A TEMPORAL EXPERT FINDING METHODOLOGY BASED ON UNITED AUTHOR-DOCUMENT-TOPIC GRAPHS

A THESIS SUBMITTED TO
THE GRADUATE SCHOOL OF INFORMATICS
OF
MIDDLE EAST TECHNICAL UNIVERSITY

BY

AHMET EMRE KILINÇ

IN PARTIAL FULFILLMENT OF THE REQUIREMENTS
FOR
THE DEGREE OF MASTER OF SCIENCE
IN
INFORMATION SYSTEMS

SEPTEMBER 2014

Approval of the thesis:

A TEMPORAL EXPERT FINDING METHODOLOGY BASED ON UNITED AUTHOR-DOCUMENT-TOPIC GRAPHS

submitted by **AHMET EMRE KILINÇ** in partial fulfillment of the requirements for the degree of **Master of Science in Information Systems Department, Middle East Technical University** by,

Prof. Dr. Nazife Baykal
Director, Graduate School of **Informatics Institute** _______________

Prof. Dr. Yasemin Yardımcı Çetin
Head of Department, **Information Systems** _______________

Assist. Prof. Dr. Tuğba Taşkaya Temizel
Supervisor, **Information Systems, METU** _______________

Examining Committee Members:

Prof. Dr. Nihan Kesim Çiçekli
Computer Engineering, METU _______________

Assist. Prof. Dr. Tuğba Taşkaya Temizel
Information Systems, METU _______________

Assoc. Prof. Dr. Banu Günel
Information Systems, METU _______________

Assoc. Prof. Dr. Aysu Betin Can
Information Systems, METU _______________

Assist. Prof. Dr. Aybar Can Acar
Medical Informatics, METU _______________

Date: _______________

I hereby declare that all information in this document has been obtained and presented in accordance with academic rules and ethical conduct. I also declare that, as required by these rules and conduct, I have fully cited and referenced all material and results that are not original to this work.

Name, Last Name: AHMET EMRE KILINÇ

Signature :

ABSTRACT

A TEMPORAL EXPERT FINDING METHODOLOGY BASED ON UNITED AUTHOR-DOCUMENT-TOPIC GRAPHS

Kılınç, Ahmet Emre
M.S., Department of Information Systems
Supervisor : Assist. Prof. Dr. Tuğba Taşkaya Temizel

September 2014, 117 pages

Expert finding is a challenging research topic due to fast paced technological development resulting in changes in people's expertise areas in time. However, the majority of the studies in the literature about expert finding systems do not take into account such temporal changes. For example, probabilistic models, which are widely used in this domain, are based on word or term associations between queries and documents. On the other hand, separated author-document graphs, which are used as baseline approach in this thesis, are based on topic modeling techniques. This approach does not take into consideration both queries and documents in the same topic modeling process, but it considers only documents in topic modeling process. As a result, it impairs relations between topic queries and documents. In this thesis, a novel expert finding system which uses domain limited Latent Dirichlet Allocation (LDA) based topic modeling and dynamic, united author-document-topic graphs is proposed. The proposed method is tested with ArnetMiner and UVT datasets and outperforms the baseline separated author-document-topic approach.

Keywords: Expert finding, Domain limited topic modeling, Temporal author-document-topic graphs

ÖZ

BİRLEŞİK YAZAR-DOKÜMAN-KONU ÇİZGELERİ TEMELLİ ZAMANA DAYALI UZMAN BULMA YÖNTEMİ

Kılınç, Ahmet Emre
Yüksek Lisans, Bilişim Sistemleri Bölümü
Tez Yöneticisi : Yrd. Doç. Dr. Tuğba Taşkaya Temizel

Eylül 2014 , 117 sayfa

Teknolojik gelişmelerdeki yüksek hızın zaman içinde insanların uzmanlık alanlarında değişimlerle sonuçlanmasından dolayı, uzman bulma zorlu bir araştırma konusudur. Ancak, literatürdeki uzman bulma sistemleriyle ilgili çalışmaların büyük çoğunluğu zamana bağlı değişiklikleri hesaba katmamaktadır. Örneğin, bu alanda çokça kullanılan olasılıksal modeller, sorgular ve dokümanlar arasındaki kelime veya terim ilişkilerini temel almaktadır. Öte yandan, bu tez çalışmasında temel yöntem olarak referans alınan ayrık doküman-yazar-konu çizgeleri, konu modellemesi tekniklerini esas almaktadır. Bu yaklaşım, konu sorgularını ve dokümanları aynı konu modellemesi sürecinde göz önünde bulundurmamakta, sadece dokümanları konu modellemesi sürecinde dikkate almaktadır. Bunun sonucunda, konu sorguları ve dokümanlar arasındaki ilişkiler zayıf kalmaktadır. Bu tezde, alan kısıtlı Latent Dirichlet Allocation (LDA) kullanan bir konu modellemesi ve zamana dayalı birleşik yazar-doküman-konu çizgeleri temelli yeni bir uzman bulma sistemi önerilmektedir. Önerilen sistem ArnetMiner ve UVT veri setleri üzerinden test edilmiş ve temel aldığı ayrık yazar-doküman-konu yaklaşımından daha iyi sonuç vermiştir.

Anahtar Kelimeler: Uzman bulma, Alan kısıtlı konu modellemesi, Zamana dayalı yazar-doküman-konu grafikleri

To my family and people who are reading this page

ACKNOWLEDGMENTS

I would like to express my thanks and gratitude to my supervisor Assist. Prof. Dr. Tuğba Taşkaya Temizel for her constant support and guidance throughout my research and academic life. Whenever I need help, she has assisted me by her invaluable suggestions and criticisms.

I would like to address my thanks to my managers and friends at TÜBİTAK YTE. They have encouraged me for the past two years to complete my research and thesis.

Lastly, I am also deeply thankful to each member of my family for their constant love, support and encouragement throughout my life.

TABLE OF CONTENTS

ABSTRACT v

ÖZ vi

ACKNOWLEDGMENTS viii

TABLE OF CONTENTS ix

LIST OF TABLES xiv

LIST OF FIGURES xv

LIST OF ALGORITHMS xviii

LIST OF ABBREVIATIONS xix

CHAPTERS

1 INTRODUCTION 1

2 BACKGROUND 5

2.1 Expert Finding 5

2.1.1 Expert Finding Systems 5

2.1.1.1 Content-Based Approach 5

2.1.1.2 Link-Analysis Based Approach 6

2.1.2 Expert Finding Tools 7

2.2 Topic Modeling 8

2.3 Weirdness 9

2.4 Temporality 9

2.5 ADT Graphs 10

2.6 Limitations of Current Studies 11

3 BASELINE APPROACH: RANKING EXPERTS USING AUTHOR-DOCUMENT-TOPIC GRAPHS 13

3.1 Overview of Author-Document-Topic Graphs 13

3.2 Experimental Settings 14

3.2.1 Datasets 14

3.2.2 Topic Query-Document Indexing 15

3.2.3 Graph Generation 15

3.2.4 Node Similarity Calculation Methods 15

3.3 Experiment Results 16

4 A TEMPORAL EXPERT FINDING METHODOLOGY BASED ON DOMAIN LIMITED TOPIC MODELING 23

4.1 Overview of The Proposed System 23

4.1.1 Topic-Document Relationship 26

4.1.2 Author-Document Relationship 26

4.2 Contributions of Proposed System 27

4.2.1 United ADT Approach 27

4.2.2 Stemming 27

4.2.3 Domain Specialty (Weirdness) 30
4.2.4 Temporality 34
4.3 Other Parameters Tested 37
4.3.1 Topic Count 37
4.3.2 LDA Alpha 38
4.3.3 LDA Beta 38
4.3.4 LDA Threshold 38
4.3.5 LDA Iteration Count 38
4.3.6 LDA Type (Hierarchical LDA) 39
4.4 Parameter Settings 40
4.5 Datasets 40
4.5.1 Sparsity Parameter 42
4.6 Experiment Setup 44
4.6.1 Experimental Environment 45
4.6.2 Experiment Progress 46
5 RESULTS AND DISCUSSION 47
5.1 Evaluation Setup and Metrics 47
5.2 United ADT Approach 47
5.3 Method 48
5.4 Topic Count 48
5.5 LDA Alpha 49

5.6 LDA Beta 49

5.7 LDA Threshold 49

5.8 LDA Iteration Count 50

5.9 Stemming 50

5.10 Temporality 50

5.11 Weirdness 51

5.12 Hierarchical Topic Modeling 51

5.13 Sparsity 51

5.14 Evaluation of Final Proposed System vs. Baseline Approach 52

5.15 Performance Evaluation and Discussion 53

5.16 Limitations 56

6 CONCLUSION AND FUTURE WORK 59

6.1 Conclusion 59

6.2 Future Work 60

REFERENCES 61

APPENDICES

A ALGORITHMS 65

B RESULTS OF METHOD AND NUMBER OF RESULTS CONTROLLED EXPERIMENTS 67

C RESULTS OF ADT TYPE CONTROLLED EXPERIMENTS 69

D RESULTS OF TOPIC COUNT CONTROLLED EXPERIMENTS . . 73

E RESULTS OF LDA ALPHA CONTROLLED EXPERIMENTS . . . 77

F RESULTS OF LDA BETA CONTROLLED EXPERIMENTS 79

G RESULTS OF LDA THRESHOLD CONTROLLED EXPERIMENTS 81

H RESULTS OF LDA ITERATION COUNT CONTROLLED EXPERIMENTS 85

I RESULTS OF STEMMING CONTROLLED EXPERIMENTS 89

J RESULTS OF TEMPORALITY CONTROLLED EXPERIMENTS . 93

K RESULTS OF WEIRDNESS CONTROLLED EXPERIMENTS . . . 99

L RESULTS OF SPARSITY CONTROLLED EXPERIMENTS 103

M RESULTS OF LDA TYPE (HIERARCHICAL / NORMAL) CONTROLLED EXPERIMENTS 105

N INDRI SEARCH ENGINE INDEX COMPARISON 109

O QUERIES OF UVT DATASET 111

P QUERIES OF ARNETMINER DATASET 117

LIST OF TABLES

TABLES

Table 3.1 Summary of Datasets Used in Baseline Approach [11] 14

Table 3.2 Baseline Sample Topics (ArnetMiner) [11] 19

Table 3.3 Baseline Sample Topics (UvT) [11] 20

Table 3.4 Baseline Performance Evaluation [11] 21

Table 4.1 ArnetMiner 10 topics without stemming 28

Table 4.2 ArnetMiner 10 topics with stemming 29

Table 4.3 ArnetMiner 10 topics without weirdness threshold 31

Table 4.4 ArnetMiner 10 topics with Weirdness Threshold = 0.8 33

Table 4.5 Parameters of Proposed System 40

Table 4.6 Properties of Datasets 41

Table 4.7 Experimental Environment 45

Table 5.1 Results using UVT dataset 52

Table 5.2 Results using ArnetMiner dataset 53

Table M.1 Example Hierarchical Topics for UVT 108

Table N.1 First 25 Document Indices for Topic 1276 (accounting) 110

Table O.1 Topic Queries of UVT Dataset 111

Table P.1 Topic Queries of ArnetMiner Dataset 117

LIST OF FIGURES

FIGURES

Figure 2.1 Probabilistic Latent Semantic Indexing (PLSI)[14] 8

Figure 2.2 Latent Dirichlet Allocation (LDA)[6] 9

Figure 2.3 Dynamic Research Interest Finding [7] 10

Figure 3.1 Baseline Author-Document-Topic Graph [11] 13

Figure 3.2 Baseline Performance with Number of Topics (ArnetMiner) [11] . . 17

Figure 3.3 Baseline Performance with Number of Topics (UVT) [11] 18

Figure 3.4 Baseline ADT Methods Performance (UVT) [11] 18

Figure 4.1 Detailed View of Baseline Approach 24

Figure 4.2 Folded View of Proposed System 25

Figure 4.3 Unfolded View of Proposed System 26

Figure 4.4 Temporality Effect = None 35

Figure 4.5 Temporality Effect = Linear 35

Figure 4.6 Temporality Effect = Quadratic 36

Figure 4.7 Temporal Link Strengths in ADT Graphs 37

Figure 4.8 Number of Topics / Log Likelihood Chart for UVT 37

Figure 4.9 United ADT Graph Example using Hierarchical LDA 39

Figure 4.10 Document Counts per Year for UVT 43

Figure 4.11 Document Counts per Year for ArnetMiner 43

Figure 4.12 All Progress 44

Figure 4.13 Interface for Creating Charts 46

Figure B.1 Method and Number of Results based Results for UVT 67

Figure B.2 Method Box Plot for UVT 68

Figure C.1 ADT Type Results According to Topic Counts for UVT 69

Figure C.2 ADT Type Box Plot for UVT 70

Figure C.3 ADT Type Time Spent According to Topic Counts for UVT 71

Figure D.1 Topic Count / Separated ADT Box Plot for UVT 73

Figure D.2 Topic Count / United ADT Box Plot for UVT 74

Figure D.3 ADT Type Results According to Topic Counts for ArnetMiner . . . 75

Figure E.1 LDA Alpha Box Plot for UVT 77

Figure E.2 LDA Alpha Time Spent Box Plot for UVT 78

Figure F.1 LDA Beta Box Plot for UVT 79

Figure F.2 LDA Beta Time Spent Box Plot for UVT 80

Figure G.1 LDA Threshold Results for UVT 81

Figure G.2 LDA Threshold Box Plot for UVT 82

Figure G.3 LDA Threshold Time Spent Box Plot for UVT 83

Figure H.1 LDA Iteration Count Results for UVT 85

Figure H.2 LDA Iteration Count Box Plot for UVT 86

Figure H.3 LDA Iteration Count Time Spent Box Plot for UVT 87

Figure I.1 Stemming Box Plot for UVT 89

Figure I.2 Stemming Box Plot for UVT 90

Figure I.3 Stemming Box Plot for ArnetMiner 91

Figure J.1 Temporality Box Plot for UVT 93

Figure J.2 Expertise Specific Temporality Results For UVT 94

Figure J.3 Temporality Box Plot for ArnetMiner 95

Figure J.4 Temporality Weirdness Based Recall for UVT 96

Figure J.5 Temporality Weirdness Based Recall for UVT 97

Figure K.1 Weirdness / Stemming based Results for UVT 99

Figure K.2 Weirdness Box Plot for UVT 100

Figure K.3 Weirdness Box Plot for ArnetMiner 101

Figure K.4 Weirdness Time Spent Results for UVT 102

Figure L.1 Sparsity Box Plot for UVT 103

Figure L.2 Sparsity Time Spent Box Plot for UVT 104

Figure M.1 LDA Type Box Plot for UVT 105

Figure M.2 LDA Type Results for UVT 106

Figure M.3 LDA Type Time Spent Box Plot for UVT 107

Figure N.1 Indri First 25 Indices for Topic 1276 (accounting) 109

LIST OF ALGORITHMS

ALGORITHMS

Algorithm 1 Temporality calculation algorithm 66

LIST OF ABBREVIATIONS

ACM	Association for Computing Machinery
ADT	Author-Document-Topic
AM	ArnetMiner
ANC	American National Corpus
API	Application Programming Interface
AVGP	Average Precision
DTM	Dynamic Topic Modeling
GLOSSEX	Glossary Extraction
HITS	Hyperlink-Induced Topic Search
LDA	Latent Dirichlet Allocation
LSI	Latent Semantic Indexing
MAP	Mean Average Precision
MRR	Mean Reciprocal Rank
PLSI	Probabilistic Latent Semantic Indexing
QREL	Query Relevance
TAT	Temporal-Author-Topic
TET	Temporal-Expert-Topic
TREC	Text Retrieval Conference
UVT	Universiteit van Tilburg (Tilburg University)
WWW	World Wide Web

CHAPTER 1

INTRODUCTION

Owing to the spread of World Wide Web (WWW), information flow speed among people has been increased dramatically. However, relevance and reliability of obtained information for a submitted inquiry may not be ensured at all times. "Information Retrieval" is about retrieving relevant information from a set of given resources. "Expert Finding" is a subtopic of "Information Retrieval" in which the goal is to retrieve relevant experts from a set of experts for a given expertise area (i.e. topic query). Expert finding systems have wide usage area. For example, in a scientific conference, reviewers are needed for assessment of submitted papers who are expected to be the experts in the research area of the conference topic. Another example is, when we look for a consultant for a project, we look for a consultant who has an authority and specialized subject of interest on project's domain. Therefore, it is crucial to find relevant experts in specific areas.

Performance of an information retrieval system can be measured by the relevance of retrieved documents for an input (i.e. precision, recall). Correspondingly, performance of an expert finding system can be figured out by the relevance of retrieved experts for the given expertise. Generally, documents that are related with a person, such as homepages, publications, course homepages, constitute the document corpus as input for the expert finding systems. When those raw documents are given as input to a topic query in an expert finding system, best performance could not be achieved because raw documents contain:

- Commonly used words of the language (stop words),
- Word variants with the same stem,

- Words that are not commonly used but not related to the corpus domain.

Relevance of an expert to an expertise is affected by temporal factors. For example, the author's interests may change over time, in other words, the author may previously be interested in data mining area but recently may become interested in computer networks. Therefore, an expert's recent research areas (i.e. publications) is significant for expert matching.

Taking into consideration those factors and limitations, in this thesis, we propose a novel expert finding system based on author-document-topic graphs. Gollapalli, Mitra, & Giles [11] have introduced author-document-topic (ADT) graphs using Latent Dirichlet Allocation (LDA) [6] topic modeling techniques through documents. We have chosen this study as baseline approach and extended the approach with the following improvements:

- **United ADT Graphs**: In the baseline approach, documents and topic queries were not entered into the same LDA process since topic queries were not considered as LDA input documents (in this thesis, we refer to this type of graphs as Separated ADT graphs). In our proposed model, we combined documents related to experts and topic queries and entered both into the same LDA process (in this thesis, we refer to this type of graphs as United ADT graphs).
- **Stemming**: Stemming was not used in our baseline approach. In our proposed system, Porter's stemming algorithm [23] is used for finding root form of the words.
- **Domain Specificity (Weirdness)**: In order to increase the significance of domain-specific words, we used "Weirdness" value that was introduced by Ahmad, Gillam, & Tostevin [1].
- **Temporality**: In order to achieve time dependency in our proposed approach, we put temporality effect as a parameter in our author-document-topic graphs between author and document nodes.

We tested our proposed system and baseline approach in 2 different datasets. The performance evaluation is done by standard information retrieval performance measures, specifically Average Precision, Average Recall, Mean Average Precision (MAP) and Mean Reciprocal Rank (MRR). Besides, we evaluated our system's runtime performance for each approach.

The rest of the thesis is organized as follows. The background studies in the literature about expert finding, topic modeling, ADT graphs and our other contributions are reviewed in Chapter 2. In Chapter 3, our baseline approach, the study of Gollapalli et al. [11] is reviewed and datasets, parameters and methods are analyzed in detail. In Chapter 4, we explained our proposed system in detail by describing all parameters, datasets and innovations added to the baseline approach. In Chapter 5, we presented our experimental results and discussed the results as well as explaining the limitations of the study. In Chapter 6, we summarized the work done and explained contributions of the study and possible improvement points of the proposed system.

CHAPTER 2

BACKGROUND

This chapter explains the previous work in the literature about expert finding, our baseline approach and improvements.

2.1 Expert Finding

"Expert finding" (or "Expert Search" or "Expert Retrieval") is a subtopic of "Information Retrieval" for finding appropriate experts in a given expertise area.

2.1.1 Expert Finding Systems

In general, expert finding systems can be divided into 2 parts according to their approaches, which are content-based and link-analysis based approaches.

2.1.1.1 Content-Based Approach

Content based expert finding systems do not take into account social network relationships between authors or experts. In this kind of approach, the experts are only related to documents and content analysis methods are important in this approach.

Balog, Azzopardi, & De Rijke [3] proposed two probabilistic models by directly creating relationships between experts, documents and query topics. In their first model, an expert's knowledge was brought out based on the documents that they are associated with. In their second model, the documents are associated with topics first and then experts are associated with those topics. They have tested those approaches using TREC Enterprise dataset and found that their second approach has outperformed the first approach which means creating reliable relationships between topics and documents can increase the performance of expert finding systems.

Macdonald & Ounis defined expert search as a voting process by adapting 11 data fusion techniques[20]. Documents related to an author are added as implicit votes to the ranks of documents that are returned from the system as a result of a query. Data fusion techniques are used for combining rankings using either scores or ranks of retrieved documents.

Author-Topic(AT) model was introduced by Rosen-Zvi, Griffiths, Steyvers, & Smyth [25] which is an extension to LDA[6]. According to this model authors and topics enter same dirichlet process and as a result each author is associated with multiple topics and each topic is associated with multiple words. ACT (Author-Conference-Topic) and ACTC (Author-Conference-Topic-Connection) models are two extensions of AT models which models papers, authors and conferences simultaneously [28]. The difference between ACTC and ACT model is, ACTC model has additional subject information of the conference latent variable.

Consequently, since content based approaches do not take into account social network relationships between authors, these approaches are appropriate if the system has access to sufficient number of documents to identify research areas of individual experts.

2.1.1.2 Link-Analysis Based Approach

In link-analysis based expert finding systems, authors connections in a social network are also considered as well as contents of documents. In this type of approach, using appropriate link-analysis methods are important. PageRank [22] and HITS [17] are two of the most commonly used link analysis algorithms that are used in information retrieval systems.

Zhang, Tang, & Li [29] used propagation-based approach for expert finding in social network systems instead of PageRank or HITS because both of them allow domination of the in-links in the network. Their experimental results suggest that instead of using only local information of an expert, social network information of a person such as co-authorship can increase the accuracy of expert finding systems.

Kardan, Omidvar, & Farahmandnia proposed PageRank based SNPageRank algorithm, in which instead of webpages the links between people in the social network are considered as hyper links [16]. Their study suggested that SNPageRank algorithm can be used to calculate expertise level of a person in a social network.

Smirnova & Balog have proposed a user-oriented expert finding system which considers hierarchy and geographical location of the user[26]. They have defined 3 networks "Organizational Network", "Geographical Network" and "Collaboration Network" and calculated the shortest path between users and experts in all 3 networks and calculated the scores of experts which also depend on the user of the system. This approach works well in local organizations because creating these 3 networks is possible in a local organization. On the other hand, this approach requires the information about the user of the system and it may not be suitable for general expert finding systems.

Consequently, since link-analysis based approaches take into account social network relationships between authors, these approaches are preferred if the system has access to social network relationships between experts.

2.1.2 Expert Finding Tools

Since "Expert Finding" is a real world problem, there are examples of expert finding tools (both commercial and noncommercial) in the real world. Some of these tools and their properties can be listed as follows:

- **LinkedIn:** LinkedIn[1] is a professional expertise based social network where profiles of experts are created by the users themselves. LinkedIn uses SNA and makes recommendations to users.
- **ArnetMiner:** ArnetMiner[2] is an academic search engine in academic social networks. The contents of ArnetMiner are gathered by integrating academic resources to extracted information from web. Besides, users can register and modify profiles of experts. It contains author and conference information of

[1] https://www.linkedin.com/
[2] http://arnetminer.org/

publications and contact and research information of people. It also visualizes the co-authorship and other links between people on the social network.

- **Microsoft Academic Search:** Microsoft Academic Search[3] contains information of academic publications along with conference proceedings and citation information to other publications.
- **DBLP:** DBLP[4] is hosted at University of Trier in Germany which contains academic papers and links to homepages of experts.

2.2 Topic Modeling

The idea of topic modeling depends on LSI (Latent Semantic Indexing) which was introduced by Dumais, Furnas, Landauer, Deerwester, & Harshman in 1988[10]. Before LSI, traditional methods employed word-based access between textual materials and user requests. LSI proposed a new technique which creates "word to text-object" semantic associations to overcome dependency on exact equality of words of queries and texts problem [10]. In 1999, Hofmann proposed PLSI (Probabilistic Latent Semantic Indexing) which is derived from LSI[14]. In addition to original LSI, PLSI depends on a latent class model based mixture decomposition instead of singular value decomposition in LSI. Figure 2.1 shows the process of PLSI where documents (d) are associated with unobserved class variables (z) and those variables are associated with vocabulary on the corpus (w).

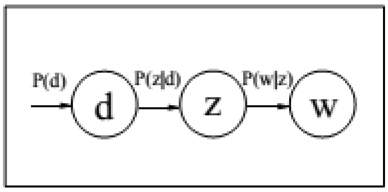

Figure 2.1: Probabilistic Latent Semantic Indexing (PLSI)[14]

In 2003, Blei, Ng, & Jordan introduced the well known topic modeling called "LDA (Latent Dirichlet Allocation)"[6]. In addition to PLSI, LDA adds a new Dirichlet parameter to each document topic distribution. This parameter is the α (alpha) parameter in LDA which stands for the number of topics per document. Similar to α

[3] http://academic.research.microsoft.com/
[4] http://dblp.uni-trier.de/

(alpha), the β (beta) parameter stands for the number of words per topic. In LDA, each document can be considered as a mixture of topics with different probabilities. Figure 2.2 shows the process of LDA where α and β parameters are added in addition to Figure 2.1.

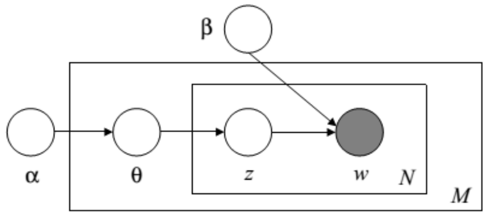

Figure 2.2: Latent Dirichlet Allocation (LDA)[6]

LDA is used by Griffiths & Steyvers to find topics in a scientific corpus[13]. Their study has shown the temporality of 3 hottest and coldest topics in 10 years by the evolution of generated topics from different years' documents.

