Fall 2014

Retrieval Models based on Linguistic Features of Verbose Queries

Jae Hyun Park
Computer Sciences

RETRIEVAL MODELS BASED ON LINGUISTIC FEATURES OF VERBOSE QUERIES

A Dissertation Presented

by

JAE HYUN PARK

Submitted to the Graduate School of the
University of Massachusetts Amherst in partial fulfillment
of the requirements for the degree of

DOCTOR OF PHILOSOPHY

September 2014

Computer Science

RETRIEVAL MODELS BASED ON LINGUISTIC FEATURES OF VERBOSE QUERIES

A Dissertation Presented

by

JAE HYUN PARK

Approved as to style and content by:

W. Bruce Croft, Chair

David A. Smith, Member

James Allan, Member

Rajesh Bhatt, Member

Lori A. Clarke, Chair
Computer Science

To my family

ACKNOWLEDGMENTS

I would like to express the deepest appreciation to my advisor, Professor W. Bruce Croft for his many years of thoughtful, patient guidance and support. I appreciate all his contributions of time, ideas, and funding to make my Ph.D. experience productive and stimulating. The insight and enthusiasm he has for information retrieval was contagious and motivational for me. Without his guidance and persistent help, this dissertation would not have been possible. I would like to thank my committee members, Professor David A. Smith and Professor James Allan whose work is another key idea of dissertation. I also appreciate their helpful comments and suggestions on all stages of my Ph.D.

I wish to express my appreciation to all the staffs of the CIIR and CS, Daniel Parker David Fisher, Jean C. Joyce, Kate Moruzzi and Leeanne M. Leclerc for their assistance. A special thank you to Laura Dietz , Ethem Can, Jeff Dalton, Van Dang, Shiri Dori-Hacohen, Sam Huston, Youngho Kim, Kriste Krstovski, CJ Lee, Weize Kong, Laura Lara, Katerina Marazopoulou, Yeun-sup Lim and all my friends at UMass who are the most nice and kind people in the world.

This work was supported in part by the Center for Intelligent Information Retrieval, in part by IBM subcontract #4913003298 under DARPA prime contract #HR001-12-C-0015, in part by NSF IIS-1160894, in part by NSF grant #IIS-0711348, and in part by ARRA NSF IIS-9014442. Any opinions, findings and conclusions or recommendations expressed in this material are those of the authors and do not necessarily reflect those of the sponsor.

ABSTRACT

RETRIEVAL MODELS BASED ON LINGUISTIC FEATURES OF VERBOSE QUERIES

SEPTEMBER 2014

JAE HYUN PARK

B.Sc., KOREA UNIVERSITY, KOREA

M.Sc., KOREA UNIVERSITY, KOREA

Ph.D., UNIVERSITY OF MASSACHUSETTS AMHERST

Directed by: Professor W. Bruce Croft

Natural language expressions are more familiar to users than choosing keywords for queries. Given that, people can use natural language expressions to represent their sophisticated information needs. Instead of listing keywords, verbose queries are expressed in a grammatically well-formed phrase or sentence in which terms are used together to represent the more specific meanings of a concept, and the relationships of these concepts are expressed by function words.

The goal of this thesis is to investigate methods of using the semantic and syntactic features of natural language queries to maximize the effectiveness of search. For this purpose, we propose the synchronous framework in which we use syntactic parsing techniques for modeling term dependencies. We use the Generative Relevance Hypothesis (GRH) to evaluate valid variations in dependence relationships between

queries and documents. This is one of the first results demonstrating that dependency parsing can be used to improve retrieval effectiveness.

We propose a method for classifying concepts in verbose queries as key concepts and secondary concepts that are used in the statistical translation model for query term expansion. Key concepts are the most important terms of queries. We use key concepts as the context for translating terms. Although secondary (key) concepts are not as important as key concepts, they are still important because they provide clues about what kinds of information users are looking for. Using concept classification results, we elaborate a translation model in which the key concepts of queries are used as the context of translation. The secondary concepts of queries are used to selectively apply the translation model to query terms.

We define the important new task of focused retrieval of answer passages that aims to immediately provide answers for users' information needs while the length of answer passage should be suitable for restricted search environments such as mobile devices and voice-based search systems.

TABLE OF CONTENTS

Page

ACKNOWLEDGMENTS . . . . . **v**

ABSTRACT . . . . . **vi**

LIST OF TABLES . . . . . **xii**

LIST OF FIGURES . . . . . **xv**

CHAPTER

1. INTRODUCTION **1**

1.1 Term Dependency in Verbose Queries 4
1.2 Variations in Dependence Relationships 7
1.3 Translation Model for Query Term Expansion 9
1.4 Answer Passage Retrieval for Verbose Queries 10
1.5 Contributions 13
1.6 Dissertation Outline 14

2. BACKGROUND AND RELATED WORK **16**

2.1 Modeling Term Dependencies 16

2.1.1 Selecting Dependent Terms 17
2.1.2 Transformations of Dependence Relationships 21

2.2 Query Term Expansion 23

2.2.1 Translation Models 25

2.3 Weighting Query Terms and Their Dependencies 28

2.3.1 Query Term Weighting 28
2.3.2 Weighting Retrieval Models 30

2.4 Focused Retrieval 33

2.4.1 Question Answering . . . 34
2.4.2 Passage Retrieval . . . 36

2.5 Natural Language Processing . . . 38

2.5.1 Parsing . . . 38
2.5.2 Ontology . . . 41

2.6 Summary . . . 42

3. DATASETS AND EVALUATION . . . 43

3.1 Test Collections . . . 43

3.1.1 TREC Collections . . . 43

3.1.1.1 Document Collections . . . 44
3.1.1.2 Topics and Relevance Judgements . . . 44

3.1.2 INEX collections . . . 46

3.1.2.1 Document Collection . . . 47
3.1.2.2 Topics and Relevance Judgements . . . 47

3.1.3 CQA Collection . . . 48

3.2 Evaluation Metrics . . . 50

3.2.1 Document-level Evaluation Metrics . . . 50
3.2.2 Passage-level Evaluation Measures . . . 53
3.2.3 Inter-Annotator Agreement . . . 54
3.2.4 Statistical Significance Test . . . 55

3.3 Summary . . . 57

4. ANSWER PASSAGE ANNOTATION . . . 58

4.1 Overview . . . 58
4.2 Two Phases of Relevance Judgments . . . 60

4.2.1 Topically Relevant Text . . . 60
4.2.2 Answer Passages . . . 63

4.3 The Process of Answer Passage Annotation . . . 69
4.4 Annotation Results . . . 71
4.5 Summary . . . 73

5. QUASI-SYNCHRONOUS FRAMEWORK 74

5.1 Overview 74
5.2 Quasi-Synchronous Framework 76

5.2.1 Quasi-Synchronous Stochastic Process 79
5.2.2 Loose Alignment Model for Quasi-Synchronous Framework 83

5.3 Predicting Optimal Parameter Settings for the Quasi-Synchronous Framework 86

5.3.1 Experimental Settings 90

5.4 Experimental Results and Analysis 91

5.4.1 Coverage of Dependent Term Pairs 91
5.4.2 Four Interpolation Strategies using the Loose Alignment Model 93
5.4.3 Exact Matching vs. Quasi Matching 95
5.4.4 Analysis By Query Length 97

5.5 Summary 100

6. MODELING VARIATIONS IN DEPENDENCE RELATIONSHIPS 102

6.1 Overview 102
6.2 Modeling Variations of Dependence Relationships 106

6.2.1 Linked Dependencies of the BIM 106
6.2.2 Generative Relevance Hypothesis of Dependence Relationship 107

6.3 Predicting Valid Variations for the Quasi-Synchronous Framework 111

6.3.1 Predicting Valid Variations of Dependence Relationships 112

6.4 Experiments and Analysis 113

6.4.1 Experimental Settings 113
6.4.2 Retrieval Performance Evaluation 114
6.4.3 Evaluating Statistical Significance Test Methods 118

6.5 Summary 119

7. CONTEXT-BASED TRANSLATION MODEL FOR QUERY TERM EXPANSION 121

7.1 Overview 121
7.2 Statistical Translation Model 124

7.2.1 Context-based Translation Model 127

7.3 Identifying Key Concepts for the Context-based Translation Model 130

7.3.1 Identifying Key Concepts 131
7.3.2 Features 131

7.4 Experiments and Analysis 133

7.4.1 Experimental Settings 133
7.4.2 Answer Retrieval 133

7.5 Summary 135

8. ANSWER PASSAGE RETRIEVAL 137

8.1 Overview 137
8.2 Passage Retrieval Model 139
8.3 Translation Model based on Conditional Mutual Information 141
8.4 Experiments and Analysis 144

8.4.1 Experimental Settings 144
8.4.2 Multi-Level Passage Retrieval Model 145
8.4.3 Evaluation of Query Term Expansion 149

8.5 Summary 153

9. SUMMARY AND FUTURE WORK 154

9.1 Conclusion and Contributions 154
9.2 Future Work 156

BIBLIOGRAPHY 159

LIST OF TABLES

Table **Page**

1.1 Example topics from the TREC Robust 2004 collection with title and description queries. *Concepts* are automatically extracted from description queries based on noun phrases. 5

1.2 The example of different expressions for the concept "chemical weapon" in relevant documents. 7

2.1 The example of question series in the TREC QA 2007 track. 35

3.1 Summary of the TREC collections for evaluating document retrieval. ***Aver. Length*** is the average length of documents in word. 44

3.2 Summary of the TREC topics for the Robust 2004, Gov2 and ClueWeb-B collections. ***Aver. Length*** is the average length of description queries in words. Numbers in parenthesis of are the number of relevant documents 45

3.3 Examples of "title", "description" and "narrative" queries from the TREC Robust 2004 collection. 46

3.4 Summary of the test collection of the INEX Ad Hoc track 2009 and 2010. Numbers in parenthesis are the numbers of XML elements or passages. 47

3.5 Example topics of the INEX Ad Hoc track. 48

4.1 Statistics of annotation results. *Relevant* and *Answer* represent relevant text fragments and answer passages, respectively. # *Relevant* is the number of relevant items in each unit. *Length* is the sum of words in relevant units. 71

5.1 Experimental results with the Robust 2004 with four interpolation strategies. Numbers in parentheses depict % improvement over the sequential dependence model. 94

5.2 Mean Average Precision of the Robust 2004 collection when we use the four interpolation strategies using the true optimal weights of the training data. In the third column, $ExactMatching$ is used for the experiments instead of $QuasiSync$. 96

5.3 Experimental results with the Gov2 collection based on an initial document set retrieved by the sequential dependence model. Numbers in parentheses depict % improvement in each evaluation measure. 98

5.4 Comparison the sequential dependence model, SDM, and $SDM + QuasiSync$. Statistics are collected from the experiments with the Robust 2004 collection. ***# queries*** is the number of queries belong to each group and ***query length*** is the average length of the queries. 98

5.5 Experimental results with the Robust 2004 according to the length of queries. *Length* is the number of terms in a query and *# queries* is the number of queries belonging to each group. Numbers in parentheses depict % improvement in each evaluation measure. 99

6.1 Retrieval effectiveness comparison with all the baselines using the mean average precision and the R-Precision. Numbers in parentheses depict % improvement over the sequential dependence model. 115

6.2 The effectiveness of evaluating the variations of dependence relationships according to statistical significance test approaches using true relevant documents. H and $\hat{H}$ are described in Eq. 6.3 and Eq. 6.4, respectively. 118

7.1 The translation results of "*grow*" with different contexts (Bold-faced). ***PLANT*** represents a WordNet category. 128

7.2 The experimental results of answer retrieval using the CQA collection. **MRR** represents Mean Reciprocal Rank. Numbers in parenthesis are relative improvements over the baseline. Significant differences with *Baseline* and *Translation* are marked by † and ‡, respectively (statistical significance was measured using the two-tailed Wilcoxon test with $p < 0.05$). 134

7.3 The number of question-answer pairs of which retrieval results were unchanged, improved and deteriorated by using the translation-based language model. 135

8.1 The effectiveness of query term expansions for the Per-Document task with top 5 documents. Mean average term precision (MAtP) and normalized discounted cumulative gain (nDCGt) are measured in word. 150

8.2 The effectiveness of query term expansions for the Per-Passage task with top 10 passages. Mean average term precision (MAtP) and normalized discounted cumulative gain (nDCGt) are measured in word. 150

8.3 Experimental results of answer passage retrieval in sentence-level precision at N. **Entire** represents that we treat a retrieval result as correct answer if the entire retrieval result overlapped answer passages. **Partial** assumes that a retrieval result was correct if more than ten percent of the retrieval result overlapped. 151

8.4 The experimental results using the true key concepts of queries for the context-based translation model. 152

LIST OF FIGURES

Figure **Page**

1.1 The example questions with "*grow*" with different contexts "hair" and "flowers" 9

1.2 A topically relevant text fragment and an answer passage in it for the information "*What are imported fire ants, and how can they be controlled?*". The bold-faced text is the answer passage. 12

2.1 Examples of syntactic parsing results. 39

3.1 Screen shot of the Yahoo! Answers service. 49

4.1 The two phases of topically relevant text fragments and answer passages. 59

4.2 A sample document describes multiple subjects about hate crimes. Bold faced text is a relevant text fragment to church arson. 61

4.3 The process of the annotation task. 69

4.4 Screen shot of the annotation toolkit. 70

5.1 Example of the quasi-synchronous alignment (Smith and Eisner, 2006) of the parent-child term pair in the source sentence to six dependence relationships in the target sentence.) 80

5.2 Four types of syntactic dependency configurations in the quasi-synchronous stochastic process. In the quasi-synchronous model matches terms in queries and documents along with transformation between these dependence relations: (a) parent-child, (b) ascendant-descendant, (c) siblings, and (d) c-commanding. 81

5.3 Example queries from the Robust 2004 collection which demonstrating better results when assigning more weight to the query likelihood model, the sequential dependence model and the quasi-synchronous model, respectively. 85

5.4 The distribution of optimal weight $P(syn_Q|T_D)$ in Eq 5.11 according to the length of queries from the Robust 2004 collection. 87

5.5 The ratio of dependent term pairs by the sequential dependence assumption and the quasi-synchronous model based on the predefined syntactic relationships. 92

5.6 Four strategies of linear interpolation with the query-likelihood model(***QL***), the sequential dependence model(***SDM***), and the quasi synchronous model(***QM***). The sequential dependence model interpolates three scores with fixed weights: the query-likelihood score f_{QL}, the ordered window score f_{OR1} and the unordered window score f_{UW8} (Metzler and Croft, 2005). **redraw the figure using a new notation.** 93

6.1 The example text fragments in which the concepts in the TREC query, "*How are young children being protected against lead poisoning from paint and water pipes?*", are used together in a sentence. 103

6.2 The $H_{\mathbb{R}}$ and H_0 of dependence relationships in relevant and non-relevant documents of ("*economy*","*Ireland*") including the their co-occurrences. 109

7.1 The example questions with "*grow*" with different contexts "hair" and "flowers" 122

8.1 The effectiveness comparison of relevant text fragment retrieval of the Gov2 collection according to the interpolation weights. nDCG is measured at the top 5 passages. 146

8.2 The effectiveness comparison of relevant text fragment retrieval of the INEX collection according to the interpolation weights. nDCG is measured at the top 5 passages. 147

8.3 The effectiveness comparison of answer passage retrieval of the Gov2 collection according to the interpolation weights. nDCG is measured at the top 5 passages. 148

CHAPTER 1

INTRODUCTION

In contrast to simple keyword-based queries, users compose verbose queries to express their information needs in detail. Users compose longer and more verbose queries in order to represent sophisticated and specific information needs, and the average query length has increased over time (Bogatin, 2006). This report emphasized that the average query length has increased. Long queries are, however, not always syntactically correct natural language queries (Buccio et al., 2013). For example, consider the following forms of the same query:

Keyword: Airport security.

Verbose: Airport security checkpoints and barriers [1]

Natural Language: A relevant document would discuss how effective government orders to better scrutinize passengers and luggage on international flights and to step up screening of all carry-on baggage has been.

The second and third queries are verbose queries for the first query. Although the second queries contains redundant terms "checkpoints" and "barriers", it is still the list of keywords. There is no explicit information about the relationships between these keywords. It is hard to infer their relationships. As the third query is written in the form of spontaneous natural language expressions, verbose natural language queries not only enumerate query terms, they explicitly represent the relationship

[1]This example query is excerpted from Buccio et al. (2013).

between the query terms using function words. The notion of *verbose natural language queries* in this thesis corresponds to that of *long queries* in (Buccio et al., 2013). For simplicity, we use the expression *verbose query* instead of *verbose natural language queries*.

Although verbose queries are more expressive, they also contain a variety of words and linguistic structures of varying importance relative to describing the query topic. For this reason, some previous research has tried to convert verbose queries to succinct keyword queries by removing less important terms (Balasubramanian et al., 2010a; Huston and Croft, 2010; Kumaran and Allan, 2007; Kumaran and Carvalho, 2009).

Kumaran and Carvalho (2009) observed that, when we use the best subset of words in verbose queries as a keyword query, the effectiveness of a retrieval model was improved almost 30%. Based on a comparative study of query processing techniques, Kumaran and Carvalho proposed a method in which the original verbose queries are replaced with sub-queries using query quality predictors. Huston and Croft (2010) concluded that the most effective approach to reduce the length of queries is by removing stop phrases. For example, when the verbose query "*a relevant document would discuss how effective government orders to better scrutinize passengers and luggage on international flights and to step up screening of all carry-on baggage has been.*" was given, query reduction approaches generate keyword queries such as "*airport security*" or "*airport security checkpoints barriers*" by selecting key concepts and removing less important query terms. These query reduction methods are, however, were unable to exploit the potential of semantic and syntactic information implied by natural language expressions.

In order to maximize the effectiveness of verbose queries, Information Retrieval (IR) systems need to make effective use of the semantic and syntactic features implied by natural language expressions, such as important concepts and their relationships. Natural language expressions are more familiar to users than choosing keywords for

queries. Given that, people can use natural language expressions to represent their sophisticated information needs. In a verbose query, there may be several concepts related to the different aspects of a user's information need. These key concepts from verbose queries has been studied in order to recognize more important query terms (Bendersky and Croft, 2008; Lee et al., 2009). Identified key concepts can be assigned to assign higher weights in the ranking process (Blanco and Lioma, 2012).

Syntactic structures of natural language expressions reveal the dependence relationships between terms and concepts in verbose queries. Although independence assumptions are used to simplify retrieval models in IR, terms are actually dependent upon each other within documents and queries. Terms are used together to express more specific meanings or, sometimes, totally different meanings. Term dependency has been studied for several decades with the aim of improving the effectiveness of information retrieval. Syntactic analysis can be used to recognize term dependencies and dependence relationships in various ways. For example, Gao et al. (2004) proposed a dependence model in which the head-modifier relationships are used to select dependent terms from queries and documents. Bendersky et al. (2009) use phrasal information to segment verbose queries.

While keyword queries suggest the relevant topic of information, verbose queries more explicitly specify details and requirements for the required information. We can use more detailed information needs for focused retrieval in which we can narrow down the text that is relevant. Documents are usually used as retrieval units, which means that users have to read through an entire document to obtain answers for their information needs. Focused retrieval methods have been studied in order to help users locate relevant information among the retrieval results (Allan, 2004). Passage retrieval systems, for example, return a ranked list of relevant text fragments instead of documents. Question-Answering (QA) systems find direct answers for specific classes of questions. As restricted search environments such as smart phones and

voice-based search systems have become more popular, focused retrieval methods have drawn increasing attention.

The aim of this thesis is to investigate methods of using the semantic and syntactic features of verbose queries to maximize the effectiveness of search. We use syntactic analysis results for finding important term dependencies (Section 1.1) and evaluating valid variations in dependence relationships between queries and documents (Section 1.2). This is one of the first results demonstrating that dependency parsing can be used to improve retrieval effectiveness. We also propose a new method for classifying concepts in verbose queries as either key or secondary. The classified concepts are used to selectively apply a translation model for bridging lexical gaps between queries and documents (Section 1.3). We also define the important new task of focused retrieval of answer passages. We apply our proposed methods using dependency parsing and translation methods to this answer passage retrieval task (Section 1.4).

1.1 Term Dependency in Verbose Queries

Term dependency has been studied for several decades in an attempt to improve the effectiveness of information retrieval. Although independence assumptions simplify retrieval models, terms are actually dependent upon each other within documents and within queries. Terms are used together to create more specific meanings. Recent work on retrieval models has achieved significant improvement over the bag-of-words assumption by considering term dependencies in verbose queries (Metzler and Croft, 2005).

There have been two issues in the consideration of dependencies between terms and concepts. The first involves determining the important dependencies between terms and concepts from the query text. The second involves matching these dependencies in document structures in order to estimate the relevance of documents.

Table 1.1. Example topics from the TREC Robust 2004 collection with title and description queries. *Concepts* are automatically extracted from description queries based on noun phrases.

TREC Topic 623	
Title	toxic chemical weapon
Description	Gather any information that mentions ricin, sarin, soman, or anthrax as a toxic chemical used as a weapon.
Concept	gather / any / mentions / ricin / sarin soman anthrax / toxic chemical / used / weapon
TREC Topic 643	
Title	salmon dams Pacific northwest
Description	What harm have power dams in the Pacific northwest caused to salmon fisheries?
Concept	harm / power dams / pacific northwest / caused / salmon fisheries

The current state-of-the-art term dependence model, the sequential dependence model (SDM), assumes that adjacent terms in queries are dependent (Metzler and Croft, 2005). Although adjacent term pairs provide good evidence for recognizing term dependencies such as noun phrases and idioms, they are limited in considering dependence relationships in longer distance. Table 1.1 shows example topics excerpted from the Robust 2004 collection in the Text Retrieval Conference (TREC). Topic 623 requests information about *"toxic chemical weapon"*. The concept *"chemical weapon"* of this topic can be captured from the title query by term dependence models using adjacent term pairs. However, the same concept *"chemical weapon"* in the description query will be ignored because *"chemical"* and *"weapon"* are placed in different clauses.

Alternatively, instead of finding specific dependence relationships from queries, one approach used in previous work treats all pairs of any query terms as dependent terms. For example, while the sequential dependence model of the Markov random field model is based on only adjacent term pairs, the full dependence variant of this model is based on the dependencies from every pair of query terms (Metzler and Croft, 2005). In previous work, there have not been significant differences between the sequential and full dependence models (Metzler and Croft, 2005; Peng et al.,

2007). This is because, while the full dependence model can consider more long-distance query term dependencies than the sequential dependence approach, some dependent terms of the full dependence approach also include incorrect dependencies of unrelated terms.

Natural language processing techniques have been used for term dependence models can also in order to identify the dependence relationships from queries. For a given query in a grammatically well-formed phrase or sentence, segmentation results of verbose queries can provide good evidence of dependent terms forming concepts (Bendersky et al., 2009; Bergsma and Wang, 2007; Croft et al., 1991). In previous work, syntactic parsing results of queries are used to capture dependence relationships beyond adjacent term pairs (Gao et al., 2004; Lee et al., 2006; Song et al., 2008).

However, previous work on dependence models using phrasal boundaries and head-modifier relationships suffices only for recognizing dependencies within a concept because most of previous work use only the head-modifier relationship in parsing trees. For the example queries in Table 1.1, the head-modifier relationship of query terms can encompass the concept *"chemical weapon"* of topic 623. The segmentation method based on noun phrases can separate important concepts, such as *"power dam"*, *"Pacific northwest"* and *"salmon fishery"*, from each other for the topic 643. However, they cannot consider dependenies between these concepts.

We use syntactic parsing techniques to capture more dependence relationships from queries. For this purpose, we propose the quasi-synchronous framework that unifies the quasi-synchronous stochastic process (Smith and Eisner, 2006) with existing retrieval models. The synchronous framework accommodates more diverse variations of dependence relationships in documents. The quasi-synchronous stochastic process have shown significant improvements not only for machine translation but also for paraphrasing (Das and Smith, 2009) and QA (Wang et al., 2007). The quasi-synchronous framework aims to capture term dependencies beyond the head-

Table 1.2. The example of different expressions for the concept "chemical weapon" in relevant documents.

TREC Topic 623	
Description	Gather any information that mentions ricin, sarin, soman, or anthrax as a toxic ***chemical*** used as a ***weapon***.
Documents	... Al-Rabta ***chemical weapons*** plant was uncovered and destroyed in a fire. ...
	... its ***chemical*** and biological ***weapons*** and nuclear program. ...
	He intends to produce not only ***chemical*** but also bacteriological ***weapons***. ...

modifier relationship. In addition, the quasi-synchronous framework takes account of the transformation of dependence relationships by allowing inexact matching of dependent terms between queries and documents.

