF	=	Term Frequency inverse Document Frequency

TVSM	=	Term Vector Space Model
VSM	=	Vector Space Model
WEKA	=	Waikato Environment for Knowledge Analysis

Appendix F

List of Resources Used in the Experiments

A 32 bit machine and another 64 bit machine both with Windows 7, are used for the experiments. RAM of 32 bit machine was 3 GB and of 64 bit machine was 16 GB.

Perl and Java is used as programming language for coding. Eclipse IDE (Integrated Development Environment) is used for program coding in Java. EPIC - Eclipse Perl Integration (*www.epic-ide.org*) is used for program coding in Perl.

Some of the classifiers, clusterers, text-filters and latent semantic indexing tool of Waikato Environment for Knowledge Analysis (WEKA) (Hall et al., 2009) are used in some of our experiment modules.

To produce the psychometric features Linguistic Inquiry and Word Count (LIWC) (Pennebaker et al. 2007) is used.

Wordnet, Stanford Parser, Stanford Named Entity Recognizer are also used for some limited experiments (refer to Chapter- 3 and 4).

Bibliography

ABS (2013). *Children's Participation in Cultural and Leisure Activities, Australia, Apr 2012 (Published in 2013).* Australian Bureau of Statistics. Retrieved in September 2014 from http://www.abs.gov.au/.

Adams, P. H., & Martell, C. H. (2008). Topic Detection and Extraction in Chat. *IEEE International Conference on Semantic Computing 2008 Second IEEE International Conference on Semantic Computing,* Santa Clara, CA, USA. p. 581-588.

Akhmatova, E., & Mollá, D. (2006). Recognizing Textual Entailment Via Atomic Propositions. In J. Quiñonero-Candela, I. Dagan, B. Magnini & F. d'Alché-Buc (Eds.), *Machine Learning Challenges. . Evaluating Predictive Uncertainty, Visual Object Classification, and Recognising Textual Entailment* (p. 385-403). Heidelberg, Germany: Springer. doi:10.1007/11736790_22.

Argamon, S., Koppel, M., Pennebaker, J. W., & Schler, J. (2009). Automatically Profiling the Author of an Anonymous Text. *Communications of the ACM, Vol. 52* (2), p. 119-123, ACM.

Armagh, D. S., & Battaglia, N. L. (2006). *Use of Computers in the Sexual Exploitation of Children.* US Dept. of Justice, Office of Justice Programs, Office of Juvenile Justice and Delinquency Prevention.

AustLII (2014). Crimes Amendment (Protection of Children) Bill 2014, Parliament of Victoria. http://www.austlii.edu.au/au/legis/vic/bill/caocb2014381/. Accessed in September 2014.

Baccianella, S., Esuli, A., & Sebastiani, F. (2010). SentiWordNet 3.0: An Enhanced Lexical Resource for Sentiment Analysis and Opinion Mining. LREC-2010, p. 2200-2204.

Baccianella, S., Esuli, A., & Sebastiani, F. (2013). Using Micro-Documents for Feature Selection: The Case of Ordinal Text Classification. *Expert Systems with Applications, Vol. 40* (11), p. 4687-4696, Elsevier.

BayesLegal (2014). Bayes and the Law. https://sites.google.com/site/bayeslegal/legal-cases-relevant-to-bayes. Accessed in September, 2014.

Bengel, J., Gauch, S., Mittur, E., & Vijayaraghavan, R. (2004). ChatTrack: Chat Room Topic Detection using Classification. *Intelligence and Security Informatics.* (p. 266-277). Heidelberg, Germany: Springer. doi:10.1007/978-3-540-25952-7_20

Bentivogli, L., Dagan, I., Dang, H. T., Giampiccolo, D., & Magnini, B. (2010). The Sixth Pascal Recognizing Textual Entailment Challenge. *Proceedings of TAC, Vol. 10,* p. 1-14, Retrieved from http://www.nist.gov/tac/2010/RTE/RTE6_Main_NoveltyDetection_Task_Guidelines.pdf.

Berliner, L. (2002). Confronting an Uncomfortable Reality. *The APSAC Advisor, Vol. 14* (2), p. 2–4, American Professional Society on the Abuse of Children.

Bezdek, J. C. (1974). Cluster Validity with Fuzzy Sets. *Journal of Cybernetics, Vol. 3* (3), p. 58-72, Taylor & Francis.

Bezdek, J. C. (1975). Mathematical Models for Systematics and Taxonomy. *Proceedings of 8th International Conference on Numerical Taxonomy,* San Francisco. *Vol. 3,* p. 143-166.

Bifet, A., & Frank, E. (2010). Sentiment Knowledge Discovery in Twitter Streaming Data. *13th International Conference on Discovery Science,* Canberra, Australia. p. 1-15.