Hierarchical topic modeling is introduced by Griffiths & Tenenbaum[12] as an extension to LDA[6]. In hierarchical topic modeling, topics are related to each other in a hierarchical manner. To the best of our knowledge, in the literature, current expert finding systems do not use hierarchical topic modeling techniques.

2.3 Weirdness

"Weirdness" is a measure for domain specialty of words which introduced by Ahmad et al. [1]. In other words, "Weirdness" refers to the proportion of distribution of a word in a specialized corpus to the distribution of the word in general corpus [18]. "Weirdness" is a well-known automatic term recognition method in the literature[18]. Ahmad, Tariq, Vrusias, & Handy also used "Weirdness" for creating corpus based dictionary[2]. To the best of our knowledge, in the literature, current expert finding systems do not have any example that uses weirdness value. Weirdness is also used as a coefficient of another well known term frequency method named "Glossex"[19].

2.4 Temporality

Temporality is a widely studied area of expert finding systems in the literature. Blei & Lafferty introduced DTM (Dynamic Topic Models) [5] which is an extension to LDA[6]. Dynamic LDA takes into account the order of the documents in addition to LDA. Daud, Li, Zhou, & Muhammad proposed Temporal-Expert-Topic (TET) approach [8] which is an again an extension of LDA by taking into account conference and year information of documents. Later, Daud also proposed Temporal-Author-Topic (TAT) approach to handle changes of researcher interests through time [7]. The output of the study is shown in Figure 2.3 where each LDA topic is related with at least 1 author and 1 year.

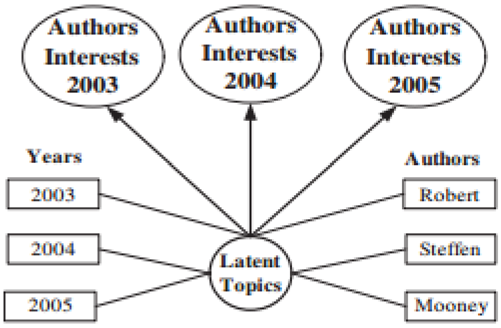

Figure 2.3: Dynamic Research Interest Finding [7]

2.5 ADT Graphs

Author-document-topic (ADT) graphs which uses topic modeling techniques were introduced by Gollapalli et al. [11]. In ADT graphs, authors are associated with documents and documents are associated with topics. The document-topic associations are generated through LDA [6]. They have tested two approaches: topic-based search and name-based search. Topic-based search is tested for finding experts of a given topic, while name-based-search is tested for finding similar experts of a given expert (author). For topic-based search, they have compared their ADT approach with Deng, King, & Lyu's [9] probabilistic model which is the extended model of Balog et al.[3] via two datasets: ArnetMiner and UVT. According to their experimental results, for ArnetMiner dataset, their proposed ADT approach performed best, however for UVT dataset, the baseline probabilistic model performed best. The details of Gollapalli et al.'s study were presented in Chapter 3 as we have chosen this study as baseline approach. The reasons that we have chosen this study as baseline approach can be listed as follows:

- It is a recent study and currently no extensions have been made on this study,
- The datasets that are used in this study (UvT and ArnetMiner), can also be

accessed currently,

- The results of the experiments were shown explicitly and numerically, therefore we can easily compare our results with this study.

2.6 Limitations of Current Studies

Some of the current studies in the literature related to expert finding use topic modeling techniques. Besides, some of these studies use dynamic topic modeling techniques and extensions to Dynamic LDA. However, to the best of our knowledge, current expert finding studies that use topic modeling do not make domain limitations on corpus before topic modeling. The variation of α and β parameters of LDA implementations do not seem to satisfy domain specificity requirement because α refers to number of topics per document and β refers to the number of words per topic. Consequently, we decided to integrate "Weirdness" value to our system for domain limitation.

Current temporal effects on expert finding systems are embedded inside of topic modeling, as a result, dynamic topic models are used. We decided to bring out a different temporality approach to expert finding systems which would be set between authors and documents apart from topic modeling. By using approach, we would be able to separate temporality from topic modeling.Therefore, even if one prefers to use different topic modeling techniques in the future, our temporality approach might still work with the new integrated topic modelling technique seamlessly.

In order to satisfy, both of these requirements, we have selected Gollapalli et al.'s study (ADT graphs)[11] as baseline approach which already uses LDA as topic modeling method. We have improved their ADT approach by introducing a new temporality aspect and a domain limited topic modeling.

CHAPTER 3

BASELINE APPROACH: RANKING EXPERTS USING AUTHOR-DOCUMENT-TOPIC GRAPHS

This chapter explains the baseline approach of Gollapalli et al. [11] which is used in our proposed system.

3.1 Overview of Author-Document-Topic Graphs

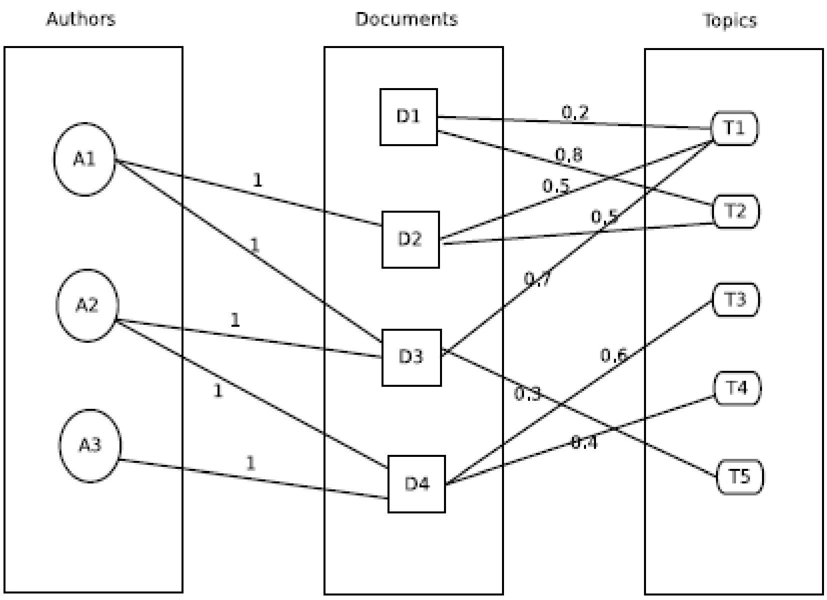

Figure 3.1: Baseline Author-Document-Topic Graph [11]

Author-document-topic graphs are tripartite graphs. Let $A = a_1, a_2...a_i$ represent the set of all authors, $D = d_1, d_2...d_j$ represent the set of all documents (i. e. publications) and $T = t_1, t_2...t_k$ represent the set of all topics. Then the tripartite graph will be $G = (V, E)$ where $V = A \cup D \cup T$ and E is the set of all edges between author-document pairs and document-topic pairs. The edges between authors and documents represent the author of document relationship. The edges between topics and documents represent the associations between topics and documents.

Before creating ADT graphs, document are indexed which is explained in Section 3.2.2. ADT graphs are used to create relationships between authors, documents and topics by assigning weights to edges which is explained in Section 3.2.3. After ADT graphs are created, three different ADT graph based node similarity calculation methods are used to calculate the scores of authors for topic queries. The details of these methods are explained in Section 3.2.4.

3.2 Experimental Settings

3.2.1 Datasets

ArnetMiner / CiteSeer and UVT Collection datasets are used for the experiments in the baseline approach. The details of datasets are shown in Table 3.1. In the table, corpus size refers to the number of documents in the dataset. Queries refer to the number of topic queries in the dataset. QRels refer to the number of manually identified experts related to the given topic queries in the dataset. Consequently, both datasets contain topic queries and list of experts related to those queries that can be used as ground truth in the experiments.

Table 3.1: Summary of Datasets Used in Baseline Approach [11]

Name	**Description**	**Corpus Size**	**Total Authors**	**Queries**	**QRels Size**
AM	ArnetMiner / CiteSeer	103838	27108	13	901
UVT	The Uvt Collection	19127	1168	203	17511

3.2.2 Topic Query-Document Indexing

In order to create topic query - document indices, the documents of both datasets are indexed using Indri[1] search engine. Indri is a local search engine that uses language modeling techniques for ranking given documents in response to given input queries. The maximum number of documents to retrieve from Indri search engine is set 100. According to the created indices, ADT graphs are created by starting from the documents that are retrieved from Indri, afterwards.

3.2.3 Graph Generation

Tripartite author-document-topic graphs are created after documents are retrieved from topic query-document indices. The creation of ADT graphs starts from documents. The weights of the edges between the authors and the documents are set to 1. In order to create the topic nodes and the weights of edges between documents and topics Latent Dirichlet Allocation (LDA) is used. The document topic proportions that are obtained from LDA are set as the weights of edges between documents and nodes. Mallet[21]'s LDA implementation was used in experiments by default values for alpha = 50.0, beta = 0.01 and threshold = 0.001.

3.2.4 Node Similarity Calculation Methods

Three schemes are studied for calculating similarity between nodes which are MaxPath, SumPaths and ProductPaths. MaxPath is the relationship between two nodes with the maximum of sum score of all paths between these nodes. SumPaths is the relationship between two nodes with the aggregation of sum score of all paths between these nodes. ProductPaths is the relationship between two nodes with the aggregation of multiplication score of all paths between these nodes. The mathematical definitions of these 3 ADT methods are as follows:

If we define p as one of the paths between nodes a and d that consists of one or more

[1] http://www.lemurproject.org/indri/

edges: $p = e_1 e_2 ... e_n$. Then, we can define *sweight* and *pweight* of path p as follows:

$$\textit{sweight(p)} = \sum_i weight(e_i) \tag{3.1}$$

$$\textit{pweight(p)} = \prod_i weight(e_i) \tag{3.2}$$

After that, according to defined *sweight* and *pweight* values; MaxPath, SumPaths and ProductPaths methods are defined as follows:

$$\textit{MaxPath(a, d)} = \max_{p \epsilon P(a,d)} sweight(p) \tag{3.3}$$

$$\textit{SumPaths(a, d)} = \sum_{p \epsilon P(a,d)} sweight(p) \tag{3.4}$$

$$\textit{ProductPaths(a, d)} = \sum_{p \epsilon P(a,d)} pweight(p) \tag{3.5}$$

3.3 Experiment Results

There were two types of experiments which are topic-based search and name-based search. Since our methodology is topic based, we are only concerned with topic-based search results. Experiments were run with different number of topics (50, 100, 150, 200, 250, 300, 350, 400, 450, 500, 550, 600, 650, 700) and number of results (10, 20, 30, 40, 50) parameters. Average Recall, Average Precision, MAP, MRR are calculated as the output of the experiments according to those equations:

If we define R_q as the set of foreknown experts for a given topic query, q. Then, if *S* refer to the set of recommendations created by the expert finding system for *q*, we can compute recall and precision values for *q* as follows:

$$\textit{Recall} = \frac{|S \cap R_q|}{|R_q|} \tag{3.6}$$

$$Precision = \frac{|S \cap R_q|}{|S|} \tag{3.7}$$

Average Precision (AvgP) value refers to the average precision with *S* after each relevant expert is retrieved from the system. MAP (mean average precision) value aggregates the average precision value over all of the topic queries (*Q*) in order to create a single measure for the precision as follows:

$$MAP = \frac{\sum_{q=1}^{Q} AvgP(q)}{|Q|} \tag{3.8}$$

MRR (mean reciprocal rank) value returns the rank at which the first correct expert is found for every topic query in *Q* as follows:

$$MRR = \frac{1}{|Q|} \sum_{q=1}^{Q} \frac{1}{rank(q)} \tag{3.9}$$

In Equation 3.9, *rank(q)* refers to the rank of the first relevant expert found for the topic query, *q*. When no relevant experts are found for a topic query, then *rank(q)* returns 0.[11].

Figure 3.2 and Figure 3.3 show the performance of the baseline approach according to number of topics alongside of Figure 3.4 which shows the performance according to ADT methods.

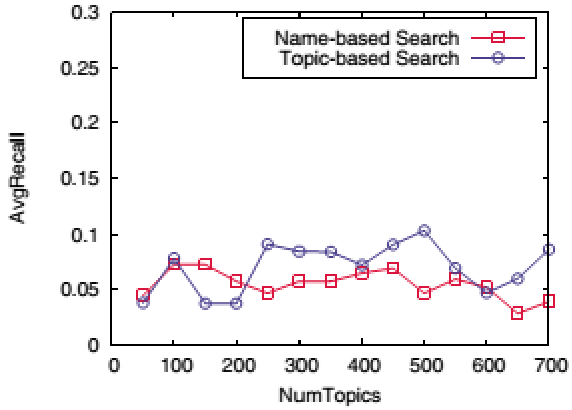

Figure 3.2: Baseline Performance with Number of Topics (ArnetMiner) [11]

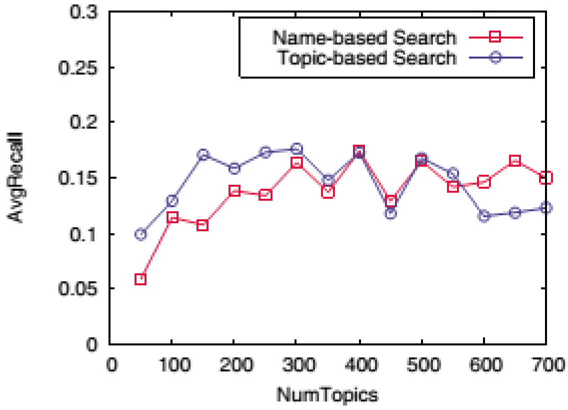

Figure 3.3: Baseline Performance with Number of Topics (UVT) [11]

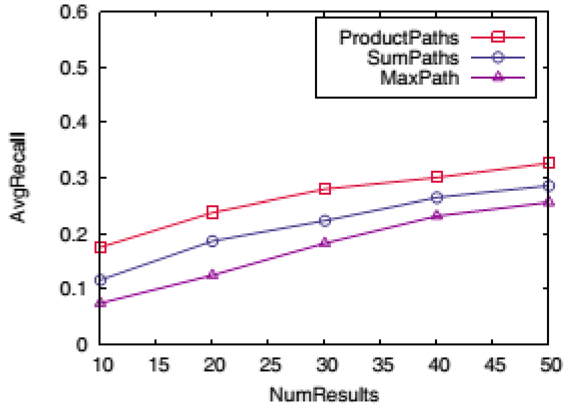

Figure 3.4: Baseline ADT Methods Performance (UVT) [11]

Table 3.2 and Table 3.3 show the sample topics from the ArnetMiner and UVT datasets respectively. By investigating these two tables, we can see that ArnetMiner is related to computer science while UVT has a broader domain. For example, in Table 3.3, topic 90 is related to computer science while topic 87 is related to economy and topic 86 is related to philosophy, all together demonstrate that UVT has a broader domain than ArnetMiner.

Table 3.2: Baseline Sample Topics (ArnetMiner) [11]

Topic No.	**Most Used Words**
305	distribution probability random distributions show number size model independent expected uniform rate average
409	knowledge learning domain reasoning system case problem acquisition machine task expert solving base process learn
448	model models bayesian probability gaussian mixture distribution estimation likelihood maximum parameters probabilistic
302	management distributed applications system systems service application support requirements dynamic services computing
414	query queries database data databases relational processing optimization evaluation join sql efficient execution support
66	mobile devices computing wireless location users device environment user access environments services network ubiquitous
408	computational complexity based algorithm paper proposed efficient algorithms cost techniques efficiency advantage reduced
109	learning training classification data supervised labeled set approach labels learn examples class task unlabeled unsupervised
439	mining data discovery patterns association rules knowledge databases database rule frequent discover large discovering
342	semantic ontology web ontologies knowledge abstract domain semantics concepts rdf language describe resources metadata

Table 3.3: Baseline Sample Topics (UvT) [11]

Topic No.	**Most Used Words**
99	estimation statistics probability regression model statistical distribution estimators methods multivariate variables
98	lines prior summary top half reflects implication patterns trends greater numbers wide variety continued portion
90	index cluster clusters space target ranking coming multi group collected clustering included mixed retrieved entry
89	mind important made sense relation make arguments common consists remarks full case interpretation existence view
87	markets industrial journal firms organization competition economics collusion oligopoly consistency letters market
86	ethics law moral ethical social legal morality politics human society theory state philosophy ideals political care
83	republic europe poland czech hungary eastern state german west east central russia french case government ten
78	asia regions areas india africa rural agricultural urban historical america agriculture spread southern cities
72	criminal crime law justice police european investigation court victims prosecution enforcement victim crimes drug
71	face brain related expressions facial cognitive emotion affective emotional expression neuroscience emotions perception

Table 3.4: Baseline Performance Evaluation [11]

ArnetMiner			
	BL(Prob)	PR	ADT
Prec@10	0.3300	0.3400	**0.4300**
MRR@10	0.5009	0.6350	**0.8433**
MAP@10	0.1844	0.2097	**0.3397**
Prec@50	0.1980	0.1680	**0.2900**
MRR@50	0.5009	0.6350	**0.8433**
MAP@50	0.0987	0.0851	**0.1986**

UvT			
	BL(Prob)	PR	ADT
Prec@10	**0.2158**	0.1856	0.1088
MRR@10	**0.5145**	0.4304	0.3021
MAP@10	**0.1506**	0.1245	0.0759
Prec@50	0.1245	**0.1246**	0.0598
MRR@50	**0.5201**	0.4393	0.3167
MAP@50	**0.1793**	0.1558	0.0943

Finally, Table 3.4 shows the general performance of ADT graphs with respect to probabilistic (BL(Prob)) and PageRank (PR) based methods. In this table "ADT" column shows the output of author-document-graph based model's results in which the ADT method is ProductPaths for both datasets. Besides, the number of topics is set to 500 for ArnetMiner and 400 for UvT. In Chapter 5, the results of our experiments will be demonstrated and our proposed system's results will be compared with these results.

CHAPTER 4

A TEMPORAL EXPERT FINDING METHODOLOGY BASED ON DOMAIN LIMITED TOPIC MODELING

This chapter explains the details of our proposed system by showing improvements and innovations on the baseline approach. Besides, the datasets used in our experiments, the experiment parameters, the experiment setup and the progress are demonstrated.

4.1 Overview of The Proposed System

In Chapter 3, we have explained the details of the baseline author-document-topic graph approach proposed by Gollapalli et al.[11]. In Figure 3.1, only authors, documents and topics of the baseline approach are shown. In the baseline approach, topic queries are off the author-document-topic graphs. Gollapalli et al. explain that they have used Indri to index documents according to the topic queries. According to Gollapalli et al., the creation of ADT graphs starts from documents which are retrieved from Indri in response to a topic query according to the indices created previously[11]. Figure 4.1 shows the baseline approach in a wide perspective in which the topic queries and their links to the documents are shown. We call this approach presented in that study as "Separated ADT" approach because contents of topic queries are not included in the author-document-topic graphs and LDA is implemented only on the documents of the datasets.

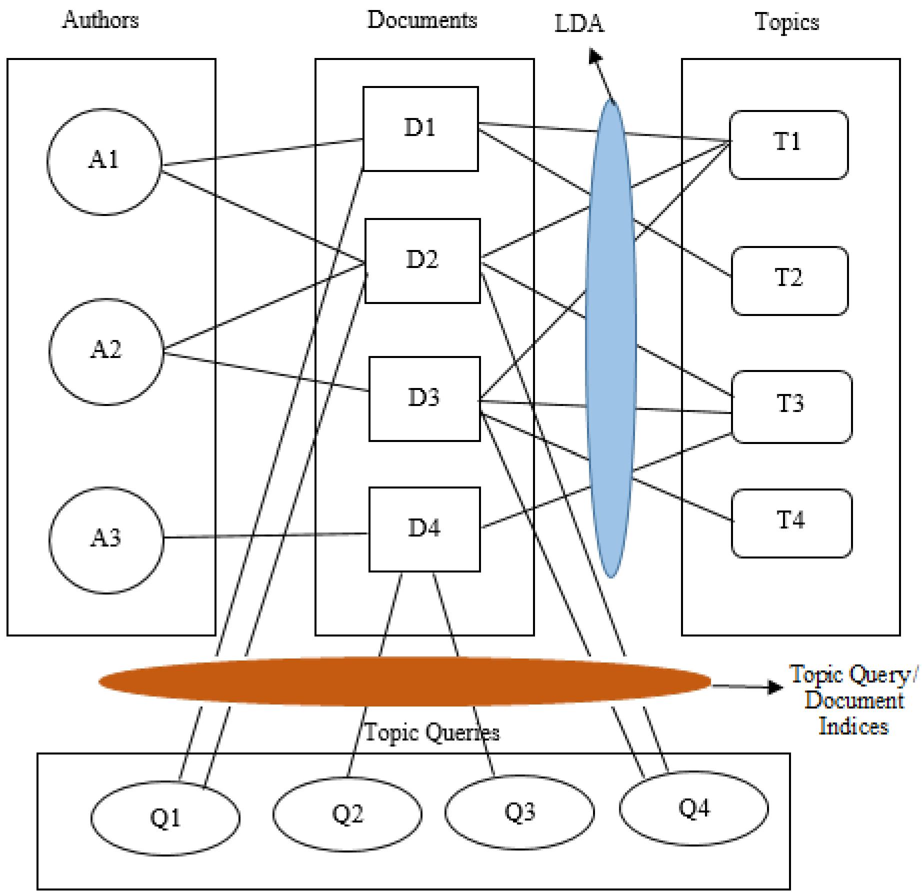

Figure 4.1: Detailed View of Baseline Approach

In our study, our primary goal is to eliminate topic query - document indices because this operation separates document-topic query relations from document - LDA topic relationships as it is shown in Figure 4.1. In order to eliminate topic query - document indices, we considered integrating topic queries and documents in LDA process by relating them through LDA topics. As a result, instead of our baseline "Separated ADT" approach, we have proposed "United ADT" approach in which topic queries are appended to the original documents during the implementation of LDA and finally, topic queries are linked to the topics instead of documents. Figure 4.2 shows our proposed united approach in folded view and Figure 4.3 shows it in unfolded view.

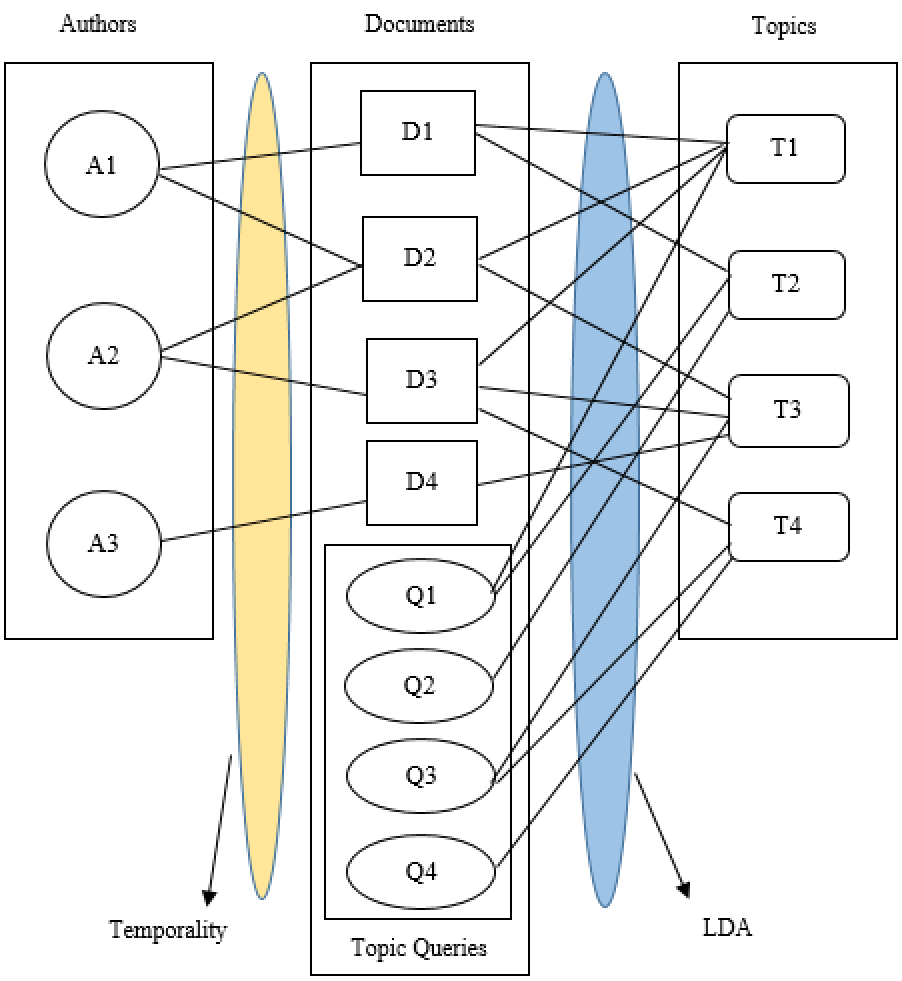

Figure 4.2: Folded View of Proposed System

In our proposed system, we calculate the similarities between nodes $a \epsilon A = a_1, a_2 \ldots a_i$ and $q \epsilon Q = q_1, q_2 \ldots q_j$ for three node similarity methods (SumPaths, ProductPaths, MaxPath) that are explained in Section 3.2.4. For a given topic query (q), we rank the author nodes (a) according to the scores of 3 methods and calculate the average recall, average precision, MAP and MRR values (Section 3.3) for different types of parameters that are explained in Section 4.4.