1.2 Variations in Dependence Relationships

Another challenging issue in considering term and concept dependencies of verbose queries is variations in dependence relationships between queries and documents. The same concepts and their relationships in queries and documents can be expressed in different ways by users and authors since the vocabularies of users and authors can be different. Table 1.2 demonstrates examples of the concept *"chemical weapon"* from the title query of topic 623 in the description query and relevant documents. Even though the concept implied by *"chemical weapons"* is similar, the orders, distances, and syntactic relations of the terms vary between the example sentences.

Recent work on term dependence models has achieved consistent improvements in effectiveness by allowing diversity in the relationships of dependent terms. The bi-term language model (Srikanth and Srihari, 2002) relaxes the constraint of order for matching query terms to documents. In the sequential dependence model (Metzler

and Croft, 2005), the unordered-window potential function allows the co-occurrence of dependent terms in any order within a window of fixed length.

However, terms within a certain distance do not always convey the same meaning. Depending on the specific syntactic relation between terms, the meaning of the terms can have different meanings. Therefore, these meanings may or may not be relevant to users' information needs. For example, both *"trade secret"* and *"secret trade"* are valid English expressions. The meaning of *"secret trade"* is not relevant to a user's information needs implied by the TREC query *"Document will discuss the theft of trade secrets along with the sources of information"*. Therefore, the valid variations in dependence relationships for a given term pair should be determined with regard to users' information needs.

Cooper argued that misunderstanding the independence assumption implied by the Binary Independence Model (BIM) lead to the failure of term dependence models Cooper (1995). He pointed out that the BIM is actually based on *linked dependence*, according to the degree of statistical dependence between terms in relevant and non-relevant documents. That is, if terms were as strongly dependent in relevant documents as in non-relevant documents, modeling term dependencies would confer no advantage over the independence assumption.

In order to identify valid variations in dependence relationships according to users' information needs, we propose a method that evaluates variations in dependence relationships based on the Generative Relevance Hypothesis (GRH). For a given information need, the GRH assumes that queries and their relevant documents can be thought of as random samples from the same latent representation space (Lavrenko, 2009). On the other hand, the null hypothesis assumes that documents and queries were drawn from unrelated populations in the representation space. We use this statistical significance test for evaluating whether a certain dependence relationship is valid for a given term pair with regards to users' information needs.

Q:	How do you get your *hair* to **grow** faster?
A:	Supposedly this works but never tried it. prenantal vitamins. they're just vitimans so they're not going to make u grow ...
Q:	How to **grow** *Columbine flowers*?
A:	Plant outside in sun or light shade, they will grow in both places. Scratch or loosen the soil lightly with a garden claw or rake. Sprinkle your seeds on and cover with the loose soil. You just cover with enough ...

Figure 1.1. The example questions with "*grow*" with different contexts "hair" and "flowers"

We apply the proposed method of evaluating valid variations in dependence relationships to the quasi-synchronous framework. For a given dependent term pair, we evaluate whether a specific syntactic relationship can represent relevant meaning to users' information needs. We refine the inexact matching process of the quasi-synchronous framework using the statistical significance test results of the GRH for specific dependence relationships of dependent terms. Instead of an arbitrary matching between dependent term pairs having different syntactic relationships, we constrain the quasi-synchronous framework to match query and document terms when having only valid syntactic relations.

1.3 Translation Model for Query Term Expansion

Query term expansion techniques have been studied to fill the lexical gap between queries and documents. A user's query is just one way of expressing an information need. Authors can compose documents using terms that are different than the user's queries. Query expansion methods reformulate an initial query using synonyms and related words (Croft et al., 2010). One approach to query term expansion is to use a statistical translation model (Berger and Lafferty, 1999).

When translating a word, we need to consider the context around the word. With different contexts, the same word can be translated into different expressions. Both

questions in Figure 7.1 contain "*grow*". The first question is about growing hair while the other question is about growing flowers. For these two queries, the translations of "*grow*" should be quite different. In order to improve the effectiveness of translation models for answer retrieval, we use the key concepts in a question as the translation context.

In verbose queries, there are key concepts that describe the most important part of the user's information need while other terms are used to express specific conditions of relevant information. Expansion results of all query terms are not always beneficial. Lee et al. (2008) pointed out the problem in previous work on translation models for query term expansion that a lack of noise control on the models. Query term expansion using translation models can cause degradation of retrieval performance because it is possible for non relevant word to be included. It is more likely for verbose queries to apply translation models to non-topical terms. Therefore, we selectively apply the translation model to terms by identifying the important concepts in a question.

In this thesis, we identify two types of concepts from verbose queries: key concepts and secondary (key) concepts. Key concepts are the most important terms of queries. We use key concepts as the context for translating terms. Although secondary (key) concepts are not as important as key concepts, they are still important because they provide clues about what kinds of information users are looking for. Based on concept classification results, we elaborate a translation model in which terms are selectively translated according to the most important context of a given query or question.

1.4 Answer Passage Retrieval for Verbose Queries

In IR, relevant judgments are usually made at the document level. Document-level relevance judgment typically assigns a level of relevance to a document based on how strongly its content is related to the information need. However, the entire

content of a relevant document may not be relevant to the information need. Users have to read retrieved documents to find specific answers to their questions. Focused retrieval methods has been studied in order to reduce this burden on users.

A focused retrieval system based on passages returns a ranked list of topically relevant text fragments. Users can more efficiently judge whether topically relevant text and corresponding documents are relevant to the topic of their information needs. A snippet in web search results is a good example of a type of passage retrieval (Arvola et al., 2010). Figure 1.4 shows a topically relevant text for the TREC topic "imported fire ants". Any part of this text could be a good passage for this topic.

However, a topically relevant text fragment may be a mixture of information that is related in varying degrees to a user's information need. Verbose queries specify detailed conditions for relevant information. For example, the title query "*imported fire ants*" of the TREC topic 820 states the topic of relevant information. On the other hand, the TREC description query "*What are imported fire ants, and how can they be controlled?*" asks for specific methods to control imported fire ants. Topically relevant text fragments do not guarantee that users will find their answers in these text fragments. In order to find an answer for this verbose query, users still need to read a retrieved passage or the whole document, if a selected passage did not contain an answer.

On the other hand, question-answering (QA) has been studied to handle a limited range of question types requesting a simple fact, the list of facts, a definition, *Wh*-questions, etc. In order to find answers for a specific type of questions, appropriate strategies and techniques are required. In IR, while the subject domain of questions are not constrained, QA systems are restricted in closed-classes of questions (Voorhees, 2001b).

In addition, when users judge a given answer, they often requires to check additional information. The TREC QA track require supporting documents in addition

... the treatment area size include the following: disposition of find site (private property, nursery, business park, etc.), method and history of introduction (if known), proximity of site to natural barriers such as dry areas, water bodies, etc., and man-made barriers.
Granular bait treatments using a metabolic inhibitor or Insect Growth Regulator (IGR) are the treatment methods of choice for red imported fire ants. These materials can be distributed by broadcast over entire areas or small applications can be made to individual mounds. Broadcast spreaders range from small hand-held units to larger hopper units. If reproductive adults are found, a soil drench of mixed pesticide may be applied to the colony to quickly kill the reproductives and prevent local spread.
In most areas of Central and Southern California, both a metabolic inhibitor, such as Amdro (hydramethylnon), and an IGR, such as Distance (pyriproxyfen) will be used to treat RIFA colonies ...

Figure 1.2. A topically relevant text fragment and an answer passage in it for the information "*What are imported fire ants, and how can they be controlled?*". The bold-faced text is the answer passage.

to answer strings. For example, "*Insect Growth Regulator (IGR)*" in Figure 1.4 is a good answer for the TREC description query. However, users must be convinced that "*Insect Growth Regulator (IGR)*" has an impact on fire ants. This means that the answer passage is required as well as the answer.

In order to compensate for the limitations of passage retrieval and question answering, we propose the new task of focused retrieval. For this purpose, we define an *answer passage* that can immediately provide answers for users' information needs while the length of answer passage should be suitable for restricted search environments such as mobile devices and voice-based search systems. In Figure 1.4, the bold-faced text is an example of answer passages.

A challenging issue in answer passage retrieval is that we need to consider two contrary characteristics of answer passages. First, answer passages have to be highly relevant to users' information needs and users should be able to recognize that the topics of answer passages are related to their information needs. Considering the

length of answer passages, query term densities in answer passages is not reliable for retrieval models to evaluate the relevancy of answer passages. Therefore, we incorporate retrieval models of varying granularities.

At the same time, answer passages should be informative. For passage-level retrieval results, if we emphasize too much the matching of concepts between queries and passages, the results may not provide novel information, which is actually what users are seeking. Therefore, while we depend on query term density for retrieval models in larger granularities, we use the translation model in Section 1.3 to measure informativeness of answer passages in passage-level retrieval models.

For evaluating the effectiveness of the proposed method for answer passage retrieval, we construct editorial data in which we manually annotated answer passages on the Gov2 collection. We investigate the different characteristics of answer passages from topically relevant text that has been used in passage retrieval systems. For a comparative study of topically relevant text fragments and answer passages, we also tagged topically relevant text fragments.

1.5 Contributions

In this section, we summarize the main contributions of this dissertation.

(a) We adopt the quasi-synchronous stochastic process that is developed for machine translation. The quasi-synchronous model for ad-hoc retrieval is able to more flexibly capture term dependencies from syntactic parsing results beyond the head-modifier relationship. The quasi-synchronous model also allow the transformation of dependence relationships between queries and documents.

(b) We propose the quasi-synchronous framework that integrates the quasi-synchronous model with other retrieval models in order to incorporate different term dependence assumptions. The quasi-synchronous framework optimizes the effectiveness

of individual retrieval models by predicting proper dependence assumptions based on the characteristics of queries.

(c) We evaluate valid variations in term dependency relationships according to the users' information needs. Using the Generative Relevance Hypothesis and evaluation results of valid variations in dependence relationships, we improve the quasi-synchronous model in matching dependence terms between queries and documents.

(d) We propose a query term expansion method using a translation model in which we selectively apply translation based on the classification results of concepts in verbose queries. Key concepts are used as the translation context in order to generate translation results of query terms based on the main topics of queries. Secondary concepts are used to prevent a translation model from introducing non-relevant expansions.

(e) We propose the new task of focused retrieval called *answer passage retrieval* that aims to provide immediate answers for more general search than factoid QA. For answer passage retrieval, we incorporate various levels of text units in which we emphasize either relevancy and informativeness.

(f) We construct manually annotated data for studying and evaluating answer passage retrieval. Using this data collection, we compare the characteristics of answer passages from topically relevant text to the typical output of passage retrieval systems.

1.6 Dissertation Outline

The remainder of this dissertation is organized as follows.

Chapter 2 presents related work, and discusses where and how the research conducted in this thesis is applicable to previous research.

Chapter 3 introduces the constituents of test collection used for empirical evaluation of our retrieval methods. In addition, we describe the evaluation metrics used in this dissertation.

Chapter 4 introduces the answer passage retrieval task. We describe the manual annotation task for answer passage retrieval.

Chapter 5 provides detailed description of the quasi-synchronous framework. We describe the integration of the quasi-synchronous stochastic process with other dependence assumptions. In addition, we discuss the characteristics of queries for predicting proper dependence assumptions.

Chapter 6 describes the evaluation method for variations in dependence relationships in order to find valid syntactic relationships of dependent terms according to users' information needs. We compare the effectiveness of the evaluation results to the arbitrary matching approach for considering the transformation of dependence relationships between documents and queries.

Chapter 7 presents the concept-based translation model. We describe the classification method of concepts for translation model. We evaluate the effectiveness of the concept-based translation model for answer retrieval from a non-factoid question-answer database.

Chapter 8 presents the answer retrieval method. We also discuss the characteristics of answer passages compared to the existing focused retrieval tasks of passage retrieval and question answering.

Chapter 9 summarizes the contributions made in this body of work and discusses potential future directions for more research in this area.

CHAPTER 2

BACKGROUND AND RELATED WORK

In this chapter we discuss background information and previous research relating to the contributions made in this thesis. We start by discussing the methods of modeling term dependencies (Section 2.1) and previously proposed methods of expanding query terms (Section 2.2) in order to capture the syntactic and semantic variations between queries and documents. Then, we survey related work on weighting query terms and dependence assumptions (Section 2.3) for evaluating these variations. We discuss previous work on focused retrieval based on QA and passage retrieval systems (Section 2.4). Finally, as background we discuss various natural language processing techniques that have been used in this thesis (Section 2.5).

2.1 Modeling Term Dependencies

Although the importance of dependence relationships between terms seems quite obvious, early work on dependency models failed to show consistent improvement over models based on the independence assumption. Many successful retrieval models including vector space models (Salton et al., 1975), probabilistic models (Robertson and Jones, 1976; Robertson and Walker, 1994) and language models (Ponte and Croft, 1998) rely on the independence assumption that ignores linguistic structures in the queries and documents instead treats queries as bags of words.

Recently, models incorporating term dependency have started to demonstrate consistent improvement (Metzler and Croft, 2005; Bendersky and Croft, 2012; Maxwell et al., 2013). There seems to be two important factors related the success of these

models: (1) The method for deciding which terms in queries are dependent and (2) how these dependent terms are used in relevant documents. In this section, we discuss previous research on selecting dependent terms from verbose queries. Then, we discuss related work that considers the transformation of relationships of dependent terms.

2.1.1 Selecting Dependent Terms

Croft et al. (1991) automatically derived term dependencies from Boolean queries in which users can manually specify term dependencies using query operators. They demonstrated the possibility that automatically extracted phrasal information from natural language queries could be as effective as manually specified dependencies by query languages.

One of the goals of this thesis is to identify term dependencies in natural language expressions using syntactic parsing results. While term dependencies in Boolean queries were explicitly expressed by ANDed adjacent term pairs, term dependencies in natural language expressions are more flexible in longer distance with various syntactic relationships. Therefore, we use the predefined syntactic relationships that dependent term pairs can have across the syntactic parsing tree.

Ideally, a retrieval model takes account of all possible dependence relationships between those terms. The Bahadur Lazarsfeld Expansion (BLE) takes account of all possible dependencies individual terms, term pairs, term triplets and so on (Losee, 1994) in $D = d_1, d_2, ..., d_n$ as follows:

$$
\begin{aligned}
Pr(D) = & \prod_{i=1}^{n} P_i^{d_i}(1-P_i)^{(1-d_i)}\Big[1+ \\
& \sum_{i<j} \varrho_{i,j} \frac{(d_i-p_i)(d_j-p_j)}{\sqrt{p_i p_j (1-p_i)(1-p_j)}} + \\
& \sum_{i<j<k} \varrho_{i,j,k} \frac{(d_i-p_i)(d_j-p_j)(d_k-p_k)}{\sqrt{p_i p_j p_k (1-p_i)(1-p_j)(1-p_k)}} + ... + \\
& \varrho_{1,2,...,n} \frac{(d_1-p_1)(d_2-p_2)...(d_n-p_n)}{\sqrt{p_1 p_2 ... p_n (1-p_1)(1-p_2)(1-p_n)}}\Big],
\end{aligned}
\tag{2.1}
$$

where d_i represent the value of term occurrences at the ith term and p_i is the probability that the ith term is relevant to users' information. ϱ is a correlation factor for normalize the probability of each order of dependence relations. The BLE measure the probability of the ith term under the dependence relationships with all possible subset of terms after the ith term.

In practice, it would be too inefficient to consider the full dependence of query terms in all degrees. Losee compared the effectiveness of the BLE according to the maximum order of dependent terms. Experimental results showed that incorporating dependence beyond degree three results in relatively little increase in performance.

Tree dependence models represent a query in a tree form in which vertices and edges are query terms and dependencies between term pairs, respectively. Yu et al. (1983) proposed a generalized term dependence model that combined the advantages of the dependence model based on the BLE and a tree dependence model in which they considered higher order dependence relationships than term pairs.

However, most previous work on term dependencies has been limited to pairs of terms. In this thesis, we also limit the basic unit of term dependencies in term pairs. Given this limitation, the simplest way to extract term dependencies is to assume that all term pairs in a query are dependent. Metzler and Croft (2005) proposed the Markov Random Field (MRF) model in which the score of a document D for a query Q is defined as follows:

$$
\begin{aligned}
P_\Lambda(Q,D) &\overset{rank}{=} \sum_{c \in C(G)} log_c \lambda f(c), \\
&= \sum_{c \in T} \lambda_T f_T(c) + \sum_{c \in B} \lambda_O f_O(c) + \sum_{c \in B} \lambda_U f_U(c),
\end{aligned}
\tag{2.2}
$$

in which T is the set of individual terms in a query Q. B is the set of dependent term pairs that is defined according to the full dependence model (FDM) and the sequential dependence model (SDM) as follows:

$$
\begin{aligned}
B_{FDM} &= \{(t_i, t_j) | 1 \leq i, j \leq |Q|\}, \\
B_{SDM} &= \{(t_i, t_{i+1}) | 1 \leq i < |Q|\},
\end{aligned}
\tag{2.3}
$$

where $|Q|$ is the length of queries. Although the full dependence assumption of B_{FDM} will not miss important pairwise dependencies between terms, it also introduces unnecessary dependencies that can hurt the effectiveness of term dependence models. Therefore, in spite of higher computational costs, the full dependence model typically does not show significant improvements over a simpler dependence assumption of B_{SDM}. The sequential dependence assumption was also used for the bi-term model (Srikanth and Srihari, 2002).

In order to avoid the full dependence assumption introducing unnecessary dependencies, Rasolofo and Savoy (2003) proposed a similar approach to the FDM where a term-proximity scoring heuristic is used to identify the more important dependencies among all the pairs of query terms.

Bendersky and Croft (2009) analyze long queries in a large scale search log in which they observed different types of queries. They proposed that a natural language processing approach should be more suitable for verb phrases while noun phrase queries might be better served by query segmentation without considering syntax. Based on the analysis results, Bendersky et al. (2009) proposed a query segmentation method for the sequential dependence model in which the sequential dependencies are constrained within the same segments.

Although the proximity of term pairs is strong evidence of a dependence relationship, modeling term dependencies based on adjacent term pairs will miss important term dependencies over longer distance. In order to capture longer distance term dependencies, dependence models based on syntactic parsing have been proposed.

Gao et al. (2004) proposed a dependence language model in which term dependencies were selected based upon the linkage structure of queries and documents. The fundamental idea behind this model is that queries and documents are represented in the form of the hidden variable, an acyclic, planar, undirected linkage graph L. The dependence language model generates not only a query but also the linkages L of the query as follows:

$$\begin{aligned} P(Q|D) =& P(L|D)P(Q|L,D) \\ =& P(L|D)p(t_h|D) \prod_{(i,j)\in L} P(t_j|t_i,L,D) \\ \text{such that } & L = argmax_L P(L|Q), \end{aligned} \tag{2.4}$$

in which q_h is the root of the parsing tree of a query Q. L is the index (i,j) of head-modifier term pairs. The probability $P(q_j|q_i,L,D)$ is computed as the point-wise mutual information of (t_i,t_j) in a document D.

Maisonnasse et al. (2007) extended this dependence language model using a syntactic and semantic analysis model. Lee et al. (2006) also suggested a language model based on dependency parse trees generated by a linguistic parser. However, these term dependence models using syntactic parsing results still constrain themselves to using the head-modifier relation. We select dependent terms using syntactic relationships across the overall topology of the parsing tree beyond the direct dependency of the head-modifier pairs.

Maxwell et al. (2013) use a term dependence model based on the catenae (Latin for the plural of 'chain') to capture dependent terms beyond the head-modifier relation.

Catenae are subsets of terms in the path from terminal nodes to the root. All catenae extracted from a query are not always useful for retrieval. Therefore, as in Rasolofo and Savoy (2003), they use a machine learning method to select important catenae in which a specific dependence path and co-occurrence features of terms are used as features. Catenae are represented by the ancestor-descendent relationship in the parsing tree. In addition, we also use the sibling and c-commanding relationship that can capture term dependencies across phrases and clauses.

2.1.2 Transformations of Dependence Relationships

The second factor of the successful term dependence models is to allow inexact matching because dependent terms conveying the same meaning will not be always used in the same form. In our quasi-synchronous framework, we allow the matching dependent terms having one of predefine syntactic relationships such as parent-child, ancestor-descendent, siblings and c-commanding. In this section, we discuss methods that were used in previous work to allow the transformation of dependence relationships between queries and documents.

Tao and Zhai (2007) explored the intuition that the proximity of matched query terms in documents can also be used to promote scores of documents. Term proximities of query terms in documents can represent the flexibility of usage of dependent terms in documents. A comparative study of five proximity measures demonstrated that the pairwise distance-based measures showed improvements for most experimental settings.

The bi-term language model (Srikanth and Srihari, 2002) allows matching between terms in different orders. The ordered-potential functions $f_O(\cdot)$ and the unordered-potential functions $f_U(\cdot)$ of the MRF model (Metzler and Croft, 2005) in Eq. 2.2 are defined as follow:

$$
\begin{aligned}
f_O(t_i, t_{i+1}) =& (1-\alpha_D)\frac{tf_{\#N}(t_i, t_{i+1}; D)}{|D|} + \alpha_D \frac{tf_{\#N}(t_i, t_{i+1}; C)}{|C|}, \\
f_U(t_i, t_{i+1}) =& (1-\alpha_D)\frac{tf_{\#uwN}(t_i, t_{i+1}; D)}{|D|} + \alpha_D \frac{tf_{\#uwN}(t_i, t_{i+1}; C)}{|C|},
\end{aligned}
\tag{2.5}
$$

in which t_i and t_{i+1} are an adjacent term pair in queries. $tf(t_i, t_{i+1}; D)$ and $tf(t_i, t_{i+1}; C)$ represent the frequencies of t_i and t_{i+1} in a document D and a collection C, respectively. $\#N$ is the frequencies of t_i and t_{i+1} in the same orders within a N word distance while $\#uwN$ means that it count the co-occurrences of t_i and t_{i+1} in any orders. The order-window potential function with #1 counts the exact matching. By setting $\#N$ and $\#uwN$, the MRF model controls the possible variations in dependence relationships in their distance and order.

Maxwell et al. (2013) use these ordered and unordered-window potential functions of the MRF model to match dependent terms to documents. While they use a computationally expensive method to select dependent terms from the syntactic analysis results of queries, they exploit a more efficient method to take account of the transformation of dependent terms based on proximity.

Peng et al. (2007) investigate term dependencies based on the proximity in the Divergence From Randomness (DFR) framework. In this work, they compared the effectiveness of window size that dependent terms can have in documents. In the experimental results, the DFF framework with the full dependence assumption does not show significant improvements when using a window size beyond a certain length. For the sequential dependence assumption, the effectiveness of the DFR framework decreased for longer window sizes.

Cui et al. (2005) proposed fuzzy relation matching for question answering in which the dependence path between term pairs are agglomerated using a machine learning algorithm. The fuzzy matching of dependence relationships reflects the degree of matching between relation paths in questions and their candidate sentences containing

answers. They attempted to match the different syntactic relationships of dependent terms.

In previous work, dependent terms were selected and variations in their relationships were predefined before a query submission. Dependence paths and co-occurrence features of dependent terms in a query (Maxwell et al., 2013) are independent from other terms in the query. We aim to model the importance of term dependencies based on models of user information needs. In Section 2.3, we will discuss related work on query evaluation methods based on users' information needs.

In previous work, dependent terms were selected and variations in their relationships were predefined before query submission. We aim to evaluate the validity of variations in dependence relationships according to users' information needs. In Section 2.3, we will discuss related work on evaluating dependence assumptions to maximize the effectiveness of retrieval models.

2.2 Query Term Expansion

As we take an account of the transformation of dependence relationships, we use query term expansion method to bridge the lexical mismatch between queries and documents. We introduce query term expansion methods according to their source of expanded concepts and then discuss a translation model that we use for query term expansion in this thesis.

Query term expansion methods have been intensively studied for bridging the lexical gaps between queries and documents. In early work, term classification results were used to to find related expansions for a given query term (Jones and Needham, 1968; Jones and Jackson, 1970). Thesauri such as WordNet have also been used for query term expansion (Voorhees, 1994).