Bogdanova, D., Rosso, P., & Solorio, T. (2014). Exploring High-Level Features for Detecting Cyberpedophilia. *Computer Speech & Language, Vol. 28* (1), p. 108-120, doi:http://dx.doi.org/10.1016/j.csl.2013.04.007.

Cambridge Dictionary (2014) . Cambridge Dictionaries Online. Cambridge University Press. http://dictionary.cambridge.org/dictionary/learner-english/evidence Accessed in September 2014.

Castillo, J. J. (2010). An Approach to Recognizing Textual Entailment and TE Search Task using SVM. *Procesamiento Del Lenguaje Natural (Natural Language Processing), Vol. 44,* pp. 139-145, Sociedad Española para el Procesamiento del Lenguaje Natural (Spanish Society for Natural Language Processing).

Castillo, J., & Cardenas, M. (2010). Using Sentence Semantic Similarity Based on WordNet in Recognizing Textual Entailment. *Advances in Artificial Intelligence–IBERAMIA 2010,* p. 366-375, Springer.

Chaski, C. E. (2005). Who's at the Keyboard? Authorship Attribution in Digital Evidence Investigations. *International Journal of Digital Evidence, Vol. 4* (1), p. 1-13, Retrieved from http://www.utica.edu/academic/institutes/ecii/publications/articles/B49F9C4A-0362-765C-6A235CB8ABDFACFF.pdf.

Choo, K. K. R. (2009). Online Child Grooming: A Literature Review on the Misuse of Social Networking Sites for Grooming Children for Sexual Offences. *AIC Reports: Research and Public Policy Series 103.* Retrieved in October 2010 from http://www.aic.gov.au/documents/3/C/1/%7B3C162CF7-94B1-4203 -8C57-79F827168DD8%7Drpp103.pdf.

Craven, S., Brown, S., & Gilchrist, E. (2006). Sexual Grooming of Children: Review of Literature and Theoretical Considerations. *Journal of Sexual Aggression, Vol. 12* (3), p. 287-299, Routledge.

Dagan, I., & Glickman, O. (2004). Probabilistic Textual Entailment: Generic Applied Modeling of Language Variability. *PASCAL Workshop on Learning Methods for Text Understanding and Mining (2004),* Grenoble, France.

de Salvo Braz, R., Girju, R., Punyakanok, V., Roth, D., & Sammons, M. (2005). Knowledge Representation for Semantic Entailment and Question-Answering. *IJCAI-05 Workshop on Knowledge and Reasoning for Question Answering,* Edinburgh, UK. p. 71-80.

de Salvo Braz, R., Girju, R., Punyakanok, V., Roth, D., & Sammons, M. (2006). An Inference Model for Semantic Entailment in Natural Language. In J.

Quiñonero-Candela, I. Dagan, B. Magnini & F. d'Alché-Buc (Eds.), *Machine Learning Challenges. Evaluating Predictive Uncertainty, Visual Object Classification, and Recognising Textual Entailment* (p. 261-286). Berlin Heidelberg, Germany: Springer. doi:10.1007/ 11736790_15.

Dekhtyar, A., & Hayes, J. (2006). Good Benchmarks are Hard to Find: Toward the Benchmark for Information Retrieval Applications in Software Engineering. In ICSM 2006 Working Session: Information Retrieval Based Approaches in Software Evolution 2007.

Dempster, A. P., Laird, N. M., & Rubin, D. B. (1977). Maximum Likelihood from Incomplete Data Via the EM Algorithm. *Journal of the Royal Statistical Society (Series B), Vol. 39* (1), p. 1-38.

de Oliveira, J. V., & Pedrycz, W. (2007). *Advances in Fuzzy Clustering and its Applications.* Wiley Online Library.

Dhruva, N., Ferschke, O., & Gurevych, I. (2014). Solving Open-Domain Multiple Choice Questions with Textual Entailment and Text Similarity Measures. *CLEF 2014 Working Notes.*

Diederich, J., Kindermann, J., Leopold, E., & Paass, G. (2003). Authorship Attribution with Support Vector Machines. *Applied Intelligence, Vol. 19* (1-2), p. 109-123, doi:10.1023/A:1023824908771. Kluwer Academic Publishers.

Dinakar, K., Reichart, R., & Lieberman, H. (2011). Modeling the Detection of Textual Cyberbullying. *International AAAI Conference on Weblogs and Social Media, Workshop "Social Mobile Web."* Barcelona.

Dubois, D., Prade, H., Esteva, F., Garcia, P., & Godo, L. (1997). A Logical Approach to Interpolation Based on Similarity Relations. *International Journal of Approximate Reasoning, Vol. 17* (1), p. 1-36, Elsevier.

Ellis, P. D. (2010). *The Essential Guide to Effect Sizes : Statistical Power, Meta-Analysis, and the Interpretation of Research Results.* (1st ed.). Cambridge, UK: Cambridge University Press.