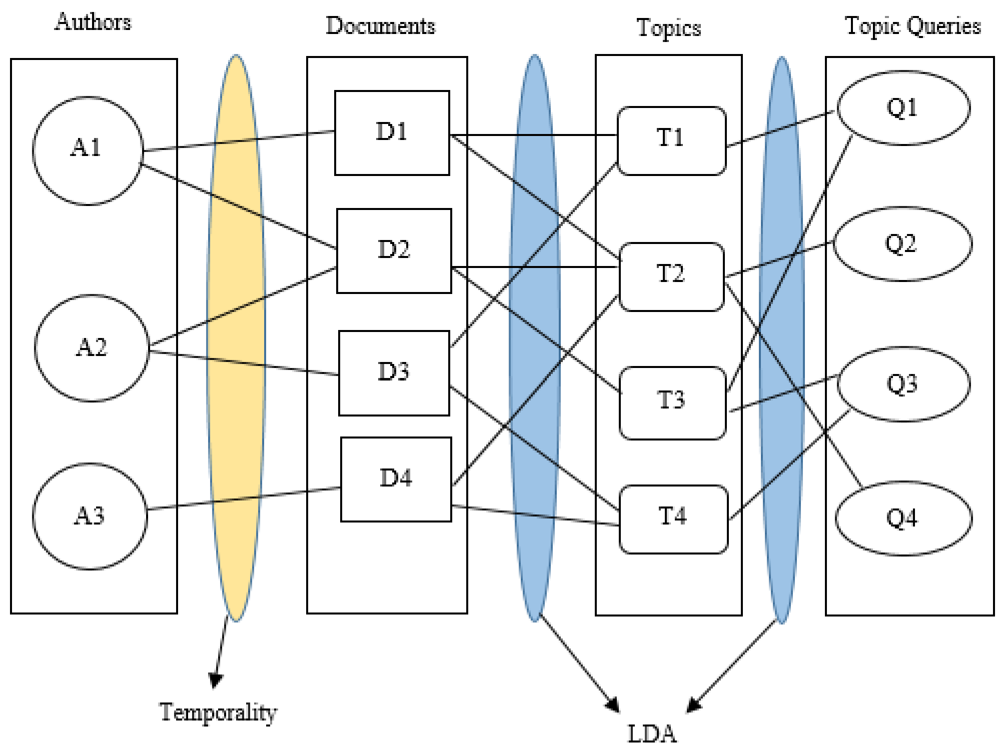

Figure 4.3: Unfolded View of Proposed System

4.1.1 Topic-Document Relationship

In United ADT approach, link strengths between documents and topics are assigned according to the LDA proportions. Similarly, values of links between topics and topic queries are assigned according to the same LDA implementation proportions as it is shown in Figure 4.2 (folded view). Differences between proposed system and baseline approach in terms of topic-document relationship are explained in Section 4.2 and Section 4.4.

4.1.2 Author-Document Relationship

In terms of author-document relationship, "United ADT" and "Separated ADT" approaches do not have any differences. In our proposed system, temporality effect is added to the link strengths between authors and documents as it can be seen in Figure 4.3). The details of temporality effect are explained in Section 4.2.4.

4.2 Contributions of Proposed System

4.2.1 United ADT Approach

We call our baseline approach as "Separated ADT" approach, in which topic queries are not directly related to the generated topics from LDA (Figure 4.1). In this approach, the relationship between documents and topic queries is provided using topic query - document indices. Our proposed system is based on a "United ADT" approach, in which documents and topic queries are both processed in same LDA process and as result, topic query - document indices are eliminated and topic queries are directly linked to the topics. By eliminating creation of topic query - document indices, we will also be able to improve the performance of our system in terms of speed. These two approaches are used as a parameter in our proposed system which is explained in Section 4.4.

4.2.2 Stemming

Porter's stemming algorithm[23] is used for finding root form of the words in documents and topic queries. In baseline approach, stemming was not used as we can see in sample topics in Table 3.2 and Table 3.3. For example, words "query", "queries", "database", "databases" are seen as sample words of topic 414 in Table 3.2 which demonstrate that stemming was not done before LDA. In our proposed system, stemming is done before running LDA on documents and topic queries. Stemming is used as a parameter for the proposed system which is explained in Section 4.4. When stemming is selected, documents and topic queries are first tokenized and stemmed, after that LDA is run on stemmed words. Table 4.1 and Table 4.2 show the effect of stemming on words after topic modeling. In these tables we printed the most used 20 words for each topic in the order of frequency of the word. Topics of Table 4.1 are created without using stemming and topics of Table 4.2 are created after using stemming. If we analyze the topics in these tables, we can see that in topic 4 of Table 4.1 there are words "query" and "queries" which derive from the same stem but are considered as different words in the same topic. If we look at topic 0 of Table 4.2, we can see the single word "queri" which is the stem of both "query" and "queries". These examples show that usage of stemming provides us to consider different words that are derived from the same stem as equivalent.

Table 4.1: ArnetMiner 10 topics without stemming

Topic No.	**Most Used Words**
0	data problem paper analysis techniques case number space small patterns classification text feature datasets key information mining process methods present
1	approach data models model based show features algorithm learning present systems plans applications paper language work set knowledge networks cost
2	web information query service semantic services agents provide based users queries agent show system paper source content proposed distributed execution
3	data mining semantic information based systems system sources web large language paper model task clustering important algorithms text approach issue
4	based algorithm database paper random query queries algorithms presents set system privacy systems user databases results length distributed peer text
5	based paper recognition image protocols system approach mobile results present number performance developed problem research secure knowledge processing images face
6	algorithms model function agents based paper results network efficient method optimal management linear group real computational distribution result data reasoning
7	learning based results accuracy methods algorithm show algorithms structure model time data machine performance problem method planning examples error retrieval
8	learning object data knowledge motion algorithm approach method images system detection view model probabilistic image objects results camera training scale

9	data learning problem domain knowledge search rules paper based problems system ontology systems task logic order design real belief models

Table 4.2: ArnetMiner 10 topics with stemming

Topic No.	**Most Used Words**
0	queri learn rule evalu method model result problem base process label system support paper data algorithm sentenc compon retriev function
1	web semant model extract map program base paper text algorithm knowledg databas inform system content time specif learn task data
2	paper system techniqu gener method set base propos approach inform data knowledg describ languag model effici design decis bayesian approxim
3	data perform algorithm base model user answer problem system show result approach dynam parallel question method present product inform distribut
4	model system analysi comput imag featur base motion inform propos algorithm match recognit present method segment multipl environ network face
5	system approach integr inform agent user ontolog activ provid queri develop gener object present data monitor domain tool logic locat
6	data cluster process mine algorithm framework base method knowledg paper perform pattern larg chang network domain relat model gener belief
7	algorithm learn base show model compar optim approach perform problem plan result structur sampl classif object class tree improv present
8	protocol problem model scheme approach data base present user servic system knowledg secur set techniqu document result attack game bound

9	agent comput plan problem search action privaci space state cost domain secur heurist solv propos polici optim learn paper data

4.2.3 Domain Specialty (Weirdness)

Our proposed system enables limiting the number of words that will be processed in LDA. In order to determine this threshold value, "Weirdness"[1] is used. Weirdness score of a word is calculated as:

$$Weirdness = \frac{\frac{w_s}{t_s}}{\frac{w_g}{t_g}} \tag{4.1}$$

where:
w_s = frequency of word in specialist language corpus
w_g = frequency of word in general language corpus
t_s = total count of words in specialist language corpus
t_g = total count of words in general language corpus

Weirdness can be thought as the domain specificity score of a word in a corpus. A more domain specialized word's weirdness value should be greater than a general word's weirdness value according to Equation 4.1. In our proposed system specialist language corpus is the documents of our datasets. For calculating general language corpus frequency of words, we have used Open American National Corpus[24]. We have used written corpus of Open ANC which has 11,406,155 words. In our experiments, *weirdness threshold* is used as a parameter of our proposed system. Firstly, words are ordered according to the weirdness score in descending order and then threshold value, between 0 and 1, limits the number of words that will be used in LDA according to the order of weirdness score. *Weirdness threshold* limits the size of the list of the words according to the equation:

$$\textit{New Corpus Size} = Original\ Corpus\ Size\ \times\ Weirdness\ Threshold \qquad (4.2)$$

For example, let's say we have a corpus of size 1000 words. After ordering these words according to their weirdness scores in descending order, if we set *weirdness threshold* value to 0.8, then it means that only first $1000 \times 0.8 = 800$ words in the corpus will be considered during the topic modeling. The remaining 200 words with smaller weirdness scores will be eliminated and corpus size will be reduced. In order to remove the weirdness effect, *weirdness threshold* value can be set to 1. Table 4.3 and Table 4.4 show the effect of using *weirdness threshold* before topic modeling. Table 4.3 shows 10 topics without *weirdness threshold* (i.e. weirdness threshold = 1.0), and Table 4.4 shows 10 topics with *weirdness threshold* = 0.8. We did not use stemming in both of the examples in order to show the words clearly. In these tables, we printed the most used 20 words for each topic in the order of frequency of the word. Looking into those topics in detail, we can see that topic#8 in Table 4.4 is equivalent to topic#0 in Table 4.3 because both of the topics contain the same words like "learning", "classification", "method", "methods". For example, topic#0 in Table 4.3 contains word "based" however topic#8 in Table 4.4 does not contain word "based" because of weirdness effect. Similarly, word "based" exists in five topics of Table 4.3 however none of the topics in Table 4.4 contain word "based". As we can see in this example, we can eliminate general words before topic modeling using *weirdness threshold*. Additionally, by shrinking corpus size, we will also be able to improve the performance of our system in terms of speed.

Table 4.3: ArnetMiner 10 topics without weirdness threshold

Topic No.	**Most Used Words**
0	learning classification method methods text machine training model features task models results show based feature tasks paper selection bayesian experiments

1	knowledge semantic language based domain support rules evaluation services system ontology paper natural rule processing process discovery source software semantics
2	data algorithm algorithms mining techniques performance results sets paper quality space privacy efficient technique experiments location structure experimental large methods
3	time based model real approach paper problem object objects multiple framework representation propose detection world temporal constraints techniques tracking important
4	problem based security provide terms present show search solving constraint document standard optimization existing general simple solutions protocols goal order
5	information web query user search users system queries patterns database sources describe databases present content retrieval access pattern management service
6	number case complexity show results graph random set function properties linear functions probabilistic error problems distribution study size adaptive independent
7	agents agent planning work approaches cost problems domains state plan decision reasoning human framework control models environment theory research multi
8	systems network distributed networks system based large applications key analysis scheme dynamic clustering scale neural effective computing provide approach architecture
9	image recognition method proposed images motion parallel local presented video models matching parameters view automatic sequences high visual question points

Table 4.4: ArnetMiner 10 topics with Weirdness Threshold = 0.8

Topic No.	**Most Used Words**
0	problem algorithm number problems case complexity show decision algorithms computational function order neural functions error vector trees results solving result
1	algorithms framework analysis techniques decision paper study cost clustering privacy provide location effective temporal quality show technique communication datasets values
2	agents distributed agent multi complex systems optimization environment services optimal time execution dynamic application human constraint constraints address software building
3	data mining results support rules patterns evaluation systems applications level paper process experimental rule knowledge statistical discovery techniques research pattern
4	knowledge domain search planning state time systems design approach approaches plan parallel reasoning problem language control plans current specific modeling
5	image results recognition features images objects space random algorithm motion input time presented set parameters proposed sequences size low visual
6	object key multiple security graph efficient video scheme matching properties protocol paper point robust estimation secure form segmentation line key
7	model system performance query approach network paper networks user service models feature describe users tracking features adaptive mobile propose computer
8	learning method methods models classification machine algorithm training accuracy experiments task paper class set sets performance examples selection structure bayesian

9	information web query semantic text queries database sources extraction ontology logic automatically databases search retrieval approach users processing user language

4.2.4 Temporality

Temporality effect is added between the documents and authors in our proposed system. To achieve it, we are required to have publication year for all documents that are related to an author. The publications in both of our datasets contain that information. However, the homepage and the research area documents in UVT dataset do not have publication year. We set year property to the maximum publication year of the known documents for these documents.

Temporality effect is used as a parameter in our system with 6 possible values (None, Logarithmic, Linear, Quadratic, Cubic, and Quartic). If parameter "None" is selected, the temporality effect is not applied to the graph which is the same situation in our baseline approach. In our baseline approach temporality effect was not included by assigning "a uniform weight of 1 to all edges between author and document nodes"[11].

Values "Logarithmic", "Linear", "Quadratic", "Cubic", and "Quartic" refer to the effect of temporality with the growth of year according to these values. Without renormalization, these values affect the link strengths of documents and authors as follows:

- **None:** $\mathrm{LS}(d, a) = 1$,
- **Logarithmic:** $\mathrm{LS}(d, a) = \log(y)$,
- **Linear:** $\mathrm{LS}(d, a) = y$,
- **Quadratic:** $\mathrm{LS}(d, a) = y^2$,
- **Cubic:** $\mathrm{LS}(d, a) = y^3$,
- **Quartic:** $\mathrm{LS}(d, a) = y^4$.

where $\mathrm{LS}(d, a)$ refers to the link strength between a document d and an author a meanwhile y refers to the year of the document d.

For example an author who is related to 4 documents, say d1, d2, d3 and d4. The years of the documents are: d1 = 1998, d2 = 2000, d3 = 2002 and d4 = 2004. Without any normalization, the effect of the temporality to each document is visualized in Figure 4.4, Figure 4.5 and Figure 4.6.

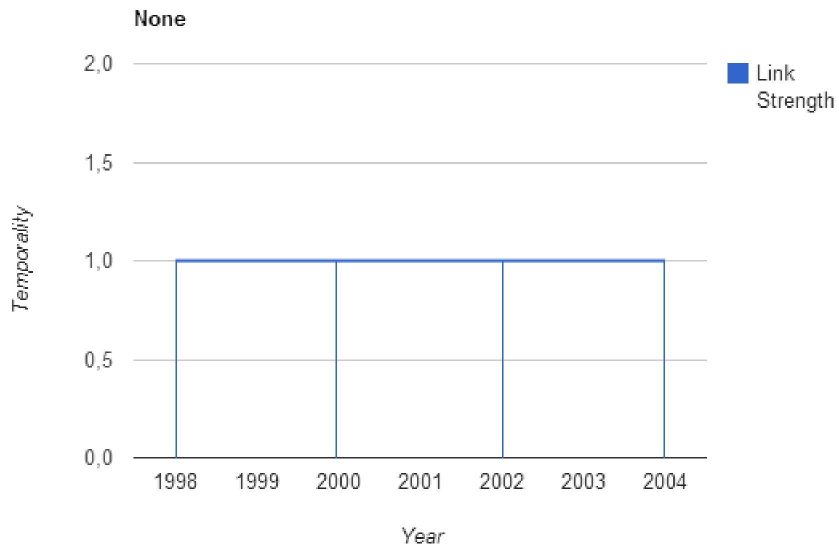

Figure 4.4: Temporality Effect = None

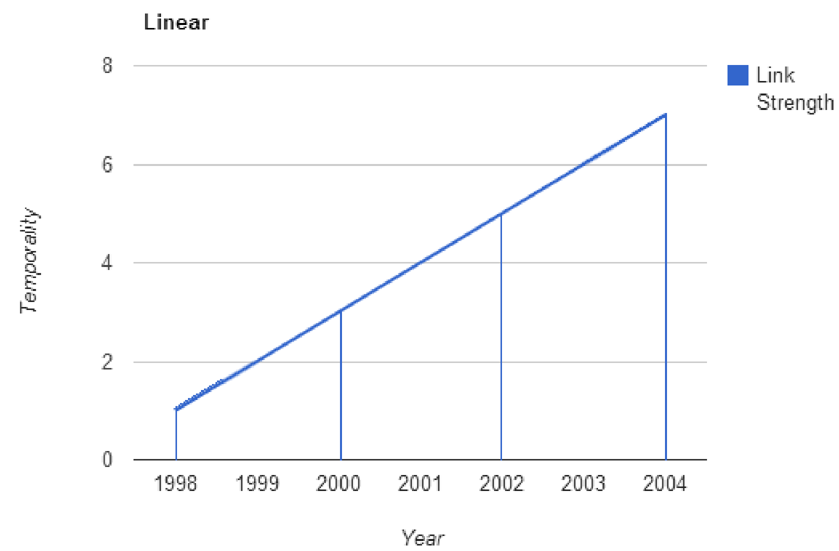

Figure 4.5: Temporality Effect = Linear

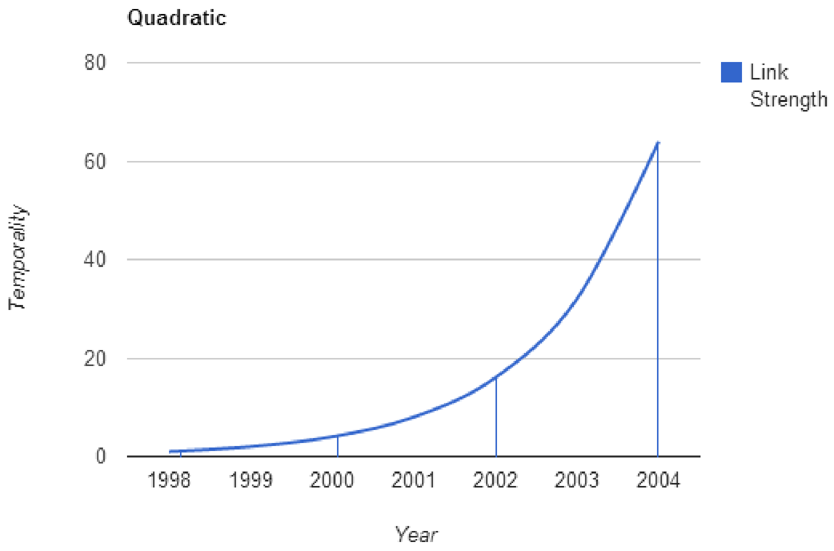

Figure 4.6: Temporality Effect = Quadratic

In order to fix the total amount of link strength between authors and documents in the ADT graphs, we renormalized the link values after applying temporality effect between documents and authors. Consequently, the area under the plots in Figure 4.4, Figure 4.5 and Figure 4.6 will be fixed and equal to the number of documents that the author is related with for each author in the dataset.

For each of the author / document pairs, temporality effects are calculated at first before the experiments once and saved for each parameter. Consequently, the link strengths in ADT graphs vary according to these temporality values. The algorithm of calculating temporality effects is shown in Algorithm 1.

After that, there will be 6 links between each document and author pairs in ADT graphs as it is shown in Figure 4.7.

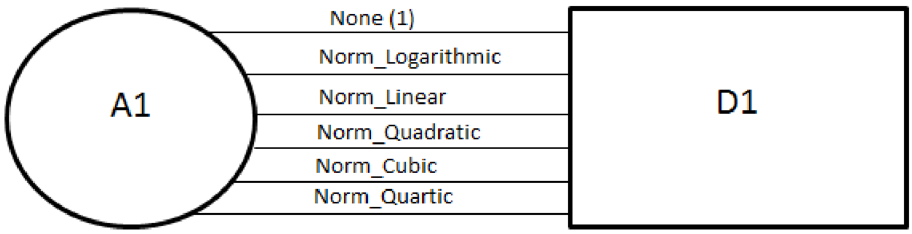

Figure 4.7: Temporal Link Strengths in ADT Graphs

4.3 Other Parameters Tested

In addition to four main contributions of our proposed system, we have also tested our system through six LDA parameters.

4.3.1 Topic Count

Topic count or *number of topics* is an input parameter of LDA. In the baseline approach, the number of topics are tested from 50 to 700 with an increment of 50[11]. In order to observe the optimum interval for number of topics, we created a log likelihood graphic using RStudio.

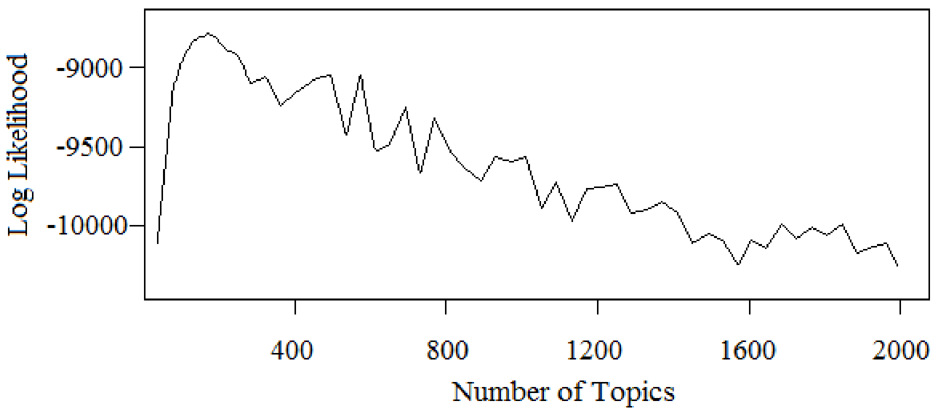

Figure 4.8: Number of Topics / Log Likelihood Chart for UVT

Figure 4.8 shows the log likelihood chart for UVT dataset for number of topics between 10 and 2000. 90% of the documents are used as training set and the remaining documents are used as test set. As it can be seen in the figure, the optimum number of topics is around 200 for UVT dataset. Since 200 is between 50 and 700, we decided to test the same number of topics as in the baseline approach as it is shown in

Table 4.5.

4.3.2 LDA Alpha

"α" (alpha) is an input parameter of LDA. It refers to the number topics per document (i.e. higher α means more number of topics per document). Mallet[21]'s LDA implementation enables us to tune α parameter. In our baseline approach, Mallet's default α value is used which is 50.0. The α values that we have tested in our proposed system are shown in Table 4.5.

4.3.3 LDA Beta

"β" (beta) is an input parameter of LDA. It refers to the number of words per topic (i. e. higher β means more number of words per topic). Mallet[21]'s LDA implementation enables us to tune β parameter. In our baseline approach, Mallet's default β value is used which is 0.01. The β values that we have tested in our proposed system are shown in Table 4.5.

4.3.4 LDA Threshold

LDA Threshold value is a parameter of Mallet[21]'s LDA implementation which refers to the elimination of document topic relations whose proportion is less than the *LDA Threshold* and renormalizing the sum of remaining topic proportions to 1.0. In our baseline approach *LDA Threshold* value was set to 0.001. The *LDA Threshold* values that we have tested in our proposed system are shown in Table 4.5.

4.3.5 LDA Iteration Count

Number of Iterations or *LDA Iteration Count* value is a parameter of Mallet[21]'s LDA implementation which refers to the number of sampling iterations while generating topics. Generally, there is a trade off between the time spent for sampling and the quality of topic model. In our baseline approach *LDA Iteration Count* value is set to 100. The *LDA Iteration Count* values that we have tested in our proposed system are shown in Table 4.5.

4.3.6 LDA Type (Hierarchical LDA)

Our baseline approach uses default LDA implementation, we called it as "Normal LDA", in which there are no direct links between generated topics as it can be seen in Figure 3.1 and in Figure 4.1. In our experiments we have tested "Hierarchical LDA" which enables creating hierarchical links between generated topics. In our experiments, we have used Mallet[21]'s Hierarchical LDA implementation. "Hierarchical LDA" implementation has α and β parameters similar to "Normal LDA" implementation. Additionally, hierarchical process has *Number of Levels* parameter which refers to the depth of leaf topics from the root topic. In addition to these parameters, our proposed system has *Level Coefficient* which refers to coefficient that will be multiplied by the level of LDA topic which relates document and topic query by the shortest path. At this point, we should note that "Hierarchical LDA" is only applied to our "United Approach" because in "Separated ADT" relates topic queries and documents wide apart from LDA topics. An example of an ADT graph that is created after using "Hierarchical LDA" is shown in Figure 4.9. In this example, *Number of Levels* parameter is 2.

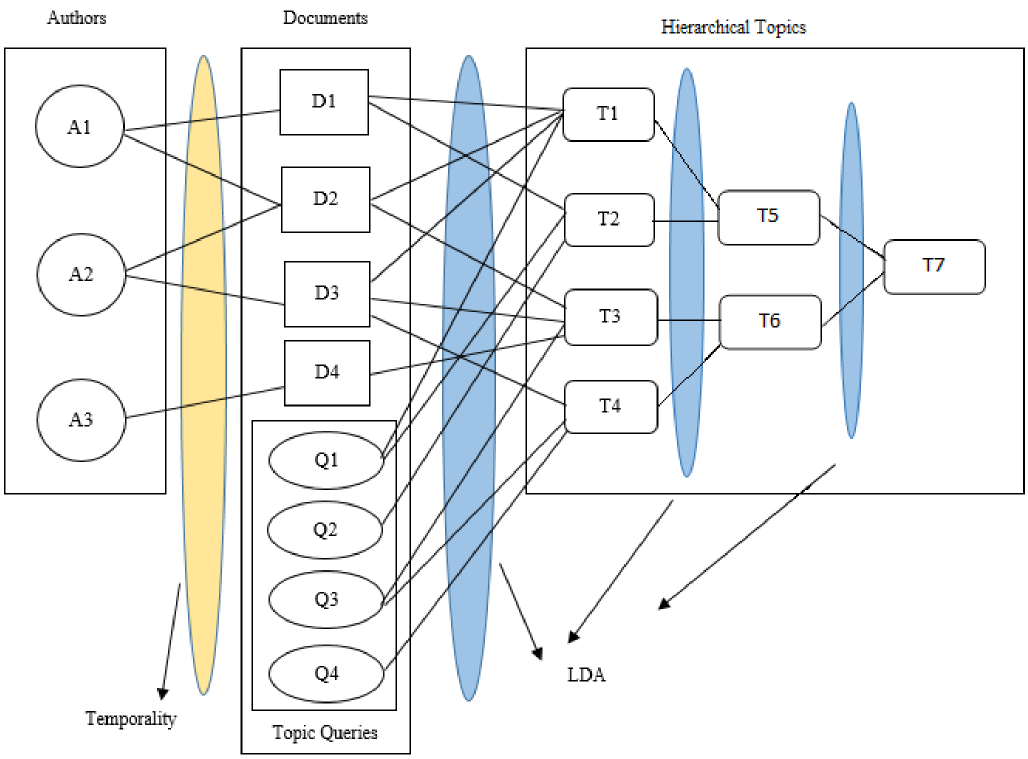

Figure 4.9: United ADT Graph Example using Hierarchical LDA

4.4 Parameter Settings

Table 4.5: Parameters of Proposed System

Parameter Name	**Baseline Approach**	**Proposed System**
ADT Type	Separated	Separated, United
Weirdness Threshold	None (1.0)	{0.60 ... 1.0}[1]
Stemming	Not Stemmed	Stemmed, Not Stemmed
Temporality	None	None, Logarithmic, Linear, Quadratic, Cubic, Quartic
Method	SumPaths, ProductPaths, MaxPath	SumPaths, ProductPaths, MaxPath
Topic Count	{50 ... 700}	{50 ... 700}[2]
LDA Alpha	50.0	{5.0 ... 100.0}[3]
LDA Beta	0.01	{0.005 ... 0.050}[4]
LDA Threshold	0.001	{0.0005 ... 0.0050}[5]
LDA Iteration Count	100	{20 ... 200}[6]
LDA Type	Normal	Normal, Hierarchical
Number of Results	{10, 20, 30, 40, 50}	{10, 20, 30, 40, 50}

4.5 Datasets

We have tested our proposed system via 2 datasets which are UVT[4] and ArnetMiner[27] which were also used in Gollapalli et al.'s study[11]. The 2 datasets, UVT and ArnetMiner, have some similar characteristics that can be listed as follows:

- All documents of both datasets are related to at least one author,
- Both datasets contain topic queries and related authors to these queries (qrels),
- Both datasets contain year information of publications.