Automatically constructed thesauri are also used for query term expansion. In this work, thesauri are constructed using co-occurrence statistics of terms (Qiu and

Frei, 1993; Salton, 1980; Schütze and Pedersen, 1997) and syntactic relationships of terms (Chen and Ng, 1995; Grefenstette, 1992). Zhou et al. (2013) use Wikipedia as world knowledge that is used to extract related words such as synonyms, polysemy, etc.

Query term expansion methods using statistics of terms in collections generate expanded concepts using global contexts. Global contexts are not changed according to queries and their initial retrieval results that can represent the characteristics of queries and corresponding documents. On the other hand, local analysis approaches rely on a given query and its initial retrieval results to generate expansions (Xu and Croft, 1996).

Instead of collecting true relevant documents to better estimate the user's information need, the initial retrieval results can be used as pseudo relevant documents. Croft and Harper (1979) used the top ranked document as pseudo relevant documents for estimating new probabilities of query terms. Jing and Croft (1994) proposed a method for automatically constructing a collection-dependent thesaurus from retrieved phrases. Similarly, Xu and Croft (1996) proposed local context analysis in which the top ranked documents are used, but the expansion is based on the best passages instead of whole documents.

Metzler and Croft (2007) proposed latent concept expansion in order to take account of term dependencies for query term expansions using the MRF model. They use features of individual terms, ordered, and unordered windows to evaluate the relation between queries and documents and between expanded concepts and documents. They also investigated multi-term concept generation, which failed to show significant improvement on query term expansion, but demonstrated the possibility of using expanded concepts for query suggestion and reformulation.

2.2.1 Translation Models

In this thesis, we use a translation model to generate expanded concepts, which will be used to measure the informativeness of passage retrieval results. In particular, our translation model exploits the key concepts of verbose queries as translation context to generate translation results related to users' information needs. In this section, we introduce research on translation models for query term expansion and discuss phrase-based translation models which can be used to take account of translation context in translation models.

Translation models have been used as statistical query term expansion methods in which expanded concepts are treated as an interpretation of the original query. While local feedback uses pseudo relevant documents or passages as source of expanded concepts, a translation model is trained from related pairs of text sentences or fragments. Translation models generate a translation table that consists of source and target term pairs and the probabilities of translations between terms. Target terms for a given source term in a translation table can be used as expanded concepts and probabilities can be used to refine query term expansion results.

In order to use translation models for ad-hoc retrieval, we need to have enough training data of related text to generate the translation table. One approach is to use queries and relevant text as the training data. Berger and Lafferty (1999) proposed a method to generate synthetic queries for a large collection of documents. They use a sampling technique for generating queries that can distinguish a document from other documents.

Alternatively, Karimzadehgan and Zhai (2010) proposed a method to estimate a translation model using normalized mutual information between words. This mutual information is estimated to reflect the self translation of a document which is more efficient in computational cost and coverage compared to the Berger and Lafferty (1999) approach.

Lu et al. (2002) used anchor-text for training a translation model for query terms such as new terminology and proper names. They assume that the anchor texts of hyperlinks pointing to the same page are alternative expressions or translations of each other.

Recently, web log data of commercial web search engines has been used to construct a parallel corpus for translation models. Gao et al. (2010) used clickthrough data in which queries and the titles of their clicked web pages are used as pairs of sentences to estimate a translation table. Similarly, Riezler et al. (2008) used queries and snippets of clicked web pages as training data for a translation model.

In addition, as manually constructed question answer pair collections have become available, these collections are being used for training translation models. Riezler et al. (2007) use 10 million question-answer pairs extracted from Frequently Asked Questions (FAQ) lists. Question-answer pairs collected from community question answering sites are also used as training data for translation models (Jeon et al., 2005; Murdock and Croft, 2005; Radev et al., 2001; Surdeanu et al., 2011; Xue et al., 2008). Bernhard and Gurevych (2009) use lexical semantic resources such as Wiki Answer, glosses and Wikipedia for training translation models.

Murdock and Croft (2005) pointed out that underestimated self-translation probabilities reduce retrieval performance by assigning low weights to question terms while overestimated self-translation probabilities in translation models for finding expanded concepts remove the benefits of the translation model in translation models for finding expanded concepts. They separate the self-translation of terms from the translation table and parameterize the weights of the self-translation. Murdock and Croft (2005) also proposed a smoothing method for translation models using document and collection models.

Xue et al. (2008) investigated the direction of translation of question-answer pairs. Unlikely, the translation between different languages, the translation results from

answers to questions can provide information that is as useful translations as the translations from questions to answers. Xue et. al. compared the effectiveness of translation models according to the direction of translation between questions and answers.

Another challenging issue is that a word-to-word translation model cannot take account of the contextual information implied by queries because the translation of a term does not affected by other term in a query. Koehn et al. (2003) proposed a phrase-level translation model that shows improvement when using phrases of up to three words.

Phrase-based translation models have been studied that compute the translation probability between multi-term phrases (Gao et al., 2010; Zhou et al., 2011, 2013). Surdeanu et al. (2011) extract bigrams from syntactic parsing trees and semantic role labelled results of question-answer pairs. In this approach, terms in phrases become a context for the translation of other terms in the same phrase. In this thesis, we use a phrase-level translation model approach in which the most important key concepts of queries are the translation context of the other terms in the query.

Mandala et al. (1999) proposed a method using different types of thesauri for query term expansion. In order to avoid incorrect expanded concepts, they used a term weighting method in which the weights rely on not only the features of terms but also on features from thesaurus. On the other hand, Lee et al. (2008) proposed a method using the TextRank algorithm to select terms. They applied a translation model for query terms based on the selected terms. Mandala et al. refined the quality of query term expansion results after generating expanded concepts while Lee at al. refined the original query terms before generating expansions.

In (Lee et al., 2008), the TextRank algorithm that is used to measure the importance of terms is evaluated within a single document, which is represented as a graph for measuring the PageRank scores of terms. We measure the importance of terms

that directly maximize the effectiveness of translation models. For this purpose, we use query term weighting methods. We will discuss in detail the estimation of the relative importance of terms in the next section.

2.3 Weighting Query Terms and Their Dependencies

Query terms have different degrees of importance. For example, inverse document frequencies are used to assign different weights to query terms based on the assumption that terms used in more documents have less power to discriminate relevant documents (Salton et al., 1975). Salton and Buckley (1988) claimed that the assignment of suitable weights to individual terms is superior to modeling term dependencies or other text representations. However, query term weighting techniques and term dependence modeling are not exclusive of each other.

In the previous section, we discussed existing work on bridging differences in query terms and their dependence relationships between queries and documents. As query term weighting methods are used for evaluating the importance of query terms, similar methods can be used for evaluating the effectiveness of expanded concepts. Furthermore, similar approaches can be used to evaluate dependent terms for a specific dependence assumption and to select suitable retrieval models with dependence assumptions.

We discuss query term weighting methods and describe how to automatically generate training data that directly maximizes the effectiveness of target retrieval models. Then, we introduce previous work on estimating the relative importance of dependence models in interpolated retrieval models of multiple dependence assumptions.

2.3.1 Query Term Weighting

Removing stopwords is one of the most commonly used method for query processing. Lo et al. (2005) propose a method for automatically finding stopwords for

a given collection. Kumaran and Allan (2007) demonstrated that retrieval models would show better results when we could select the optimal subset of a query. Based on this observation, methods for reducing long queries by removing less important terms have been studied (Balasubramanian et al., 2010a,b; Kumaran and Carvalho, 2009).

Huston and Croft (2010) compared query processing techniques to preprocess verbose queries for web search engines. In this work, Huston and Croft proposed a method for automatically selecting stop structure that showed similar performance compared to manually identified stop structure.

Bendersky and Croft (2008) proposed a method identifying key concepts of verbose queries. Instead of removing less important terms, they used the identification of key concepts to assign higher weights. They use a machine learning method that is trained on manually annotated key concepts of verbose queries. Similarly, we can use this approach to identify important dependent term pairs.

Xue et al. (2010) proposed a method in which the conditional random field model is used to treat the query terms selection problem as a sequential labeling problem. Using the sequential labeling setting, they reflect local and global dependencies between query terms.

Instead of relying on manually annotated training data, other approaches have been studied for identifying important terms in queries. Query difficulty (Amati et al., 2004), clarity scores (Cronen-Townsend et al., 2002), and performance prediction results of queries (Balasubramanian et al., 2010b) have been used to predict the importance of query terms.

Hauff et al. (2008) assume that query performance prediction algorithms fall into two categories: pre-retrieval prediction and post retrieval prediction. In pre-retrieval prediction, the query is evaluated before the retrieval step without considering the ranked list of results (He and Ounis, 2004). On the other hand, post retrieval predic-

tion algorithms either compare the ranked list to the collection as a whole, or different rankings produced by perturbing the query or documents.

Lee et al. (2009) proposed a method to rank query terms in which the rank of a query term was decided to maximize a target evaluation measure. As in Xue et al. (2010), they took into account of the effect of other terms when measuring the effectiveness of query terms. For this purpose, they iteratively selected the best query terms among the remaining terms at each iteration.

Katz and Lin (2003) proposed a method that selectively apply linguistic techniques based on the classification results of semantic symmetry and ambiguous modifications. In the Lee et al. (2009) approach, an evaluation measure for a retrieval model is used to automatically generate training labels because the goal of their query term weighting method is to maximize the effectiveness of retrieval models. In this thesis, we can set up the process of generating training labels using different criteria according to the target problem. For example, in order to select good query terms to which we apply a translation model, we can use the effectiveness of expanded concepts with a target evaluation measure.

2.3.2 Weighting Retrieval Models

In this thesis, we evaluate the validity of variations in dependence relationships for a given term pairs according to users' information needs. In previous work, the effectiveness of different dependence assumptions and retrieval models are measured according to the characteristics of target collections and queries.

Metzler (2007) proposed an automatic feature selection method to combine multiple features including the BM25 model and language models with three potential functions. For a given collection, the automatic feature selection method iteratively adds new features with an interpolation weight that shows the best performance in each iteration. In this automatic feature selection method, the importance of

dependence assumptions is measured according to the characteristics of document collections but are independent from the users' information needs implied by queries.

Bendersky and Croft (2012) proposed the *query hypergraph* in which local and global factors are used to consider the importance of individual concepts and specific dependence structures as follows:

$$sc(Q, D) = \sum_{\sigma \in \Sigma^Q} \lambda(\sigma) \sum_{\kappa \in \sigma} f(\kappa, D) + \max_{\pi \in \Pi_D} \sum_{\sigma \in \Sigma^Q} \lambda(\sigma, \Sigma^Q) \sum_{\kappa \in \sigma} f(\kappa, D), \tag{2.6}$$

where κ is a concept that can be an individual term or dependent term pairs. Σ^Q is a set of *hypergraph structures* implying different dependent assumptions. The matching function $f(\kappa, X)$ is weighted by the local factor $\lambda(\sigma)$ and global factor $\lambda(\sigma, \Sigma^Q)$. The values of these two factors are selected per hypergraph structure as follows:

$$\begin{aligned} &\forall \kappa_i, \kappa_j \in \sigma : \lambda(\kappa_i) = \lambda(\kappa_j) = \lambda(\sigma) \\ &\forall \kappa_i, \kappa_j \in \sigma : \lambda(\kappa_i, K^Q) = \lambda(\kappa_j, K^Q) = \lambda(\sigma, \Sigma^Q) \end{aligned} \tag{2.7}$$

where $\lambda(\cdot)$ is same for all concepts κ in the same hypergraph structure σ. where $\lambda(\kappa, K^Q)$ is the weight of the concept κ in the context of the entire set of query concepts K^Q. Similarity, in the quasi-synchronous framework that interpolated the quasi-synchronous stochastic process with other dependence models, we also predict optimal weights of individual dependence models for a given queries in order to maximize their strengths.

In their earlier work, Bendersky et al. (2010) proposed the weighted dependence model that estimates the importance of dependence assumptions for individual de-

pendent term pairs. Their full parametric form of the weighted dependence model is as follows:

$$
\begin{aligned}
P(D|Q) \overset{rank}{=} & \sum_{i=1}^{k_u} w_i^u \sum_{q \in Q} g_u^u(q) f_T(q, D) + \\
& \sum_{i=1}^{k_b} w_i^b \sum_{q_j, q_{j+1} \in Q} g_b^u(q_j, q_{j+1}) f_O(q_j, q_{j+1}, D) + \\
& \sum_{i=1}^{k_b} w_i^b \sum_{q_j, q_{j+1} \in Q} g_b^u(q_j, q_{j+1}) f_U(q_j, q_{j+1}, D)
\end{aligned}
\tag{2.8}
$$

where w_i^u and w_i^b are interpolation weights measured per dependence assumption while $g_u^u(q)$ and $g_b^u(q_j, q_{j+1})$ are weighted per query term and term pair, respectively. Bendersky et al. use collection-dependent and collection-independent features to predict $g_u^u(q)$ and $g_b^u(q_j, q_{j+1})$. These features are still determined for a given query. Therefore, it cannot evaluate valid dependence assumptions for individual term pairs.

In this thesis, we aim to evaluate the validity of variations in dependence relationships for a given term pairs. Therefore, suitable syntactic relationships can differ for term pairs in a query although an information need implied by the query is the same for these term pairs. In (Song et al., 2008), the training data consists of dependent term pairs in which the training labels are selected per dependent term pairs. Song et al. proposed *variability* that represents the inverse strength of the head-modifier term pairs. In the case of a strongly-tied headword and a modifier, e.g., "mutual → fund", terms will have the same head-modifier relationship in relevant documents. On the other hand, the variability of weakly-tied pairs, e.g., "overcrowded → prison", will be higher than that of strongly-tied pairs. Song et al. used the variability for smoothing as follows:

$$v(t_i) = P(w_{h_i} = 0 | w_i \in R, w_i \rightarrow w_{h_i} \in Q) \tag{2.9}$$

in which, $w_i \rightarrow w_{h_i}$ represents the head-modifier relation of w_i and its headword w_{h_i}. The conditional probability v_i is estimated from relevant documents, that is the strength of dependence relationships in relevant text to users' information needs.

Although variability can represents the strength of head-modifier term pairs, it might not affect the effectiveness of term dependence model if the strength of dependence relationships in the non-relevant class were stronger than that in the relevant class. We use the GRH to evaluate the relative strength of dependence relationships by comparing the statistics of dependent term pairs between relevant documents and the entire collection.

2.4 Focused Retrieval

We study a focused retrieval system that can return retrieval results from which users can find answers for their information needs, while we keep the size of retrieval results being suitable the restricted search environments such as the small screen of mobile devices or voice-based search systems. Focused retrieval aims to provide more precise retrieval results instead of the list of documents. The retrieval units of focused retrieval systems can be defined at various granularities. Question answering systems return direct answers for the specific types of questions. However, users need additional evidence in order to confirm that returned results are true for their questions.

On the other hand, passage retrieval systems extract highly relevant text fragments from documents (Salton et al., 1993). Although retrieved passages can help users be convinced whether documents that the passages are extracted from are relevant to their information needs, it does not guarantee that retrieved passages contain answers for users' questions. In this section, we introduce related work on QA and

passage tasks and discuss the limitation in this work to provide direct answers without additional evidence.

It is hard to define question answering and passage retrieval tasks separately. Most question answering systems rely on passage retrieval results (Roberts and Gaizauskas, 2004). As a first step, an information retrieval system is used to extract the candidate documents or passages that seem to contain answers. Then, more complex and expensive techniques such as pattern matching and natural language processing are applied to these candidates to find answers. The two-stage architecture of question answering systems makes use of the efficiency of information retrieval systems when selecting candidates from large-size document collections while still allowing more expensive techniques.

In this section, we will investigate focused retrieval systems from two directions. From the viewpoint of question answering systems, we introduce question types, from simple factoid questions to questions about more general information needs. Then, we describe retrieval systems using passage-level evidence and related issues.

2.4.1 Question Answering

Answering natural language questions has long history in the fields of natural language processing and information retrieval (e.g. Simmons, 1965). Question answering systems aim to find direct answers from document collections for specify classes of questions such as a single fact, a list of facts, a definition, *Wh*-questions, etc.

Table 2.1 shows example questions and their types (Dang et al., 2007). Factoid questions request a simple fact such as a name, place, date etc. The list questions of the TREC QA track assemble answers from multiple sources (Voorhees, 2001a). For example, the question "Which US government officials accepted his claims regarding Iraqi weapons labs?" requires a list of officials' names. The definition task of the TREC QA track asks information about a given target (Voorhees, 2004). This kind

Table 2.1. The example of question series in the TREC QA 2007 track.

QID	Type	Question
291		Pakistan earthquakes of October 2005
219.1	FACTOID	What year did Curveball defect?
219.2	FACTOID	What was Curveballs profession?
219.3	FACTOID	What is Curveballs real name?
219.4	FACTOID	Which intelligence service employed Curveball?
219.5	LIST	Which US government officials accepted his claims regarding Iraqi weapons labs?
219.6	FACTOID	Where does Curveball now live?
219.7	OTHER	

of question can be interpreted as "Tell me other interesting things about this target that I dont know enough to ask directly" (Voorhees, 2005). This type is referred to as "other" in Table 2.1.

The early TREC QA tracks asked for ranked-lists of documents containing answers (Voorhees, 2000). The TREC 2003 QA track required the pair of an answer string and its support document. An answer string is supposed to be a precise answer to the question and the support document contains information that supports the answer (Voorhees, 2005). We can see the direction of this task from the judgment criteria as follows:

- **incorrect**: the answer string does not contain a right answer or the answer is not responsive;
- **not supported**: the answer string contains a right answer but the document returned does not support that answer;
- **not exact**: the answer string contains a right answer and the document supports that answer, but the string contains more than just the answer or is missing bits of the answer;

- **correct**: the answer string consists of exactly the right answer and that answer is supported by the document returned.

In this criteria, not only missing part of an answer but also including unnecessary text in answer strings were penalized. Even if an answer were right, this answer string without a supporting document would not be treated as a correct answer because users may not be convinced.

Researcher have worked on finding answers for questions beyond simple factoid questions. As one of the approaches, non-factoid QA systems find answers by searching for similar questions in a question answer database instead of finding answers from unstructured raw text (Cong et al., 2008; Jeon et al., 2005; Murdock and Croft, 2005; Surdeanu et al., 2011; Xue et al., 2008).

In the question-answer pair database, an answer is an independent text providing an answer for a specific question. Therefore, it does not require further analysis to locate exact answer from retrieval results as we treat each answer or question-answer pairs as a single document. If users' questions were perfectly matched with questions in retrieval results, it can provide concise and convenient answers for users. a retrieved answer, however, cannot provide any useful information for a user's question if the main topic of an original question for the retrieved answer is different from the topic of the user's question even if the user's question and is similar to the original question.

2.4.2 Passage Retrieval

Our answer passage retrieval system aims to retrieve answers from unstructured documents as we find answers from manually composed answers in community question-answer database. Passage retrieval systems retrieve parts of documents instead of the entire documents (Salton et al., 1993). In the High Accuracy Retrieval from Documents (HARD) track 2004 of the TREC, it was investigated whether passage retrieval can be used to increase accuracy of retrieval systems by eliminating non rel-

evant text (Allan, 2004). The INitiative for the Evaluation of XML retrieval (INEX) was set up with the aim to establish retrieval systems for the structured data of XML documents (Gövert and Kazai, 2002). The INEX Ad Hoc Track investigated whether the document structure helps to identify where the relevant information is within a document (Fuhr et al., 2008).

Furthermore, the INEX 2010 Ad Hoc Track suggested a search environment in which available resources are restricted such as a small screen in mobile devices (Arvola et al., 2011). For restricted search environments, NTCIR also organized the 1-Click track that aims to satisfy the user with a single textual output, immediately after the user clicks on the SEARCH button (Sakai et al., 2011).

Callan (1994) studied how passages can be defined, how they can be ranked, and how passage evidence can be incorporated into document retrieval. Passages were used both as independent retrieval units and as evidence that could be used to modify a document ranking.

Bendersky and Kurland (2008) proposed a method that estimates the weight of passage retrieval results based on the similarity between a document and its passages. In this work, they assign higher weight on a document model when passages in a document are similar to each other. They compare methods to measure similarities between passages and documents.

One way to define passages is to use the structural information of documents (Callan, 1994; Hearst and Plaunt, 1993; Salton et al., 1993). Specific markup information for a document, such as section, empty line, text indent, period etc., can be used as passage boundaries. Callan (1994) proposed the bounded paragraph based on the hypothesis that one can provide more consistent paragraphs by merging short paragraphs and by dividing large paragraphs. Salton et al. (1993) compared the effectiveness of sentence, section and paragraphs as passages.

Hearst and Plaunt (1993) use orthographically marked segments to select passages. They also used the TextTiling method that split a document into coherent units. Text segmentation methods have been studied to decompose documents to identify the structure of documents (Salton et al., 1996; Ponte and Croft, 1997).

Arbitrary passages are defined by the overlapped windows (Kaszkiel and Zobel, 1997, 2001). Kaszkiel and Zobel (1997) used arbitrary passages based on fixed-length windows. They also compared fixed-length windows and variable-length windows. For variable-length windows, the size of window is selected that shows the highest score by a passage retrieval model. Liu and Croft (2002) compared half-overlapped windows and arbitrary passages.

As an alternative approach, Lv and Zhai (2009) proposed the positional retrieval model in which language models are derived for each position of a document. When the positional retrieval model compute scores at a certain position, term frequencies are counted based on a proximity-based density function that discounted the term frequencies according to the distance of terms from a given position.

2.5 Natural Language Processing

2.5.1 Parsing

In this thesis, we aim to identify term dependencies in the sentence structure of queries and documents. For this purpose, we use the dependency parsing results of queries and documents. A parser produces useful structures over arbitrary sentences (Manning and Schütze, 1999, Chapter 8). A dependency parser is a probabilistic parsing technique where sentence structures are based on the dependency relation.

Figure 2.1 demonstrates the example parsing results of"*John liked the dog in the pen*". Figure 2.1.(a) is the PCFG parsing tree in which the sub-structures of the sentence are represented by non-terminal nodes. On the other hand, Figure 2.1.(b)

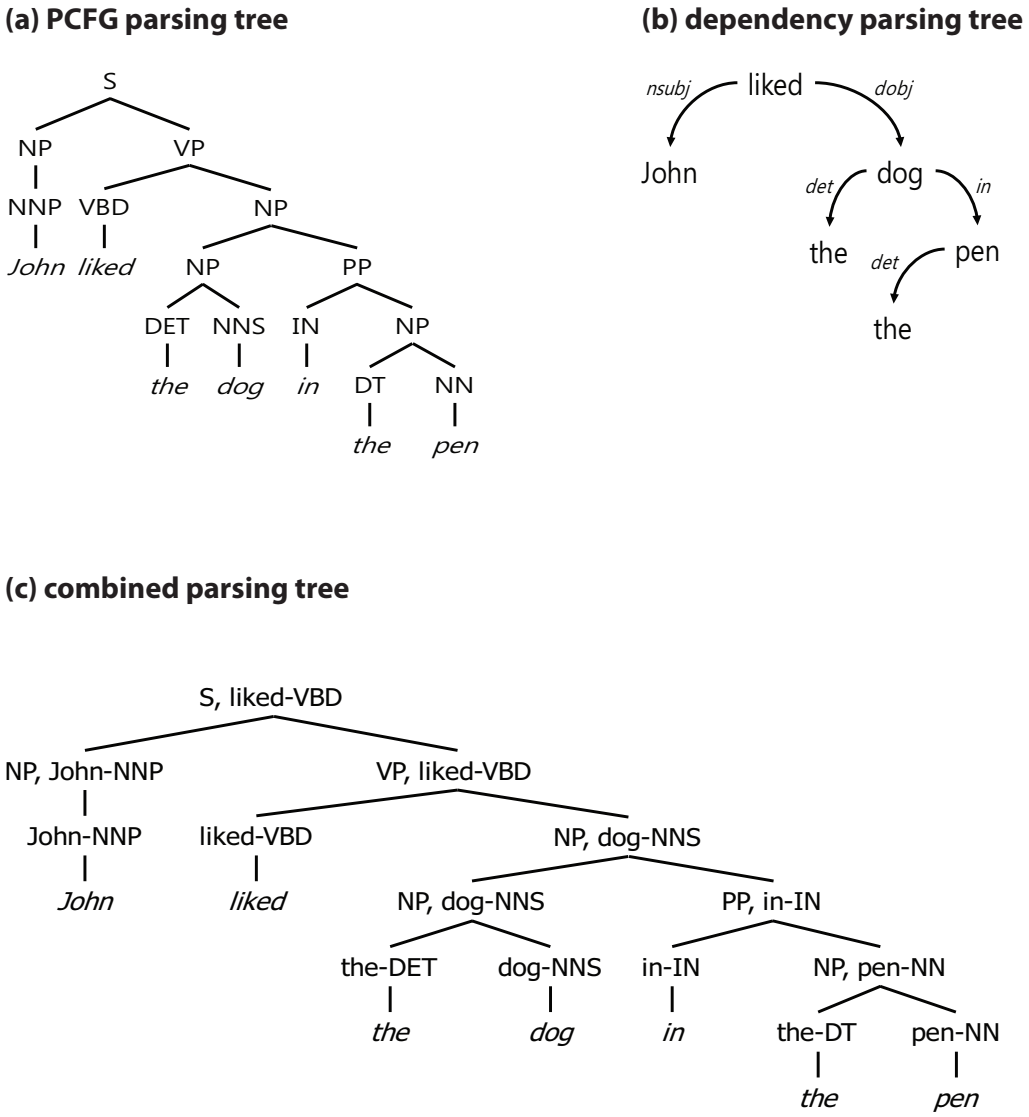


Figure 2.1. Examples of syntactic parsing results.

is the dependency parsing tree of the sentence in which nodes and edges represent words and their relationships, respectively. The labels of edges are the grammatical relationships between dependent words in the example dependency parsing tree.