Eltzeroth, R., & Elzerman, T. (1981). *The Crime Scene Technician Manual.* The University of Illinois and Illois, Department of Law Enforcement.

Esteva, F., Garcia, P., Godo, L., & Rodríguez, R. (1997). A Modal Account of Similarity-Based Reasoning. *International Journal of Approximate Reasoning, Vol. 16* (3), p. 235-260, Elsevier.

Esteva, F., Godo, L., Rodrıguez, R. O., & Vetterlein, T. (2010). On the Logics of Similarity-Based Approximate and Strong Entailment. *Proceedings of the 15th Spanish Congress on Fuzzy Logic and Technology, ESTYLF,* p. 187-192.

Esteva, F., Godo, L., Rodríguez, R. O., & Vetterlein, T. (2012). Logics for Approximate and Strong Entailments. *Fuzzy Sets and Systems, Vol. 197,* p. 59-70, Elsevier.

Facebook-statistics (2014). http://www.Facebook.com/press/info.Php?Statistics. Accessed in September 2014.

Fellbaum, C. (1998). *Wordnet: An Electronic Lexical Database*. Cambridge: MIT Press.

Figueroa, A., & Neumann, G. (2014). Category-Specific Models for Ranking Effective Paraphrases in Community Question Answering. *Expert Systems with Applications, Vol. 41*(10), p. 4730-4742, Elsevier.

Finkel, J. R., Grenager, T., & Manning, C. (2005). Incorporating Non-Local Information into Information Extraction Systems by Gibbs Sampling. *Proceedings of the 43rd Annual Meeting on Association for Computational Linguistics,* p. 363-370.

Fisher, D., Beech, A., & Browne, K. (1999). Comparison of Sex Offenders to Nonoffenders on Selected Psychological Measures. *International Journal of Offender Therapy and Comparative Criminology, Vol. 43*(4), p. 473, Sage Publications.

Forsyth, E. N., & Martell, C. H. (2007). Lexical and Discourse Analysis of Online Chat Dialog. *First IEEE International Conference on Semantic Computing 2007,* Irvine, California, USA. p. 19-26. doi:10.1109/ICSC.2007.55.

Gansterer, W., & Pölz, D. (2009). E-Mail Classification for Phishing Defense. In M. Boughanem, C. Berrut, J. Mothe & C. Soule-Dupuy (Eds.), *Advances in Information Retrieval.* (p. 449-460). Berlin Heidelberg, Germany: Springer. doi:10.1007/978-3-642-00958-7_40.

Glickman, O. (2006). *Applied Textual Entailment.* Doctoral dissertation, Bar Ilan University, Israel.

Godo, L., & Rodríguez, R. O. (2008). Logical Approaches to Fuzzy Similarity-Based Reasoning: An Overview. *Preferences and Similarities. Vol. 504,* p. 75-128, Springer Vienna.

Hall, M., Frank, E., Holmes, G., Pfahringer, B., Reutemann, P., & Witten, I. H. (2009). The WEKA Data Mining Software: An Update. *ACM SIGKDD Explorations Newsletter, Vol. 11* (1), p. 10-18, ACM.

Witten, I. H., Frank, E., & Hall, M. A. (2011). Data Mining: Practical Machine Learning Tools and Techniques. (3rd ed.) Morgan Kaufmann.

Hastie, T., Friedman, J., & Tibshirani, R. (2009). *The Elements of Statistical Learning.* Springer.

Havenstein, H. (2007). Meet the Virtual Generation. *Pcworld,* Retrieved in September 2014 from http://www.pcworld.com/article/ 139748/article.html.

Heilman, M., & Smith, N. A. (2010). Tree Edit Models for Recognizing Textual Entailments, Paraphrases, and Answers to Questions. *Human Language Technologies: The 11th Annual Conference of the North American Chapter of the Association for Computational Linguistics,* Los Angeles, CA, USA. p. 1011-1019.

Hoerl, A. E., & Kennard, R. W. (1970). Ridge Regression: Biased Estimation for Nonorthogonal Problems. *Technometrics, Vol. 12(1),* p. 55-67, Taylor & Francis Group.

Hope, D. (2008). *Java Wordnet Similarity Library.* Retrieved from http://www.sussex.ac.uk/Users/drh21/.

Horswell, J. (Ed.). (2004). *The practice of crime scene investigation*. CRC Press. USA.

Howitt, D. (1995). Pornography and the Paedophile: Is it Criminogenic? *British Journal of Medical Psychology, Vol. 68 (1),* p. 15-27, Wiley Online Library.

Inches, G., & Crestani, F. (2012). Overview of the International Sexual Predator Identification Competition at PAN-2012. *CLEF 2012 Evaluation Labs and Workshop-Working Notes Papers.* Rome, Italy.