[1] with an increment of 0.1
[2] with an increment of 50
[3] with an increment of 5.0
[4] with an increment of 0.005
[5] with an increment of 0.0005
[6] with an increment of 20

The datasets are compared to each other in Table 4.6.

Table 4.6: Properties of Datasets

Property	**UVT**	**ArnetMiner**
Domain	General (Table O.1)	Specific (Computer Science) (Table P.1)
Corpus Size	14,702 documents	155,418 documents
Total Authors	1,168	25,854
Document per Author	12.58	6.01
Queries	203(Table O.1)	13(Table P.1)
Qrels Size	1,751	920
Vocabulary Size	165,979	12,026,201
Word per Document	11.28	77.38
Document Types	Publication, Research Area, Course Homepage, Personal Homepage	Publication
Time Info	Year info only in Publication typed documents	Year info in all documents
Year Range	1975 - 2006	1967 – 2006

UVT dataset is constant and the data did not change since 2007 meanwhile ArnetMiner is growing over time. In order to create similar conditions to Gollapalli et al.'s study[11], we have selected UVT dataset as the primary dataset of our experiments.

UVT dataset can be obtained by downloading the "uvt-expert-collection-v1.4.tgz" file from UVT's website[7]. The dataset contains both documents and qrels (Table O.1) in the .tgz file. We have inserted all those information into our database tables in order to use in our experiments. Document counts per year for UVT dataset are shown in Figure 4.10.

[7] http://ilk.uvt.nl/uvt-expert-collection/

Obtaining ArnetMiner dataset was difficult than UVT dataset. We have crawled ArnetMiner's website[8] and downloaded 2.327.387 paper names and abstracts and 1.307.111 author names related with those papers. After that, in order to set similar conditions to Gollapalli et al.'s study[11] we did not take into account papers whose year is greater than 2006. However, still our corpus size was nearly 10 times larger than the size of the dataset collected by Gollapalli et al. because they created a subset of those document abstracts from CiteSeer[9] corpus by matching venue names that are obtained from Wikipedia [10][11]. This final operation was not reproducible for our dataset because the details of this operation was not given.

4.5.1 Sparsity Parameter

Since we could not reproduce matching CiteSeer and ArnetMiner corpuses, our ArnetMiner dataset remained to be bigger than Gollapalli et al.'s. In order to shrink our corpus size, we have created 3 subsets from our ArnetMiner dataset according to the publication counts of authors. We called these subsets as "sparse weighted", "normal weighted" and "heavy weighted". For "sparse weighted" subset we took into account first 25,854 authors having the least count of publications. For "heavy weighted" subset we took into account first 25,854 authors having the most count of publications. Finally, for "normal weighted" subset we randomly took into account 25,854 authors and their publications from our ArnetMiner dataset. Document counts per year for ArnetMiner's "heavy weighted" dataset are shown in Figure 4.11. In Section 5.13 the results of experiments using these 3 subsets are discussed. ArnetMiner's ground truth 13 topic queries(Table P.1) and manually identified authors related to these 13 queries are listed on its website [11].

[8] arnetminer.org
[9] http://citeseerx.ist.psu.edu/index
[10] http://en.wikipedia.org/wiki/List_of_computer_science_conferences
[11] http://arnetminer.org/lab-datasets/expertfinding/

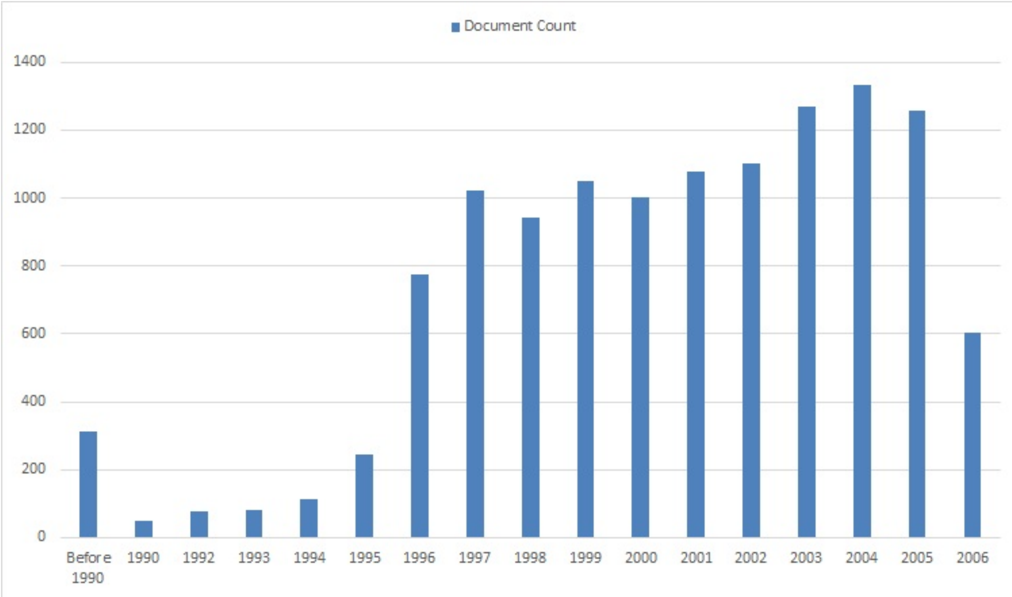

Figure 4.10: Document Counts per Year for UVT

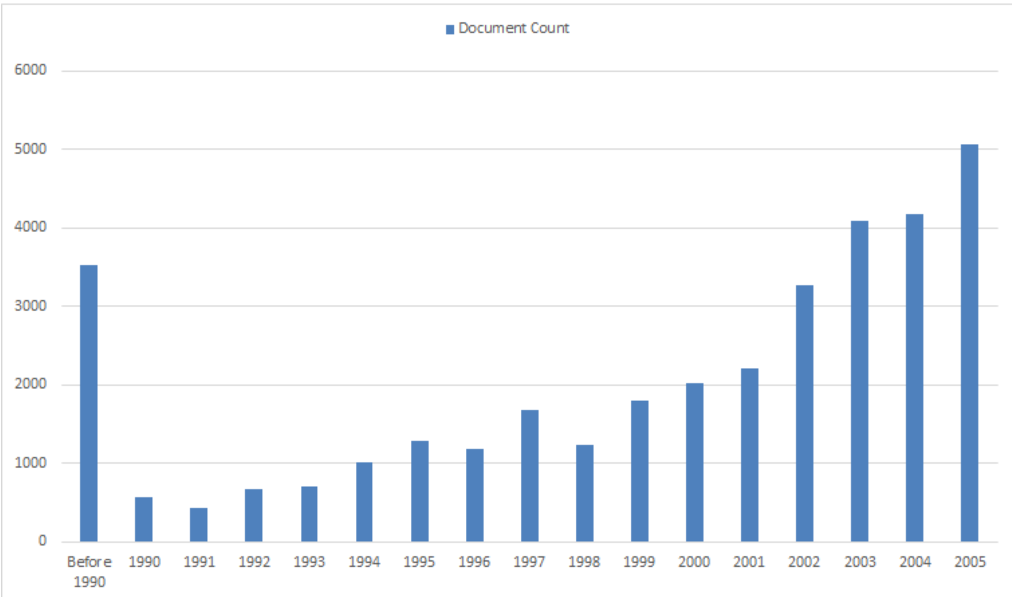

Figure 4.11: Document Counts per Year for ArnetMiner

4.6 Experiment Setup

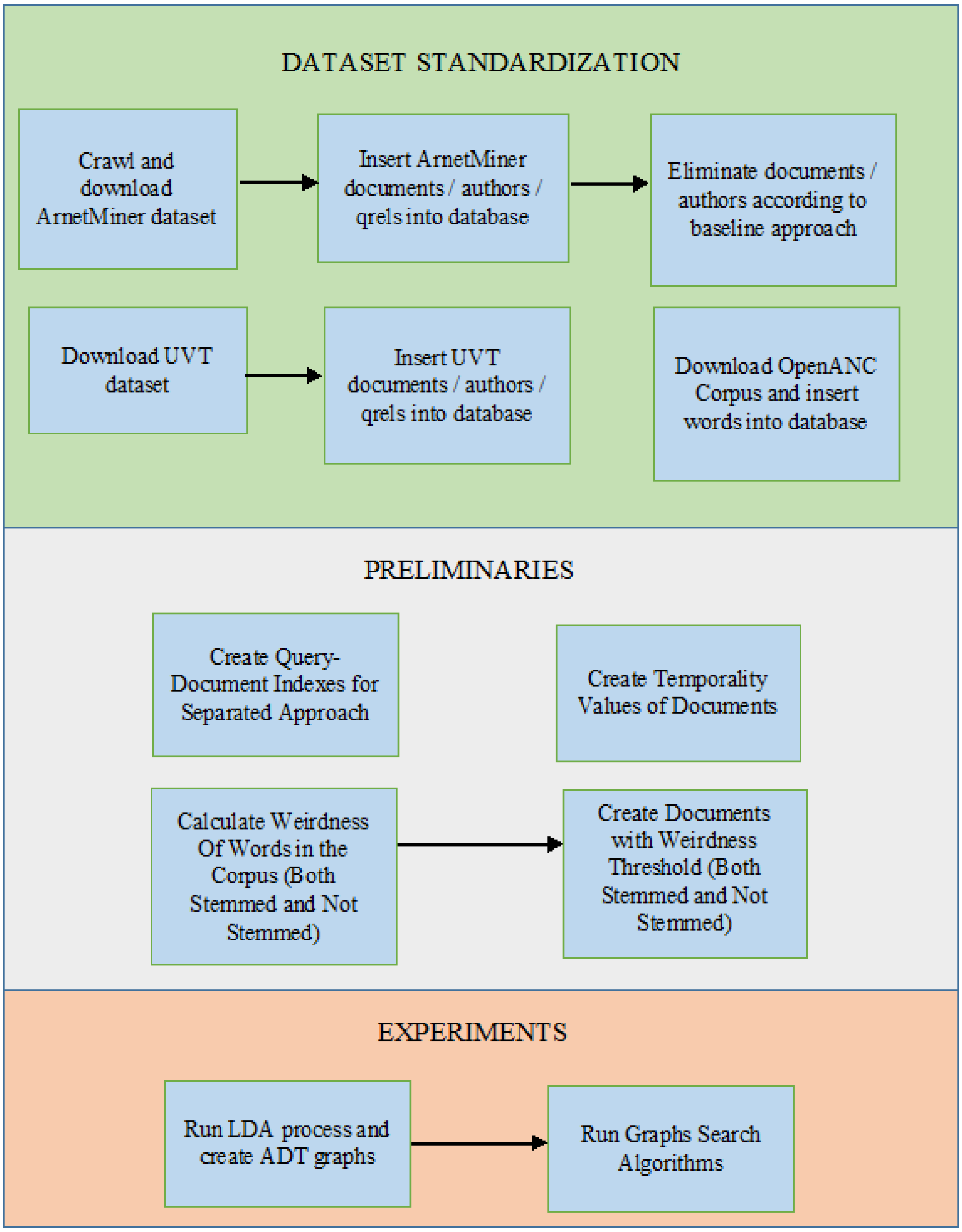

Figure 4.12: All Progress

Figure 4.12 shows the major procedures of our experiments' progress. First main section of our experiment is to set the standardization of our datasets according to our baseline approach. The details of this section are explained in Section 4.5. Second section is the preliminary works before our experiments. The operations of "Preliminaries" section are run once after first section and their outputs can be used many times in the final "Experiments" section. Final section contains the main operations of our experiments. This section can be run many times according to different parameters that are introduced in Section 4.4. The arrow marks in the figure represent the prerequisite for operations. For example, graph search algorithms can not be run before running LDA and creating ADT graphs.

4.6.1 Experimental Environment

The environmental requirements (framework, programming language, API, database) for creating our proposed system are shown in Table 4.7.

Table 4.7: Experimental Environment

Requirement	**Version**	**Purpose of Use**
Java	1.7	Implementation
Netbeans	7.3	Implementation Framework
JSoup	1.7.2	Crawling
Mallet	2.0.7	LDA Implementation
JFreeChart	1.0.17	Creating charts
Oracle	11.2.0	Storage
RStudio	0.98.1062	Log likelihood chart
R	3.1.1	Log likelihood chart

The implementation phase is completed using Java programming language in Netbeans IDE. The crawling phase of ArnetMiner dataset is done using JSoup library. For creating LDA implementation, Mallet toolkit is used. The experimental results are stored in Oracle database. Finally, for interpretation of experimental results, we have also developed an interface to create charts and box plots. The screenshot of the developed interface is shown in Figure 4.13.

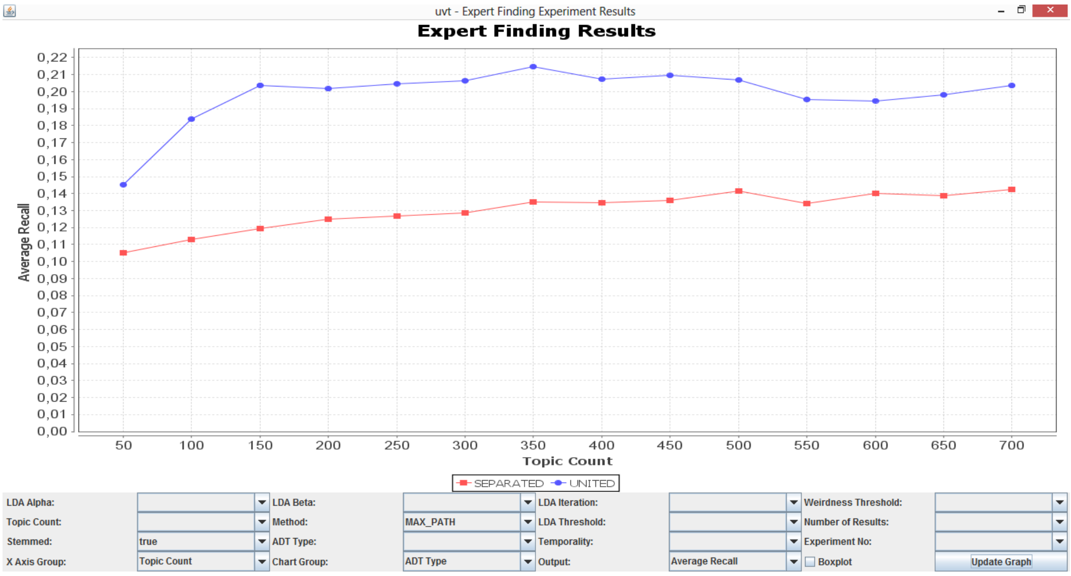

Figure 4.13: Interface for Creating Charts

4.6.2 Experiment Progress

Number of possible results of our proposed system is calculated as the production of count of all possible values of all parameters that are defined in Section 4.4 that would give us more than 1,000,000,000 possible results. Since dealing with that amount of information would be difficult and unmanageable, we followed controlled experiments strategy. First, we selected a parameter to be controlled and fixed other parameters' values. After that, for each possible value of selected parameter, we have calculated the performance of our proposed system and selected the value which returns the best result. In order to ensure the results' reliability, we repeated same experiments for sufficient times to create box plots and we considered average results of those same experiments. We have repeated this method for all parameters and finally selected parameters which return the best performance as our proposed system's parameters.

CHAPTER 5

RESULTS AND DISCUSSION

This chapter demonstrates the results of our experiments and evaluates the performance of our proposed system according to the baseline approach.

5.1 Evaluation Setup and Metrics

We have evaluated our proposed system's performance for 4 different performance evaluation methods (Average Recall, Average Precision, MAP, MRR) at five different number of retrieved results (k = 10, 20, 30, 40, 50) as they were explained in Section 3.3. Those methods and values are also the performance evaluation parameters of the baseline approach, too[11].

In addition to those four performance evaluation methods, we have also tested our system's performance by *time spent* property. At this point, we should note that we have only calculated *time spent* in "Experiments" section of Figure 4.12. In other words, preliminary works were not taken into account while calculating *time spent* for an experiment. However, not all the parameters are related to preliminary works (for example: LDA Alpha, LDA Beta, Topic Count etc.). Consequently, we have decided that *time spent* property can help us to measure the performance of our proposed system.

5.2 United ADT Approach

One of the main contributions of our proposed system is "United ADT" approach. In order to evaluate the performance of "United ADT", we compared its results in terms of Average Recall with the "Separated ADT". Figure C.1 shows the performance of these two ADT types according to 13 different topic counts. This figure indicates that "United Approach" outperforms "Separated Approach" in all counts of topics between 50 and 700. For further analysis, in figure Figure C.2 we put the best performances of both approaches (i. e. topic count = 200 for United and topic count = 650 for Separated) in box plot. The boxplot explicitly demonstrates that "United ADT" approach outperforms "Separated Approach".

If we compare both approaches' performance in terms of *time spent* property, Figure C.3 shows that "United ADT" performs better than "Separated ADT" in all number of topics. At this point, we should note that "Separated ADT" approach also requires a preliminary work before running experiments which is "Create Document-Query Indexes" in Figure 4.12. The time spent for this operation is not calculated in that plot. Even though this preliminary operation's time is not added to "Separated ADT"'s time, "United ADT" approach still outperforms "Separated ADT" in terms of *time spent*.

5.3 Method

We used node similarity calculation methods that are explained in Section 3.2.4. Those methods were introduced in our baseline approach[11]. Figure B.1 shows the performance of methods according to the number of results. This figure seems to be similar to Figure 3.4 which proves that our system runs ADT methods correctly. Figure B.2 shows the results of ADT method tests in a box plot. In the box plot, we can see that that ProductPaths method outperforms other methods. According to these results, we have chosen ProductPaths as our proposed system's ADT method.

5.4 Topic Count

Figure C.1 and Figure C.3 show that there is strong relationship between number of topics and ADT Type parameters and both ADT type approaches perform the best in different number of topics for UVT dataset. Similar situation can be seen in Figure D.3 for ArnetMiner dataset. Figure D.2 and Figure D.1 performance of both approaches in different number of topics in box plots. Those figures indicate that

"Separated ADT" performs best for 650 topics and "United ADT" performs best for 250 topics in UVT dataset. Since "United ADT" is our proposed system's main contribution, we have selected 250 as our proposed system's *Number of Topics* value for UVT dataset. For ArnetMiner, we have selected 650 for baseline approach and 150 as our proposed system's *Number of Topics* value, according to the results.

5.5 LDA Alpha

In Section 4.3.2, we have explained the function of "α" (alpha) parameter in LDA process. Figure E.1 shows the performance of system for different alpha values in terms of average recall, and Figure E.2 shows it in terms of time spent. In terms of time spent, smaller alpha values performs better, however in terms of average recall, values between 35.0 and 50.0 performs better than other values. Since time spent is not our priority and average recall box plot does not show clear improvements for any of the values, we have chosen 50.0, which is same as our baseline approach, as our proposed system's *LDA Alpha* value.

5.6 LDA Beta

In Section 4.3.3, we have explained the function of "β" (beta) parameter in LDA process. Figure F.1 shows the performance of system for different beta values in terms of average recall, and Figure F.2 shows it in terms of time spent. In terms of bot average recall and time spent, smaller beta values seem to perform better. According to those results, we have chosen 0.01, which is same as our baseline approach, as our proposed system's *LDA Beta* value.

5.7 LDA Threshold

In Section 4.3.4, we have explained the function of *LDA Threshold* parameter in LDA process. Figure G.1 shows the performance of the system according to different *LDA Threshold* values in 5 different experiments and Figure G.2 demonstrates the same results in box plot. Figure G.3 shows *LDA Threshold* performance in terms of time spent. According to these results, in terms of time spent different *LDA Threshold* values do not make significant differences, however in terms of average recall higher

LDA Threshold values perform better than lower values. According to those results, we have chosen 0.0050 as our proposed system's *LDA Threshold* value.

5.8 LDA Iteration Count

In Section 4.3.5, we have explained the function of *LDA Iteration Count* parameter in LDA process. Figure H.1 and Figure H.2 show the performance of *LDA Iteration Count* parameter in terms of average recall and Figure H.3 shows it in terms of time spent. As it was expected, higher *LDA Iteration Count* values perform better than in terms of average recall while they perform worse in terms of time spent. Since time spent is not our priority, we have selected 180 as *LDA Iteration Count* of our proposed system.

5.9 Stemming

In Section 4.2.2, we have explained the effect of stemming to our LDA topics. Figure I.2 shows the effect of stemming to average recall for different topic counts and it shows that for most of the topics stemming improves performance. Figure I.1 explicitly shows the improvement of performance by stemming in a box plot where number of topics is set to 200 for UVT dataset. Similarly, Figure I.3 shows the improvement of performance for ArnetMiner dataset. According to those results, we decided to integrate Porter's stemming algorithm[23] to our proposed system.

5.10 Temporality

Figure J.1 and Figure J.3 show the performance of our proposed system for all 6 of the temporality parameters defined. According to the both box plots, it can be seen that all temporality effects outperform our baseline approach (i. e. parameter = NONE) in both datasets. Other parameters' outputs cannot be distinguished from each other except parameter NONE. Figure J.4 shows the same result in different plot which again shows that all other parameters outperform parameter NONE. This plot also demonstrates that parameter QUARTIC performs the best in terms of average recall in UVT. Figure J.5 shows that QUARTIC performs better than NONE in ArnetMiner, too. According to those results, we have selected QUARTIC as our proposed system's temporality parameter.

5.11 Weirdness

In Section 4.2.3 we have explained the effect of weirdness threshold to our LDA topics. Figure K.1 shows the effect of both weirdness threshold and stemming in terms of average recall. According to this figure, weirdness threshold should be between 0.80 and 0.85 in order to gain the best performance. Figure K.2 and Figure K.3 show the performance in a box plot with more number of experiments and less number of weirdness threshold inputs for both datasets. Those figures explicitly demonstrate that weirdness threshold values between 0.80 and 0.85 improve performance of the system. Figure K.4 shows the effect in terms of time spent which indicates that lower weirdness threshold performs better in terms of time spent (note that weirdness threshold also requires a preliminary work and time). According to these results, we have selected 0.80 for UVT and 0.83 for ArnetMiner as our proposed system's *Weirdness Threshold* value.

5.12 Hierarchical Topic Modeling

In Section 4.3.6 we have explained hierarchical LDA's details. Figure M.1 shows the comparison of performances of "Hierarchical LDA" and "Normal LDA" in terms of average recall and Figure M.3 shows the performance in terms of time spent. Figure M.2 shows each of the 7 experiment's results individually. These outputs show that, "Hierarchical LDA" performs worse than "Normal LDA" in our proposed system. According to those results, we selected "Normal LDA" as our proposed system's *LDA Type* parameter.

5.13 Sparsity

Sparsity was only used in ArnetMiner dataset as a parameter in order to generate similar conditions to baseline approach. The reasons for using *sparsity* parameter were explained in Section 4.5.1. Figure L.1 shows the performance of our system for the three sparsity parameters in terms of average recall and Figure L.2 shows the performance in terms of time spent. Because heavy weighted subset contains more documents than others, it required more time to complete experiments. Figure L.1 shows that "Normal Weighted" subset performs the best in these 3 subsets. Therefore, for ArnetMiner dataset we have selected "Normal Weighted" subset as dataset of our experiments.

5.14 Evaluation of Final Proposed System vs. Baseline Approach

Table 5.1: Results using UVT dataset

UVT					
Output	**Baseline (Sep. ADT)**	**Uni. ADT**	**Uni. ADT + Stem.**	**Uni. ADT + Stem. + Weir.**	**Uni. ADT + Stem. + Weir. + Temp.**
AvgRecall@10	0.126	0.130	**0.135**	0.133	0.134
AvgPrec@10	0.108	0.107	**0.115**	0.113	0.113
MRR@10	**0.325**	0.223	0.230	0.241	0.182
MAP@10	**0.069**	0.056	0.060	0.059	0.049
AvgRecall@50	0.278	0.286	0.290	0.301	**0.303**
AvgPrec@50	0.048	0.048	0.051	0.052	**0.052**
MRR@50	**0.338**	0.236	0.242	0.255	0.196
MAP@50	**0.086**	0.076	0.080	0.080	0.070

Table 5.1 shows the performance of our proposed system using UVT dataset according to different performance evaluation methods at different "number of results" values. In this table, when weirdness is selected, Weirdness threshold is set to 0.80; when temporality is selected, temporality is set to QUARTIC. The topic count is set to 650 for "Separated ADT" and to 250 for "United ADT" approaches. The other parameters are as follows: Method: ProductPaths, LDA Alpha: 50.0, LDA Beta: 0.01, LDA Threshold: 0.005, LDA Iteration Count: 100, Number of Results: 50.