Klein and Manning (2003) proposed the factored model for natural language parsing which generates the combine parsing tree of a phrase structure tree T and a dependency tree D. They assumed the dependency and phrase structure need not be modeled jointly, therefore, they factor the model as $P(T, D) = P(T)P(D)$.

The rules for lexicalized PCFG parsing of the factored model looks like $S, x \rightarrow NP, y\ VP, x$ of which the score is computed by joining of PCFG score for $S \rightarrow NPVP$ and the dependency score for x taking y as a dependent and the left and right STOP scores of y. Klein and Manning use A* algorithm in which the PCFG parser is used to find scores $PCFG(e)$ for each edge and the dependency parser is used to find outside scores $DEP(e)$, separately. Then, the combined outside estimation $a(e) = PCFG(e)+DEP(e)$ is used for A* algorithm to more efficiently prune candidate edges while exploiting the advantages of the PCFG and dependency parsing approaches.

Verbose queries are usually written in imperative and interrogative sentences that can be more likely parsed incorrectly because of the proportion of sentential types in the training data. We observed that rephrasing queries to declarative sentences can improve the parsing results of queries.

Specific syntactic relationships between terms are used to identify predicate-argument relationships for semantic role labeling (Hacioglu, 2004). Dependency relationships of terms are also used for query term selection (Park and Croft, 2010). Balasubramanian and Allan (2009) proposed the SVM weighting method in which the Subject-Verb-Object relationships of terms were used to assign weight to query terms.

We are interested in the structural information between terms and do not consider the internal substructures in grammar that are represented by non-terminal nodes of PCFG parsing results. Therefore, we select a dependency parser (De Marneffe et al., 2006) instead of the PCFG syntactic parser. Dependency paths in parsing results are used to capture the dependence relationships between terms (Cui et al., 2005; Aktolga et al., 2011). The dependency path of a term pair is the set of dependency relationships in the path from one term to the other. For example, the dependency path of "*John*" and "*dog*" in Figure 2.1 is $nsubj--dobj$. There are too many possible dependency paths that term pairs can have. Cui et al. (2005) proposed a passage retrieval model for question answering in which they restricted the maximum length

of dependency paths and used the IBM translation model to measure the matching scores between different dependency paths. It is not practical for modeling term dependencies to consider each dependency path and all possible specific variations. Therefore, in this paper, we do not consider the types of dependency relationships but use the topological relationships of dependent terms in dependency parsing results for modeling term dependencies.

2.5.2 Ontology

As we discussed in Section 2.3, thesauri have been used for finding relationships between words. WordNet is is one of the large lexical databases of English (Fellbaum, 2010) in which nouns, verbs, adjectives and adverbs are grouped into sets of cognitive synonyms (*synsets*) that representing distinct concepts. Synsets are interlinked by means of conceptual-semantic and lexical relations.

Synsets are interlinked by means of conceptual-semantic and lexical relations. In the resulting network, 117,000 synsets is linked to other synsets by means of conceptual relations [1]. The most frequently-used relation for noun synsets is the super-subordinate relation including hyperonymy, hyponymy and ISA relation that more general synsets like "*furniture*" to specific ones like "*bed*" and "*bunkbed*". All noun hierarchies go up the root node "*entity*".

Verb synsets are also arranged into hierarchies. The lower level of verb synsets express increasingly specific manners, e.g. "*communicate*"-"*talk*"-"*whisper*". The more specific manner are defined according to the semantic field. Volume is one dimension along which verbs can be elaborated. Others are speed (move-jog-run) or intensity of emotion (like-love-idolize). For more detail explanation about the relationships of WordNet, please refer to Fellbaum (2010)

---

[1] `http://http://wordnet.princeton.edu/wordnet`

Ciaramita and Johnson (2003) proposed the supersense tagger that is an extended named-entity recognition using the sematic categories for the lexicographer developing WordNet. They also used data in WordNet to training classifier.

We use the named-entity recognition results in order to solve data sparseness problem in query term expansion using a translation model. In Chapter 1, we introduce the example of translations of "*grow*" based on the context "*Columbine flowers*". A problem is that "*Columbine flowers*" is rarely used. Therefore, the data for "*Columbine flowers*" is not enough to estimate translation probabilities. We solve data sparseness problem by using the named-entity recognition result "*PLANT*" of "*Columbine flowers*".

2.6 Summary

In this chapter, we summarized the background and the previous work related to this thesis. We described the methods of modeling syntactic variations in term dependence relationships (Section 2.1) and lexical variations in query terms (Section 2.2). We, then, described the related work on weighting query terms and dependence assumptions (Section 2.3) for evaluating these variations. We discuss previous work on focused retrieval. Finally, we discuss natural language processing techniques that have been used in this thesis.

In the next chapter, we will describe the data collections and the evaluation metrics used for empirical evaluation of our retrieval models.

CHAPTER 3

DATASETS AND EVALUATION

In this chapter, we describe the data collections and evaluation measures that we use in the remainder of this dissertation. In Section 3.1, we describe test collections that we use for the evaluation. Then, in Section 3.2, we explain the evaluation criteria and metrics used to measure the performance of the document and passage retrieval results.

3.1 Test Collections

3.1.1 TREC Collections

Text REtrieval Conference (TREC) [1] aims to support research of the IR community and provides the infrastructure for the large-scale evaluation of text retrieval methods and tasks. TREC has produced a number of test collections over the years. These test collections have been used by the IR community to enable the development of retrieval models, query processing techniques, and evaluation measures for a broad range of IR applications.

Table 3.1 shows a summary of the three TREC collections that we use in this dissertation. TREC collections consist of a document collection, topics and the set of relevance judgments for the topics. In this section, we describe the characteristics of documents, topics and corresponding judgments for these three TREC collections in detail.

[1] `http://trec.nist.gov`

Table 3.1. Summary of the TREC collections for evaluating document retrieval. ***Aver. Length*** is the average length of documents in word.

Collection	Documents	# Doc.	# Topic	Aver. Length
Robust04	News articles	528,155	301∼405, 601∼700	510.5
Gov2	.gov documents	25,205,179	701∼850	937.3
ClueWeb-B	Web pages	50,220,423	1∼150	804.8

3.1.1.1 Document Collections

We use three TREC test collections extracted from different sources. The Robust 2004 document collection consists of news articles from the Financial Times, the Federal Register, the LA Times, and the Foreign Broadcast Information Service that are part of TREC disks 4 & 5 issued in 2002. In the Gov2 collection, documents were crawled from .gov documents in 2004. ClueWeb-B is a collection of web pages first used in TREC 2009.

The average length of news articles in the Robust 2004 collection are relatively short compared to documents in the other collections. The ClueWeb-B collection contains a higher proportion of low-quality spam documents than other collections. Therefore, we applied a spam filter to documents in ClueWeb-B (Cormack et al., 2011). Approximately 40% of the documents were filtered out.

3.1.1.2 Topics and Relevance Judgements

A TREC collection contains a set of predefined topics, which can viewed as representations of users' information needs, for which retrieval systems are supposed to find documents that satisfy these information needs. Table 3.2 shows a summary of the TREC topics for the three test collections. The topics are designed to reflect information needs that users may have when they use documents in corresponding collections. The topics of the ClueWeb-B collection are general informational queries

Table 3.2. Summary of the TREC topics for the Robust 2004, Gov2 and ClueWeb-B collections. ***Aver. Length*** is the average length of description queries in words. Numbers in parenthesis of are the number of relevant documents

Collection	Documents	# Topics	Aver. Length	# Relevance Judgments
Robust04	News articles	250	7.5	311,409 (17,412)
Gov2	governmental web pages	150	5.7	135,352 (26,917)
ClueWeb-B	Web pages	150	4.8	42,044 (8,754)

because ClueWeb-B contains general web pages. On the other hand, the topics for the Gov2 collection are related to governance and international relationships that users may search for inc government web pages.

The topics for the Robust 2004 and Gov2 collections are made of three types of queries: "title", "description" and "narrative". The "title", "description" and "narrative" queries of a topic represent the same information need. Table 3.3 shows an example of these three types of queries. "Title" queries are short keywords, while a "description" query is a verbose natural language description of the information needs. We focus on the description queries in this dissertation. Although "narrative" queries are also written in the form of natural language expressions, theyh include not only conditions for information to be relevant but also negative conditions about the kinds of information that are not relevant. These negative conditions are not the focus of this dissertation. Therefore, we use description queries from the Robust 2004 and Gov2 collections for experiments.

The topics of the TREC Web Track 2009, 2010, and 2011 for the ClueWeb-B collection consist of the set of "title", "description" and "subtopic" queries. As with the topics of the Robust 2004 and Gov2 collections, "title" queries are keyword queries, while "description" queries are verbose natural language descriptions about users'

Table 3.3. Examples of "title", "description" and "narrative" queries from the TREC Robust 2004 collection.

TREC Topic 643	
Title	salmon dams Pacific northwest
Description	Compile a list of mammals that are considered to be endangered, identify their habitat and, if possible, specify what threatens them.
Narrative	Any document identifying a mammal as endangered is relevant. Statements of authorities disputing the endangered status would also be relevant. A document containing information on habitat and populations of a mammal identified elsewhere as endangered would also be relevant even if the document at hand did not identify the species as endangered. Generalized statements about endangered species without reference to specific mammals would not be relevant.

information needs. "Subtopics" of a topic represent different aspects of information needs, in which the types of subtopics are classified into two categories: information and navigational. "Subtopics" are used to evaluate the diversity of retrieval results.

To evaluate retrieval results, TREC collections provide a set of documents that are manually judged for relevance. Different categories of relevance are used for different tasks. For the Robust 2004 collections, binary relevance judgments (relevant vs. non-relevant) are used, while the Gov2 and ClueWeb-B documents are judged on a graded scale. The Gov2 collection uses a five-point scale of grades for relevant judgments. Documents in the ClueWeb-B collection were judged on a three-point scale as being "relevant", "highly relevant" or "not relevant". We map grades of relevance judgments to binary relevant judgments. Table 3.2 shows the number of documents that are classified as relevant for each collection.

3.1.2 INEX collections

The INitiative for the Evaluation of XML retrieval (INEX) aims at providing an infrastructure to evaluate the effectiveness of focused retrieval systems for XML docu-

Table 3.4. Summary of the test collection of the INEX Ad Hoc track 2009 and 2010. Numbers in parenthesis are the numbers of XML elements or passages.

Collection	# Doc.	# Topic	Aver. # Rel.
INEX 2009	2,666,190	68	71 (117)
INEX 2010	(101,917,424)	52	66 (112)

ments (Gövert and Kazai, 2002). Although the INEX Ad Hoc track claims to support the internal document structure (mark-up) for retrieving relevant information, it also provides topics and relevance judgments for focused retrieval systems based on raw text.

3.1.2.1 Document Collection

From 2009, INEX used a document collection extracted from Wikipedia (Geva et al., 2010). Table 3.4 shows a summary of the three INEX collections that are used in this dissertation. The original Wiki pages were converted into the XML format. The Wiki pages are the English Wikipedia articles dumped on 8 October 2008. In order to use the XML documents for evaluating passage retrieval systems based on raw text, they provide an XML converter to TXT format. [2]

3.1.2.2 Topics and Relevance Judgements

The topics of the INEX Ad Hoc track consist of five types of queries: titles, CAS-titles, phrase-titles, descriptions and narratives. Title queries are simple keyword queries. The content and structure (CAS) title queries specify the relevant XML structure information while phrase-title queries provide the phrasal information in title queries. Description queries are expressed in the form of natural language expressions. We use description queries in this dissertation.

[2] `https://code.google.com/p/inex/`

Table 3.5. Example topics of the INEX Ad Hoc track.

INEX Ad Hoc Track Topic 2009114	
Title	self-portrait
CAS-Title	//painter//figure[about(.//caption, self-portrait)]
Phrase-Title	"self portrait"
Description	Find self-portraits of painters.
Narrative	I am studying how painters visually depict themselves in their work. Relevant document components are images of works of art, in combination with sufficient explanation (i.e., a reference to the artist and the fact that the artist him/herself is depicted in the work of art). Also textual descriptions ...

Relevance judgments were based on XML elements. For evaluating passage retrieval results of raw text fragments, the relevance judgments are also provided in the file-offset-length (FOL) format in which numbers are based on characters.

3.1.3 CQA Collection

In the Yahoo! Answers service, users register questions that consist of titles and descriptions. Other users give answers for a question. The questioners select the best answer for their questions that gives additional incentive to users who write the best answers. Figure 3.1 shows a screen shot of the Yahoo! Answers service.

The community-based QA (CQA) collection, the Yahoo! Answers Comprehensive Questions and Answers version 1.0 [3], was collected from the Yahoo! Answers service. The CQA collection contains about 1M question-answer pairs. A pair consists of a question and its answers in which the best answer of the question is marked.

In the setting of IR tasks, questions and answers are used as queries and documents, respectively. Each answer is a document. In the service, because any user

[3]http://webscope.sandbox.yahoo.com/

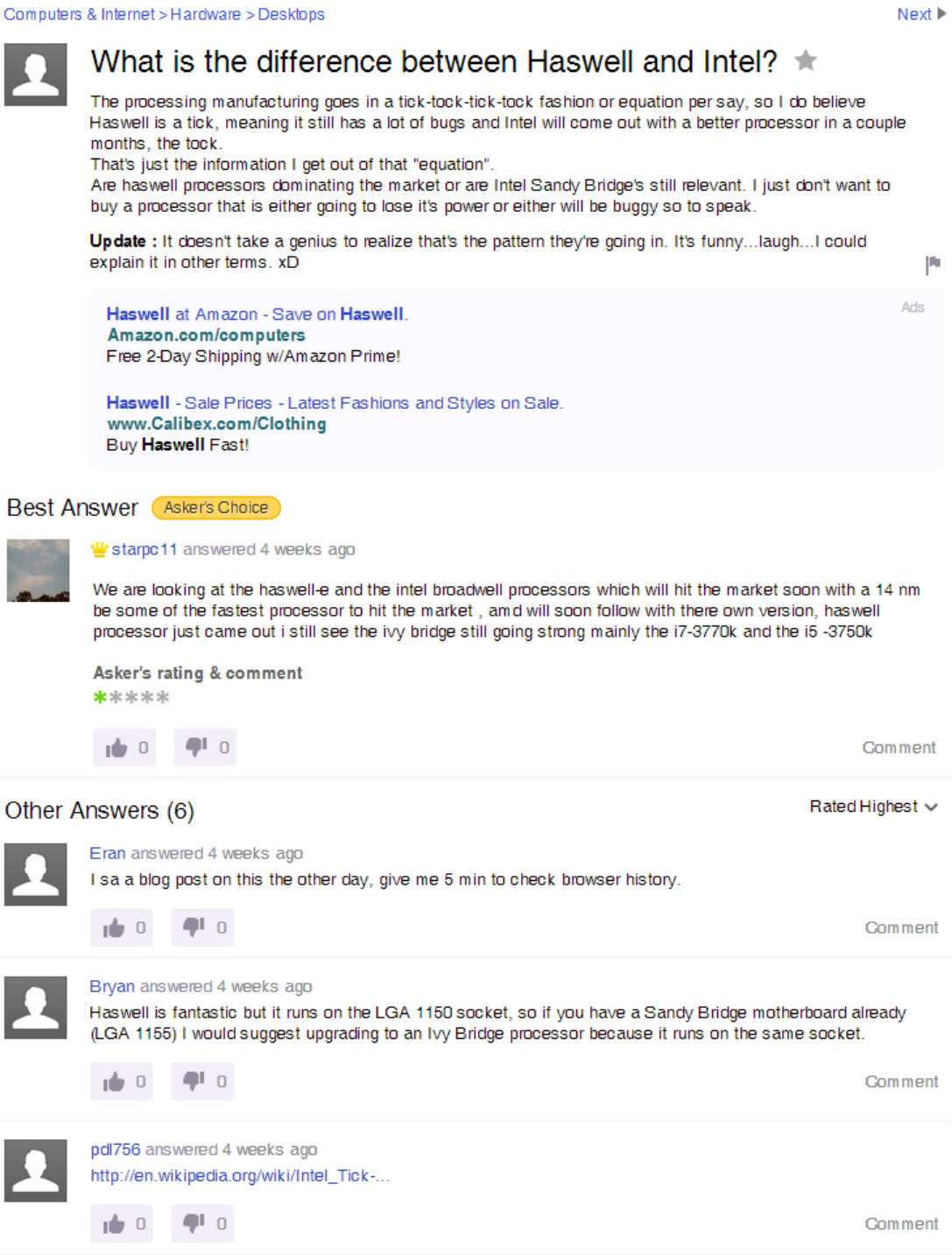


Figure 3.1. Screen shot of the Yahoo! Answers service.

can write an answer, some answers have low quality. Therefore, we only use the best answers that were selected by questioners.

3.2 Evaluation Metrics

We evaluate our methods using several evaluation metrics. These metrics can be classified into two groups according to what kinds of retrieval units are evaluated. For evaluating document-level retrieval results, we use precision at N (Prec@N), average precision (MAP), normalized discounted cumulative gain (nDCG) and reciprocal rank (R-rank). On the other hand, for focused retrieval results, we use character-based average precision (McAP), interpolated average precision (MiAP) and generalized average precision (MgAP).

3.2.1 Document-level Evaluation Metrics

In the case of binary relevance judgments, the set of relevance judgments R consists of the list of relevant documents for a given query Q. The recall and precision of a retrieval system S are measured as follows:

$$\begin{aligned} Precision(S) &= \frac{r}{n}, \\ Recall(S) &= \frac{r}{|R|}, \end{aligned} \tag{3.1}$$

in which n is the number of documents retrieved by S and r is the number of relevant documents in the n retrieved documents. $|R|$ is the number of relevant documents. Recall and precision are two widely used evaluation metrics for evaluating classification results. Compared to the classification problem, retrieval results consist of the ranked list of documents. We assume that users will start to read documents from the top, which means a document at a higher rank has move chanced to be read by users than documents in lower ranks. Therefore, the evaluation metrics of the ranked

list of documents should take account of the ranks of relevant documents in retrieval results.

Precision at N (Prec@N): Users are not able to read the entire retrieval results. They might only be interested in examining up to a certain cutoff k. For example, in the case of web search, users usually read only search results in the first page. *Precision at N* is used to evaluate how many relevant documents a retrieval system S returned in retrieval results that users typically examine. Precision at N is defined as follows:

$$Prec@N(S) = \frac{\sum_{i=1}^{N} R_i}{N}, \tag{3.2}$$

in which R_i is one when ith document is relevant.

Mean Average Precision (MAP): The precision at N only takes into account the top N documents. The problem with the precision at N is that the value of N can differ according to various factors such as the search environment, the characteristics of the task, the types of users and so on. In web search results, N can be the number of documents that are presented to users in the first page of search results. However, for the smaller screen of a smart phone, N should be smaller. In retrieval tasks such as patent search, users are likely to check more documents. Therefore, we still need an evaluation metric for the ranked list of documents in general.

Average precision (AP) can be thought of as a weighted precision measure that gives higher weight to relevant documents that appear near the top of the ranked list. The measure is computed by averaging the sum of the precision at N for every position at N where a relevant document is retrieved in the ranked list of documents where the size is large enough for most kinds of retrieval systems. The top 1,000 documents are typically used for average precision. The average precision measure implicitly accounts for both precision and recall because positions at N for relevant

documents ranked lower than the 1,000 documents are zero. The average precision is defined as follows:

$$AP(S) = \frac{\sum_{k=1}^{1,000} Prec@k}{|R|}, \tag{3.3}$$

in which we compute the average precision in the top 1,000 documents. Mean Average Precision (MAP) represents the mean value of the average precisions of all queries.

Normalized Discounted Cumulative Gain (nDCG): For TREC web corpora that contain graded relevance judgments, it is reasonable to reflect the grades of relevant documents in the evaluation rather than just using binary metrics of precision and recall. The *normalized discounted cumulative gain* (nDCG) was proposed to take account of the grade of relevant documents (Järvelin and Kekäläinen, 2002)

Discounted cumulative gain (DCG) assigns higher scores when a retrieved document has a higher grade as follows:

$$DCG_k(S) = \sum_{i=1}^{k} \frac{2^{rel_i} - 1}{log_2(i+1)}, \tag{3.4}$$

in which rel_i is a grade of a ith document. The *Idear discounted cumulative gain* (IDCG) measures the DCG value when we have an ideal order of relevant documents for a given query. That is, the most relevant documents are ranked at the top of a ranked list, the second most relevant documents are ranked next and so on. nDCG is the normalized value of DCG divided by IDCG as follows:

$$nDCG_k(S) = \frac{DCG_k}{IDCG_k}, \tag{3.5}$$

Mean Reciprocal Rank (MRR): We use the reciprocal rank for evaluating question answer retrieval. MAP uses all retrieved relevant documents by using the sum of precision at N. In the setting of our question answer retrieval task, there is only one relevant document. Therefore, we use the reciprocal rank in order to consider the rank of the answers in retrieval results. Mean Reciprocal Rank (MRR) is defined as follows:

$$MRR(S) = \frac{1}{|Q|} \sum_{i=1}^{|Q|} \frac{1}{rank_i}, \tag{3.6}$$

in which $rank_i$ is the rank of an answer for the ith query.

3.2.2 Passage-level Evaluation Measures

term-based MAP (MAtP) and nDCG (nDCGt): While a document is an explicitly separated unit, there is no single definition of passages that is used consistently in IR experiments. Therefore, evaluation metrics for passage retrieval need to be able to compare different types of passages. Therefore, smaller units such as terms or characters are used for evaluating passage-level retrieval results. Post processed text from the same documents can be differently represented according to tokenizers and stemmers. Therefore, a character-based index of the original text can be used as a basis for passage-level evaluation metrics. The TREC 2004 HARD Track (Allan, 2004) used character-based precision and recall that count each character as retrieved documents. In this thesis, we use the Indri and Lemur toolkit for indexing and retrieval (Strohman et al., 2005) in which passage retrieval results are returned in the offset of terms. Therefore, we use terms as a unit for the evaluation of passage retrieval results. Precision and recall in terms are measured as follows:

$$
\begin{aligned}
tPrec@k(S) &= \frac{\sum_{i=1}^{k} tR_i}{tN}, \\
tRecall@k(S) &= \frac{\sum_{i=1}^{k} tR_i}{|tR|},
\end{aligned}
\tag{3.7}
$$

in which tN and tR represents the number of terms in the retrieval results and relevant passages, respectively. When we count terms, we include stopwords because stopwords are also shown as retrieval results. As with the regular MAP, this measure assigned zero precision for characters not in the rankings. Then, average term-Precision (AtP) is defined as follows:

$$
AtP(S) = \frac{\sum_{k=1}^{1,000} tPrec@k}{|tR|}. \tag{3.8}
$$

In the same way, we measure normalized Discounted Cumulative Gain in term (nDCGt) in Eq. and in which each term is treated a retrieved document.