Irclog. http://www.irclog.org/. Accessed in September 2014.

Ivkovic, S., & Ma, L. (2009). Discovering Problems, Emotions, Expectations and Lexical Patterns from Project Related E-Mail Communication. *IADIS International Conference on Applied Computing 2009,* Rome, Italy. *, VII*(2) p. 176-180.

James, R. E., Meloan, C. E., & Saferstein, R. (1980). *Laboratory Manual for Criminalistics.* Prentice-Hall.

Jezek, K., & Hynek, J. (2007). The Fight Against Spam-A Machine Learning Approach. *Conference on Electronic Publishing,* Vienna, Austria. p. 381-392.

Jiang, J. J., & Conrath, D. W. (1997). Semantic Similarity Based on Corpus Statistics and Lexical Taxonomy. *Proceedings of International Conference on Research in Computational Linguistics,* Taiwan. p. 19-33.

Joachims, T. (1998). Text Categorization with Support Vector Machines: Learning with Many Relevant Features. *10th European Conference on Machine Learning,* Chemnitz, Germany. *, 1398* p. 137-142. doi:10.1007/BFb0026683.

Kerlikowske, R. G., & Wilson, M. (2007). NetSmartz: A Comprehensive Approach to Internet Safety and Awareness. *Police Chief Magazine, Vol.74*, p.46. Retrieved from http://policechiefmagazine.org/magazine/index.cfm?fuseaction=display&article_id=1157&issue_id=42007

Klein, D., & Manning, C. D. (2003). Accurate Unlexicalized Parsing. *Proceedings of the 41st Annual Meeting on Association for Computational Linguistics, Vol. 1,* Morristown, NJ, USA. p. 423-430.

Kontostathis, A., Edwards, L., Bayzick, J., McGhee, I., Leatherman, A., & Moore, K. (2009). Comparison of Rule-Based to Human Analysis of Chat Logs. *1st International Workshop on Mining Social Media Programme, Conferencia De La Asociación Española Para La Inteligencia Artificial (2009), Vol.* 8, p. 2.

Krone, T. (2005). Queensland Police Stings in Online Chat Rooms. *Trends & Issues in Crime and Criminal Justice Series*. Accessed in November 2010 from http://www.aic.gov.au/documents/B/C/E/%7BBCEE2309-71E3-4EFA-A533-A39661BD1D29%7Dtandi301.pdf.

Kucukyilmaz, T., Cambazoglu, B. B., Aykanat, C., & Can, F. (2008). Chat Mining: Predicting User and Message Attributes in Computer-Mediated

Communication. *Information Processing & Management, Vol. 44(4),* p. 1448-1466, Elsevier.

Lamb, M. E., & Brown, D. A. (2006). Conversational Apprentices: Helping Children Become Competent Informants about their Own Experiences. *British Journal of Developmental Psychology, Vol. 24(1),* p. 215-234, British Psychological Society.

Landauer, T. K., Foltz, P. W., & Laham, D. (1998). An Introduction to Latent Semantic Analysis. *Discourse Processes, Vol. 25 (2-3),* p. 259-284, Taylor & Francis.

Leacock, C., & Chodorow, M. (1998). Combining Local Context and WordNet Similarity for Word Sense Identification. *WordNet: An Electronic Lexical Database, Vol. 49 (2),* p. 265-283.

Leatherman, A. (2009). *Luring Language and Virtual Victims: Coding Cyber-Predators on-Line Communicative Behavior.* PhD, Media and Communication Studies, Ursinus College.

Lee, H., Peirsman, Y., Chang, A., Chambers, N., Surdeanu, M., & Jurafsky, D. (2011). Stanford's Multi-Pass Sieve Coreference Resolution System at the CoNLL-2011 Shared Task. *Proceedings of the Fifteenth Conference on Computational Natural Language Learning: Shared Task,* p. 28-34.

Lee, K., Palsetia, D., Narayanan, R., Patwary, M. M. A., Agrawal, A., & Choudhary, A. (2011). Twitter Trending Topic Classification. Data Mining Workshops (ICDMW), 2011 IEEE 11th International Conference on, p. 251-258.

Lenhart, A. (2007). *Social Networking Websites and Teens: An Overview.* Pew Internet & American Life Project. Retrieved from http://www.pewinternet.org/~/media//Files/Reports/2007/PIP_SNS_Data_Memo_Jan_2007.pdf.

Lesk, M. (1986). Automatic Sense Disambiguation using Machine Readable Dictionaries: How to Tell a Pine Cone from an Ice Cream Cone. *Proceedings of the 5th Annual International Conference on Systems Documentation,* New York, NY, USA. p. 24-26.