Table 5.2: Results using ArnetMiner dataset

ArnetMiner					
Output	**Baseline (Sep. ADT)**	**Uni. ADT**	**Uni. ADT + Stem.**	**Uni. ADT + Stem. + Weir.**	**Uni. ADT + Stem. + Weir. + Temp.**
AvgRecall@10	0.337	0.390	0.405	**0.410**	0.390
AvgPrec@10	0.402	0.460	0.472	**0.479**	0.455
MRR@10	0.792	0.782	0.874	**0.895**	0.442
MAP@10	0.254	0.296	**0.318**	0.314	0.223
AvgRecall@50	0.644	0.614	0.634	0.662	**0.673**
AvgPrec@50	0.153	0.145	0.150	0.156	**0.158**
MRR@50	0.792	0.783	**0.874**	0.827	0.443
MAP@50	0.340	0.365	0.395	**0.404**	0.324

Table 5.2 shows the performance of our proposed system using ArnetMiner dataset. In this table, when weirdness is selected, Weirdness threshold is set to 0.83; when temporality is selected, temporality is set to QUARTIC. The topic count is set to 650 for "Separated ADT" and to 150 for "United ADT" approaches. Other parameters are as follows: Method: ProductPaths, LDA Alpha: 50.0, LDA Beta: 0.01, LDA Threshold: 0.005, LDA Iteration Count: 100, Number of Results: 50.

5.15 Performance Evaluation and Discussion

Table 5.1 and Table 5.2 show that our contributions to the baseline approach remarkably improves the performance of baseline approach, especially in terms of AvgRecall and AvgPrec methods.

In Table 5.1, it can be seen that our proposed system can not improve MRR@10, MAP@10, MRR@50 and MAP@50 values in all cases for UvT dataset. This problem might have been emerged as both MAP Equation 3.8 and MRR Equation 3.9

methods attach highest importance to first "true positive" results, even MRR only considers first relevant expert retrieved. On the other hand, AvgRecall and AvgPrec methods attach equal importance to each relevant document retrieved. According to these results, we can say that if someone wants to find only one expert in a topic by only looking at first few results, our proposed system may not be the best choice for this kind of requirement. However, if more than one expert is demanded for a topic, then our proposed system will be a better choice than baseline approach.

On the other hand, Table 5.2 shows that our contributions have made certain improvements on ArnetMiner dataset because the results of all performance evaluation methods returned highest scores on one of our proposed contributions. However, baseline results in our experiments for ArnetMiner dataset (Table 5.2) are not so similar to Gollapalli et al.'s results [11] (Table 3.4). Possible reasons that caused these problems are:

- ArnetMiner dataset is not constant and it grows over time, therefore the dataset that we accessed has been changed since the time it was accessed by Gollapalli et al.
- Gollapalli et al. created a subset of those document abstracts from CiteSeer corpus as it is explained in Section 4.5. Since we were unable to reproduce that operation, our dataset may remain different from theirs.

If we discuss the performance of our contributions, firstly, we can see that our proposed "United ADT" approach clearly outperforms the baseline "Separated ADT" approach in terms of retrieval performance and time spent. "United ADT" not only improves performance, but also eliminates preliminary topic query - document indexing operation which makes our proposed system straightforward.

Our experiments showed that there is a strong relationship between performance of ADT graph and number of topics. For UVT, higher number of topics improves performance of our baseline approach, on the other hand, our proposed system's performance converges to maximum for number of topics between 100 and 200 for both datasets. The reason that UVT requires more number of topics than ArnetMiner might be the domain size. As it was shown in Table 4.6, UVT's domain is broader and larger while ArnetMiner's domain is limited to Computer Science.

Our experiments suggested that tuning parameters (alpha, beta, iteration count, threshold) of LDA implementation can slightly make improvements on retrieval and time spent performances.

The significance of stemming is also demonstrated by our experiments. Stemming improved performances of both UVT and ArnetMiner datasets. On the other hand, stemming requires a preliminary work which can slightly reduce time spent performance in total.

Our experiments demonstrated that temporal effects of documents can slightly increase performance of expert finding systems. The reason that temporality did not bring out higher improvements can be dataset limitations. Figure J.3 and Figure J.1 show the effect of temporality difference in ArnetMiner and UVT datasets. The reason that temporality in ArnetMiner performed better than temporality in UVT may be that ArnetMiner has higher ratio of documents with time info than UVT which is explained in Table 4.6. Another reason of this effect might be attributed to the domain characteristics. ArnetMiner is only related to Computer Science and people's interests in Computer Science might change significantly compared to the other domains over time.

According to Figure J.4, our temporality parameters are listed according to their performance as: QUARTIC, CUBIC, QUADRATIC, LINEAR, LOGARITHMIC and NONE. That result concludes that exponential temporal relationships between authors and documents give better results than linear temporal relationships.

Figure J.2 shows temporality results for some specific queries of UVT dataset. The minimum year of papers of authors that have expertise for query (expertise) id = 8581 is 2003. The minimum year of papers of authors that have expertise for query (expertise) id = 9362 is 1996. The maximum year of both queries is 2006. In the plot it can be seen that, query id = 8581 did not gain improvement by temporality while query id = 9362 gained performance improvement in QUARTIC temporal parameter. That detailed plot demonstrates that higher differences between maximum and minimum year of papers will bring out higher performance improvements with temporality.

Weirdness effect on topic modeling is also shown in our experiments. Both our datasets performed better for weirdness threshold values between 0.80 and 0.85. Since ArnetMiner's domain is specific while UVT's domain is general, we expected that weirdness would make higher performance improvements in ArnetMiner. However, our experiments demonstrated that weirdness threshold has made similar performance improvements in both datasets. This can be caused by the general words in the names of the topic queries. For example in Table P.1, there is a topic query named as "Planning" which is a general word while it is related to Computer Science. When we looked expertise based results of our experiments setting weirdness threshold to 0.83 for "Semantic Web" improves performance from 0.620 to 0.684 in terms of average recall while for "Planning" performance is improved from 0.561 to 568 which is negligible.

Hierarchical topic modeling did not perform well in our experiments. In all cases, "Normal LDA" performed better than "Hierarchical LDA". Table M.1 shows examples of hierarchical LDA topics for UVT. At the bottom of table, the two topics 14474 and 14648 both have exactly the same most common words. Besides, most of the common words of topic 11418 is same as these two topics. Even though, we changed alpha and gamma parameters of LDA, we again faced same problems. These topics show that hierarchical LDA approach is not suitable for our datasets.

5.16 Limitations

The limitations that affected the performance of our experiments can be listed as follows:

- ArnetMiner dataset that was used in this study was larger than Gollapalli et al.'s study[11]. In order minimize this effect, we have used sparsity parameter which is explained in Section 4.5.1.
- Only publication type documents of UvT documents contain year information. Research Area, Course Homepage, Personal Homepage type documents in UvT dataset do not have year information (Table 4.6). In order to use these documents in our temporality experiments, we set their year values to 2006, the maximum year of documents in UvT.
- Some of the authors listed in qrel lists of both datasets do not have any related

documents. For example, the author named "Ko van der Sloot" is related to 2 topic queries as ground truth which are "topic5695:programming languages" and "topic5801:computer linguistics" (Table O.1). However, in the dataset "Ko van der Sloot" does not have any related documents. There are more than 50 similar examples in both datasets. Consequently, this problem decreases our performance results.

- We were not able to improve Indri search engine's source codes in order to insert its indices to our databases. Therefore, we have created our own topic query - document indexes by our term frequency based algorithm. Samples indices of both Indri and our method are shown in Figure N.1 (Indri) and in Table N.1 (our method) for topic query "accounting" of UvT dataset. We can see in the figure and table that first 10 indexed documents are the same for both results. On the other hand, there are some differences in these results. For example, in Figure N.1 Indri returned document no. 12693 in 22^{th} order while our method did not return document no. 12693 in first 25 results in Table N.1.
- Some of the publication titles in UvT dataset are incorrect. For example, in file "profile.english.120146.xml" the title of 10^{th} publication was written as "to-ber 1)" which should be "Measuring the effectiveness of a problem structuring method". We have manually fixed similar problems in the dataset.

CHAPTER 6

CONCLUSION AND FUTURE WORK

6.1 Conclusion

In this study, we have presented a novel temporal expert finding methodology along with weirdness effect. We have tested the performance of our proposed system in two different datasets, UVT and ArnetMiner, via 5 performance evaluation methods and 12 different parameters. Our proposed system performed better than the baseline approach in most of the cases for both datasets. The main contributions of our study are summarized as follows:

- A "United ADT" approach is suggested that eliminates preliminary operations for creating topic-document indices.
- The performance of LDA number of topics is proportional to domain diversity and size of a dataset which means a collection that contains documents related to diverse topics requires higher number of LDA topics for better performance.
- Stemming can increase performance of LDA topic modeling.
- Taking into account temporality of a document can increase the performance of an expert finding system. Additionally, creating exponential relationships proportional to year of documents can give better performance than linear relationships.
- Weirdness value of words in a corpus can be used for assigning weights to words according to this value and using a weirdness threshold before topic modeling can increase performance of an expert finding system.
- Creating hierarchical relations between LDA topics and usage of these relations in ADT graphs does not increase performance of an expert finding system.

6.2 Future Work

For future work, possible extension points of our study can be listed as follows:

- A different hierarchical LDA implementation can be used for creating accurate hierarchical LDA topics.
- Hierarchical relations of LDA generated topics can be created using an external topic hierarchy such as ACM [1] or Wikipedia [2].
- The relationships between topics may not necessarily be hierarchical. A relational similarity based topic modeling can be used in order to use the effect of similarity between topics.
- Dynamic topic modeling (Dynamic LDA) can be used for creating temporal relationship between topics and documents. By this way, temporal relationship can be moved from author-document links to topic-document links in ADT graph.
- Document citation information can be used for creating inter-document relationships (In order to achieve this extension, different datasets that contain citation information should be used).
- Social Network Analysis techniques can be used for creating inter-author relationships. For example centrality of an author can be used to effect score of the author to a topic query.
- "Random walk on graphs" approach can be used in ADT graphs.
- The documents' "number of authors" property can be used to change the link strengths between authors and documents.
- Different stemming algorithms or lemmatizing that are compared by Jivani[15] can used instead of Porter's stemming algorithm[23].
- Different word/term frequency methods that are compared by Knoth, Schmidt, Smrz, & Zdrahal[18] can be used as threshold values instead of Weirdness. Different methods can be integrated to each other and the output values can be used as a threshold values as well.

[1] http://dl.acm.org/ccs_flat.cfm
[2] http://en.wikipedia.org/wiki/Category:Areas_of_computer_science

REFERENCES

[1] Khurshid Ahmad, Lee Gillam, Lena Tostevin, et al. University of surrey participation in trec8: Weirdness indexing for logical document extrapolation and retrieval (wilder). In *TREC*, 1999.

[2] Khurshid Ahmad, Mariam Tariq, Bogdan Vrusias, and Chris Handy. *Corpus-based thesaurus construction for image retrieval in specialist domains*. Springer, 2003.

[3] Krisztian Balog, Leif Azzopardi, and Maarten De Rijke. Formal models for expert finding in enterprise corpora. In *Proceedings of the 29th annual international ACM SIGIR conference on Research and development in information retrieval*, pages 43–50. ACM, 2006.

[4] Krisztian Balog, Toine Bogers, Leif Azzopardi, Maarten De Rijke, and Antal Van Den Bosch. Broad expertise retrieval in sparse data environments. In *Proceedings of the 30th annual international ACM SIGIR conference on Research and development in information retrieval*, pages 551–558. ACM, 2007.

[5] David M Blei and John D Lafferty. Dynamic topic models. In *Proceedings of the 23rd international conference on Machine learning*, pages 113–120. ACM, 2006.

[6] David M Blei, Andrew Y Ng, and Michael I Jordan. Latent dirichlet allocation. *the Journal of machine Learning research*, 3:993–1022, 2003.

[7] Ali Daud. Using time topic modeling for semantics-based dynamic research interest finding. *Knowledge-Based Systems*, 26:154–163, 2012.

[8] Ali Daud, Juanzi Li, Lizhu Zhou, and Faqir Muhammad. Temporal expert finding through generalized time topic modeling. *Knowledge-Based Systems*, 23(6):615–625, 2010.

[9] Hongbo Deng, Irwin King, and Michael R Lyu. Formal models for expert finding on dblp bibliography data. In *Data Mining, 2008. ICDM'08. Eighth IEEE International Conference on*, pages 163–172. IEEE, 2008.

[10] Susan T Dumais, George W Furnas, Thomas K Landauer, Scott Deerwester, and Richard Harshman. Using latent semantic analysis to improve access to textual information. In *Proceedings of the SIGCHI conference on Human factors in computing systems*, pages 281–285. ACM, 1988.

[11] Sujatha Das Gollapalli, Prasenjit Mitra, and C Lee Giles. Ranking experts using author-document-topic graphs. In *Proceedings of the 13th ACM/IEEE-CS joint conference on Digital libraries*, pages 87–96. ACM, 2013.

[12] DMBTL Griffiths and MIJJB Tenenbaum. Hierarchical topic models and the nested chinese restaurant process. *Advances in neural information processing systems*, 16:17, 2004.

[13] Thomas L Griffiths and Mark Steyvers. Finding scientific topics. *Proceedings of the National academy of Sciences of the United States of America*, 101(Suppl 1):5228–5235, 2004.

[14] Thomas Hofmann. Probabilistic latent semantic indexing. In *Proceedings of the 22nd annual international ACM SIGIR conference on Research and development in information retrieval*, pages 50–57. ACM, 1999.

[15] Anjali Ganesh Jivani et al. A comparative study of stemming algorithms. *Int. J. Comp. Tech. Appl*, 2(6):1930–1938, 2011.

[16] Ahmad Kardan, Amin Omidvar, and Farzad Farahmandnia. Expert finding on social network with link analysis approach. In *Electrical Engineering (ICEE), 2011 19th Iranian Conference on*, pages 1–6. IEEE, 2011.

[17] Jon M Kleinberg. Authoritative sources in a hyperlinked environment. *Journal of the ACM (JACM)*, 46(5):604–632, 1999.

[18] Petr Knoth, Marek Schmidt, Pavel Smrz, and Zdenek Zdrahal. Towards a framework for comparing automatic term recognition methods. 2009.

[19] Lev Kozakov, Youngja Park, T Fin, Youssef Drissi, Yurdaer Doganata, and Thomas Cofino. Glossary extraction and utilization in the information search and delivery system for ibm technical support. *IBM Systems Journal*, 43(3):546–563, 2004.

[20] Craig Macdonald and Iadh Ounis. Voting for candidates: adapting data fusion techniques for an expert search task. In *Proceedings of the 15th ACM international conference on Information and knowledge management*, pages 387–396. ACM, 2006.

[21] Andrew Kachites McCallum. Mallet: A machine learning for language toolkit. http://mallet.cs.umass.edu/, 2002.

[22] Lawrence Page, Sergey Brin, Rajeev Motwani, and Terry Winograd. The pagerank citation ranking: Bringing order to the web. 1999.

[23] Martin F Porter. An algorithm for suffix stripping. *Program: electronic library and information systems*, 14(3):130–137, 1980.

[24] Randi Reppen and Nancy Ide. The american national corpus overall goals and the first release. *Journal of English Linguistics*, 32(2):105–113, 2004.

[25] Michal Rosen-Zvi, Thomas Griffiths, Mark Steyvers, and Padhraic Smyth. The author-topic model for authors and documents. In *Proceedings of the 20th conference on Uncertainty in artificial intelligence*, pages 487–494. AUAI Press, 2004.

[26] Elena Smirnova and Krisztian Balog. A user-oriented model for expert finding. In *Advances in Information Retrieval*, pages 580–592. Springer, 2011.

[27] Jie Tang, Jing Zhang, Limin Yao, Juanzi Li, Li Zhang, and Zhong Su. Arnetminer: extraction and mining of academic social networks. In *Proceedings of the 14th ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 990–998. ACM, 2008.

[28] Jianwen Wang, Xiaohua Hu, Xinhui Tu, and Tingting He. Author-conference topic-connection model for academic network search. In *Proceedings of the 21st ACM international conference on Information and knowledge management*, pages 2179–2183. ACM, 2012.

[29] Jing Zhang, Jie Tang, and Juanzi Li. Expert finding in a social network. In *Advances in Databases: Concepts, Systems and Applications*, pages 1066–1069. Springer, 2007.

APPENDIX A

ALGORITHMS

for all author a in **A (set of all authors) do**

{initialization}

$authorPaperCount \leftarrow 0$

$logarithmicSum \leftarrow 0$

$linearSum \leftarrow 0$

$quadraticSum \leftarrow 0$

$cubicSum \leftarrow 0$

$quarticSum \leftarrow 0$

for document d of a **do**

{calculation}

$y \leftarrow yearofd$

$authorPaperCount \leftarrow authorPaperCount + 1$

$d.logarithmicYear \leftarrow log(y)$

$d.linearYear \leftarrow y$

$d.quadraticYear \leftarrow y^2$

$d.cubicYear \leftarrow y^3$

$d.quarticYear \leftarrow y^4$

$logarithmicSum \leftarrow logarithmicSum + d.logarithmicYear$

$linearSum \leftarrow linearSum + d.linearYear$

$quadraticSum \leftarrow quadraticSum + d.quadraticYear$

$cubicSum \leftarrow cubicSum + d.cubicYear$

$quarticSum \leftarrow quarticSum + d.quarticYear$

end for

for document d of a **do**

{normalization}

$d.normLogarithmic \leftarrow d.logarithmicYear \div logarithmicSum$

$d.normLinear \leftarrow d.linearYear \div linearSum$

$d.normQuadratic \leftarrow d.quadraticYear \div quadraticSum$

$d.normCubic \leftarrow d.cubicYear \div cubicSum$

$d.normQuartic \leftarrow d.quarticYear \div quarticSum$

end for

end for

Algorithm 1: Temporality calculation algorithm

APPENDIX B

RESULTS OF METHOD AND NUMBER OF RESULTS CONTROLLED EXPERIMENTS

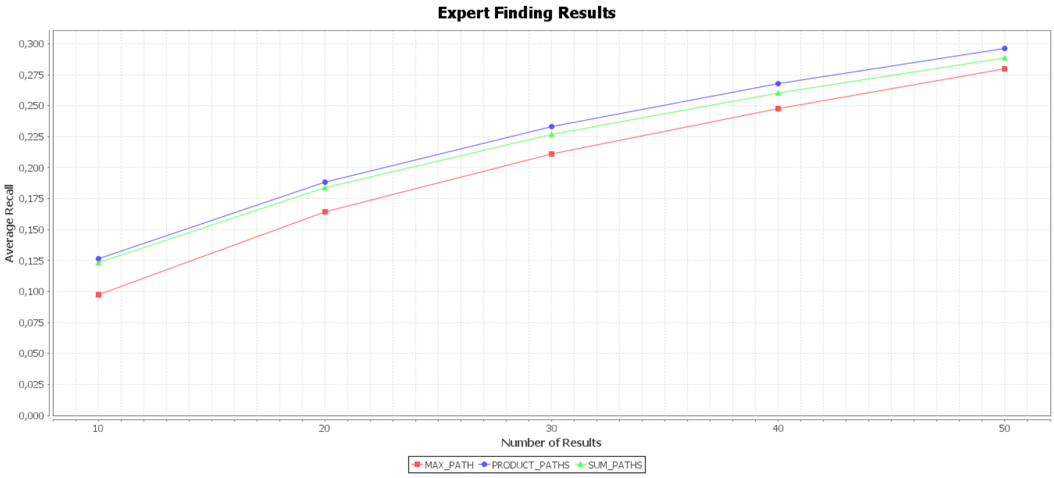

Figure B.1: Method and Number of Results based Results for UVT

The parameters of Figure B.1 are as follows:

- **Dataset**: UVT
- **LDA Alpha**: 50.0
- **LDA Beta**: 0.01
- **LDA Iteration Count**: 100
- **LDA Threshold**: 0.001
- **Weirdness Threshold**: 1.0
- **Stemmed**: false
- **ADT Type**: United
- **Topic Count**: 50 - 700 (average of all)
- **Method**: MaxPath, SumPaths, ProductPaths (varies in plot)
- **Number of Results**: 10 - 50 (x axis values)

- **Output**: Average Recall
- **Temporality**: NONE
- **LDA Type**: Normal
- **Experiment No**: 1 – 5 (average of all)

Figure B.2: Method Box Plot for UVT

The parameters of Figure B.2 are as follows:

- **Dataset**: UVT
- **LDA Alpha**: 50.0
- **LDA Beta**: 0.01
- **LDA Iteration Count**: 100
- **LDA Threshold**: 0.001
- **Weirdness Threshold**: 1.0
- **Stemmed**: false
- **ADT Type**: United
- **Topic Count**: 50 - 700 (average of all)
- **Method**: MaxPath, SumPaths, ProductPaths (x axis values)
- **Number of Results**: 10 - 50 (average of all)
- **Output**: Average Recall
- **Temporality**: NONE
- **LDA Type**: Normal
- **Experiment No**: 1 – 5 (varies in box plot)

APPENDIX C

RESULTS OF ADT TYPE CONTROLLED EXPERIMENTS

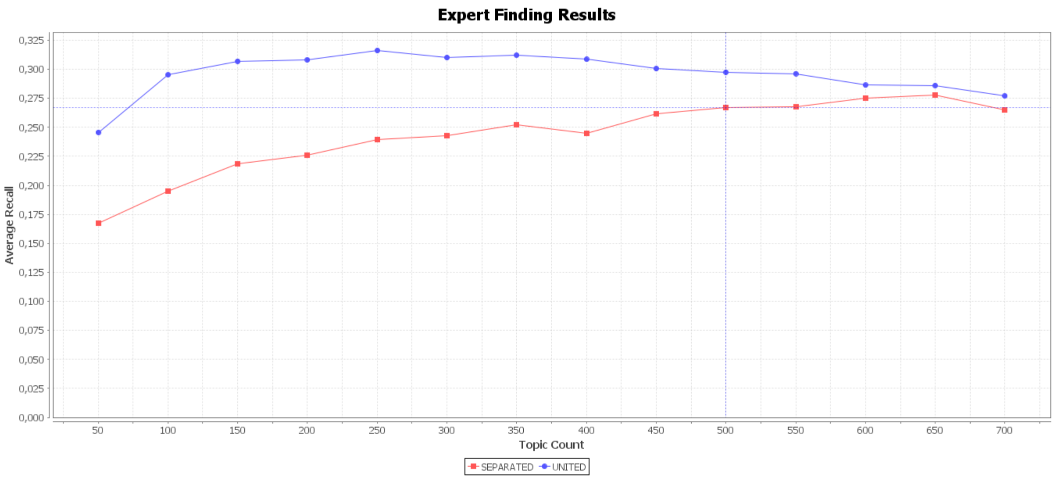

Figure C.1: ADT Type Results According to Topic Counts for UVT

The parameters of Figure C.1 are as follows:

- **Dataset**: UVT
- **LDA Alpha**: 50.0
- **LDA Beta**: 0.01
- **LDA Iteration Count**: 100
- **LDA Threshold**: 0.001
- **Weirdness Threshold**: 1.0
- **Stemmed**: false
- **ADT Type**: United, Separated (varies in plot)
- **Topic Count**: x axis values (50 – 700)
- **Method**: ProductPaths
- **Number of Results**: 50
- **Output**: Average Recall

- **Temporality**: NONE
- **LDA Type**: Normal
- **Experiment No**: 1 – 5 (average of all)

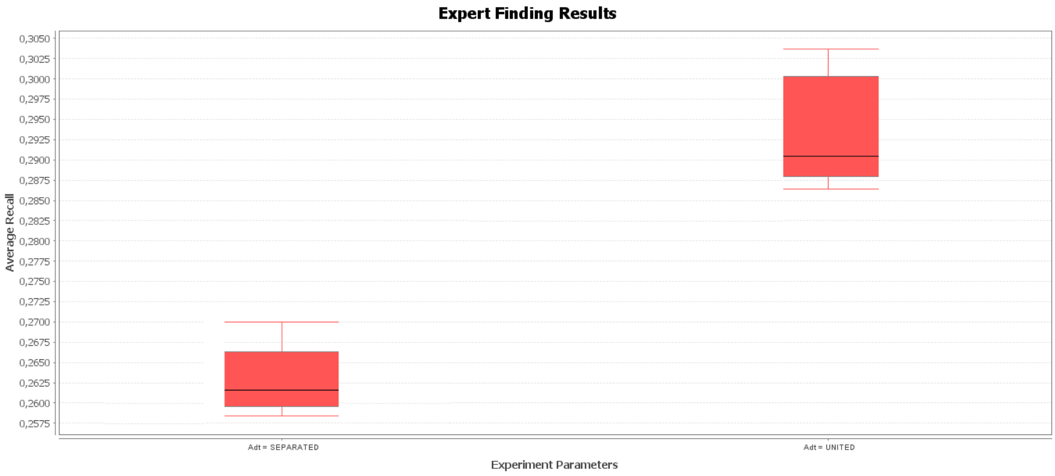

Figure C.2: ADT Type Box Plot for UVT

The parameters of Figure C.2 are as follows:

- **Dataset**: UVT
- **LDA Alpha**: 50.0
- **LDA Beta**: 0.01
- **LDA Iteration Count**: 100
- **LDA Threshold**: 0.001
- **Weirdness Threshold**: 1.0
- **Stemmed**: false
- **ADT Type**: United, Separated (x axis values)
- **Topic Count**: 650
- **Method**: ProductPaths
- **Number of Results**: 50
- **Output**: Average Recall
- **Temporality**: NONE
- **LDA Type**: Normal
- **Experiment No**: 1 – 5 (varies in box plot)

Figure C.3: ADT Type Time Spent According to Topic Counts for UVT

The parameters of Figure C.3 are as follows:

- **Dataset**: UVT
- **LDA Alpha**: 50.0
- **LDA Beta**: 0.01
- **LDA Iteration Count**: 100
- **LDA Threshold**: 0.001
- **Weirdness Threshold**: 1.0
- **Stemmed**: false
- **ADT Type**: United, Separated (varies in plot)
- **Topic Count**: x axis values (50 – 700)
- **Method**: ProductPaths
- **Number of Results**: 50
- **Output**: Time Spent
- **Temporality**: NONE
- **LDA Type**: Normal
- **Experiment No**: 1 – 5 (average of all)

APPENDIX D

RESULTS OF TOPIC COUNT CONTROLLED EXPERIMENTS

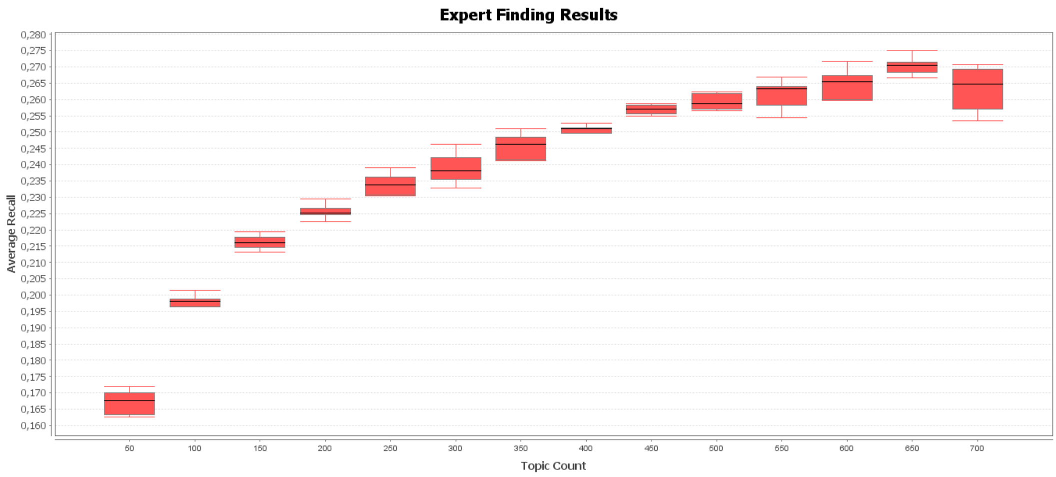

Figure D.1: Topic Count / Separated ADT Box Plot for UVT

The parameters of Figure D.1 are as follows:

- **Dataset**: UVT
- **LDA Alpha**: 50.0
- **LDA Beta**: 0.01
- **LDA Iteration Count**: 100
- **LDA Threshold**: 0.001
- **Weirdness Threshold**: 1.0
- **Stemmed**: false
- **ADT Type**: Separated
- **Topic Count**: 50 - 700 (x axis values)
- **Method**: ProductPaths
- **Number of Results**: 50
- **Output**: Average Recall

- **Temporality**: NONE
- **LDA Type**: Normal
- **Experiment No**: 1 – 5 (varies in box plot)

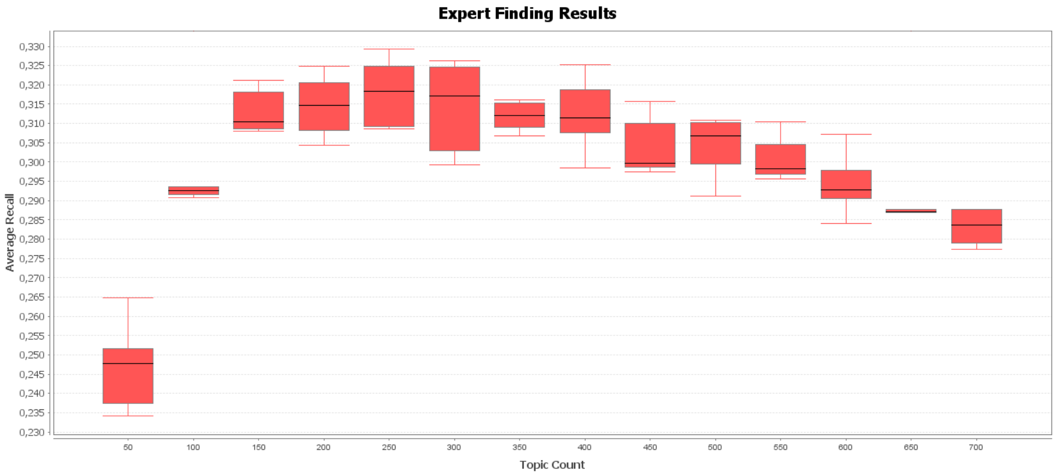

Figure D.2: Topic Count / United ADT Box Plot for UVT

The parameters of Figure D.2 are as follows:

- **Dataset**: UVT
- **LDA Alpha**: 50.0
- **LDA Beta**: 0.01
- **LDA Iteration Count**: 100
- **LDA Threshold**: 0.001
- **Weirdness Threshold**: 1.0
- **Stemmed**: false
- **ADT Type**: United
- **Topic Count**: 50 - 700 (x axis values)
- **Method**: ProductPaths
- **Number of Results**: 50
- **Output**: Average Recall
- **Temporality**: NONE
- **LDA Type**: Normal
- **Experiment No**: 1 – 5 (varies in box plot)

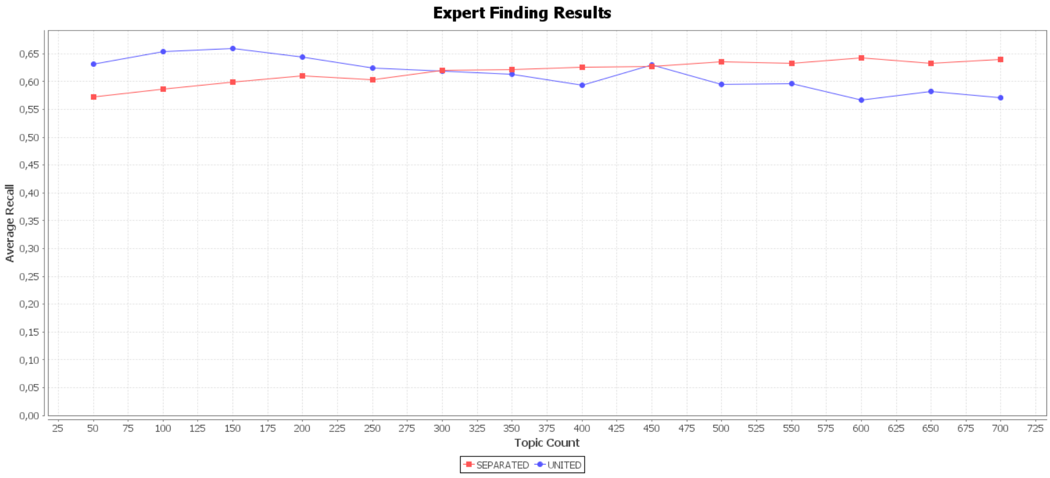

Figure D.3: ADT Type Results According to Topic Counts for ArnetMiner

The parameters of Figure D.3 are as follows:

- **Dataset**: ArnetMiner
- **LDA Alpha**: 50.0
- **LDA Beta**: 0.01
- **LDA Iteration Count**: 100
- **LDA Threshold**: 0.001
- **Weirdness Threshold**: 1.0
- **Stemmed**: false
- **ADT Type**: United, Separated (varies in plot)
- **Topic Count**: x axis values (50 – 700)
- **Method**: ProductPaths
- **Number of Results**: 50
- **Output**: Average Recall
- **Temporality**: NONE
- **LDA Type**: Normal
- **Experiment No**: 1 – 5 (average of all)

APPENDIX E

RESULTS OF LDA ALPHA CONTROLLED EXPERIMENTS

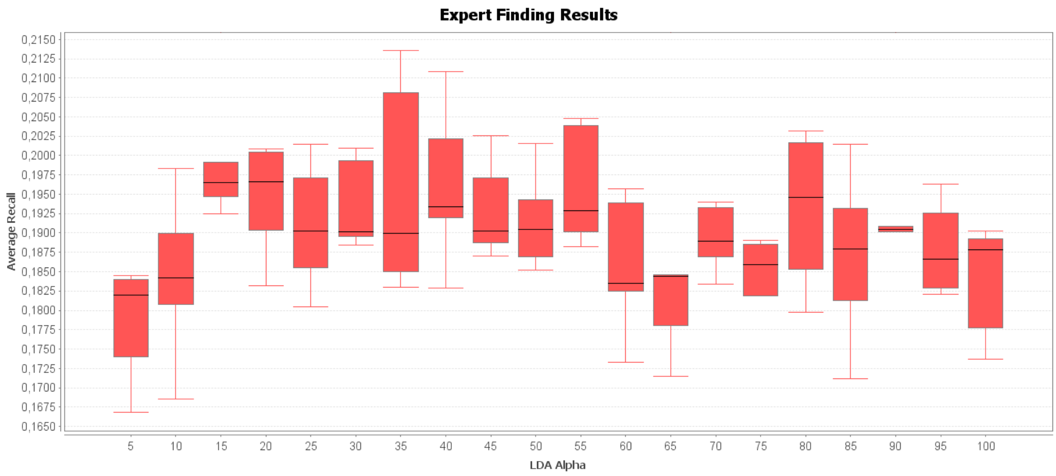

Figure E.1: LDA Alpha Box Plot for UVT

The parameters of Figure E.1 are as follows:

- **Dataset**: UVT
- **LDA Alpha**: 5.0 - 100.0 (x axis values)
- **LDA Beta**: 0.01
- **LDA Iteration Count**: 100
- **LDA Threshold**: 0.001
- **Weirdness Threshold**: 1.0
- **Stemmed**: false
- **ADT Type**: United
- **Topic Count**: 200
- **Method**: ProductPaths
- **Number of Results**: 50
- **Output**: Average Recall

- **Temporality**: NONE
- **LDA Type**: Normal
- **Experiment No**: 1 – 5 (varies in box plot)

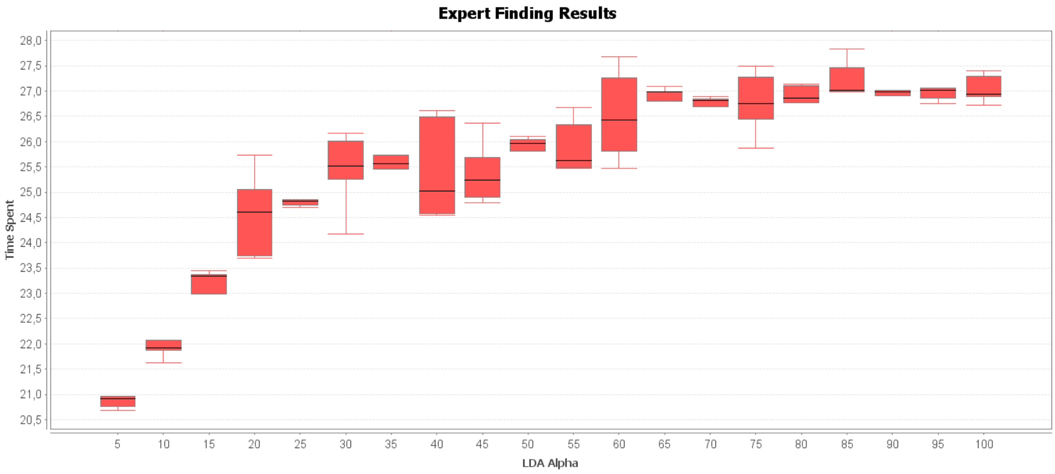

Figure E.2: LDA Alpha Time Spent Box Plot for UVT

The parameters of Figure E.2 are as follows:

- **Dataset**: UVT
- **LDA Alpha**: 5.0 - 100.0 (x axis values)
- **LDA Beta**: 0.01
- **LDA Iteration Count**: 100
- **LDA Threshold**: 0.001
- **Weirdness Threshold**: 1.0
- **Stemmed**: false
- **ADT Type**: United
- **Topic Count**: 200
- **Method**: ProductPaths
- **Number of Results**: 50
- **Output**: Time Spent
- **Temporality**: NONE
- **LDA Type**: Normal
- **Experiment No**: 1 – 5 (varies in box plot)

APPENDIX F

RESULTS OF LDA BETA CONTROLLED EXPERIMENTS

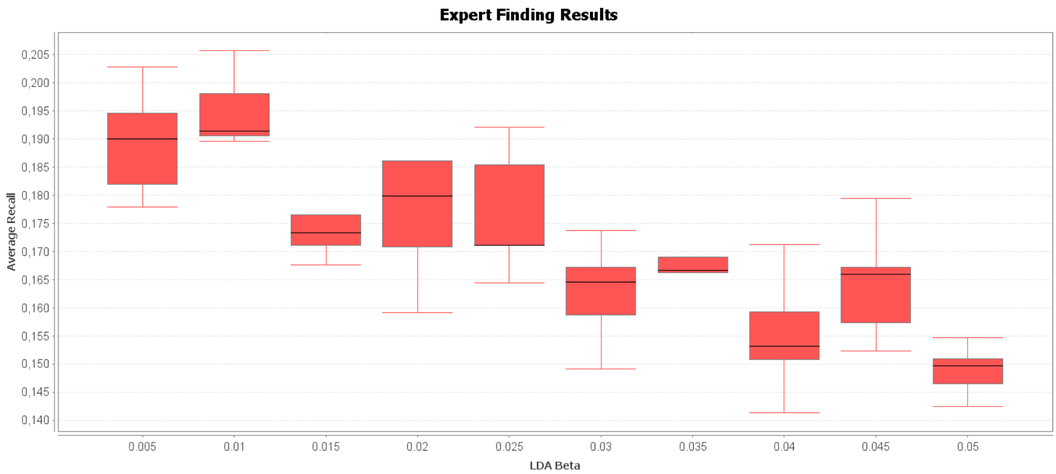

Figure F.1: LDA Beta Box Plot for UVT

The parameters of Figure F.1 are as follows:

- **Dataset**: UVT
- **LDA Alpha**: 50.0
- **LDA Beta**: 0.005 - 0.050 (x axis values)
- **LDA Iteration Count**: 100
- **LDA Threshold**: 0.001
- **Weirdness Threshold**: 1.0
- **Stemmed**: false
- **ADT Type**: United
- **Topic Count**: 200
- **Method**: ProductPaths
- **Number of Results**: 50
- **Output**: Average Recall

- **Temporality**: NONE
- **LDA Type**: Normal
- **Experiment No**: 1 – 5 (varies in box plot)

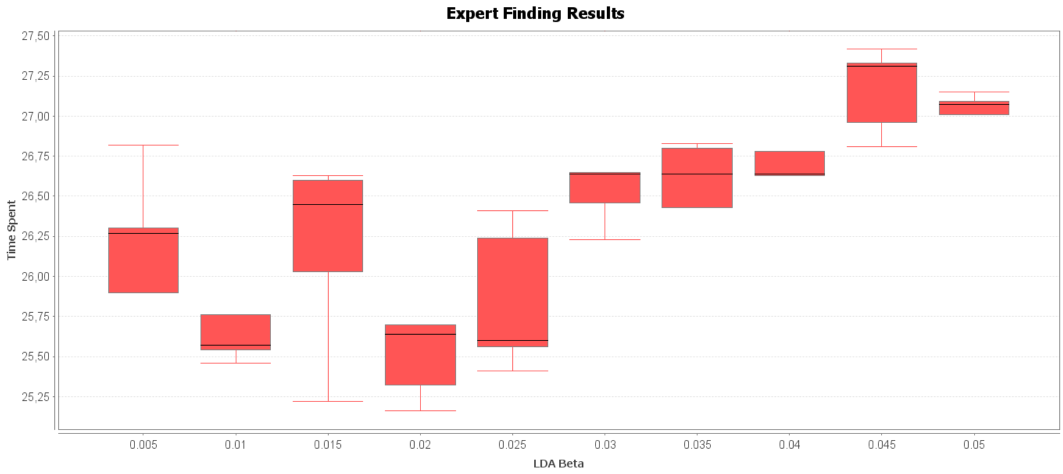

Figure F.2: LDA Beta Time Spent Box Plot for UVT

The parameters of Figure F.2 are as follows:

- **Dataset**: UVT
- **LDA Alpha**: 50.0
- **LDA Beta**: 0.005 - 0.050 (x axis values)
- **LDA Iteration Count**: 100
- **LDA Threshold**: 0.001
- **Weirdness Threshold**: 1.0
- **Stemmed**: false
- **ADT Type**: United
- **Topic Count**: 200
- **Method**: ProductPaths
- **Number of Results**: 50
- **Output**: Time Spent
- **Temporality**: NONE
- **LDA Type**: Normal
- **Experiment No**: 1 – 5 (varies in box plot)

APPENDIX G

RESULTS OF LDA THRESHOLD CONTROLLED EXPERIMENTS

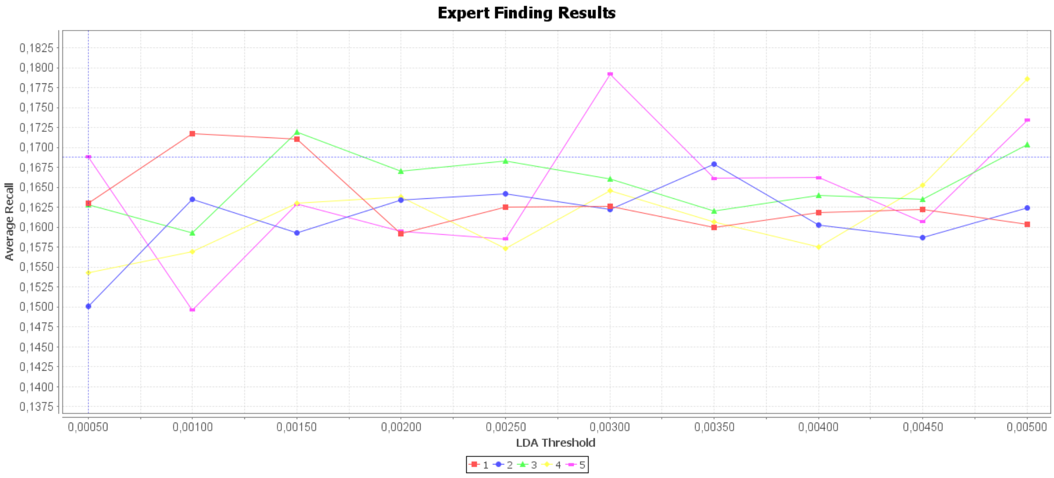

Figure G.1: LDA Threshold Results for UVT

The parameters of Figure G.1 are as follows:

- **Dataset**: UVT
- **LDA Alpha**: 50.0
- **LDA Beta**: 0.01
- **LDA Iteration Count**: 100
- **LDA Threshold**: 0.0005 - 0.0050 (x axis values)
- **Weirdness Threshold**: 1.0
- **Stemmed**: false
- **ADT Type**: United
- **Topic Count**: 200
- **Method**: ProductPaths
- **Number of Results**: 50

- **Output**: Average Recall
- **Temporality**: NONE
- **LDA Type**: Normal
- **Experiment No**: 1 – 5 (varies in plot)

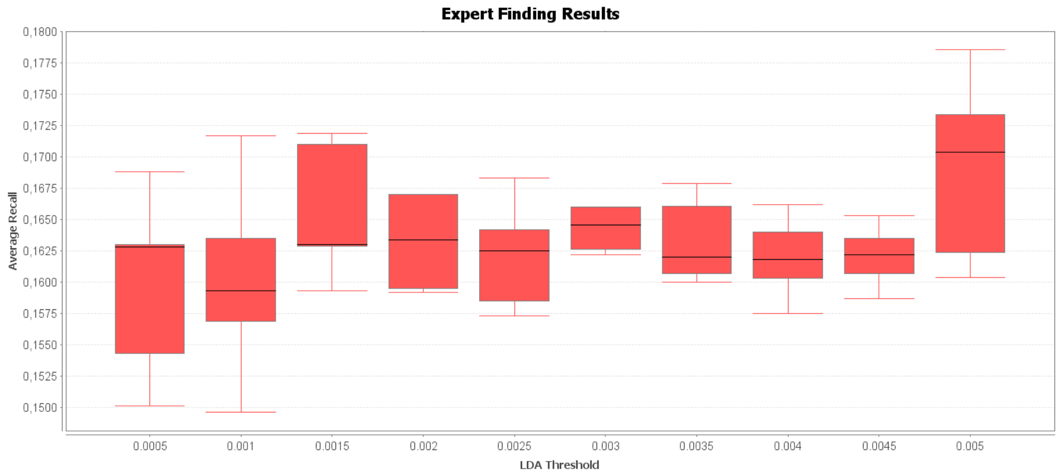

Figure G.2: LDA Threshold Box Plot for UVT

The parameters of Figure G.2 are as follows:

- **Dataset**: UVT
- **LDA Alpha**: 50.0
- **LDA Beta**: 0.01
- **LDA Iteration Count**: 100
- **LDA Threshold**: 0.0005 - 0.0050 (x axis values)
- **Weirdness Threshold**: 1.0
- **Stemmed**: false
- **ADT Type**: United
- **Topic Count**: 200
- **Method**: ProductPaths
- **Number of Results**: 50
- **Output**: Average Recall
- **Temporality**: NONE
- **LDA Type**: Normal
- **Experiment No**: 1 – 5 (varies in box plot)

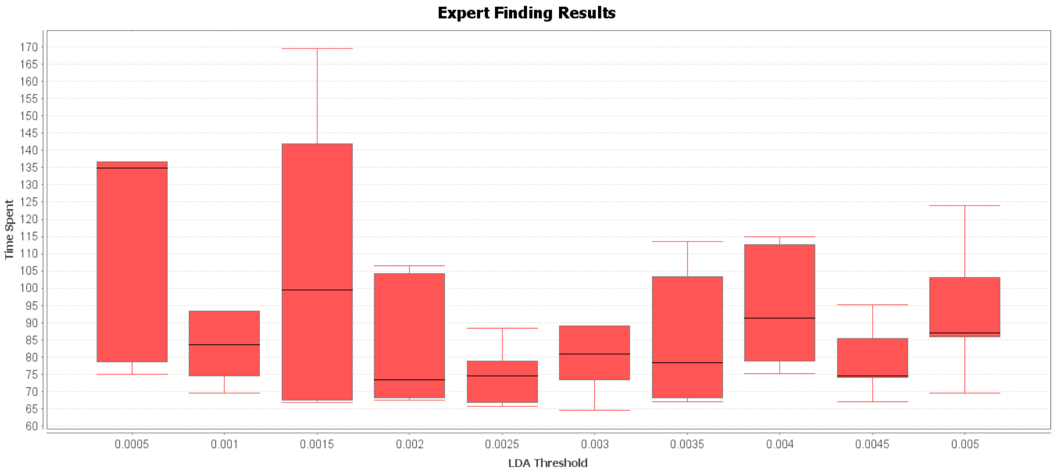

Figure G.3: LDA Threshold Time Spent Box Plot for UVT

The parameters of Figure G.3 are as follows:

- **Dataset**: UVT
- **LDA Alpha**: 50.0
- **LDA Beta**: 0.01
- **LDA Iteration Count**: 100
- **LDA Threshold**: 0.0005 - 0.0050 (x axis values)
- **Weirdness Threshold**: 1.0
- **Stemmed**: false
- **ADT Type**: United
- **Topic Count**: 200
- **Method**: ProductPaths
- **Number of Results**: 50
- **Output**: Time Spent
- **Temporality**: NONE
- **LDA Type**: Normal
- **Experiment No**: 1 – 5 (varies in box plot)

APPENDIX H

RESULTS OF LDA ITERATION COUNT CONTROLLED EXPERIMENTS

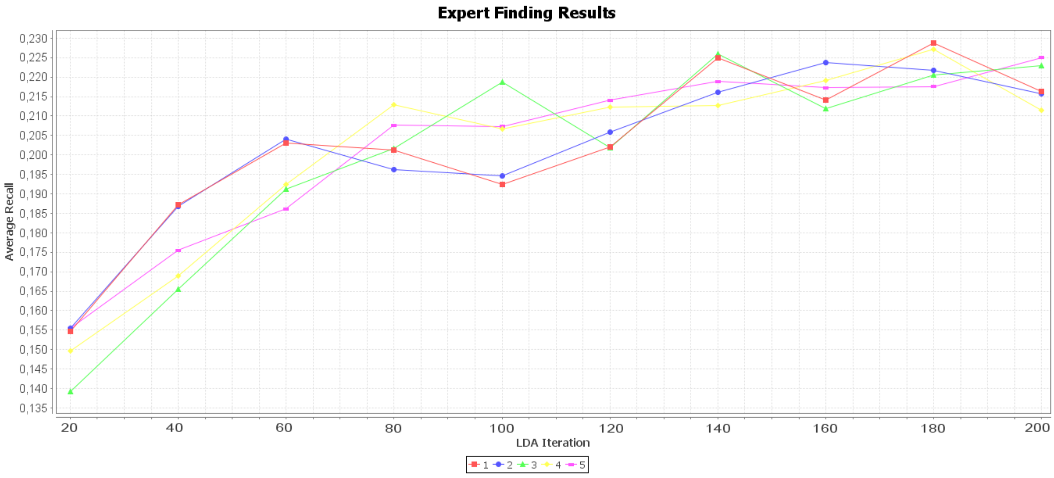

Figure H.1: LDA Iteration Count Results for UVT

The parameters of Figure H.1 are as follows:

- **Dataset**: UVT
- **LDA Alpha**: 50.0
- **LDA Beta**: 0.01
- **LDA Iteration Count**: 20 - 200 (x axis values)
- **LDA Threshold**: 0.005
- **Weirdness Threshold**: 1.0
- **Stemmed**: false
- **ADT Type**: United
- **Topic Count**: 200
- **Method**: ProductPaths
- **Number of Results**: 50

- **Output**: Average Recall
- **Temporality**: NONE
- **LDA Type**: Normal
- **Experiment No**: 1 – 5 (varies in plot)

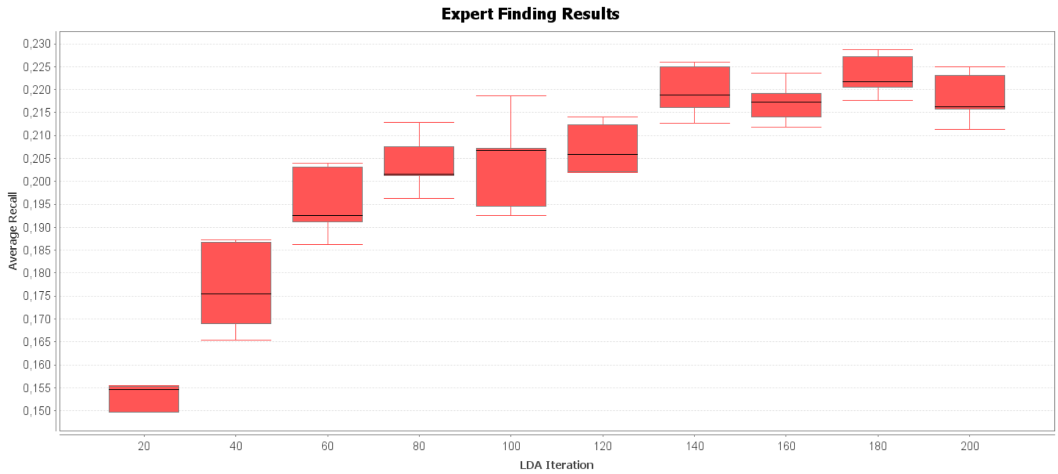

Figure H.2: LDA Iteration Count Box Plot for UVT

The parameters of Figure H.2 are as follows:

- **Dataset**: UVT
- **LDA Alpha**: 50.0
- **LDA Beta**: 0.01
- **LDA Iteration Count**: 20 - 200 (x axis values)
- **LDA Threshold**: 0.005
- **Weirdness Threshold**: 1.0
- **Stemmed**: false
- **ADT Type**: United
- **Topic Count**: 200
- **Method**: ProductPaths
- **Number of Results**: 50
- **Output**: Average Recall
- **Temporality**: NONE
- **LDA Type**: Normal
- **Experiment No**: 1 – 5 (varies in box plot)

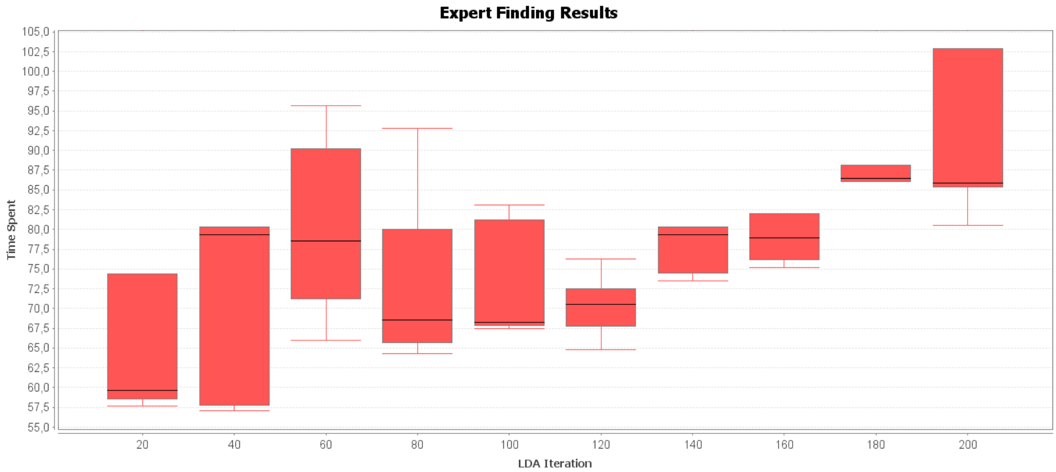

Figure H.3: LDA Iteration Count Time Spent Box Plot for UVT

The parameters of Figure H.3 are as follows:

- **Dataset**: UVT
- **LDA Alpha**: 50.0
- **LDA Beta**: 0.01
- **LDA Iteration Count**: 20 - 200 (x axis values)
- **LDA Threshold**: 0.005
- **Weirdness Threshold**: 1.0
- **Stemmed**: false
- **ADT Type**: United
- **Topic Count**: 200
- **Method**: ProductPaths
- **Number of Results**: 50
- **Output**: Time Spent
- **Temporality**: NONE
- **LDA Type**: Normal
- **Experiment No**: 1 – 5 (varies in box plot)

APPENDIX I

RESULTS OF STEMMING CONTROLLED EXPERIMENTS

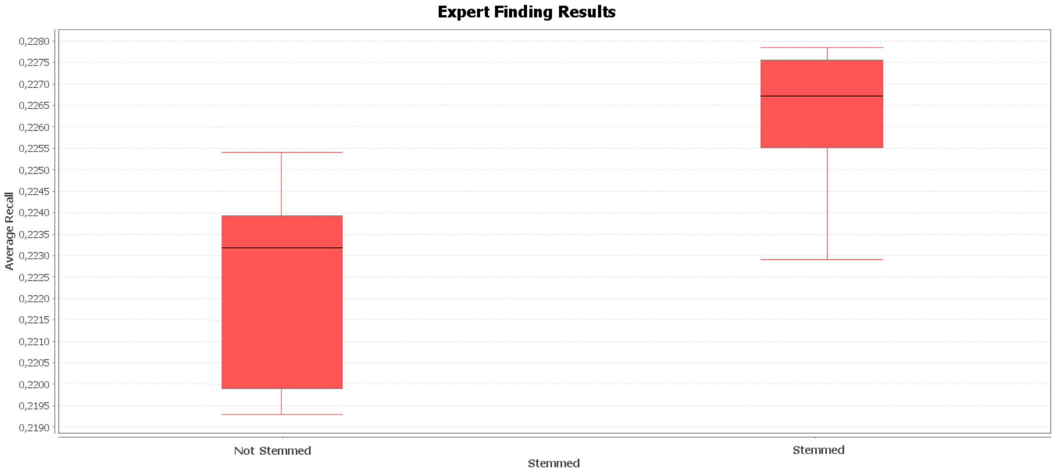

Figure I.1: Stemming Box Plot for UVT

The parameters of Figure I.1 are as follows:

- **Dataset**: UVT
- **LDA Alpha**: 50.0
- **LDA Beta**: 0.01
- **LDA Iteration Count**: 100
- **LDA Threshold**: 0.005
- **Weirdness Threshold**: 1.0
- **Stemmed**: true, false (x axis values)
- **ADT Type**: United
- **Topic Count**: 200
- **Method**: ProductPaths
- **Number of Results**: 50
- **Output**: Average Recall

- **Temporality**: NONE
- **LDA Type**: Normal
- **Experiment No**: 1 – 5 (varies in box plot)

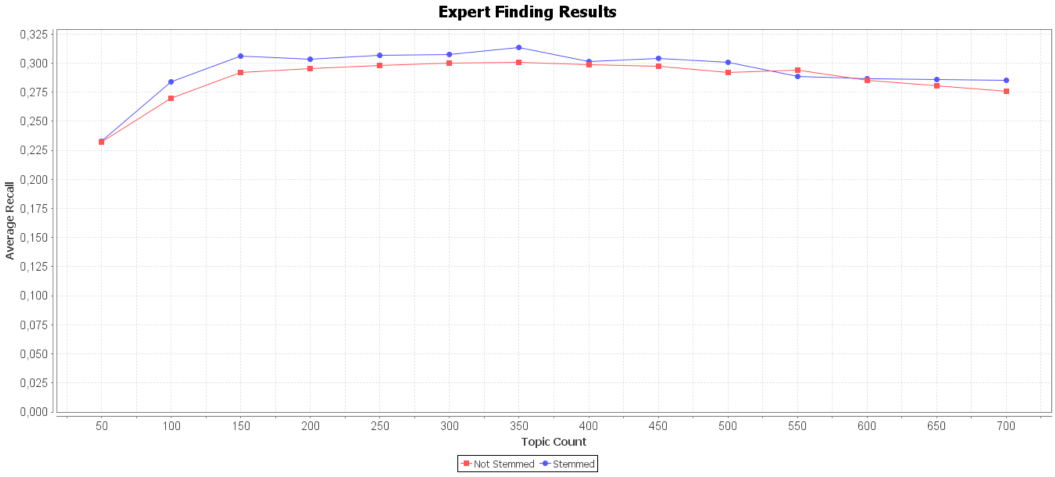

Figure I.2: Stemming Box Plot for UVT

The parameters of Figure I.2 are as follows:

- **Dataset**: UVT
- **LDA Alpha**: 50.0
- **LDA Beta**: 0.01
- **LDA Iteration Count**: 100
- **LDA Threshold**: 0.005
- **Weirdness Threshold**: 1.0
- **Stemmed**: true, false (varies in plot)
- **ADT Type**: United
- **Topic Count**: 50 - 700 (x axis values)
- **Method**: ProductPaths
- **Number of Results**: 50
- **Output**: Average Recall
- **Temporality**: NONE
- **LDA Type**: Normal
- **Experiment No**: 1 – 5 (average of all)

Figure I.3: Stemming Box Plot for ArnetMiner

The parameters of Figure I.3 are as follows:

- **Dataset**: ArnetMiner
- **LDA Alpha**: 50.0
- **LDA Beta**: 0.01
- **LDA Iteration Count**: 100
- **LDA Threshold**: 0.005
- **Weirdness Threshold**: 1.0
- **Stemmed**: true, false (x axis values)
- **ADT Type**: United
- **Topic Count**: 200
- **Method**: ProductPaths
- **Number of Results**: 50
- **Output**: Average Recall
- **Temporality**: NONE
- **LDA Type**: Normal
- **Experiment No**: 1 – 5 (varies in box plot)

APPENDIX J

RESULTS OF TEMPORALITY CONTROLLED EXPERIMENTS

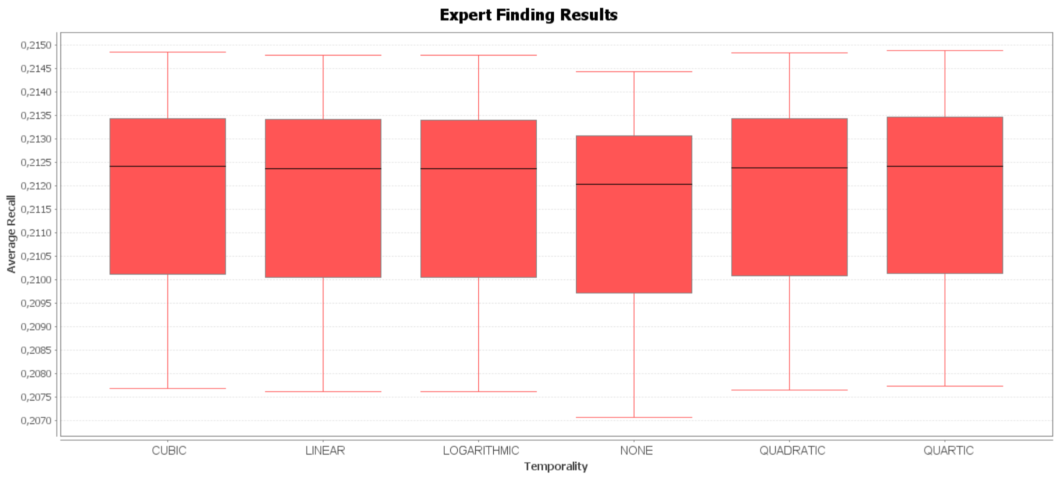

Figure J.1: Temporality Box Plot for UVT

The parameters of Figure J.1 are as follows:

- **Dataset**: UVT
- **LDA Alpha**: 50.0
- **LDA Beta**: 0.01
- **LDA Iteration Count**: 100
- **LDA Threshold**: 0.005
- **Weirdness Threshold**: 1.0
- **Stemmed**: false
- **ADT Type**: United
- **Topic Count**: 50 – 700 (average of all)
- **Method**: MaxPath, SumPaths, ProductPaths (average of all)
- **Number of Results**: 10 – 50 (average of all)

- **Output**: Average Recall
- **Temporality**: None, Logarithmic, Linear, Quadratic, Cubic, Quartic (x axis values)
- **LDA Type**: Normal
- **Experiment No**: 1 – 5 (varies in box plot)

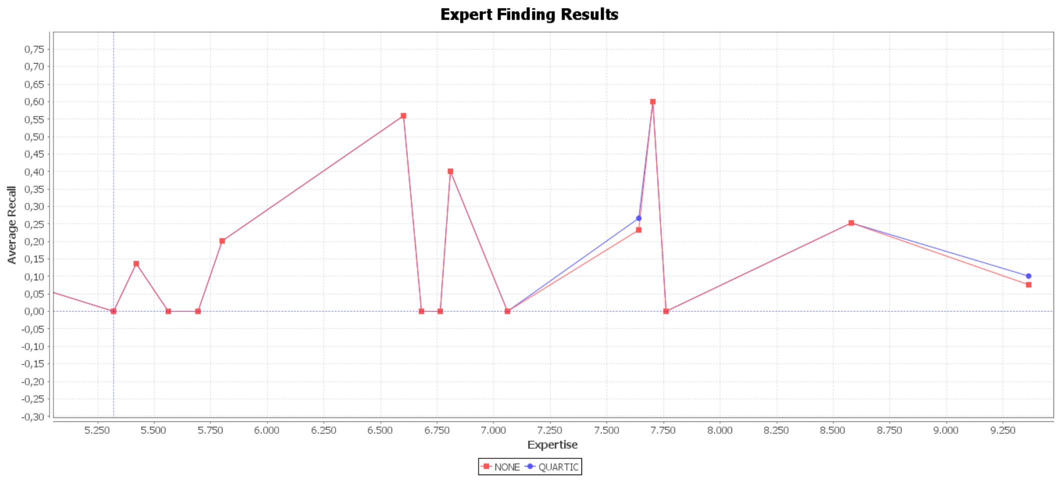

Figure J.2: Expertise Specific Temporality Results For UVT

- **Dataset**: UVT
- **LDA Alpha**: 50.0
- **LDA Beta**: 0.01
- **LDA Iteration Count**: 100
- **LDA Threshold**: 0.005
- **Weirdness Threshold**: 1.0
- **Stemmed**: false
- **ADT Type**: United
- **Topic Count**: 200
- **Method**: ProductPaths
- **Number of Results**: 10
- **Output**: Average Recall
- **Temporality**: None, Quartic (varies in box plot)
- **LDA Type**: Normal
- **Experiment No**: 1
- **Topic Query ID**: 5301 - 9362 (x axis values)

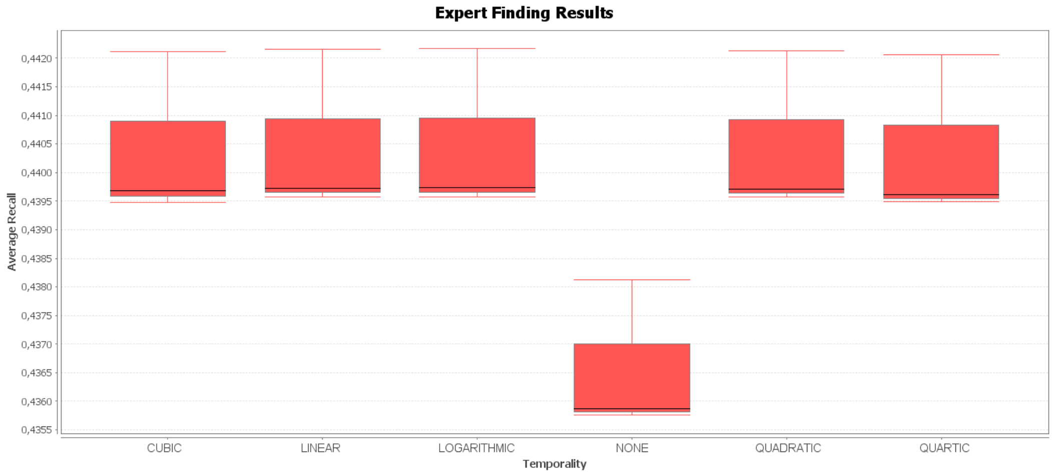

Figure J.3: Temporality Box Plot for ArnetMiner

The parameters of Figure J.3 are as follows:

- **Dataset**: ArnetMiner
- **LDA Alpha**: 50.0
- **LDA Beta**: 0.01
- **LDA Iteration Count**: 100
- **LDA Threshold**: 0.005
- **Weirdness Threshold**: 1.0
- **Stemmed**: false
- **ADT Type**: United
- **Topic Count**: 50 – 700 (average of all)
- **Method**: MaxPath, SumPaths, ProductPaths (average of all)
- **Number of Results**: 10 – 50 (average of all)
- **Output**: Average Recall
- **Temporality**: None, Logarithmic, Linear, Quadratic, Cubic, Quartic (x axis values)
- **LDA Type**: Normal
- **Experiment No**: 1 – 5 (varies in box plot)

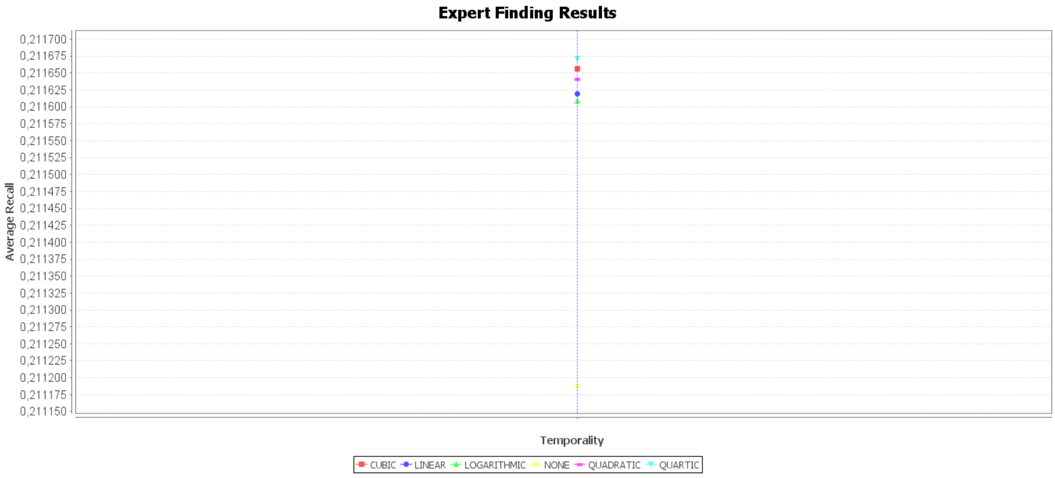

Figure J.4: Temporality Weirdness Based Recall for UVT

The parameters of Figure J.4 are as follows:

- **Dataset**: UVT
- **LDA Alpha**: 50.0
- **LDA Beta**: 0.01
- **LDA Iteration Count**: 100
- **LDA Threshold**: 0.005
- **Weirdness Threshold**: 1.0
- **Stemmed**: false
- **ADT Type**: United
- **Topic Count**: 50 – 700 (average of all)
- **Method**: MaxPath, SumPaths, ProductPaths (average of all)
- **Number of Results**: 10 – 50 (average of all)
- **Output**: Average Recall
- **Temporality**: None, Logarithmic, Linear, Quadratic, Cubic, Quartic (varies in plot)
- **LDA Type**: Normal
- **Experiment No**: 1 – 5 (average of all)

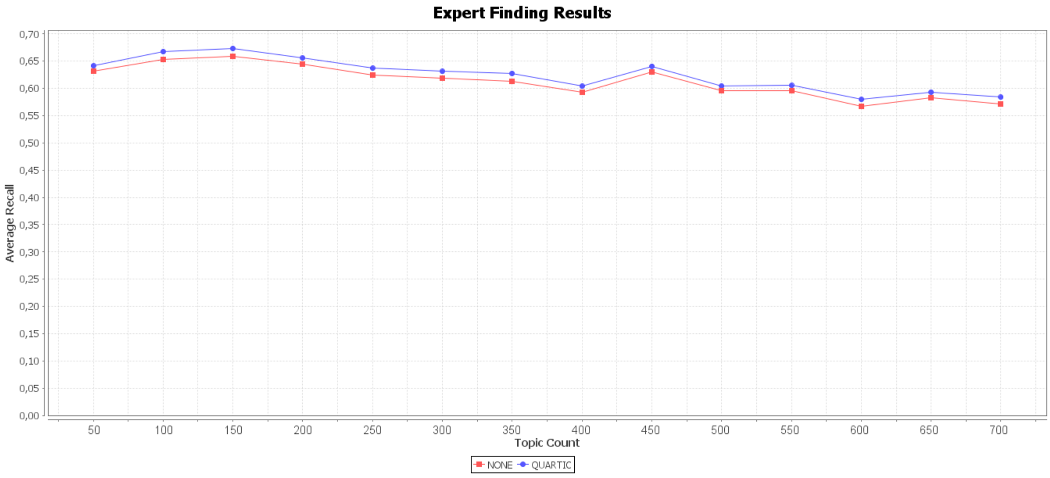

Figure J.5: Temporality Weirdness Based Recall for UVT

The parameters of Figure J.5 are as follows:

- **Dataset**: ArnetMiner
- **LDA Alpha**: 50.0
- **LDA Beta**: 0.01
- **LDA Iteration Count**: 100
- **LDA Threshold**: 0.005
- **Weirdness Threshold**: 1.0
- **Stemmed**: false
- **ADT Type**: United
- **Topic Count**: 50 – 700 (x axis values)
- **Method**: ProductPaths
- **Number of Results**: 50
- **Output**: Average Recall
- **Temporality**: None, Quartic (varies in plot)
- **LDA Type**: Normal
- **Experiment No**: 1 – 5 (average of all)

APPENDIX K

RESULTS OF WEIRDNESS CONTROLLED EXPERIMENTS

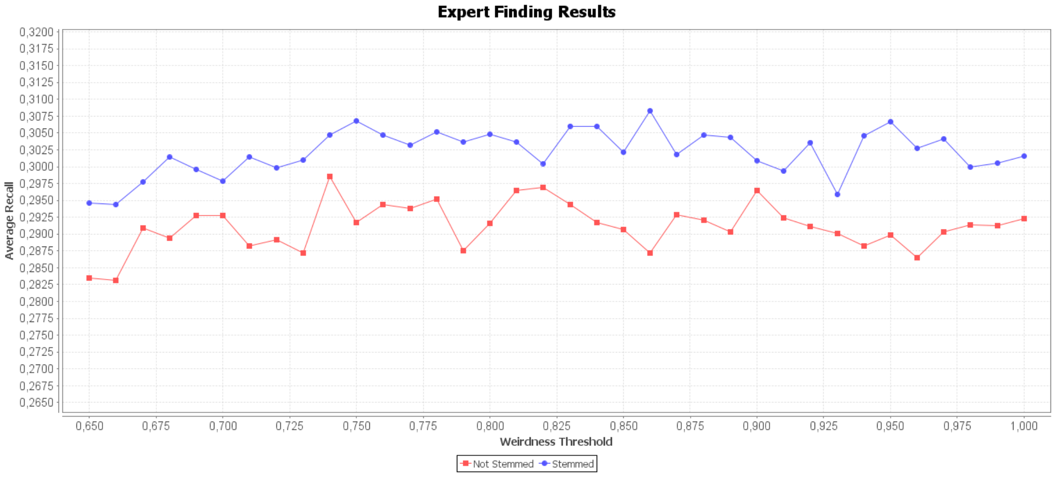

Figure K.1: Weirdness / Stemming based Results for UVT

The parameters of Figure K.1 are as follows:

- **Dataset**: UVT
- **LDA Alpha**: 50.0
- **LDA Beta**: 0.01
- **LDA Iteration Count**: 100
- **LDA Threshold**: 0.005
- **Weirdness Threshold**: 0.65 - 1.0 (x axis values)
- **Stemmed**: true, false (varies in plot)
- **ADT Type**: United
- **Topic Count**: 200
- **Method**: MaxPath, SumPaths, ProductPaths (average of all)
- **Number of Results**: 50
- **Output**: Average Recall