3.2.3 Inter-Annotator Agreement

Cohen's κ coefficient is used to measures the inter-annotator agreement. Inter-annotator agreement measures are used to estimate the difficulty of tasks and to evaluate the reliability of annotation results. κ takes into account the simple percent of agreement over the agreement occurring by chance as follows:

$$
\kappa = \frac{P(A) - P(E)}{1 - P(E)}, \tag{3.9}
$$

where $P(A)$ is the observed agreement among annotation results and $P(E)$ is the hypothetical probability of chance agreement using the observed data. For example, suppose that we have binary annotation results of annotator A and B as follows:

		A	
		Yes	No
B	Yes	45	15
	No	20	15

The probability of agreement is $P(A) = (45 + 15)/100 = 0.60$. Annotator A tags "*yes*" on the 65% cases and B tags "*yes*" on the 60% cases. Therefore, the probability that both annotator would randomly tag "*yes*" is $0.60 \times 0.65 = 0.39$ and the probability that both annotator would randomly tag "*no*" is $0.40 \times 0.35 = 0.14$. The probability of random agreement $P(E)$ is $0.39 + 0.14 = 0.53$. Cohen's κ is as follows:

$$\kappa = \frac{0.60 - 0.53}{1 - 0.53} = \frac{0.07}{0.47} = 0.15. \tag{3.10}$$

3.2.4 Statistical Significance Test

Most information retrieval experiments compare two retrieval systems: a proposed system A and a baseline system B. As in other scientific experiments, the outcome of an IR experiment can be affected by random errors. As a result, we cannot conclude that a proposed system A is better than a baseline system B based on small differences in performance. We need to determine whether the candidate retrieval system A is indeed better than a baseline retrieval system B, as hypothesized, and whether this difference is statistically significant. To determine statistically significant difference, it is not sufficient just to compare the average of evaluation metrics such as the mean average precision of all queries.

Statistical significance methods are used to compare a candidate system A and a baseline system B. There are statistical significance testing methods that can be used

to compare the effectiveness of retrieval systems including Wilcoxon signed rank test, sign test, t-test and others (Smucker et al., 2007). In this dissertation, we use the *t-test* to evaluate the effectiveness of our methods compared to baseline systems.

The basic idea of the *t-test* is assumed that a proposed system A and a baseline system B are equally good. Under this assumption, we estimate the probability (p_value) that we observe differences between the performances of the two systems. For n queries $q_1, q_2, \ldots, q_n$, we define a random variables as follows:

$$D_i = AP(q_i, A) - AP(q_i, Y), \tag{3.11}$$

in which $AP(q_i, X)$ is the average precision of q_i using a system X. The $t - test$ assumes that $D_1, D_2, \ldots, D_n$ of queries follow the same normal distribution. The assumption T is defined as the t-distribution with $n-1$ degrees of freedom as follows:

$$T = \frac{\overline{D}}{\sqrt{\frac{1}{n-1}\sum_{i=0}^{n}(D_i - \overline{D})^2}}, \tag{3.12}$$

in which $\overline{D}$ is the average of Ds. The p_value is defined as follows:

$$p\_value = 1.0 - F(n-1, T), \tag{3.13}$$

where F is the cumulative distribution function. The smaller the p_value is, the less likely that these two techniques are equally good. If p_value were lower than a threshold α, we could reject the assumption and conclude that there is a statistically significant difference between a proposed system A and a baseline system B.

3.3 Summary

In this chapter, we described data collections that we will use to evaluate proposed methods in this dissertation. In particular, we described the Robust04 , Gov2 and ClueWeb-B TREC collections for the document retrieval task, which consist of different types of documents, topics and information needs. We also described the INEX collection for focused retrieval for focused retrieval task.

In the second part of this chapter, we introduce the evaluation metrics for document and passage-level retrieval results. We introduce *t-test*, a statistical significance test that is used to distinguish between the performance of the retrieval systems throughout this dissertation.

In addition to the TREC and INEX test collection, we also build our own test collection for evaluating the new focused retrieval task of answer passage retrieval. In order to explain the detail information of building this test collection, in the next section, we describe the guideline, the toolkit and the process of answer passage annotation.

CHAPTER 4

ANSWER PASSAGE ANNOTATION

4.1 Overview

A novel task that we address in this thesis is answer passage retrieval. We define *answer passages* as short text fragments from which users can find direct answers for their questions without requiring additional information. As restricted search environments such as smart phones, GPS, voice-based interface, and it may not available for users to read the entire documents in retrieval results, focused retrieval systems draw attention. Focused retrieval systems can help users locate relevant parts in the retrieval results. For example, QA systems return the list of answers instead of the list of documents. Using QA systems, users can find direct answers. However, QA systems are limited to specific types of questions. Although researchers have tried to overcome the limitation of QA systems using manually constructed question-answer data collections, the size of available collections is limited.

On the other end of the research spectrum for the focused retrieval task, passage retrieval systems have been studied. Passage retrieval methods are used in order to provide highly accurate retrieval results by eliminating non relevant text (Allan, 2004). Similarly, XML retrieval (Gövert and Kazai, 2002; Trotman and Geva, 2006) focuses on returning XML elements of structured documents. Passage retrieval models can be used not only for generating passage-level retrieval results but also for providing evidence for document retrieval and QA systems. Although the content of a topically relevant text fragment is definitely related to a user's information need, it can be a

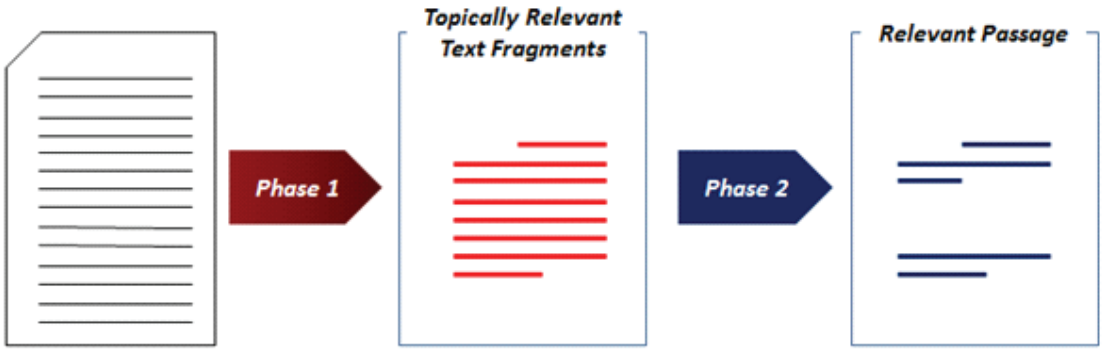


Figure 4.1. The two phases of topically relevant text fragments and answer passages.

mixture of information that is related to a user's information need. Users still have to read relevant text fragments to find answers in retrieved text.

"add reference to the chapter where you study this in more detail" - Do you mean Chapter 2 about the related work? We propose an answer passage retrieval task to overcome the limitations of focused retrieval systems. In this chapter, we introduce an annotation task that is used to help evaluate*answer passage* retrieval systems. We construct a version of the GOV2 data collection where we manually annotate answer passages. Relevance assessment of text fragments for the answer passage retrieval task is conducted in two phases. The first phase of relevance annotation to extract *topically relevant text fragments* from a relevant document. Then, in the second phase, we select answer passages in topically relevant text fragments. Figure 4.3 shows the two phases of topically relevant text fragments and answer passages.

The content of a relevant document may not be perfectly matched with a user's information need. While a document can describe multiple subjects, a user is looking for one of these subjects. Therefore, the annotation of *topically relevant text fragments* will identify passages similar to existing data collection for evaluating passage retrieval systems such as the relevance judgments of the INEX Ad Hoc track. In the topically relevant text fragments, we additionally annotate relevant answer passages that can

immediately provide answers for a user's information need as expressed in the query. These passages will be particularly useful in restricted search environments such as mobile search where the bandwidth for result display is limited.

The rest of this chapter is organized as follows. Section 4.2 describes the guideline for annotating topically relevant text fragments and answer passages. Section 4.3 introduce the overall process of the annotation and an annotation toolkit. In Section 4.4, we show the statistics of annotation results.

4.2 Two Phases of Relevance Judgments

4.2.1 Topically Relevant Text

A *topically relevant text fragment* is a continuous text fragment in a document. One relevant document can have several topically relevant text fragments, separated by significant amounts of non-relevant text. This process is clear when a document describes multiple subjects and only one of these subjects is related to a user's information need. For example, Figure 7.1 shows the part of a document describing examples of hate crimes. For the query "Identify any specific instances of church arson", the text fragment about "church arson" is topically relevant to the user's information need.

These annotations are based on the TREC description and narrative queries, in which a user's information need is described in detail rather than just using keywords. Annotators check the following points:

Is the content of text specific enough for a given topic? It is possible that a document introduces the general idea of a subject while a user has an interest in a more specific focus. Topically relevant text fragments should include only text fragments related to this specific issue.

For example, the query, "What information is available on the involvement of the North Korean Government in counterfeiting of US currency." is asking about a

... one count of Title 18, U.S.C., Section 924(c) (Use of a firearm while committing a crime of violence). Subsequently, Anderson entered into a plea agreement with the government.
On May 5, 1998, Anderson was sentenced to 27 months imprisonment for violating Title 18, U.S.C., Section 247; and 120 months imprisonment for violating Title 18, U.S.C., Section 924(c).

Mobile, Alabama:

On July 1, 1997, St. Joseph Baptist Church was discovered burned to the ground. Shortly thereafter, Tate Chapel A.M.E. Church, located approximately a quarter mile from St. Joseph Baptist Church and on the same rural road, was discovered vandalized with evidence also present of an attempted arson. A joint investigation by the local National Church Arson Task Force was immediately initiated. St. Joseph Baptist Church and Tate Chapel A.M.E. Church host African American congregations.
On July 31, 1997, subjects Alan Odom, Michael Woods, Brandy Boone, and % Kenneth Cumbie were indicted in connection with the arson of St. Joseph Baptist Church. A second count of the indictment charged Alan Odom and Jeremy Boone with the attempted arson of the Tate Chapel A.M.E. Church.
On November 3, 1997, Alan Odom, Brandy Boone, and John Kenneth Cumbie were found guilty of violating Title 18, U.S.C., Section 371 and Odom was also found guilty of violating Title 18, U.S.C., Sections 844(h)(1) and 844(i) regarding the St. Joseph Baptist Church arson. Previously, on October 27, 1997, defendant Michael Woods pled guilty to one count each of the same arson related statutes.

Louisville, Kentucky:

On September 12, 1997, numerous copies of a threatening flyer were found lying in the yard of an African-American family. The family, the only African-Americans living in this small rural community, had ...

Figure 4.2. A sample document describes multiple subjects about hate crimes. Bold faced text is a relevant text fragment to church arson.

specific illegal activity of North Korea. A text about the report of North Korean activities in the international market place is related to the topic of North Korea but the text describes not only the counterfeiting of US currency but also other illegal activities such as drug trafficking, trading nuclear weapon techniques and so on. Topically relevant text fragments have to include only the part of text describing counterfeiting of US currency.

Considering contextual information, does a text fragment satisfy the conditions expressed in a query? Topically relevant text fragments are evaluated from a system-oriented viewpoint. That is, we allow annotators to exploit background knowledge represented by a document but not included in the relevant text fragments.

For example, in the case of the query, "What restrictions are placed on older persons renewing their drivers' licenses in the U.S.?", users are looking for the information for renewing drivers license in the U.S. The following text, which is extracted from a web page from Florida department of highway safety & motor vehicle, satisfies the conditions in different ways.

> ... January 1, 2004, all drivers who are 80 years of age or older must pass a vision test before renewing their driver license. The test may be administered at the driver license office at no additional charge or your licensed health care practitioner, such as your medical doctor, osteopath or a vision examination report must be completed and submitted to the department ...

We know this regulation is limited to the U.S. because the text is extracted from the web site of the Florida government. Annotators are supposed to take account of this contextual information for tagging topically relevant text fragments. This is

a major difference to the criteria for annotating relevant answer passages. We will discuss this issue in more detail in the next section.

Is a condition exclusive or comprehensive? The conditions mentioned in description queries can be classified into two categories: exclusive or comprehensive. The previous example about renewing drivers' licenses specifies exclusive conditions. The condition of "the U.S." limits the relevant information to a certain geographical area. The condition of "renewing" disqualifies text about "getting a new driving license".

On the other hand, comprehensive conditions exemplify relevant information related to the users' information need. For example, in the topic "What kinds of harm do cruise ships do to sea life such as coral reefs, and what is the extent of the damage?", the condition of " such as coral reefs " does not limit the relevant information to "coral reefs". The condition of "the extent of the damage" expands the range of relevant information to the indirect effects of cruise ships.

We emphasize again that the goal of annotating topically relevant text fragments is to find strongly topically relevant text fragments in documents previously marked as "relevant" by filtering out non-relevant or partially relevant content.

4.2.2 Answer Passages

A topically relevant text fragment is similar to raw mineral ore before processing. The content of a topically relevant text fragment is definitely related to a user's information need. However, it can be a mixture of various information that is related to a user's information need. Users will still have to read relevant text fragments to find answers in that text. In this next step, we will identify answer passages that are succinct answers to a user's question.

While the concept of relevance for a topically relevant text passage is system-oriented, the concept of relevance for an answer passage is based on a user-centric

viewpoint. In Section 4.2.1, the fact that a text fragment is extracted from the web page of Florida governmental web site is contextual information. For annotating topically relevant text fragments, this contextual information can be obtained from characteristics of the document that may not be explicit in the actual text. On the other hand, the annotation of answer passages should be based strictly on the text content.

A topically relevant text fragment can contain zero, one, or more answer passages. The size of an answer passage will vary according to the characteristics of topics and the content of relevant text. For a simple factoid question such as "when was Mozart born?", an answer passage could be a single sentence. Our target queries generally require more complex answers. Generally, we expect the size of an answer passage to be several contiguous sentences (i.e., 2-4). However, this is flexible and annotators can tag more or less sentences as an answer passage based on their judgment.

Annotators identify answer passages using three criteria: *completeness*, *conciseness* and *unity*.

Completeness of an answer passage means that a user, using his or her own background knowledge, can find an answer without additional information or inference. For example, if a user is looking for information about church arson in the following text,

(a) St. Joseph Baptist Church was discovered burned to the ground.

(b) Tate Chapel A.M.E. Church, located approximately a quarter mile from St. Joseph Baptist Church and on the same rural road, was discovered vandalized with evidence also present of an attempted arson.

With only (a), an annotator cannot conclude that it is spontaneous combustion or church arson. On the other hand, (b) explicitly mentions arson. Therefore, the

annotator decides that (a) is an example of church arson. Therefore, both (a) and (b) are tagged together as an answer passage.

TREC description queries express various conditions about relevant information. Annotators are supposed to check whether an answer passage satisfies these conditions. The example topic of Section 2.1, "What restrictions are placed on older persons renewing their drivers' licenses in the U.S.?", specifies the following conditions:

- Renewing drivers licenses
- In the U.S.
- Older person

In terms of answer passages, consider the same text extracted from a web page from Florida Department of Highway Safety & Motor Vehicle.

> ... January 1, 2004, all drivers who are 80 years of age or older must pass a vision test before renewing their driver license. The test may be administered at the driver license office at no additional charge or your licensed health care practitioner, such as your medical doctor, osteopath or a vision examination report must be completed and submitted to the department if your vision test is administered by your doctor.

Annotators need to check that an answer passage explicitly satisfies each condition. The example text satisfies the first condition by " ... renewing their driver license ...". On the other hand, as mentioned in the previous section, contextual information is needed to know that this regulation is limited to the U.S. Therefore, the example text does not explicitly satisfy the second condition as an answer passage.

The definition of older person is a subjective concept in general and the example text defines this condition by mentioning "80 years of age or older." This is only valid for Florida. On the other hand, the condition of other states for renewing drivers licenses is as follows:

- **Missouri**: To all applicants for a license or renewal to transport persons or property classified in section 302.015 who are at least twenty-one years of age and under the age seventy,
- **Colorado**: a fee of three dollars and fifty cents at the time of application for an identification card or renewal of an identification card; or three dollars and fifty cents for a duplicate card; except that, for applicants sixty years of age or older and applicants referred by any county department of social services pursuant to section
- **Oregon**: This rule establishes the requirement for a vision check every eight years for a person 50 years of age or older. The amendments resolve conflicts with OAR 735-062-0050 and

Therefore, annotators must take account of possible variations that satisfy a condition according to contextual information. But, we assume that it does not require users inference because users have background knowledge about the definition of "older" and, so, they can compose the description query, or the definition of "older" itself is the part of their information needs.

Conciseness of answer passages indicates that there is little or no irrelevant information in the answer passages. Consider the following sentence for the query "What information is available on the involvement of the North Korean Government in counterfeiting of US currency."

(a) In addition to seeking a solution through multilateral diplomacy, the United States, working with other countries, has taken steps to curtail dangerous and

illicit North Korean activities such as drug smuggling, counterfeiting, and trade in WMD and missiles.

Although (a) mentions counterfeiting, it is just one of the illegal activities of North Korea. Therefore, (a) is not concise. Note that this text may still be the best answer passage.

Unity of answer passages means that the content of a answer passage consists of a single instance of an answer for a query. For the topic "Identify any specific instances of church arson.", one topically relevant text fragment consists of a series of incidents as follows:

> ... Pilot Knob Lutheran Church - July 3, 2000 **The first fire occurred at 11:20 p.m. on July 3, 2000 at the Pilot Knob Lutheran Church, located in Hancock County, and having a RR Forest City address**. Somber Lutheran Church - July 4, 2000 **The second fire occurred at 1:05 a.m. on July 4, 2000 at the Somber Lutheran Church, located in Worth County, and having a RR Lake Mills address**. Bethel Lutheran Church - July 29, 2000 **The third fire occurred at 1:13 a.m. on July 29, 2000 at the Bethel Lutheran Church, located in Worth County, and having a RR Joice address.** ...

For this topically relevant text fragment, annotators should tag each instance separately as answer passages. Consider another example query "Describe the Javelina or collared peccary and its geographic range."

... spread simultaneously with the replacement of Arizona's native grasslands by scrub and cactus. **The collared peccary has one of the greatest latitudinal ranges of any New World game animal, occurring from Arizona to Argentina.** *The range of the peccary is still expanding, primarily northwestward. The collared peccary, which occurs in the United States only in Arizona, Texas, and New Mexico, currently occupies approximately 34 percent of Arizona with an estimated population of 60,000 animals ...*

The current geographic range (Boldfaced) and expected geographic range (Italic) of the collared peccary is tagged as separate answer passages. The current geographic range describes information about the geographical range of the collared peccary while the expansion of its geographical range tells a different story.

One of the challenging issues in annotating answer passages is that there may be only a few complete and concise answer passages. As shown by the example topic about renewing drivers license, answer passages cannot explicitly satisfy some conditions. Therefore, we relax the criteria of completeness and conciseness of answer passages using four classification categories for passages as follows:

- **PERFECT**: A passage is complete and concise.
- **EXCELLENT**: A passage requires only simple inference that can be made by users using their background knowledge. For this case we assume that a user has enough background knowledge to derive an answer. When there is still some unrelated information in a passage, we classify a passage as EXCELLENT.
- **GOOD**: The passage requires more extensive inference to derive an answer.

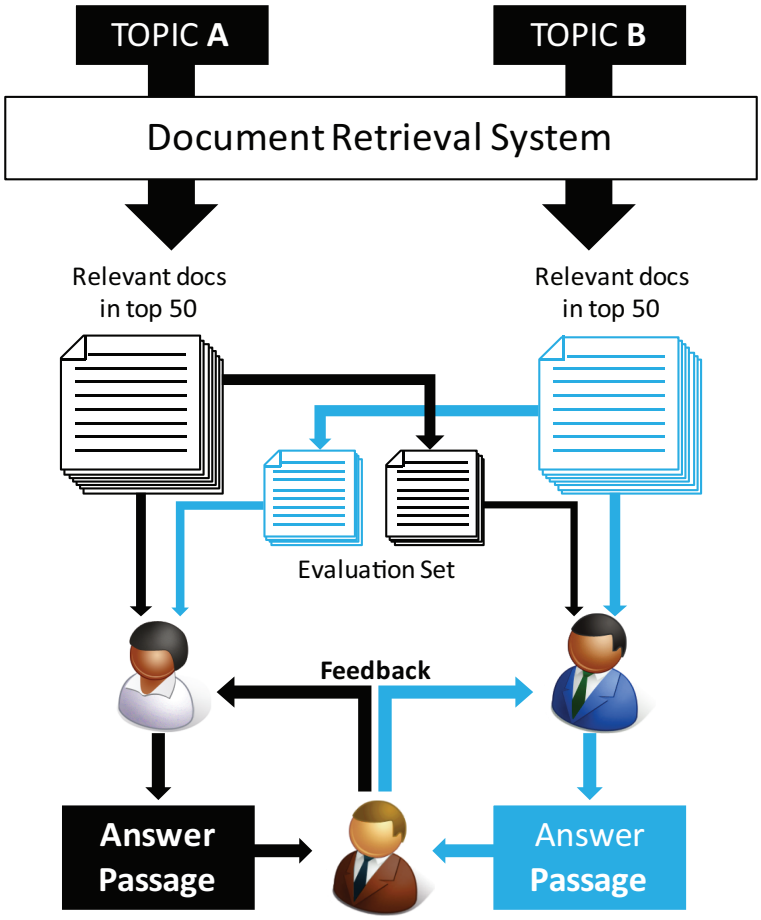


Figure 4.3. The process of the annotation task.

- **FAIR**: A user can guess that a passage is an answer for his or her information need. But, in order to confirm that it is an answer, a user has to read the entire document or other documents.

4.3 The Process of Answer Passage Annotation

For the answer passage annotation, we retrieved the top 50 documents for each topic retrieved using SDM. We annotated relevant text fragments and answer passages for these documents. Figure 4.3 shows the process of answer passage annotation. We hired three undergraduate students to annotate answer passages. Annotators were assigned per topic, which means that one annotator was supposed to tag all relevant

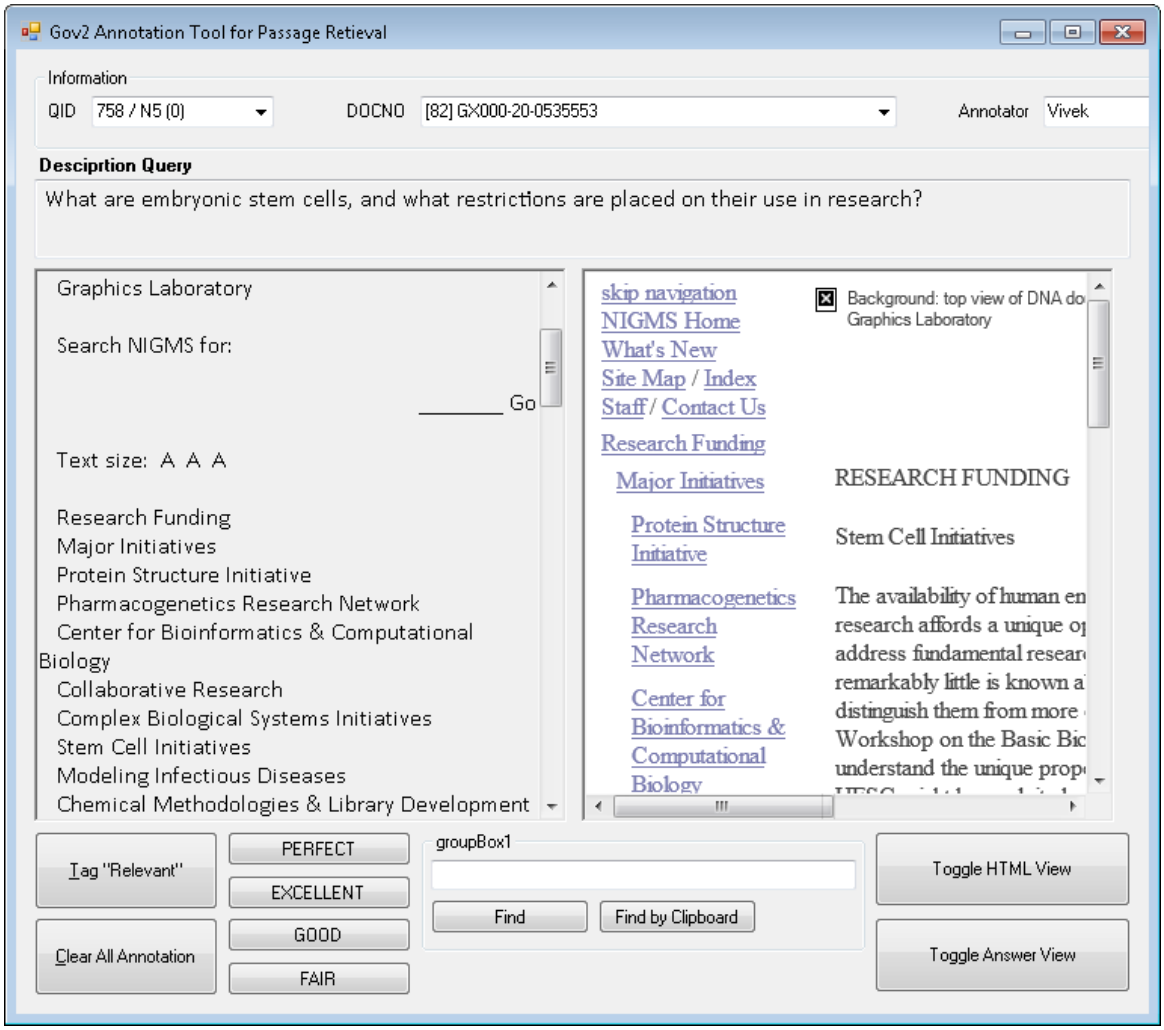


Figure 4.4. Screen shot of the annotation toolkit.

documents for one topic. Annotation results were checked by proofreader who gives feedback to annotators. Proofreaders gave comments on annotation results instead of directly modifying the results. In addition, the top 5 relevant documents for each topic were annotated by another annotator in order to validate the annotation results.