Li, Y., McLean, D., Bandar, Z. A., O'Shea, J. D., & Crockett, K. (2006). Sentence Similarity Based on Semantic Nets and Corpus Statistics. *IEEE Transactions on Knowledge and Data Engineering, Vol. 18 (8)*, p. 1138-1150.

Lin, D. (1998). An Information-Theoretic Definition of Similarity. *Proceedings of the 15th International Conference on Machine Learning,* Madison, Wisconsin, USA. p. 296-304.

LPR.OxfordJournal (2014). *Law, Probability & Risk*. http://lpr.oxfordjournals.org/. Accessed in September 2014.

MacQueen, J. (1967). Some Methods for Classification and Analysis of Multivariate Observations. *Proceedings of the Fifth Berkeley Symposium on Mathematical Statistics and Probability, Vol. 1(14),* p. 281-297.

Manning, C. D., Raghavan, P., & Schütze, H. (2009). *An Introduction to Information Retrieval.* Cambridge University Press.

Marshall, W. L., Barbaree, H. E., & Fernandez, Y. M. (1995). Some Aspects of Social Competence in Sexual Offenders. *Sexual Abuse: A Journal of Research and Treatment, Vol. 7 (2)*, p. 113-127, SAGE Publications. doi:10.1177/1079063 29500700202.

McGhee, I., Bayzick, J., Kontostathis, A., Edwards, L., McBride, A., & Jakubowski, E. (2011). Learning to Identify Internet Sexual Predation. *International Journal of Electronic Commerce, Vol. 15 (3)*, p. 103-122, ME Sharpe.

McNulty, P. J. (2007, March 2007). Project Safe Childhood. *Police Chief Magazine, Vol. 74*, p. 36. Retrieved from http://policechiefmagazine.org/magazine/index.cfm?fuseaction=display_arch&article_id=1138&issue_id=32007.

Mehdad, Y., Moschitti, A., & Zanzotto, F. M. (2010). Syntactic/Semantic Structures for Textual Entailment Recognition. *Human Language Technologies: The 11th Annual Conference of the North American Chapter of the Association for Computational Linguistics,* Los Angeles, CA, USA. p. 1020-1028.

Miah, M. W. R., Yearwood, J., & Kulkarni, S. (2011). Detection of Child Exploiting Chats from a Mixed Chat Dataset as a Text Classification Task. *Proceedings of the*

Australasian Language Technology Association Workshop 2011, Canberra, Australia. p. 157-165.

Miah, M. W. R., Yearwood, J., & Kulkarni, S. (2014). Constructing an inter-post Similarity Measure to Differentiate the Psychological Stages in Offensive Chats. *Journal of the Association for Information Science and Technology,* doi:10.1002/asi.23247. John Wiley & Sons, Ltd.

Mihalcea, R., Corley, C., & Strapparava, C. (2006). Corpus-Based and Knowledge-Based Measures of Text Semantic Similarity. *Proceedings of the Twenty-First National Conference on Artificial Intelligence (AAAI-06) ,* Boston, Massachusetts, USA. p. 775-780.

Morris, J., & Hirst, G. (1991). Lexical Cohesion Computed by Thesaural Relations as an Indicator of the Structure of Text. *Computational Linguistics, Vol. 17 (1),* p. 21-48, MIT Press.

NAA (2014). Evidence Law in Australia. http://www.naa.gov.au/records-management/strategic-information/standards/records-in-evidence /evidence-law-australia.aspx. Accessed in September 2014.

O'Connell, R. (2003). A Typology of Child Cybersexploitation and Online Grooming Practices. Retrieved in August 2010 from http://image.guardian.co.uk/sys-files/Society /documents/2003/07/17/Groomingreport.pdf.

Ofoghi, B., & Yearwood, J. (2011). Learning Parse-Free Event-Based Features for Textual Entailment Recognition. In J. Li (Ed.), *AI 2010: Advances in Artificial Intelligence.* p. 184-193, Berlin Heidelberg, Germany: Springer. doi:10.1007/978-3-642-17432-2_19.

Ofoghi, B., & Yearwood, J. L. (2009). From Lexical Entailment to Recognizing Textual Entailment using Linguistic Resources. *Australasian Language Technology Association Workshop 2009,* Sydney, Australia. *, Vol. 7,* p. 119-123.

Olson, L. N., Daggs, J. L., Ellevold, B. L., & Rogers, T. K. K. (2007). Entrapping the Innocent: Toward a Theory of Child Sexual Predators' Luring Communication. *Communication Theory, Vol. 17 (3),* p. 231-251, doi:10.1111/j.1468-2885.2007.00294.x. Wiley-Blackwell.

Omegle. http://omegle.inportb.com/. Cited in Inches and Crestani (2012).

Ortony, A., Clore, G. L., & Foss, M. A. (1987). The Referential Structure of the Affective Lexicon. *Cognitive Science, Vol. 11(3)*, p. 341-364, doi:http://dx.doi.org/10.1016/S0364-0213(87)80010-1.