- **Temporality**: NONE
- **LDA Type**: Normal
- **Experiment No**: 1

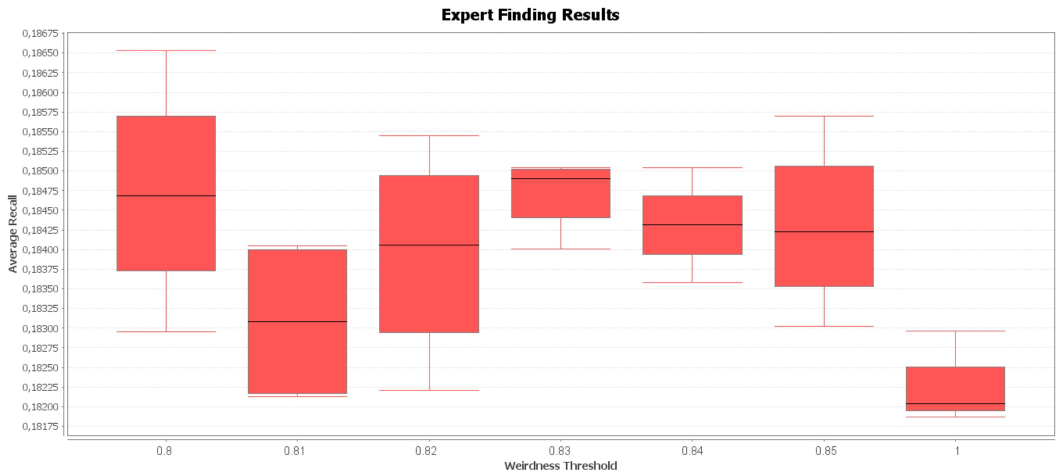

Figure K.2: Weirdness Box Plot for UVT

The parameters of Figure K.2 are as follows:

- **Dataset**: UVT
- **LDA Alpha**: 50.0
- **LDA Beta**: 0.01
- **LDA Iteration Count**: 100
- **LDA Threshold**: 0.005
- **Weirdness Threshold**: 0.80 - 0.85, 1.0 (x axis values)
- **Stemmed**: true
- **ADT Type**: United
- **Topic Count**: 200
- **Method**: MaxPath, SumPaths, ProductPaths (average of all)
- **Number of Results**: 50
- **Output**: Average Recall
- **Temporality**: NONE
- **LDA Type**: Normal
- **Experiment No**: 1 – 5 (varies in box plot)

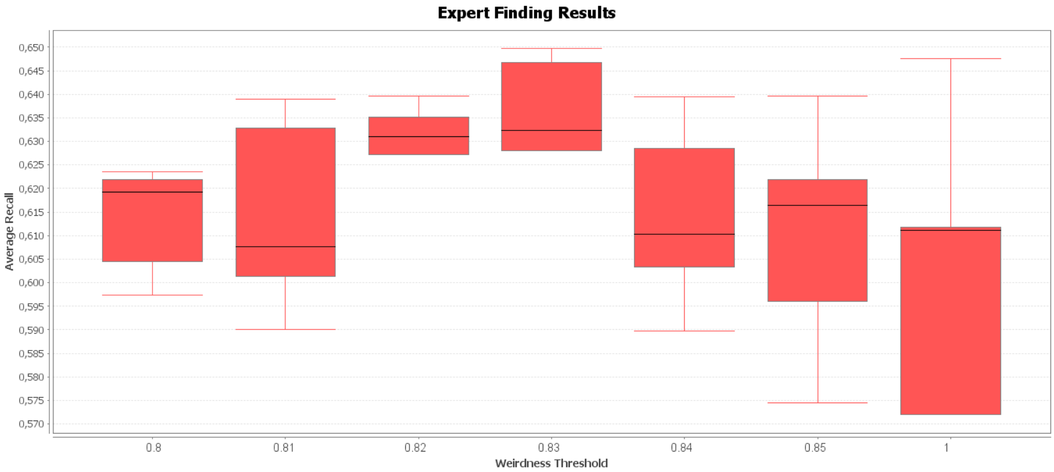

Figure K.3: Weirdness Box Plot for ArnetMiner

The parameters of Figure K.3 are as follows:

- **Dataset**: ArnetMiner
- **LDA Alpha**: 50.0
- **LDA Beta**: 0.01
- **LDA Iteration Count**: 100
- **LDA Threshold**: 0.005
- **Weirdness Threshold**: 0.80 - 0.85, 1.0 (x axis values)
- **Stemmed**: true
- **ADT Type**: United
- **Topic Count**: 200
- **Method**: MaxPath, SumPaths, ProductPaths (average of all)
- **Number of Results**: 50
- **Output**: Average Recall
- **Temporality**: NONE
- **LDA Type**: Normal
- **Experiment No**: 1 – 5 (varies in box plot)

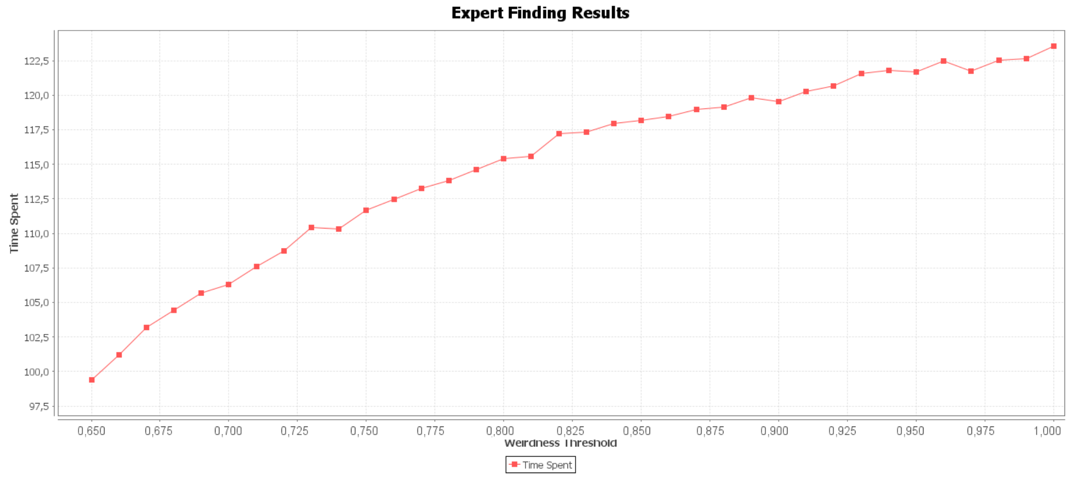

Figure K.4: Weirdness Time Spent Results for UVT

The parameters of Figure K.4 are as follows:

- **Dataset**: UVT
- **LDA Alpha**: 50.0
- **LDA Beta**: 0.01
- **LDA Iteration Count**: 100
- **LDA Threshold**: 0.005
- **Weirdness Threshold**: 0.65 - 1.0 (x axis values)
- **Stemmed**: true
- **ADT Type**: United
- **Topic Count**: 200
- **Method**: MaxPath, SumPaths, ProductPaths (average of all)
- **Number of Results**: 50
- **Output**: Time Spent
- **Temporality**: NONE
- **LDA Type**: Normal
- **Experiment No**: 1

APPENDIX L

RESULTS OF SPARSITY CONTROLLED EXPERIMENTS

Figure L.1: Sparsity Box Plot for UVT

The parameters of Figure L.1 are as follows:

- **Dataset**: ArnetMiner
- **LDA Alpha**: 50.0
- **LDA Beta**: 0.01
- **LDA Iteration Count**: 100
- **LDA Threshold**: 0.005
- **Weirdness Threshold**: 1.0
- **Stemmed**: true
- **ADT Type**: United
- **Topic Count**: 200
- **Method**: ProductPaths
- **Number of Results**: 50
- **Output**: Average Recall

- **Temporality**: NONE
- **LDA Type**: Normal
- **Sparsity**: SPARSE, NORMAL, HEAVY (x axis values)
- **Experiment No**: 1 – 5 (varies in box plot)

Figure L.2: Sparsity Time Spent Box Plot for UVT

The parameters of Figure L.2 are as follows:

- **Dataset**: ArnetMiner
- **LDA Alpha**: 50.0
- **LDA Beta**: 0.01
- **LDA Iteration Count**: 100
- **LDA Threshold**: 0.005
- **Weirdness Threshold**: 1.0
- **Stemmed**: true
- **ADT Type**: United
- **Topic Count**: 200
- **Method**: ProductPaths
- **Number of Results**: 50
- **Output**: Time Spent
- **Temporality**: NONE
- **LDA Type**: Normal
- **Sparsity**: SPARSE, NORMAL, HEAVY (x axis values)
- **Experiment No**: 1 – 5 (varies in box plot)

APPENDIX M

RESULTS OF LDA TYPE (HIERARCHICAL / NORMAL) CONTROLLED EXPERIMENTS

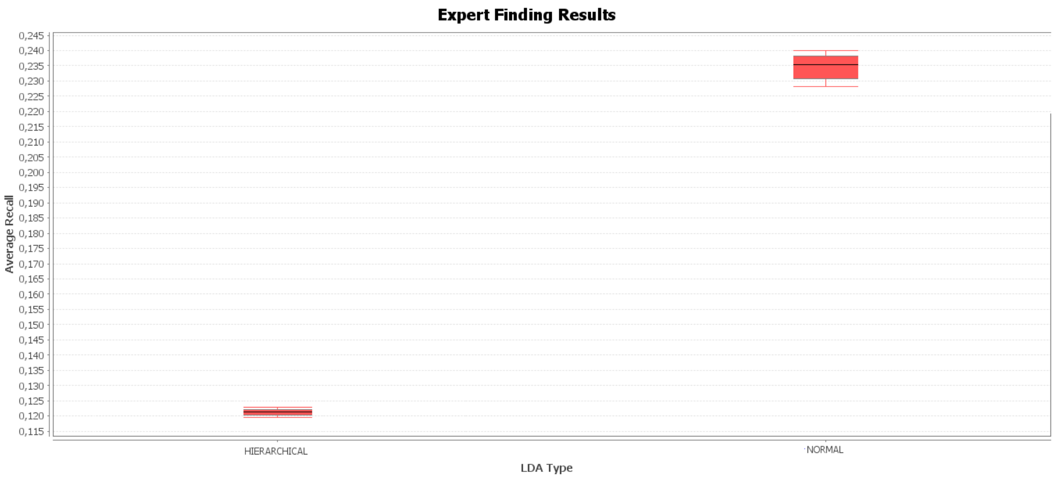

Figure M.1: LDA Type Box Plot for UVT

The parameters of Figure M.1 are as follows:

- **Dataset**: UVT
- **LDA Alpha**: 50.0
- **LDA Beta**: 0.01
- **LDA Iteration Count**: 100
- **LDA Threshold**: 0.005
- **Weirdness Threshold**: 1.0
- **Stemmed**: false
- **ADT Type**: United
- **Topic Count**: 200
- **Method**: ProductPaths
- **Number of Results**: 50

- **Output**: Average Recall
- **Temporality**: None
- **LDA Type**: Normal, Hierarchical (x axis values)
- **Number of levels**: 5
- **Level Coefficient**: 0.5
- **Experiment No**: 1 – 7 (varies in box plot)

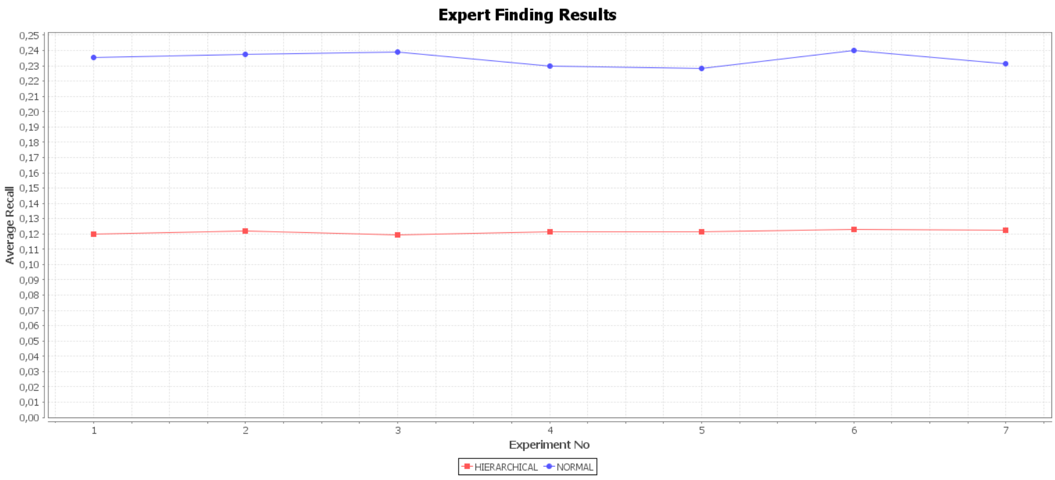

Figure M.2: LDA Type Results for UVT

The parameters of Figure M.2 are as follows:

- **Dataset**: UVT
- **LDA Alpha**: 50.0
- **LDA Beta**: 0.01
- **LDA Iteration Count**: 100
- **LDA Threshold**: 0.005
- **Weirdness Threshold**: 1.0
- **Stemmed**: false
- **ADT Type**: United
- **Topic Count**: 200
- **Method**: ProductPaths
- **Number of Results**: 50
- **Output**: Average Recall
- **Temporality**: None
- **LDA Type**: Normal, Hierarchical (varies in plot)
- **Number of levels**: 5

- **Level Coefficient**: 0.5
- **Experiment No**: 1 – 7 (x axis values)

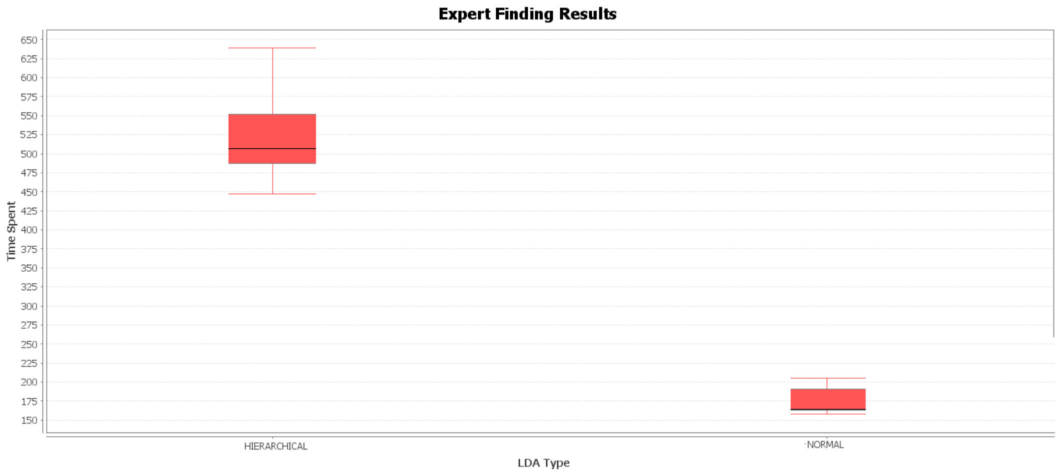

Figure M.3: LDA Type Time Spent Box Plot for UVT

The parameters of Figure M.3 are as follows:

- **Dataset**: UVT
- **LDA Alpha**: 50.0
- **LDA Beta**: 0.01
- **LDA Iteration Count**: 100
- **LDA Threshold**: 0.005
- **Weirdness Threshold**: 1.0
- **Stemmed**: false
- **ADT Type**: United
- **Topic Count**: 200
- **Method**: ProductPaths
- **Number of Results**: 50
- **Output**: Time Spent
- **Temporality**: None
- **LDA Type**: Normal, Hierarchical (x axis values)
- **Number of levels**: 5
- **Level Coefficient**: 0.5
- **Experiment No**: 1 – 7 (varies in box plot)

Table M.1: Example Hierarchical Topics for UVT

ID	**ID**	**ID**	**ID**	**ID**	**Most Used Words**
0					models perspective technology analysis problems study governance evidence business change
	1				social systems law based review policy information structure political research
		2			management contents information theory international attention effects study related policy
			3		aw students cultural analysis european social data economics international research
				4	students language contents theory research empirical topics concepts law specifics
					...
					...
				11418	electronic physics electroencephalogram physiology unpaid volunteer triad patents israeli deconstruction
				14474	physics electroencephalogram physiology unpaid volunteer triad patents israeli deconstruction scandinavia
				14648	physics electroencephalogram physiology unpaid volunteer triad patents israeli deconstruction scandinavia

APPENDIX N

INDRI SEARCH ENGINE INDEX COMPARISON

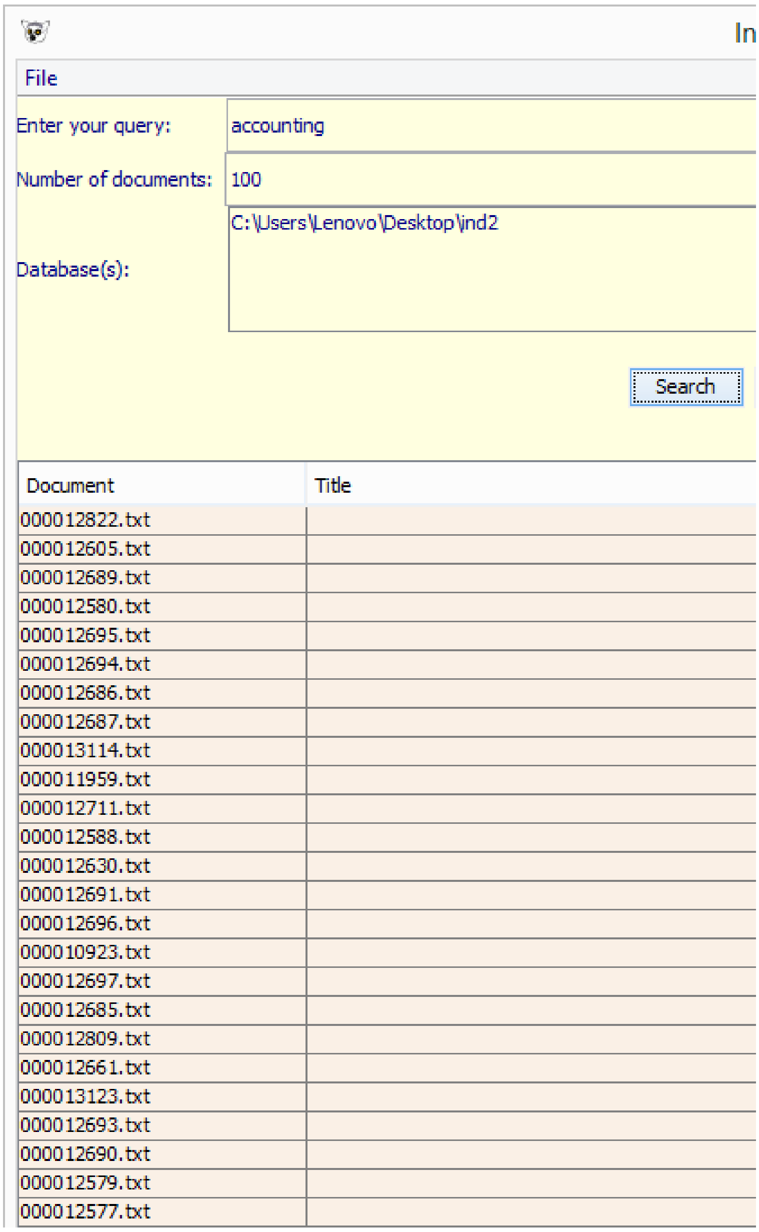

Figure N.1: Indri First 25 Indices for Topic 1276 (accounting)

Table N.1: First 25 Document Indices for Topic 1276 (accounting)

Topic Query ID	**Document Id**	**Score**
1276	12822	13
1276	12605	13
1276	12689	12
1276	12580	11
1276	12695	11
1276	12694	9
1276	12686	8
1276	12687	5
1276	13115	5
1276	12711	4
1276	12588	4
1276	12685	3
1276	11959	3
1276	12691	3
1276	12630	3
1276	12809	2
1276	12661	2
1276	12688	2
1276	12697	2
1276	13124	2
1276	12696	2
1276	10923	2
1276	12692	1
1276	12577	1
1276	12579	1

APPENDIX O

QUERIES OF UVT DATASET

Table O.1: Topic Queries of UVT Dataset

Query ID	Topic Query (Expertise)	Query ID	Topic Query (Expertise)
1274	accountancy	1760	social security
1276	accounting	1765	sociolinguistics
1282	general economics	1766	sociology
1285	foreigners	1771	game theory
1296	labour economics	1773	municipal law
1301	labour law	1774	statistics
1302	industrial/labour relations	1775	statistical methods
1327	tax law	1778	law of criminal procedure/criminal adjective law
1330	policy and management	1780	criminal law
1337	information management and technology	1783	strategic decision-making
1341	public administration	1784	syntax
1342	administrative law	1787	systematic/methodical theology
1361	communication	1788	language
1378	consumer behaviour	1789	language analysis
1383	corporate governance	1790	language and artificial intelligence

1392	cultural psychology	1791	language and minorities
1393	cultural sociology	1796	language technology
1395	databases	1797	language acquisition
1398	democracy	1799	language studies/science
1408	dogmatic theology	1803	discourse studies
1413	econometrics	1804	theology
1414	economics	1812	bilingualism and multilingualism
1418	psychology of economic behaviour	1816	corporate income tax
1430	english	1830	inventory management
1432	ethics	1833	leisure
1441	european law	1843	invalidity benefits shortfall/wao shortfall
1449	experimental economics	1852	mathematics
1450	expert systems	1929	emotions
1459	philosophy	1951	multicultural society
1462	philosophy of mind	1985	turkish migrants
1467	financial markets	2052	brain and behaviour
1470	finance	2061	quality of life
1488	behaviour	2066	stress and disease
1490	memory	2072	internet use by consumers
1495	history	2101	psychophysiology
1499	health psychology	2124	personality and health
1504	religion	2146	group processes
1524	computer science	2153	innovation
1525	computer science law	2160	heart and vascular diseases
1526	information management	2186	rule of law
1527	information law	2201	labour market

1528	information systems	2206	stress
1529	information technology	2226	personality
1533	income tax	2264	human resource management
1534	insolvency law	2266	organisational change
1540	intercultural communication	2293	shame
1546	private international law	2311	property rights
1548	international law	2341	knowledge management
1549	international criminal law	2517	poverty
1552	internet	2536	taxes
1566	knowledge representation	2542	decision theory
1567	knowledge technology	2577	civil society
1575	artificial intelligence	2598	computer simulation
1582	leadership	2625	cooperation
1585	literature	2636	data mining
1586	study of literature	2643	decision support systems
1589	logistics	2670	e-commerce
1590	logistics management	2680	non profit sector
1593	power	2689	economic growth
1595	macroeconomics	2787	health care
1596	management	2792	globalisation
1597	management accounting	2814	ict
1599	marketing	2815	ict in education
1603	market research	2941	machine learning
1607	competition law	2994	multilingualism
1608	media	3013	environmental economics
1610	human rights	3095	operations research
1614	microeconomics	3099	optimisation
1618	migrants	3104	organisation theory

1619	environmental policy	3107	outsourcing
1629	minorities	3130	programming
1634	moral theology	3185	speech technology
1639	dutch as a second language	3188	talking computer
1640	dutch for foreigners	3204	strategic management
1641	networks	3259	telecommnications
1652	corporate finance	3281	bilingualism
1653	company law	3382	sustainable development
1658	methodology and statistics	3466	culture
1674	organisational sociology	3721	liability law
1678	government policy	3901	man-machine interaction
1695	politics	3937	reasoning
1697	political philosophy / political ethics	3983	legal skills
1701	practical theology	4032	securities law
1702	private law	4165	identity
1704	privacy	4227	face recognition
1705	procedural law	4233	social capital
1711	psycholinguistics	4310	web-based application design
1712	psychology	5321	jurisprudence
1714	psychonomics	5421	entrepreneurship
1717	public sector	5564	victimology
1719	law and informatization/-computerization	5695	programming languages
1721	philosophy of law	5801	computer linguistics
1722	history of law	6603	labour market policy
1726	legal theory	6681	cognitive processes
1727	comparative law	6762	organisation studies

1743	semantics	6807	decision making
1745	simulation	7061	elderly people
1746	social law	7642	care
1754	social philosophy / social ethics	7703	brain
1758	social politics	7761	depression
1759	social psychology	8581	inter-organisational relationships
		9362	anxiety

APPENDIX P

QUERIES OF ARNETMINER DATASET

Table P.1: Topic Queries of ArnetMiner Dataset

Query ID	**Topic Query (Expertise)**
1	Ontology Alignment
2	Semantic Web
3	Data Mining
4	Information Extraction
5	Boosting
6	Support Vector Machine
7	Planning
8	Intelligent Agents
9	Machine Learning
10	Natural Language Processing
11	Cryptography
12	Computer Vision
13	Neural Networks

TEZ FOTOKOPİ İZİN FORMU

ENSTİTÜ

Fen Bilimleri Enstitüsü ☐

Sosyal Bilimler Enstitüsü ☐

Uygulamalı Matematik Enstitüsü ☐

Enformatik Enstitüsü X

Deniz Bilimleri Enstitüsü ☐

YAZARIN

Soyadı : Kılınç
Adı : Ahmet Emre
Bölümü : Bilişim Sistemleri

TEZİN ADI (İngilizce) : A TEMPORAL EXPERT FINDING METHODOLOGY BASED ON UNITED AUTHOR-DOCUMENT-TOPIC GRAPHS

TEZİN TÜRÜ : Yüksek Lisans X Doktora ☐

1. Tezimin tamamı dünya çapında erişime açılsın ve kaynak gösterilmek şartıyla tezimin bir kısmı veya tamamının fotokopisi alınsın. X
2. Tezimin tamamı yalnızca Orta Doğu Teknik Üniversitesi kullanıcılarının erişimine açılsın. (Bu seçenekle tezinizin fotokopisi ya da elektronik kopyası Kütüphane aracılığı ile ODTÜ dışına dağıtılmayacaktır.) ☐
3. Tezim bir (1) yıl süreyle erişime kapalı olsun. (Bu seçenekle tezinizin fotokopisi ya da elektronik kopyası Kütüphane aracılığı ile ODTÜ dışına dağıtılmayacaktır.) ☐

Yazarın imzası .... Tarih ....