Figure 4.4 is a screen shot of the annotation toolkit. Annotators tag relevant text fragments and answer passages in raw text that was generated from HTML files using

Table 4.1. Statistics of annotation results. *Relevant* and *Answer* represent relevant text fragments and answer passages, respectively. *# Relevant* is the number of relevant items in each unit. *Length* is the sum of words in relevant units.

		# Relevant	Length
Document		2,380	11,009,861
Relevant		3,479	1,564,086
Answer	PERFECT	4,258	180,395
	EXCELLENT	2,940	146,295
	GOOD	731	30,990
	FAIR	150	6,105
	TOTAL	8,079	363,785

a text-based web browser[1]. In addition to raw text, the annotation toolkit shows an original HTML page, as seen in the right pane of the annotation toolkit example.

4.4 Annotation Results

From 150 topics of the Gov2 collection, we remove three topics that there is no relevant document in the top 50 documents retrieved using SDM. We also remove queries requiring lists or totals as answers and vague queries that obviously require longer answers. We annotated answer passages for 110 topics of the Gov2 collection. Table 4.1 shows the annotation results. Among the 5,500 documents, 2,380 documents are relevant. The average length of relevant documents is 564.2 words. 188 documents do not have relevant text fragments. For example, for the topic "*What was the role of Portugal in World War II?*", annotators does not tag a relevant text fragment to a document that just mentions about Portugal as follows:

[1]http://en.wikipedia.org/wiki/Lynx_(web_browser)

> ... Germany's wartime trade, with only brief mention of other countries, the supplement, entitled U.S. and allied wartime and postwar relations and negotiations with Argentina, Portugal, Spain, Sweden, and Turkey on looted gold and other assets stolen or hidden by Germany during World War II, ...

Annotators select 14.2% (1.5M words) of relevant documents as relevant text fragments. The average number of relevant text fragments per document is 1.46.

8,079 answer passages were tagged. There are 4,258 (52.7%) perfect answers, 2,940 (36.4%) excellent answers, 731 (9%) good answers and 150 (2%) fair answers. The proportion of good and fair answers is lower than perfect and excellent answers. Annotators tend to avoid taggin good and fair answers because it is hard for them to recognize relevant information in these types of text fragments. The average length of answer passages is 45.4 words, including stopwords.

To evaluate the annotation results, we measure the inter-annotator agreement of relevant text fragment and answer passage annotation results using κ coefficient in Eq. 3.10 in word. The inter-annotator agreement is used to measure the quality of annotation results and the difficulty of an annotation task. In the inter-annotator agreement of answer passages, we did not consider the categories of answer passages. The κ of relevant text fragments is 0.45 while the κ of answer passages is 0.29. The κ coefficient of answer passages is relatively lower than that of relevant text fragments because annotators tend to select one answer passage per relevant text fragment. In particular, when a relevant text fragment describes detailed information about a specific topic, there might be several candidate sentences which can be used as answer passages. For example, consider the basic structure of a well-organized paragraph. A paragraph starts with topic sentences. The body of a paragraph then describes about this topic in detail. The last sentences of a paragraph summarize the topic. When this paragraph is tagged as a topically relevant text fragment, both the first and last

sentences would be good candidates for answer passages. However, we observe that annotators selected one of them as answer passages and ignored the others to avoid the repetition of the same answers. The sentence-lvel annotation results of the TREC novelty track showed similar tendencies Harman (2002). If annotators had selected different parts of relevant text fragments to tag answer passage, the annotation results would totally disagree. This decreases the inter-agreement of annotation results of answer passages.

4.5 Summary

In this chapter, we describe the annotation of answer passages. We annotate relevant text in two levels. First, relevant text fragments similar to existing passage-level relevant judgments are annotated. Then, annotators tag answer passages within relevant text fragments. For annotating answer passages, There are three criteria: completeness, conciseness and unity. Using these criteria, we aim to select answer passages that can satisfy users' information needs without external information while being succinct enough to be used for restricted search environments.

CHAPTER 5

QUASI-SYNCHRONOUS FRAMEWORK

5.1 Overview

Term dependence models for IR have been intensively studied to consider the term dependencies and even concept relationships in verbose queries. Terms are used together to express specific concepts of which meanings can differ from the meanings of individual terms. Concepts are used together in a grammatically well-formed phrase or sentence that represents the relationships between concepts. Term dependence models need to recognize related terms and concepts from queries and be able to match these concepts to documents.

However, existing dependence models rely on a limited set of dependencies, such as adjacent term pairs (Metzler and Croft, 2005; Srikanth and Srihari, 2002) in queries or head-modified pairs (Gao et al., 2004; Lee et al., 2006) in the parsing results of queries. Other alternatives assume arbitrary dependencies between any pair of terms in queries (Metzler and Croft, 2005; Rasolofo and Savoy, 2003). Adjacent term pairs are useful to identify the relationships between terms in the same phrase, but they cannot cover the dependence relationships between concepts. On the other hand, arbitrary term pairs include not only dependent term pairs but also unimportant or even harmful dependent terms.

Syntactic parsing techniques are used to identify term dependencies to overcome the limitation in the dependence assumption between adjacent terms (Gao et al., 2004; Lee et al., 2006; Song et al., 2008). However, because previous work used only the head-modifier relations, term dependence models based on syntactic pars-

ing results are still deficient in identifying term dependencies of verbose queries at a longer distance. Moreover, this previous work takes account only of matching between the head-modifier term pairs in queries and documents. They are less flexible towards considering variations in relationships of dependent term pairs compared to the unordered-window potential function in Eq 2.5. Therefore, term dependence models based on syntactic parsing results failed to show consistent improvement over retrieval models based on term proximity.

In this dissertation, we propose a term dependence model inspired by the quasi-synchronous stochastic process developed by (Smith and Eisner, 2006). Synchronous grammars were proposed for machine translation to generate translated expressions or identify translation examples by aligning a parse tree in a source language to a parse tree in a target language (Shieber and Schabes, 1990). Because of inherent incompatibility between a source language and target language, syntactic and lexical variations occur during translating from a source sentence to a target sentence. Thus, a synchronous model should be able to align a translation unit in a source language to different forms than that of the original (Gupta and Chatterjee, 2001). Smith and Eisner suggested several different types of syntactic configurations to which the head-modifier term pairs in source sentences can be aligned in target sentences. Using these predefined syntactic configurations for identifying dependent term pairs, we model dependence relationships at a longer distance. In addition, we take account of the transformations of dependence relationships between queries and document by allowing matching between different syntactic configurations.

The quasi-synchronous framework unifies the quasi-synchronous stochastic process with existing dependence models. For the quasi-synchronous framework, we select appropriate dependence assumptions according to not just the importance of individual terms but rather the stability of dependent terms and concepts. For example, the dependence relationships of terms in proper nouns are more stable than other terms.

It means that terms in proper nouns will be used in the same form by queries and documents. Similarly, dependencies of terms within a concept are stronger and more stable than dependencies between concepts. The quasi-synchronous framework analyzes the characteristics of verbose queries. We use the analyzed results for assigning suitable weights on individual dependence relationships.

The remainder of this chapter is organized as follows. First, we define the basic structure of the quasi-synchronous framework. Then, we describe the quasi-synchronous stochastic process in detail. In Section 5.3, we explain supervised methods to predicate optimal parameter values that are used in the quasi-synchronous framework to unify multiple retrieval models. Section 5.4 describes experimental settings and evaluation measures for evaluating the effectiveness of the quasi-synchronous framework. We give the experimental results and analysis in Section 5.5.

5.2 Quasi-Synchronous Framework

Like the language model framework, the basic idea of the quasi-synchronous framework is to rank a document using the probability that a query is generated by the document model. However, we infer a document model from the dependency tree T_D of a document rather than the raw term sequence D as in the dependence language model (Gao et al., 2004). The document model generates not an individual term or a dependent term pair but a fragment of the parsing tree T_Q of a query Q through the loose alignment A.

$$\begin{aligned} P_{Quasi}(Q, D) &\approx P(T_Q, A|T_D) \\ &= P(A|T_D)P(T_Q|T_D, A), \end{aligned} \tag{5.1}$$

in which we estimate the probability that a query Q is generated from a document model D using the parsing results of a query T_Q and a document T_D. The loose align-

ment A intermediate different syntactic relationships between queries and documents. Dependence relationships in queries and documents can be different. Therefore, the conditional probability of $P(A|T_D)$ synchronizes the different dependence relationships between queries and documents as follows:

$$A = \{(syn_Q, syn_D)|syn_Q \in SYN, syn_D \in SYN\}, \tag{5.2}$$

in which SYN is the set of dependence relationships that dependent term pairs can have in queries and documents. SYN is defined based on the dependence assumptions of term dependence models in the quasi-synchronous framework. For example, we express the SDM (Metzler and Croft, 2005) in terms of the quasi-synchronous framework. The set of dependence relations for queries SYN_Q and documents SYN_D consists of the following dependencies:

$$\begin{aligned} SYN_Q^{SDM} &= \{syn_{ql}, syn_{seq}\}\text{and} \\ SYN_D^{SDM} &= \{syn_{ql}, syn_{\#ow1}, syn_{\#uw8}\}. \end{aligned} \tag{5.3}$$

with which the set of dependent terms of queries and documents are defined as follow:

$$\begin{aligned} T_{Q,syn_{ql}} &= \{q_i|\forall 1 < i < m\}, \\ T_{Q,syn_{seq}} &= \{(q_i, q_{i+1})|\forall 1 \leq i \leq m-1\}, \\ T_{D,syn_{ql}} &= \{w_i|\forall 1 < i < m\}, \\ T_{D,syn_{\#ow1}} &= \{(w_i, w_{i+1})|\forall 1 \leq i \leq n-1\} \text{ and} \\ T_{D,syn_{\#uw8}} &= \{(w_i, w_j)|\forall 1 \leq i, j \leq n-1, |i-j| < 8\}, \end{aligned} \tag{5.4}$$

in which ordered and unordered-window potential functions use the window sizes of one and eight, respectively.

The language model in Eq 5.1 is decomposed according to the syntactic relationships of query terms as follows:

$$\begin{aligned} & P(A|T_D)P(T_Q|T_D,A) \\ & = \prod_{syn_Q \in SYN} \prod_{(t_i,t_j)\in T_{Q,syn_Q}} P(A|T_D)P(t_i,t_j|T_D,A), \\ & \overset{rank}{\approx} \sum_{syn_Q \in SYN} \sum_{(t_i,t_j)\in T_{Q,syn_Q}} logP(A|T_D)P(t_i,t_j|T_D,A), \end{aligned} \tag{5.5}$$

in which T_{Q,syn_Q} is the set of term pairs that have a syntactic relation syn_Q in a query. The loose alignment A represents a loose alignment between syn_Q and syn_D that are the syntactic relationships of dependent term pairs (t_i, t_j) in a query and documents, respectively. In this dissertation, we use the sum of log probabilities.

Eq. 5.5 for the specific combination of a loose alignment $A = (syn_Q, syn_D)$ is defined as follows:

$$\begin{aligned} & P(A|T_D)P(t_i,t_j|T_D,A) \\ & \quad = \sum_{syn_D \in SYN} P(syn_Q,syn_D|T_D)P(t_i,t_j|T_{D,syn_D}), \end{aligned} \tag{5.6}$$

where T_{D,syn_D} is a document model that is inferred from term pairs having a syntactic relationship syn_D in a document D. $P(t_i, t_j|T_{D,syn_D})$ is the probability that the document model T_{D,syn_D} generates a term pair (t_i, t_j) in a query. $P(t_i, t_j|T_{D,syn_D})$ is computed using the language model with smoothing as follows:

$$\begin{aligned} & P(t_i,t_j|T_{D,syn_D}) \\ & \quad = \alpha \frac{tf_{t_i,t_j,syn_D}}{|D|} + (1-\alpha)\frac{cf_{t_i,t_j,syn_D}}{|C|}, \end{aligned} \tag{5.7}$$

where tf_{t_i,t_j,syn_D} and cf_{t_i,t_j,syn_D} are the term frequency of term pairs t_i and t_j with the syntactic relation syn_D in a document D and a collection, respectively. α is the parameter for smoothing a document model using a collection model.

In the next section, we describe the quasi-synchronous stochastic process and derive the loose alignment model $P(syn_Q, syn_D|T_D)$ using the predefined syntactic configurations of the quasi-synchronous stochastic process.

5.2.1 Quasi-Synchronous Stochastic Process

Synchronous grammars, originally proposed for machine translation (Shieber and Schabes, 1990), jointly generate trees of a source and target sentence. Depending on the size and complexity of the rewrite rules in a synchronous grammar, the source and target trees can diverge more or less in their structures.

Smith and Eisner (2006) pointed out the problem in the quasi-synchronous process that the parsing trees in a target language do not always perfectly match with the parsing tree of a source language. Therefore, the synchronous process must relax the requirement of synchronous grammar formalism. Smith and Eisner introduce methods proposed by previous work to allow exceptions in the synchronous process. For example, there are methods on an unaligned node (Yamada and Knight, 2001), alignment of duplicated children (Gildea, 2003), alignments between elementary tree in different size using multiple rules(Ding and Palmer, 2005; Eisner, 2003; Melamed et al., 2004). Even for the translation of similar languages such as English, German and French that belong to the same language tribe, it is required for synchronous grammars to take account of possible linguistic divergences between a source language and a target language.

Smith and Eisner proposed the quasi-synchronous stochastic process in order to allow the synchronous grammar be able to cover any permutations as the IBM translation models 3-5 (Brown et al., 1993). Figure 5.1 demonstrates the example of inexact

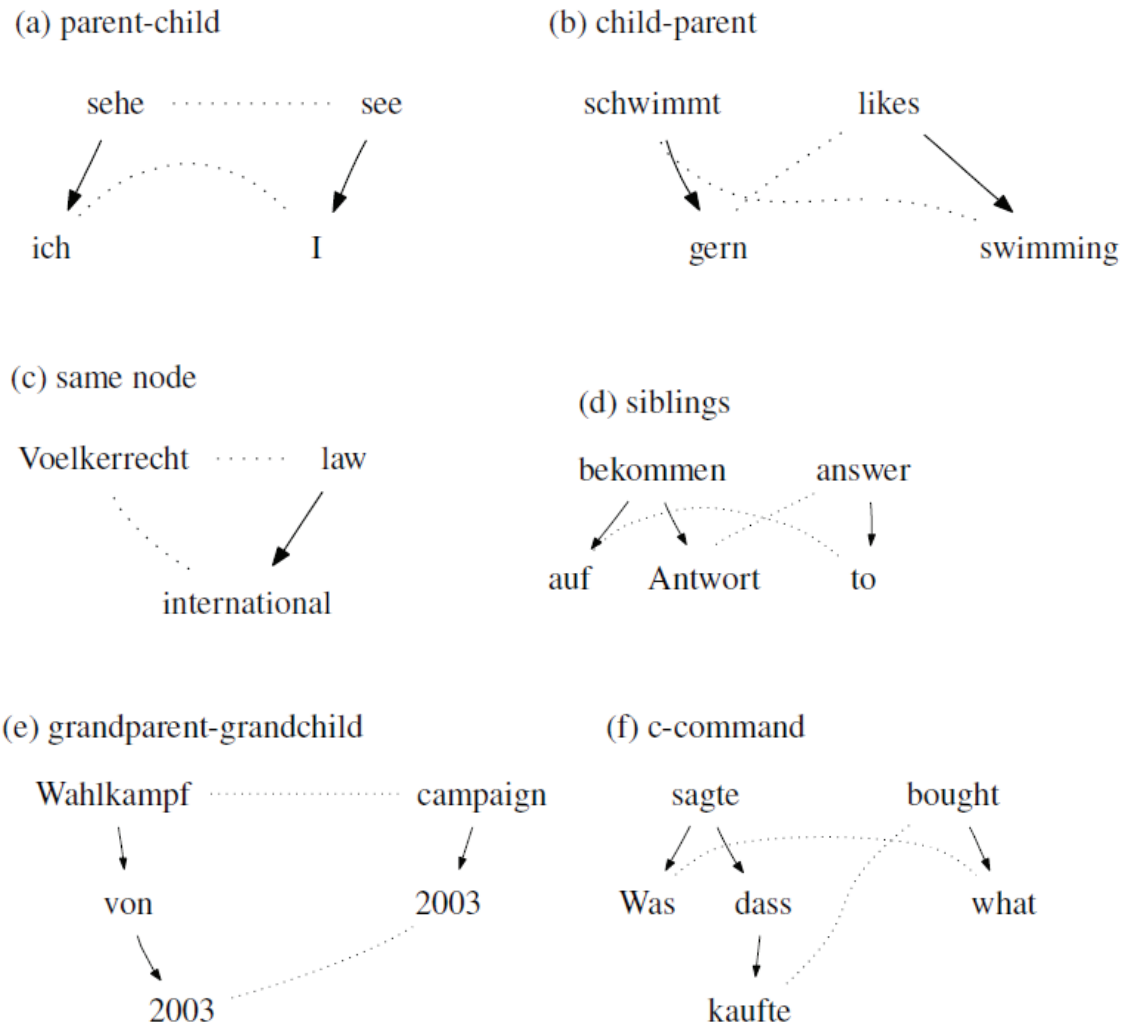


Figure 5.1. Example of the quasi-synchronous alignment (Smith and Eisner, 2006) of the parent-child term pair in the source sentence to six dependence relationships in the target sentence.)

alignments from the parent-child term pairs in source sentences to term pairs having six different dependence relationships in target sentences. In these examples, a German word "voelkerrecht" are expressed a noun phrase "international law" in English. Figure 5.1.(b) shows an alignment between an English phrase "like swimming" and a German phrase in which a direct object "swimming" and a verb "likes" are aligned to a verb "schwimmt" and a adverb "gern". In Figure 5.1.(e) and (f), English phrases are expressed in different sentences with additional functional words in German.

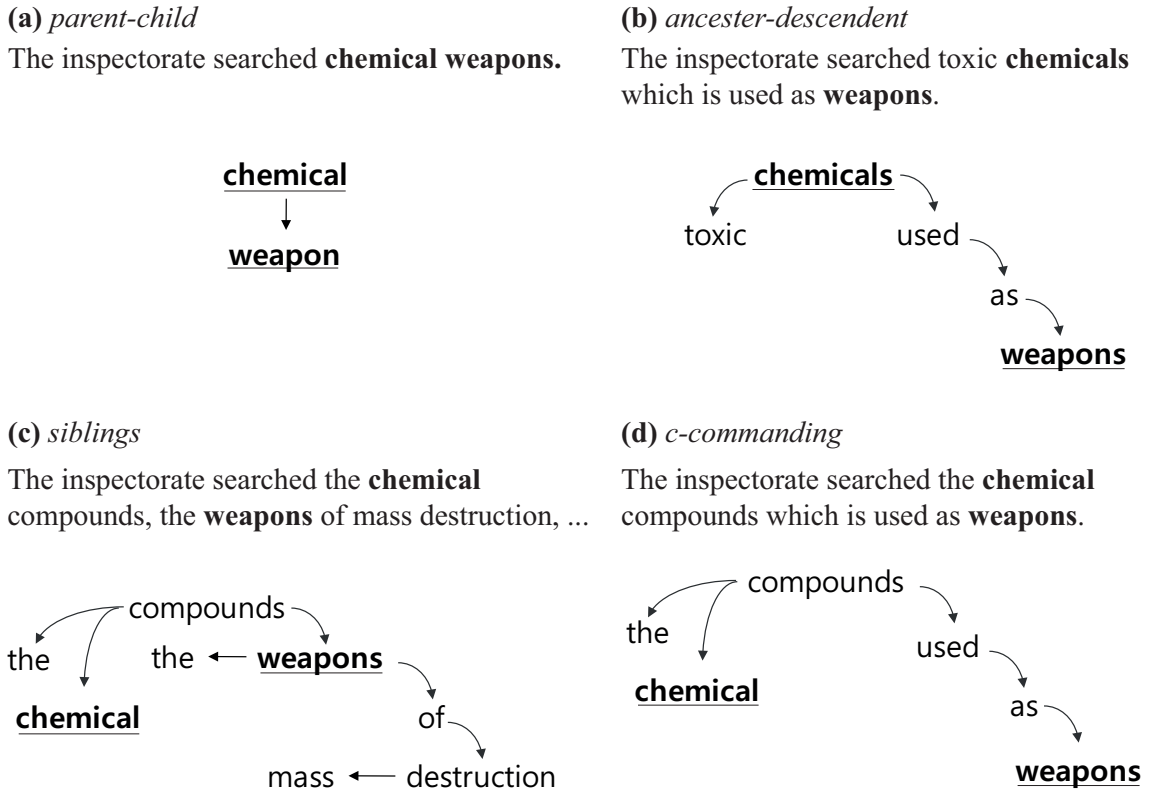


Figure 5.2. Four types of syntactic dependency configurations in the quasi-synchronous stochastic process. In the quasi-synchronous model matches terms in queries and documents along with transformation between these dependence relations: (a) parent-child, (b) ascendant-descendant, (c) siblings, and (d) c-commanding.

This kind of a variation can happens even within English. For example, a noun phrase "information retrieval" can be expressed a clause "retrieve information". This inexact matching process of the quasi-synchronous model has also shown significant improvements for other tasks such as open domain QA (Das and Smith, 2009) and paraphrasing (Wang et al., 2007). The processes of selecting answer sentences and paraphrased sentences are interpreted as a free translation process between sentences in the same language. In a similar way, we adopt the quasi-synchronous stochastic approach and generalize it for information retrieval, where a target sentence (a query) is generated from a set of sentences (a document) instead of a single source sentence.

Compared to previous work, the quasi-synchronous framework for IR has two different settings in the usage of the quasi-synchronous stochastic process. First, we do not consider the alignment from dependent term pairs to individual terms that is shown in Fig 5.1.(c) because the alignment to individual terms can be covered by a retrieval model using the independence assumption. Second, the quasi-synchronous models for machine translation, paraphrasing and QA consider the head-modifier relation in source sentences or queries and predefined syntactic configurations are used for identify dependent terms from target sentences or answers. On the other hand, we use the predefined syntactic configurations for both queries and documents.

In the quasi-synchronous framework, four syntactic configurations are used as follows:

- Parent-Child
- Ancestor-Descendent [1]
- Siblings
- C-Commanding [2].

Figure 5.2 depicts the examples of the above four relationships for terms, "*chemical*" and "*weapon*". The parent-child relation represents a direct relationship in a parsing tree. The ancestor-descendent is the expanded relation of the parent-child in which terms between the root of a tree to a given term are the ancestors of the given term. In the Maxwell et al. (2013) approach, dependence paths corresponded to parent-child and ancestor-descendent relations.

In addition to the parent-child and ancestor-descendent relations, the predefined syntactic configurations include siblings and c-commanding relationships. Terms

[1]We expanded the syntactic configuration, "grandparent-grandchild", in the original work.

[2]In this thesis a term has a c-commanding relation with terms which are descendants of the given term's parent node.

sharing the same parent node are siblings. In the siblings relation, a term t has the c-command relation with terms whose ancestors are the parents of the term t Haegeman (1991). The siblings and c-command relations represent term dependencies across phrases and clauses.