Osherenko, A. (2008). Towards Semantic Affect Sensing in Sentences. *Affective Language in Human and Machine: AISB 2008 Convention on Communication, Interaction and Social Intelligence, Vol. 2,* p. 41-44, Aberdeen, Scotland.

Osman, D., Yearwood, J., & Vamplew, P. (2009). Weblogs for Market Research: Finding More Relevant Opinion Documents using System Fusion. *Online Information Review, Vol. 33 (5),* p. 873-888, doi:10.1108/14684520911001882. Emerald Group Publishing Limited.

Osman, D., Yearwood, J., & Vamplew, P. (2010). Automated Opinion Detection: Implications of the Level of Agreement between Human Raters. *Information Processing & Management, Vol. 46 (3)*, p. 331-342, Elsevier.

Osman, D. J., & Yearwood, J. L. (2007). Opinion Search in Web Logs. *Eighteenth Australasian Database Conference,* Ballarat, Victoria, Australia.*, Vol. 63,* p. 133-139.

Oxford Dictionary (2014). Oxford English Dictionary, Oxford University Press. http://public.oed.com/history-of-the-oed/dictionary-facts/. Accessed in September 2014.

Padó, S., Noh, T., Stern, A., Wang, R., & Zanoli, R. (2013). Design and Realization of a Modular Architecture for Textual Entailment. *Natural Language Engineering, Vol.1 (1)*, p. 1-34, Cambridge University Press.

Pan (2014). http://pan.webis.de/. Accessed in September 2014.

Panton, J. H. (1979). MMPI Profile Configurations Associated with Incestuous and Non-Incestuous Child Molesting. *Psychological Reports, Vol. 45* (1), p. 335-338, MEDLINE.

Pedersen, T., Patwardhan, S., & Michelizzi, J. (2004). WordNet:: Similarity: Measuring the Relatedness of Concepts. *Demonstration Papers at HLT-NAACL 2004,* p. 38-41.

Perverted-Justice.Com(PJ). http://www.perverted-justice.com/. Accessed in September 2014.

Penas, A., & Rodrigo, A. (2011). A Simple Measure to Assess Non-Response. *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies, Vol. 1,* p. 1415-1424.

Pendar, N. (2007). Toward Spotting the Pedophile Telling Victim from Predator in Text Chats. *Proceedings of the First IEEE International Conference on Semantic Computing,* Irvine, California, USA, p. 235-241.

Pennebaker, J. W., Mehl, M. R., & Niederhoffer, K. G. (2003). Psychological Aspects of Natural Language use: Our Words, our Selves. *Annual Review of Psychology, Vol. 54,* (1), pp. 547-577.

Pennebaker, J. W., Chung, C. K., Ireland, M., Gonzales, A., & Booth, R. J. (2007). *The Development and Psychometric Properties of LIWC2007.* Austin, TX, USA. Retrieved in August 2010 from http://www.liwc.net/LIWC2007LanguageManual.pdf.

pjfi.org. Perverted Justice Foundation Incorporated. http://pjfi.org/. Accessed September 2014.

Popescu, M., & Grozea, C. (2012). Kernel Methods and String Kernels for Authorship Analysis. *CLEF 2012 Evaluation Labs and Workshop - Working Notes Papers,* Rome, Italy.

Queensland Criminal Code. (2014). Retrieved in November, 2014 from https://www.legislation.qld.gov.au/LEGISLTN/CURRENT/C/CriminCode.pdf.

Quinlan, J. R. (1993). *C4. 5: Programs for Machine Learning.* Morgan kaufmann.

Rand, W. M. (1971). Objective Criteria for the Evaluation of Clustering Methods. *Journal of the American Statistical Association, Vol. 66*(336), p. 846-850, Taylor & Francis.

Resnik, P. (1995). Using Information Content to Evaluate Semantic Similarity. *Proceedings of the 14th International Joint Conference on Artificial Intelligence, Vol. 1,* p. 448-453. Montreal, Québec, Canada.

Rijsbergen, V. (1979). *Information Retrieval* (2nd ed.). Newton, MA, USA: Butterworth-Heinemann.

Ropelato, J. (2007). Internet Pornography Statistics. *Top Ten Reviews*. Retrieved in August 2010 from http://internet-filter-review.toptenreviews.com/internet-pornography-statistics.html.

Rosa, K. D., & Ellen, J. (2009). Text Classification Methodologies Applied to Micro-Text in Military Chat. *Eighth IEEE International Conference on Machine Learning and Applications,* p. 710-714, Miami Beach, Florida, USA.

Ross, M. K., Lin, K., Truong, K., Kumar, A., & Conway, M. (2013). Text Categorization of Heart, Lung, and Blood Studies in the Database of Genotypes and Phenotypes (dbGap) Utilizing n-Grams and Metadata Features. *Biomedical Informatics Insights, Vol. 6,* p. 35, Libertas Academica.