Because we distinguish the order of terms in the relationships, there are eight relationships that a term pair can have in queries and documents as follows:

$$SYN^{quasi} = \{syn_{PC}, syn_{AD}, syn_{SS}, syn_{CC}, syn_{PC-R}, syn_{AD-R}, syn_{SS-R}, syn_{CC-R}\}, \tag{5.8}$$

in which syn_{PC-R}, syn_{AD-R}, syn_{SS-R} and syn_{CC-R} represents the reverse order of syn_{PC}, syn_{AD}, syn_{SS}, syn_{CC}, respectively.

We use these predefined syntactic configurations to identify dependent relationships from verbose queries beyond the head-modifier relation and consider their variations in relevant documents. Then, we define the loose alignment model in Eq. 5.6 with the predefined syntactic configurations in which we compare the exact and inexact matching approaches for the quasi-synchronous model.

5.2.2 Loose Alignment Model for Quasi-Synchronous Framework

The loose alignment model $P(A = (syn_Q, syn_D)|T_D)$ in Eq. 5.8 represents the probability that dependent term pairs having a syntactic relationship syn_Q in queries will have a syntactic relationship syn_D in relevant documents. The alignment model for the SDM with fixed interpolation weights can be defined as follows:

$$P^{SDM}(syn_Q, syn_D|T_D) = \begin{cases} \frac{0.85}{|T_Q,ql|} & \text{if } syn_Q = syn_{ql} \text{ and } syn_D = syn_{ql} \\ \frac{0.10}{\cdot|T_Q,seq|} & \text{if } syn_Q = syn_{seq} \text{ and } syn_D = syn_{\#ow1} \\ \frac{0.05}{\cdot|T_Q,seq|} & \text{if } syn_Q = syn_{seq} \text{ and } syn_D = syn_{\#uw8} \\ 0 & \text{otherwise,} \end{cases} \tag{5.9}$$

in which $|T_Q, ql|$ and $|T_Q, seq|$ are the number of terms and adjacent term pairs, respectively. $P(syn_Q, syn_D|T_D)$ can be used to linearly interpolate the quasi-synchronous model with the SDM in the quasi-synchronous framework as follows:

$$P(syn_Q, syn_D|T_D) = \begin{cases} \alpha P^{SDM}(syn_Q, syn_D|T_D) & \text{if } syn_Q \in SYN^{SDM} \\ (1-\alpha)P^{quasi}(syn_Q, syn_D|T_D) & \text{if } syn_Q \in SYN^{quasi}, \end{cases} \tag{5.10}$$

in which α is a weight assigned to the SDM and the rest of weight $(1-\alpha)$ is assigned to the quasi-synchronous model. Then, the weight assigned to the SDM is redistributed to potential functions of the SDM by Eq. 5.9.

The conditional probability of the alignment model can be separated as follows:

$$P(A = (syn_Q, syn_D)|T_D) = P(syn_Q|T_D)P(syn_D|T_D, syn_Q), \tag{5.11}$$

in which $P(syn_Q|T_D)$ represents the weight of a specific dependent assumption that assigns weights to dependence models in the quasi-synchronous framework. We will describe a method to predict $P(syn_Q|T_D)$ according to a given query.

On the other hand, $P(syn_D|T_D, syn_Q)$ is the probability that dependent term pairs having a syntactic relation syn_Q will have a dependence relationship syn_D in relevant

What is the prognosis *for* new drugs?
Find *ways of* measuring creativity.
What are commercial *uses of* Magnetic Levitation?
What drugs *are being used in the* treatment *of* Alzheimer's Disease
What are the arguments *for and against* Great Britain's approval *of* women *being* ordained *as* Church *of* England priests?
What are the industrial *or* commercial uses *of* cyanide *or its* derivatives?

Figure 5.3. Example queries from the Robust 2004 collection which demonstrating better results when assigning more weight to the query likelihood model, the sequential dependence model and the quasi-synchronous model, respectively.

documents. We adopt the quasi-synchronous process in order to not only identify dependent term pairs from verbose queries beyond the head-modifier relation, but also consider variations in dependence relationships between queries and documents. In order to evaluate the effectiveness of modeling variations in dependence relationships, we compare the two settings of $P(syn_D|T_D, syn_Q)$. First the exact matching approach of the alignment model is defined as follows:

$$P^{exact}(syn_D|T_D, syn_Q) = \begin{cases} 1 & syn_D = syn_Q \\ 0 & Otherwise, \end{cases} \tag{5.12}$$

in which the alignment model ignores when the syntactic relationships of dependent terms in documents is not same as the syntactic relationships in queries. On the other hand, the inexact matching approaches of the alignment model is as follows:

$$P^{inexact}(syn_Q, syn_D|T_D) = \frac{1}{|SYN_{quasi}|}, \tag{5.13}$$

in which the alignment model assigns weights for the combinations of syntactic relationships based on the uniform distribution. In experiments, we will compare the effectiveness of these two approaches for matching dependent term pairs between queries and documents

5.3 Predicting Optimal Parameter Settings for the Quasi-Synchronous Framework

Although the predefined syntactic configurations of the quasi-synchronous model are capable of encompassing syntactically important dependence relationships for retrieving relevant documents, independence assumptions or simpler dependence assumptions is still effective and may outperform our model based on the quasi-synchronous process.

Figure 5.3 shows some example queries. In the first group of queries, the important terms are not expected to be used with specific dependencies. In these queries, individual query terms such as "prognosis" and "creativity" is the most important key concepts. Therefore, it is sufficient for retrieval model to regard these terms individually. Dependent terms "Magnetic Levitation" and "Alzheimer's Disease" in the second group of queries are placed near each other and they are supposed to be used in the same way as they are used in queries. Therefore, the quasi synchronous model may be unnecessary for these queries in modeling term dependencies. The quasi-synchronous model aims to identify dependent terms from example queries in the third group and model variations in relationships of identified dependent terms.

Metzler (2007) suggests an automatic feature selection model which determines the optimal weight of a linear feature-based model by using a greedy procedure. Metzler selected weights of features in the retrieval model for a given document collection. Therefore, these weights are independent from the characteristics of queries.

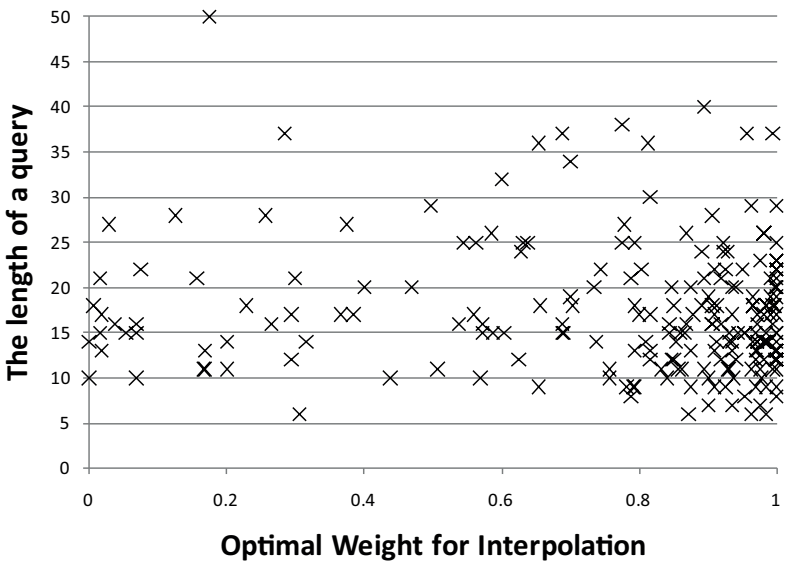


Figure 5.4. The distribution of optimal weight $P(syn_Q|T_D)$ in Eq 5.11 according to the length of queries from the Robust 2004 collection.

Zhai and Lafferty (2002) emphasis that the optimal settings of retrieval parameters not only depend on document collections but also can be affected by the characteristics of queries. In the quasi-synchronous framework, we unify the quasi-synchronous process with the SDM using the alignment model in Eq. 5.10. When we combine several retrieval models, it is important to assign a proper weight to each retrieval model according to the characteristics of the queries. Instead of select the fixed weights α in Eq. 5.10 for a given collection, we predict optimal weights for the linear interpolation according to the characteristics of a given query.

Figure 5.4 presents the distribution of optimal weights α in Eq. 5.10 according to the length of description queries in the Robust 2004 collection. This result demonstrates that the SDM and the quasi-synchronous model have their own advantages for different types of queries. If we use a fixed parameter for all queries, the quasi synchronous model improves the effectiveness for some queries but also adversely affects the performance for other queries.

We can find the optimal parameter setting for a new task and document collection even though this may require excessive tuning. On the other hand, it is impossible to optimize a parameter setting for an unseen query. To address this, we exploit a machine learning approach to predict the optimal parameter settings for individual retrieval models based on a given query. Machine learning methods have been intensively studied for query term ranking approaches to predict the importance of an individual term or a set of terms in a given query (Bendersky and Croft, 2008; Lee et al., 2009; Park and Croft, 2010; Xue and Croft, 2011). We expect these machine learning methods work to measure the effectiveness of retrieval methods for each query and extend this general approach to weighting the different combinations of retrieval models.

We train a prediction model to measure λ_Q. Training data consists of a query and the optimal weights of retrieval models.

$$(x_1, w_{i1}), \ldots, (x_n, w_{in})$$

where x_j is a ith query and its feature vector. w_{ij} is the optimal parameter setting of a jth sub retrieval model for the ith query that were selected empirically. First, we retrieve initially a ranked list of documents for a specific query using a baseline retrieval model. We initially retrieve a large number of documents because we wish the training data to cover documents out of the ranking which may be retrieved by other parameter settings. We retrieved 2,000 documents for each query while 1,000 documents are retrieved for actual retrieval experiments. Then, we choose optimal parameter values of retrieval models which maximize the performance with respect to a retrieval metric. In this thesis, we used mean average precision (MAP) as the retrieval metric. The weights in Figure 5.4 were chosen in this way.

We use the Support Vector Regression (Chang and Lin, 2001) to estimate optimal weights for the interpolation of the retrieval models of the quasi-synchronous frame-

work. Features for predicting optimal weights for a query can be classified into two categories: statistical features and syntactic features.

Statistical Features : Statistical features are the aggregation of features representing the characteristics of terms in a query.

- ***length***: the length of a query in word.
- ***average TF***, ***average DF*** and ***average TDxIDF***: : This feature is the averages of term frequency (TF), document frequency (DF) and TFxIDF of terms in a query, respectively. We did not count stopwords.
- ***NOUN ratio***, ***ADJ ratio*** and ***VERB ratio*** : In order to consider the ratio of context words, we measure the ratio of nouns, adjectives and verbs in a query per the query length.
- ***key concept ratio*** : We measure the ratio of terms which are selected by query term selection method. When there are many key concepts across a query, a term dependence model need to take account of dependence relationships at a longer distance.
- ***average score*** : We use the scores measured by a query term selection method. This feature is the average of query term selection scores of terms in a query.
- ***stopword ratio*** : This feature is the ratio of stopwords in a query.

Syntactic Features : Syntactic features reflect the syntactic structures of queries.

- ***Is question***: This feature is a Boolean indicator whether a query is a question.
- ***Wh-question***: This feature is a Boolean indicator whether a query os a wh-question.

- ***# NP*** and ***NP ratio***: These features are the number of a noun phrases in a query. We use the absolute number of noun phrases and the ratio of noun phrases in a query as features.
- ***# clause*** and ***caluse ratio***: If a query is a complex sentence, terms at a long distance across clauses can have a dependence relationships. Therefore, we measure the number of clauses in a query.
- ***average depth***: Similar to the number of NP and clauses in a query, the higher the depth of a parsing tree, the more complex the syntactic structure of a query. Therefore, we use the average depth of key concept terms in the parsing tree of a query.
- ***height tree***: This feature is the height of a parsing tree of a query.
- ***PC ratio***, ***AD ratio***, ***SB ratio*** and ***CC ratio***: The ratio of dependent term pairs which have parent-child, ancestor-descendent, siblings and c-commanding relations in a query, respectively. These features reflect the number of dependent term pairs that can be captured by the quasi synchronous model.

5.3.1 Experimental Settings

We evaluate the effectiveness the quasi-synchronous framework using of the TREC Robust 2004 and Gov2 collections. We used Indri, an open-source search engine Strohman et al. (2005), for indexing and retrieval. For the Robust 2004 collection, all documents and queries were parsed using the Stanford dependency parser (Klein and Manning, 2003). Because the quasi synchronous model needs an acyclic tree structure, we use the basic dependency representation form instead of the Stanford parser's collapsed representation.

For the Gov2 collection, it is impractical to parse all documents. Existing syntactic and dependence parser take an hour or more to parse one million words. The Gov2 collection consists of 27M documents with thousands of millions words and the most

of the documents in the Gov2 collection are not related to any topic. Therefore, we retrieve an initial document set using a baseline retrieval model. We parsed documents in the initial set and evaluate the quasi synchronous model upon only the initial document set. Because the dependency parser accepts raw text format, we used ***lynx***[3] to convert the documents of the Gov2 collection in the TREC web format to raw text format before parsing documents.

To predict optimal weights for the interpolation of retrieval models, we used the Support Vector Regression Method (Chang and Lin, 2001) which predicts the approximate target value based on a given feature vector. We trained the regression model for each query using leave-one-out cross-validation in which one query was used for test data and the others were used for training data.

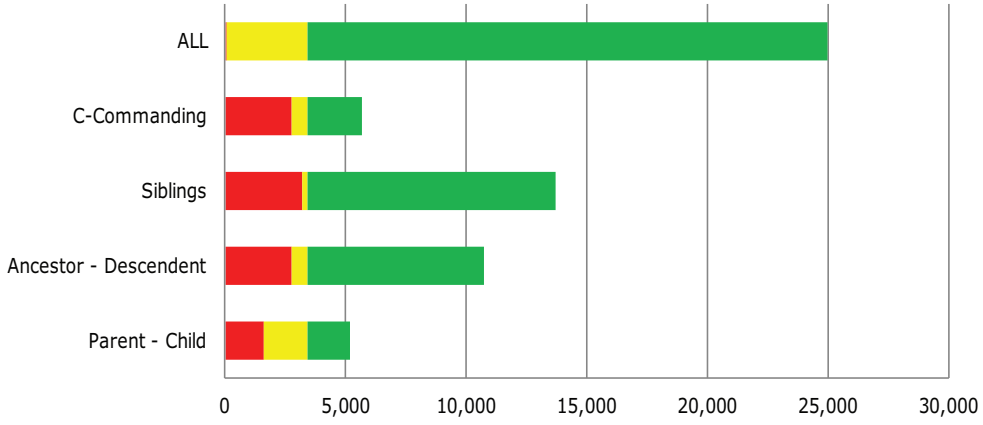
5.4 Experimental Results and Analysis

5.4.1 Coverage of Dependent Term Pairs

The predefined syntactic relationships of the quasi-synchronous model are used to identify additional dependent term pairs from verbose queries. We compare the coverage of the predefined syntactic relationships to the adjacent dependence assumption and the head-modifier relation for select dependent term pairs.

Figure 5.5 demonstrates the number of term pairs that are adjacent to each other and have one of the predefined syntactic relationships. There are 3,423 adjacent term pairs in which 3,365 (98%) have one of the predefined syntactic relationships and only 58 adjacent term pairs were not covered by the predefined syntactic relationships of the quasi-synchronous model. In addition to adjacent term pairs, the quasi-synchronous model identifies 21,556 dependent term pairs.

[3]http://en.wikipedia.org/wiki/Lynx_(web_browser)



	Parent - Child	Ancestor - Descendent	Siblings	C-Commanding	ALL
# Unamtching pairs of SDM	1,584	2,761	3,220	2,762	58
# of Overlapped Pairs	1,839	662	203	661	3,365
# Unamtching pairs in Quasi Model	1,743	7,306	10,281	2,226	21,556

Figure 5.5. The ratio of dependent term pairs by the sequential dependence assumption and the quasi-synchronous model based on the predefined syntactic relationships.

Among 24,921 dependent term pairs of the quasi-synchronous model, there are 3,582 (14%) parent-child term pairs, 7,968 (32%) ancestor-descendent pairs, 2,887 (12%) siblings and 10,484 (42%) c-commanding term pairs. 3,582 parent-child term pairs are the dependent term pairs that were used in the term dependence model based on the syntactic parsing results in previous work. 1,839 adjacent term pairs are parent-child term pairs in our syntactic parsing results. However, 1,584 adjacent term pairs are not covered by the parent-child relation although additional 1,743 term pairs are introduced by using the parent-child relationships for selecting dependent term pairs. Therefore, using only the parent-child relationship, although we may cover term dependencies at a longer distance, overall number of dependent term pairs having the parent-child relationship is similar to the number of adjacent term pairs that have significantly improved the effectiveness of retrieval models in previous work.

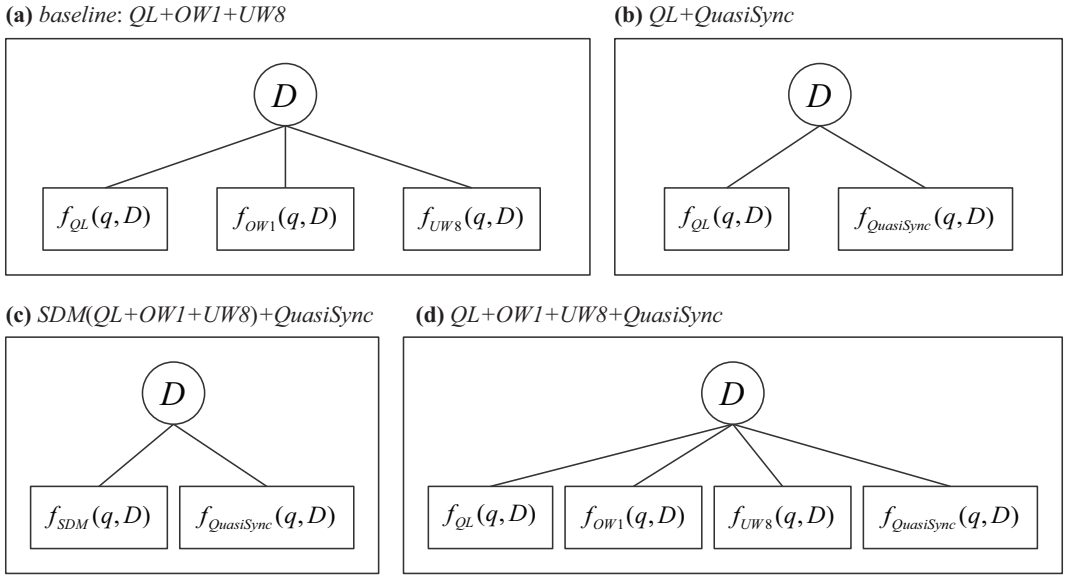


Figure 5.6. Four strategies of linear interpolation with the query-likelihood model(***QL***), the sequential dependence model(***SDM***), and the quasi synchronous model(***QM***). The sequential dependence model interpolates three scores with fixed weights: the query-likelihood score f_{QL}, the ordered window score f_{OR1} and the unordered window score f_{UW8} (Metzler and Croft, 2005). **redraw the figure using a new notation.**

5.4.2 Four Interpolation Strategies using the Loose Alignment Model

We cannot posit that every dependent term pair newly introduced by the quasi-synchronous model will improve the effectiveness of modeling term dependencies. More positive dependent terms may be introduced for some queries while unnecessary or harmful dependent terms can be introduced for other queries. As we described in Section 5.3, we estimate optimal weights of the quasi-synchronous model for a given query compared to other retrieval models.

We compare the effectiveness of the quasi-synchronous framework to an independence assumption and a sequential dependency. For this purpose, we interpolated the quasi synchronous model with the query likelihood model (Ponte and Croft, 1998) and the SDM (Metzler and Croft, 2005). We combine the quasi synchronous model with these two baseline retrieval models using four different interpolation

Table 5.1. Experimental results with the Robust 2004 with four interpolation strategies. Numbers in parentheses depict % improvement over the sequential dependence model.

	MAP	**nDCG**	**Prec@10**
QL	0.2414	0.5061	0.4096
SDM	0.2477	0.5097	0.4217
$QL + OW1 + UW8$	$0.2462_{\star}$ (-0.61%)	0.5067 (-0.59%)	0.4177 (-0.95%)
Quasi Synchronous Matching			
$QL + QuasiSync$	$0.2754^{\dagger}_{\star}$ (11.18%)	$\mathbf{0.5473}^{\dagger}_{\star}$ (7.38%)	$0.4606^{\dagger}_{\star}$ (9.22%)
$SDM + QuasiSync$	$\mathbf{0.2786}^{\dagger}_{\star}$ (12.47%)	$0.5472^{\dagger}_{\star}$ (7.36%)	$\mathbf{0.4614}^{\dagger}_{\star}$ (9.41%)
$QL + OW1 + UW8$ $+QuasiSync$	$0.2765^{\dagger}_{\star}$ (11.63%)	$0.5440^{\dagger}_{\star}$ (6.73%)	$0.4582^{\dagger}_{\star}$ (8.66%)
Exact Matching			
$QL + QuasiSync$	$0.2553^{\dagger}_{\star}$ (3.07%)	$0.5231^{\dagger}_{\star}$ (2.63%)	$0.4273^{\dagger}_{\star}$ (1.33%)
$SDM + QuasiSync$	$0.2590^{\dagger}_{\star}$ (4.54%)	$0.5234^{\dagger}_{\star}$ (2.70%)	$0.4345^{\dagger}_{\star}$ (3.05%)
$QL + OW1 + UW8$ $+QuasiSync$	$0.2583^{\dagger}_{\star}$ (4.27%)	$0.5218^{\dagger}_{\star}$ (2.39%)	$0.4361_{\star}$ (3.43%)

$\star$ means statistically significant difference with QL

$\dagger$ means statistically significance difference with SDM

strategies. Figure 5.6 depicts these four linear interpolation strategies. In the four interpolation strategies, the three potential functions of the sequential dependence model are interpolated in two ways. The "$SDM + QuasiSync$" strategy interpolates three factors using fixed weights–$f_{SDM} = 0.85 \cdot f_{QL} + 0.10 \cdot f_{OW1} + 0.05 \cdot f_{UW8}$–and, then, interpolates f_{SDM} and $f_{QuasiSync}$ using predicted optimal weights. The "$QL + OW1 + UW8 + QuasiSync$" strategy use predicted weights for all the individual factors: f_{QL}, f_{OW1} and f_{UW8}.

Table 5.1 shows the experimental results of the four interpolation strategies using the Robust 2004 collection. In these experiments, we compare the two alignment models for the quasi-synchronous model: the exact match in Eq. 5.12 and the quasi match in Eq. 5.13. For all interpolation strategies, the quasi synchronous approach shows better results than the exact match. Among the four interpolation strategies in Figure 5.6, all the interpolation strategies with the quasi synchronous model show significant improvements over a stat-of the-art baseline model, the SDM, except the $QL+OW1+UW8+QuasiSync$ with the exact match. $SDM+QuasiSync$ achieves the most improvement in the Pred10. On the other hand, predicting the weights for the factors of the SDM fails to show improvement.

The quasi matching approach shows better results than the exact matching approach for all the evaluation measures. Although the exact matching approach can capture more dependent term pairs from verbose queries, it cannot match these dependent term pairs to documents. Therefore, the margin of possible improvement would be limited when using the exact matching for the quasi-synchronous model

5.4.3 Exact Matching vs. Quasi Matching

For the further comparison of the exact and quasi matching approaches, we compare the experimental results when we assume that we know the true optimal weights of the alignment model for the interpolation. To see the potential of the quasi-

Table 5.2. Mean Average Precision of the Robust 2004 collection when we use the four interpolation strategies using the true optimal weights of the training data. In the third column, $ExactMatching$ is used for the experiments instead of $QuasiSync$.

	MAP	
	Quasi	**Exact**
SDM	0.2477	
$QL + OW1 + UW8$	0.2725	
$QL + QuasiSync$	0.3013	0.2699
$SDM + QuasiSync$	0.3022	0.2724
$QL + OW1 + UW8 + QuasiSync$	**0.3165**	0.2936

synchronous model, we evaluate the four interpolation strategies using the training labels as the interpolation weights. Table 5.2 shows the experimental results with the Robust 2004 collection.