Ruspini, E. H. (1991). On the Semantics of Fuzzy Logic. *International Journal of Approximate Reasoning, Vol. 5* (1), p. 45-88, Elsevier.

RvAdams. (1996). Regina Vs Dennis John Adams. Court of Appeal of England and Wales. Accessed from http://www.bailii.org/cgi-bin/markup.cgi?doc=/ew/cases/EWCA/Crim/2006/222.html&query=bayes&method=boolean.

Salton, G., Wong, A., & Yang, C. (1975). A Vector Space Model for Automatic Indexing. *Communications of the ACM, Vol. 18* (11), p. 613-620, ACM.

Salton, G., & Buckley, C. (1988). Term-Weighting Approaches in Automatic Text Retrieval. *Information Processing & Management, Vol. 24*(5), p. 513-523, doi:10.1016/0306-4573(88)90021-0.

Schell, B. H., Martin, M. V., Hung, P. C. K., & Rueda, L. (2007). Cyber Child Pornography: A Review Paper of the Social and Legal Issues and Remedies--and a Proposed Technological Solution. *Aggression and Violent Behavior, Vol. 12* (1), p. 45-63, Elsevier.

Sebastiani, F. (2002). Machine Learning in Automated Text Categorization. *ACM Computing Surveys, Vol. 34*(1), p. 1-47, Association for Computing Machinery.

Shannon, C. E. (1948). A Mathematical Theory of Communication. *Bell System Technical Journal, Vol. 27,* p. 379-423 and 623-656, Retrieved in September 2014 from http://cm.bell-labs.com/cm/ms/what/shannonday/paper.html.

Skabar, A., & Abdalgader, K. (2011). Clustering Sentence-Level Text using a Novel Fuzzy Relational Clustering Algorithm. *IEEE Transactions on Knowledge and Data Engineering, Vol. 25* (1), p. 62-75, doi:10.1109/TKDE.2011.205. IEEE Xplore.

Sleator, D., Temperley, D., & Lafferty, J. (2004). *Link Grammar Parser (LGP)*. Accessed from http://www.Link.Cs.Cmu.edu/link/submit-Sentence-4.html.

Stern, A., & Dagan, I. (2014). Recognizing Implied Predicate-Argument Relationships in Textual Inference. *Proceedings of ACL-2014. Association for Computational Linguistics.*

Strapparava, C., & Mihalcea, R. (2007). Semeval-2007 Task 14: Affective Text. *Proceedings of the 4th International Workshop on Semantic Evaluations,* p. 70-74.

Strapparava, C., & Valitutti, A. (2004). WordNet Affect: An Affective Extension of WordNet. *LREC, Vol. 4,* p. 1083-1086.

Tatar, D., Serban, G., Mihis, A., & Mihalcea, R. (2009). Textual Entailment as a Directional Relation. *Journal of Research and Practice in Information Technology, Vol. 41* (1), p. 53, Australian Computer Society Inc.

Tatar, D., Mihis, A., & Lupsa, D. (2008). Text Entailment for Logical Segmentation and Summarization. *Natural Language and Information Systems.* (p. 233-244). Berlin Heidelberg, Germany: Springer. doi:10.1007/978-3-540-69858-6_24.

TeenChatDecoder.com.http://www.teenchatdecoder.com/. Accessed August 2010.

Thelwall, M., Buckley, K., Paltoglou, G. Cai, D., & Kappas, A. (2010). Sentiment strength detection in short informal text. Journal of the American Society for Information Science and Technology, 61(12), 2544–2558.

Thelwall, M., Buckley, K., & Paltoglou, G. (2011). Sentiment in Twitter Events. Journal of the American Society for Information Science and Technology, Vol. 62, (2), pp. 406-418, Wiley Online Library.

Tibshirani, R. (1996). Regression Shrinkage and Selection Via the Lasso. *Journal of the Royal Statistical Society.Series B (Methodological),* p. 267-288, JSTOR.

Tsuboi, Y., & Matsumoto, Y. (2002). Authorship Identification for Heterogeneous Documents. *IPSJ SIG Notes, Vol. NL-148-3,* p. 17-24, Information Processing Society of Japan.

Turney, P. D., & Mohammad, S. M. (2013). Experiments with Three Approaches to Recognizing Lexical Entailment. *Natural Language Engineering,* p. 1-40, Cambridge Univ Press.

Uğuz, H. (2011). A Two-Stage Feature Selection Method for Text Categorization by using Information Gain, Principal Component Analysis and Genetic Algorithm. *Knowledge-Based Systems, Vol. 24* (7), p. 1024-1032, Elsevier.

Valdes, R. (2007). *Facebook and the Emerging Social Platform Wars.* Gartner Inc. Retrieved from http://images.dinnosaur.multiply.multiplycontent.com/attachment/0/R3RYCAoKClAAAC-ETkE1/Face%2BBook%2Band%2Bthe%2BEmerging%2BSocial%2BPlatform%2BWars.pdf?key=dinnosaur:journal:192&nmid=74697100.

Villatoro-Tello, E., Juárez-González, A., Escalante, H. J., Montes-y-Gómez, M., & Pineda, L. V. (2012). A Two-Step Approach for Effective Detection of Misbehaving Users in Chats. *CLEF2012 (Online Working Notes/Labs/PAN Workshop),* Rome, Italy.

Walls, H. J. (1968). *Forensic Science.* London: Sweet & Maxwell.

Wang, R., & Zhang, Y. (2008). Recognizing Textual Entailment with Temporal Expressions in Natural Language Texts. *IEEE International Workshop on Semantic Computing and Applications 2008,* Incheon, South Korea, p. 109-116.

Ward Jr, J. H. (1963). Hierarchical Grouping to Optimize an Objective Function. *Journal of the American Statistical Association, Vol. 58,* (301), pp. 236-244, Taylor & Francis.

Ward, T., Hudson, S. M., & Marshall, W. L. (1996). Attachment Style in Sex Offenders: A Preliminary Study. *Journal of Sex Research, Vol. 33* (1), p. 17-26, Routledge.

Ward, T., McCormack, J., & Hudson, S. M. (1997). Sexual Offenders' Perceptions of their Intimate Relationships. *Sexual Abuse: A Journal of Research and Treatment, Vol. 9* (1), p. 57, SAGE Publications.

Witten, I. H., Frank, E., & Hall, M. A. (2011). *Data Mining: Practical Machine Learning Tools and Techniques.* (3rd ed.) Morgan Kaufmann.

Wittgenstein, L. (1958). *Philosophical Investigations.* (G. E. M. Anscombe Trans.). Oxford , UK: Basil Blackwell Ltd.

Wu, Z., & Palmer, M. (1994). Verbs Semantics and Lexical Selection. *Proceedings of the 32nd Annual Meeting on Association for Computational Linguistics,* Stroudsburg, PA, USA. p. 133-138. doi:10.3115/981732.981751.

Wu, T., Khan, F. M., Fisher, T. A., Shuler, L. A., & Pottenger, W. M. (2005). Posting Act Tagging using Transformation-Based Learning. In T. Young Lin, S. Ohsuga, C. Liau, X. Hu & S. Tsumoto (Eds.), *Foundations of Data Mining and knowledge Discovery* (p. 319-331). Springer Berlin Heidelberg Germany. doi:10.1007/11498186_18.

Wu, X., Kumar, V., Quinlan, J. R., Ghosh, J., Yang, Q., Motoda, H., McLachlan, G. J., Ng, A., Liu,B., Yu, P. S. (2008). Top 10 Algorithms in Data Mining. *Knowledge and Information Systems, Vol. 14* (1), p. 1-37, Springer.

Yang, Y., & Joachims, T. (2008). Text Categorization. *Scholarpedia, 3*(5), 4242. Accessed from http://www.scholarpedia.org/article/Text_categorization, Accessed in 11 September, 2014.

Yang, Y. (1999). An Evaluation of Statistical Approaches to Text Categorization. *Information Retrieval, Vol. 1* (1-2), p. 69-90, Springer. doi:10.1023/A:1009982220290.

Young, K. (2005). Profiling Online Sex Offenders, Cyber Predators, and Pedophiles. *Journal of Behavioral Profiling, Vol. 5* (1), p. 1-18, ABP.

Zanzotto, F. M., Pennacchiotti, M., & Moschitti, A. (2009). A Machine Learning Approach to Textual Entailment Recognition. *Natural Language Engineering, Vol. 15,* (04), pp. 551-582. Cambridge Univ Press. doi:10.1017/S1351324909990143.

Zhang, J. (2008). Benchmarks and Evaluation Criteria for Information Retrieval Visualization. In J. Zhang (Ed.), *Visualization for Information Retrieval.* (p. 239-254). Berlin Heidelberg: Springer. doi:10.1007/978-3-540-75148-9_11.

Zhang, X., Zhou, Z., & and Wu, M. (2009). Positive, Negative, Or Mixed? Mining Blogs for Opinions. *The Fourteenth Australasian Document Computing Symposium,* Sydney, Australia. p. 141-144.

Zheng, R., Li, J., Chen, H., & Huang, Z. (2006). A Framework for Authorship Identification of Online Messages: Writing-Style Features and Classification Techniques. *Journal of the American Society for Information Science and Technology, Vol. 57* (3), p. 378-393, doi:10.1002/asi.20316. ASIS&T.