When we use the training label or the true optimal weight, the $QL + OW1 + UW8 + QuasiSync$ strategy using the quasi matching approach demonstrates the best results. The $QL + OW1 + UW8$ strategy is also better than the baseline. This demonstrates that the sequential dependence model still has a considerable margin for being improved by using a proper parameter setting instead of a fixed weight for interpolating its potential functions.

Comparing the MAP value of $QL + QuasiSync$ or $SDM + QuasiSync$ with $QL + OW1 + UW8$, the quasi-synchronous model has higher potential for taking into an account term dependencies than the sequential dependency model. Meanwhile, $QL + OW1 + UW8 + QuasiSync$ shows considerable improvement compared to $SDM + QuasiSync$. $SDM + QuasiSync$ assigns the same weights to adjacent term pairs while $QL + OW1 + UW8 + QuasiSync$ gives different weights based on the query. This means that certain types of dependency could prove superior for a given query. Thus, we expect further improvement by using a different probability distribution for the alignment $P(syn_D, syn_Q|T_{D,syn_D})$ in Eq 5.2.

On the other hand, the exact matching approach fails to show the potential to improve the effectiveness of a retrieval model even though $QL + OW1 + UW8 + QuasiSync$ shows a significant improvement over $QL + OW1 + UW8$. The sequential dependence model can take account of long-distance term dependencies on the document side by the unordered window factor $UW8$ and the exact matching approach considers long-distance term dependencies on the query side by extracting dependent terms having a parent-child, ancestor-descendent, siblings or c-commanding relations. Because the exact matching approach does not consider the possibility of the transformation of dependency relations between queries and documents, the gap of MAP values between $QL + ExactMatching$ and $SDM + ExactMatching$ is bigger than that of $QL + QuasiSync$ and $SDM + QuasiSync$. Only $QL + OW1 + UW8 + ExactMatching$ achieves similar improvement to $SDM + QuasiSync$ using the quasi matching approach.

5.4.4 Analysis By Query Length

We also applied the quasi-synchronous model to a web collection, the Gov2 collection. For the Gov2 collection, it is impractical to parse all documents. We retrieve an initial document set (1,000 documents) using SDM and then run experiments against this initial document set.

Table 5.3 shows the experimental results for the Gov2 collection. The performances of the interpolation strategies do not show as much improvement as the Robust 2004 collection. Still, $SDM + QuasiSync$ and $QL + OW1 + UW8 + QuasiSync$ interpolation strategies improve the effectiveness significantly.

Compared to the sequential dependence model, the quasi synchronous model aims to capture long distance dependencies in queries. To test the impact of the quasi synchronous model on long distance dependencies, we analyze queries for which quasi synchronous model shows better or worse results. Table 5.4 demonstrates the compar-

Table 5.3. Experimental results with the Gov2 collection based on an initial document set retrieved by the sequential dependence model. Numbers in parentheses depict % improvement in each evaluation measure.

	Gov2		
	MAP	**nDCG**	**Prec@10**
SDM	0.2654	0.5234	0.5195
QL+OW1+UW8	0.2674 (0.75%)	0.5246 (0.22%)	0.5228 (0.65%)
$SDM + QuasiSync$	0.2755 (3.81%)	**0.5352** (2.25%)	**0.5443** (4.78%)
$QL + OW1 + UW8 +$ $QuasiSync$	**0.2764** (4.14%)	0.5342 (2.06%)	0.5396 (3.88%)

$\star$ means statistically significant difference with QL

† means statistically significance difference with SDM

Table 5.4. Comparison the sequential dependence model, SDM, and $SDM + QuasiSync$. Statistics are collected from the experiments with the Robust 2004 collection. ***# queries*** is the number of queries belong to each group and ***query length*** is the average length of the queries.

	# queries	query length
$SDM \geq QL + OW1 + UW8$	150	17.12
$SDM < QL + OW1 + UW8$	86	16.77
$SDM \geq SDM + QuasiSync$	98	15.14
$SDM < SDM + QuasiSync$	151	18.21

ison. The first two rows are the comparison of the sequential dependence model with fixed and predicted weights. For the sequential dependence model itself, the length of queries does not matter. On the other hand, the lower three rows are the comparison between the sequential dependence model, SDM, and $SDM + QuasiSync$. This comparison shows that queries improved by the quasi synchronous model tend to longer than the other queries.

Table 5.5 shows another comparison in which we calculate MAP of four interpolation strategies based upon the length of queries. The interpolation strategies

Table 5.5. Experimental results with the Robust 2004 according to the length of queries. *Length* is the number of terms in a query and *# queries* is the number of queries belonging to each group. Numbers in parentheses depict % improvement in each evaluation measure.

	Robust 2004 (MAP)		
Length	**$\sim$ 10**	**11 $\sim$ 20**	**21 $\sim$**
# queries	**43**	**147**	**59**
SDM	0.3062	0.2398	0.2248
$QL + OW1 + UW8$	0.3056 (-0.19%)	0.2380 (-0.75%)	0.2232 (-0.71%)
$QL + QuasiSync$	0.3268 (6.72%)	0.2613 † (8.98%)	0.2731 † (21.51%)
$SDM + QuasiSync$	0.3255 (6.29%)	0.2668 † (11.27%)	0.2738 † (21.81%)
$QL + OW1 + UW8 +$ $QuasiSync$	0.3251 (6.17%)	0.2649 † (10.47%)	0.2698 † (20.04%)

† means statistically significance difference with SDM

containing the quasi synchronous model demonstrate a clear tendency to show larger improvements for longer queries while $QL + OW1 + UW8$ does not. The quasi synchronous model extracts term dependencies across a query based on its parsing results. The longer queries are, the more chance that the quasi synchronous model extracts term dependencies from the query that are not extracted by the sequential dependence model.

We use uniform distribution alignment between different syntactic relations in queries and documents. However, intuitively, dependent terms are expected to be used less frequently in a certain syntactic configurations and more frequently in others. Thus, employing a weighted alignment model could improve the effectiveness of the quasi synchronous model. In the next chapter, we propose a method to evaluate the valid variations in dependence relationships between queries and documents. We use the evaluation results for a weighted alignment model instead of a uniform distribution.

5.5 Summary

We have proposed the quasi-synchronous framework, inspired by a quasi-synchronous stochastic process which constructs an inexact matching of syntactic relations between source and target sentences. As in query term expansion techniques that address lexical variation between query and document, we aim to support syntactic divergence of term dependencies from documents to queries using an inexact matching approach. We generalize these ideas from machine translation to the information retrieval task in which matching occurs between a sentence and an entire document. Experimental results show that the quasi-synchronous model can significantly improve effectiveness compared to a strong state-of-the-art retrieval model.

Each retrieval model, however, has its own strengths and weaknesses, which can differ query by query. A simpler retrieval model may be superior to a more sophisticated model depending on the query. This is why most previous work using term dependencies has had problems showing consistent improvement. To address this issue, we used a machine learning approach to find an optimal parameter setting for a combination of retrieval models. By using a predicated optimal weight, we optimized the overall performance of the interpolation of several retrieval models. This interpolation technique, we found, is necessary for achieving the best results with the quasi-synchronous framework.

We use a uniform distribution over alignments between different syntactic relations in queries and documents. Intuitively, however, dependent terms are expected to be used less frequently in a certain syntactic configurations and more frequently in others. For example, dependent terms in fixed phrases such as technical terminology, proper names, etc., will be used in the same way by both searchers and authors. Moreover, as shown in the experimental results, certain syntactic configurations could prove more important for evaluating the relevance of documents. In the next chapter, instead

of a uniform distribution, we will propose a method to predict a weighted alignment model $P(syn_D|T_D, syn_Q)$ in Eq 5.11.

CHAPTER 6

MODELING VARIATIONS IN DEPENDENCE RELATIONSHIPS

6.1 Overview

SDM uses two potential functions of ordered and unordered windows with different window sizes in order to distinguish different dependent relationships of term pairs in documents. By assigning higher weights on the ordered window potential function with length one, the SDM can emphasize term pairs that are used in documents in the same form as in queries. This is reasonable because co-occurrences of terms do not always convey the same meaning. Depending on the syntactic relation of terms, the meaning of terms may or may not be relevant to users' information needs.

For example, both *"trade secret"* and *"secret trade"* are valid English expressions. In our test collection, *"secret trade"* is more frequently observed than *"trade secret"*. However, the meaning of *"secret trade"* is not relevant to the user's information need implied by the TREC query *"Document will discuss the theft of trade secrets along with the sources of information ..."*.

In the previous chapter, we predict optimal weights of individual retrieval model according to given queries that can indirectly take account of valid variations in the relationships of dependent terms. When a query contained term pairs where the meaning could change such as the example term pair *"trade secret"*, the optimal weight for the ordered window potential function would be higher than weights for other factors. However, this method cannot distinguish valid variations in dependent relationships for an individual term pair. The proposed method predicting optimal

TREC Topic 656	
Description	How are ***young children*** being ***protected*** against ***lead poisoning*** from ***paint*** and water pipes?
Documents	***Young children*** also may be ***poisoned*** during teething by mouthing on window sills that contain ***leaded paint***. ... We had a law passed in 1988 to ***protect kids*** in school from ***lead***, and the EPA and ... ... what youll use to ***protect*** your vehicles ***paint*** is like going to the ice cream stand: some go with plain ...

Figure 6.1. The example text fragments in which the concepts in the TREC query, "*How are young children being protected against lead poisoning from paint and water pipes?*", are used together in a sentence.

weights for the retrieval models assigns the same weights for a given query. The quasi matching process of the quasi-synchronous model treats every combination of the predefined syntactic relationships in the same way.

In addition, we proposed a query term ranking method where the ranking results are used to remove unnecessary dependent term pairs because all the dependent term pairs introduced by the quasi-synchronous model are not always beneficial to modeling term dependency. For this purpose, we used the method that selects the rankings of query terms according to the the effectiveness of individual terms in terms of the target evaluation measure. However, just because individual terms in a dependent term pair are important key concepts of queries, it does not always mean that the dependent term pair is also important.

Let's consider another TREC topic, "*How are young children being protected against lead poisoning from paint and water pipes?*", in Figure 6.1. The example topic contains several concepts including "*young children*", "*protected*", "*paint*", "*paint*" or "*water*" in order to express detailed criteria for relevance. Figure 6.1 also shows the three text fragments containing these concepts. The first two text fragments are

about how to protect children against lead poisoning from paint and water pipes. On the other hand, the last text fragment is talking about products to protect paint from scratches, bugs, dings etc. that is not relevant to the information need. Therefore, the dependence relationships between concepts "*paint*" and "*protect*" is not valid when "*paint*" is used as the direct object of the verb "*protect*".

Rasolofo and Savoy (2003) used a term-proximity scoring heuristic to select important dependent term pairs. This method reflects the collection statistics of arbitrary term pairs in queries to evaluate the validity of dependence relationships of query terms. Therefore, the validity of dependent term pairs measured by this method is independent from users' information needs. When we apply this method to *"trade secret"* and *"secret trade"*, *"secret trade"* will receive a higher score than *"trade secret"* because *"secret trade"* is more frequently used in the test collection. The validity of dependent relationships and their variations in relevant documents should be evaluated with regards to users' information needs.

Song et al. (2008) proposed a method to evaluate the strength of the head-modifier relation of query terms within relevant documents. They make the observation that, in the relevant documents of a query containing "*mutual fund*", "*mutual*" has a head-modifier relationship with "*fund*" when "*mutual*" is used in the relevant documents. On the other hand, in the relevant documents of a query containing "*overcrowded prison*", "*overcrowded*" has the head-modifier relationship with other terms as many times as "*prison*". Based on this observation, they proposed the variability that represents the strength of the head-modifier relationship in the relevant class.

However, the evaluation result of the strength of term dependencies in the relevant class is not always proportional to relevance scores. This is one of the incorrect assumptions about modeling term dependencies in the long history of IR. Cooper argued that misunderstanding the independence assumption implied by the Binary Independence Model (BIM) led to the failure of term dependence models (Cooper,

1995). He pointed out that the BIM is actually based on *linked dependence* according to the degree of statistical dependence associated between terms in relevant and non-relevant documents. That is, if terms were as strongly dependent in relevant documents as in non-relevant documents, modeling term dependencies would confer no advantage over the independence assumption.

In the same way, even if two dependent terms are strongly correlated when they have a certain syntactic relationship, it would not be beneficial to explicitly model this dependency when this dependency is as strong in non-relevant documents as relevant documents. In related work, Lavrenko proposed the Generative Relevance Hypothesis (GRH) (Lavrenko, 2009) in which a statistical significance test between a relevance hypothesis against its null hypothesis was used to evaluate the correlation between an original query term and its expansions in terms of users' information needs.

In this chapter, we propose a method that evaluates valid variations in dependence relationships based on the GRH. In previous research, weights for different dependence assumptions are predicted according to the characteristics of query in order to validate dependence relationships for a given query (Bendersky and Croft, 2012). For a given information need, the GRH assumes that queries and their relevant documents can be thought of as random samples from the same latent representation space (Lavrenko, 2009). On the other hand, the null hypothesis assumes that documents and queries were drawn from unrelated populations in the representation space. The statistical significance test of the GRH against the null hypothesis can be interpreted as a measure of whether the assumption of the GRH is statistically true or not. The statistical significance test of the GRH has been used for ad-hoc retrieval, relevance feedback, cross-language retrieval, handwriting retrieval, etc. We use this statistical significance test for evaluating whether a certain dependence relationship is valid for a given term pair with regard to users' information needs.

We apply the proposed method of evaluating valid variations in dependence relationships to the quasi-synchronous model. The quasi-synchronous model allows the alignment of dependent terms in queries and documents even if they have different syntactic relationships. For this purpose, the quasi-synchronous model allows inexact matching between any combinations of predefined syntactic configurations. Using the statistical significance test results of the GRH for specific dependence relationships of dependent terms, we elaborate on this inexact matching process of the quasi-synchronous model that link between only valid variations in dependence relationships for term pairs according to users' information needs.

The rest of this chapter is organized as follows. Section 6.2 describes Cooper's argument and the GRH. In Section 6.3, we describe a method to assess the valid variations of dependence relationships given a user's information need using pseudo relevant documents. In Section 6.4, we present the experimental results on the effectiveness of the proposed method for modeling variations of dependence relationships.

6.2 Modeling Variations of Dependence Relationships

6.2.1 Linked Dependencies of the BIM

Cooper suggested possible reasons for the repeated failures of modeling term dependencies, based on incorrect modeling assumptions for the Binary Independence Model (BIM). For the BIM, Robertson and Jones (1976) assumed conditional independence between terms. The strength of term dependency between t_i and t_j given relevance or non-relevance classes are defined as follows:

$$\begin{aligned} P(t_i, t_j|R=1) =& k_1 \cdot P(t_i|R=1) \cdot P(t_j|R=1), \\ P(t_i, t_j|R=0) =& k_0 \cdot P(t_i|R=0) \cdot P(t_j|R=0). \end{aligned} \tag{6.1}$$

When two constants are $k_1 = 1$ and $k_0 = 1$, t_i and t_j are conditionally independent. Cooper pointed out that the BIM does not require k_1 and k_0 to be 1. The BIM with t_i and t_j is expressed as follows:

$$\frac{P(t_i, t_j|R = 1)}{P(t_i, t_j|R = 0)} = \frac{k_1 P(t_i|R = 1)P(t_j|R = 1)}{k_0 P(t_i|R = 0)P(t_j|R = 0)}. \tag{6.2}$$

Cooper shows $k_1 = k_0$ is sufficient for Eq. 6.2 to be true (Cooper, 1995). In the BIM, $k_1 = k_0$ does not require t_i and t_j to be independent conditioned by relevance and non-relevance classes, that is, k_1 and k_0 are 1. The BIM only assumes that the strength of their dependencies must be same in both relevant and non-relevant documents. Therefore, although t_i and t_j depend on each other, the BIM can reflect this dependency as long as its strength in the relevant class (k_1) is the same as the non-relevant class (k_0).

Cooper suggested that this misunderstanding about the BIM originated from the name of the model. He suggested that "*Linked dependence*" would be a better name because it represents the proportional correlation of dependent terms in relevant and non-relevant documents (Cooper, 1995). Modeling term dependency can affect the effectiveness of retrieval only if the relative strength of term dependencies in the relevant class is stronger than in the non-relevant class.

6.2.2 Generative Relevance Hypothesis of Dependence Relationship

The GRH postulates that relevant documents are randomly sampled from the same latent presentation space of a query. In order to test whether a document D and a query Q were drawn from the same population, Lavrenko used a statistical significance test between two competing hypothesises:

- $H_{\mathbb{R}}$:Relevant hypothesis is that D and Q are drawn from the same population.

- H_0 : Null hypothesis is that the document D and the query Q are drawn from different, unrelated populations in the representation space.

If a significance test demonstrates that $H_{\mathbb{R}}$ is a significantly stronger hypothesis than H_0, the GRH indicates D is relevant to query Q. To model the variations of dependence relationships, we derive the GRH as when (t_i, t_j) in a query and a document are drawn from the same population, their syntactic relation will be syn_D in a document. As the strength of term dependencies in Eq. 6.2 under the relevant and non-relevant classes, we use the pairwise mutual information as follows:

$$\begin{aligned} H_{\mathbb{R}}(t_i, t_j, syn_D) &= \frac{P_{\mathbb{R}}(t_i, t_j, syn_D)}{P_{\mathbb{R}}(t_i)P_{\mathbb{R}}(t_j)}, \\ H_0(t_i, t_j, syn_D) &= \frac{P_0(t_i, t_j, syn_D)}{P_0(t_i)P_0(t_j)}. \end{aligned} \tag{6.3}$$

$H_{\mathbb{R}}$ and H_0 represent the strength of the dependence relationship syn_D for t_i and t_j compared to the independence assumption under the relevant class ($P_{\mathbb{R}}$) and the non-relevant class (P_0). The probability $P_{\mathbb{R}}$ of relevance class is measured from relevant documents while the probability P_0 of non-relevant is measured from the collection statistics as follows:

$$\begin{aligned} P_{\mathbb{R}}(t_i, t_j, syn_D) &= \frac{tf_{t_i,t_j,syn_D}(D_{\mathbb{R}})}{|D_{\mathbb{R}}|}, \\ P_0(t_i, t_j, syn_D) &= \frac{cf_{t_i,t_j,syn_D}}{|C|}, \end{aligned} \tag{6.4}$$

in which $D_{\mathbb{R}}$ is the set of relevant documents for a given query Q. $tf_{t_i,t_j,syn_D}(D_{\mathbb{R}})$ is the frequency of a term pair (t_i, t_k) having a syntactic relation syn_D. $P_{\mathbb{R}}(t_i)$ and $P_0(t_i)$ for an individual term are also computed using the same set of documents.

The value of $H_{\mathbb{R}}$ is in approximately inversely proportional to the variability of Eq. 2.9. However, using only $H_{\mathbb{R}}$ may mislead the evaluations of the valid variations

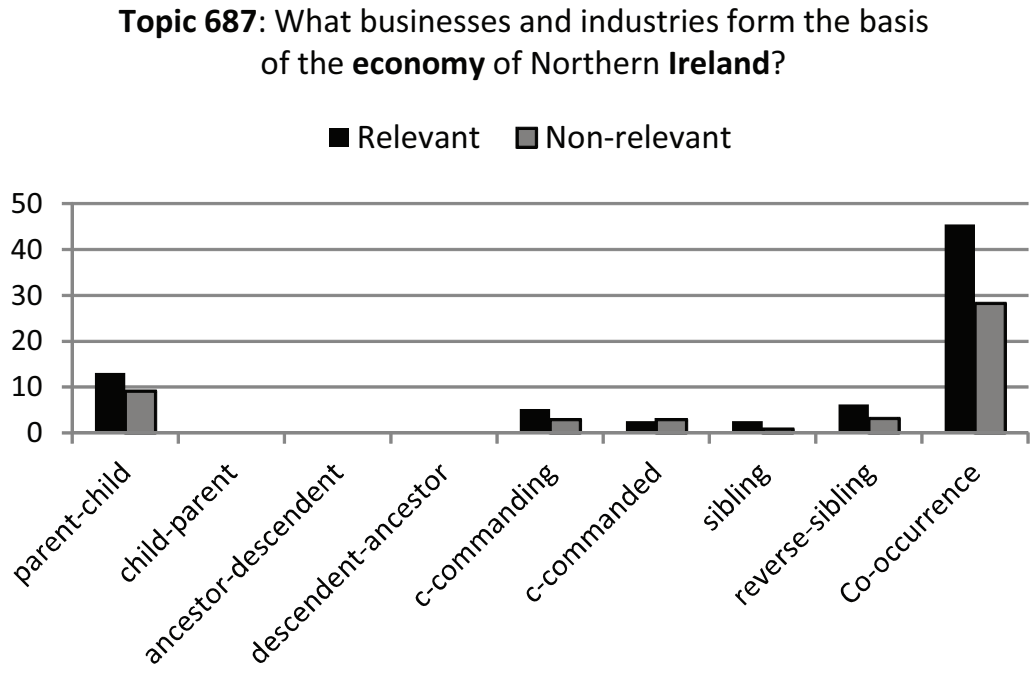


Figure 6.2. The $H_{\mathbb{R}}$ and H_0 of dependence relationships in relevant and non-relevant documents of ("*economy*","*Ireland*") including the their co-occurrences.

in dependence relationships . For example, Figure 8.1 shows $H_{\mathbb{R}}$ and H_0 of "*software*" and "*producing*" for the eight syntactic relationships of SYN^{quasi} in Eq. 5.8. "*Software*" and "*producing*" are strongly dependent in relevant documents when having parent-child, ancestor-descendent or c-command relations. However, the strength of their dependencies is stronger under the non-relevant class as confirmed by the statistics of the entire collection. The evaluation result of the valid dependence relationships of "*software*" and "*producing*" using only the relevance hypothesis $H_{\mathbb{R}}$ may impair the effectiveness of retrieval models, which is why we use a statistical significant test of the relevance hypothesis $H_{\mathbb{R}}$ over the null hypothesis H_0.

We evaluate valid variations in dependence relationships for the quasi-synchronous framework in order to take account of the different meanings of dependent terms when having different syntactic relationships. This model, however, is not always

more effective than a simpler dependence assumption. Figure 8.2 shows the pairwise mutual information of "*economy*" and "*Ireland*" according to their syntactic relationships. "*economy*" and "*Ireland*" have stronger dependencies for parent-child, sibling and c-commanding relations in relevant documents than in non-relevant documents. However, "*economy*" and "*Ireland*" show stronger dependencies under the relevance class when we just tabulate their co-occurrence. The simple dependence assumption of co-occurrence is sufficient or better for the retrieval model to incorporate the dependency of "*economy*" and "*Ireland"* for the information need of the example topic. The unordered-window potential function of the SDM corresponds to "*co-occurrence*" in Figure 2.(b). This example shows that, when evaluating a specific syntactic relationship, we need to compare the relationship to the simpler dependence assumption that is used in the interpolated retrieval model. The unordered-window potential function of the SDM assumes terms co-occurring in documents are dependent while the quasi-synchronous model considers a specific syntactic relation syn_D. We compare the strengths of the syntactic and proximity dependencies using the following hypothesis:

$$\begin{aligned} H_{\mathbb{R}}^{co-occur}(t_i, t_j, syn_D) &= \frac{H_{\mathbb{R}}(t_i, t_j, syn_D)}{H_{\mathbb{R}}(t_i, t_j, syn_{co-occur})}, \\ H_0^{co-occur}(t_i, t_j, syn_D) &= \frac{H_0(t_i, t_j, syn_D)}{H_0(t_i, t_j, syn_{co-occur})}, \end{aligned} \tag{6.5}$$

where $syn_{cooccur}$ represents the co-occurrences of (t_i, t_j). Therefore, $H_{\mathbb{R}}^{cooccur}(syn_D, t_i, t_j)$ and $H_0^{cooccur}(syn_D, t_i, t_j)$ are the relative strength of each syntactic relationship over co-occurrences.

6.3 Predicting Valid Variations for the Quasi-Synchronous Framework

We use $H_{\mathbb{R}}^{co-occur}$ and $H_{0}^{co-occur}$ for the loose alignment model of the quasi-synchronous framework. We separated the alignment model as follows in Eq. 5.11 as follows:

$$P(A = (syn_Q, syn_D)|T_D) = P(syn_Q|T_D)P(syn_D|T_D, syn_Q). \tag{6.6}$$

We predict the probability of $P(syn_Q|T_D)$ as interpolation weights of retrieval models in order to consider the appropriate dependence assumptions for a given query. In this section, we define $P(syn_D|T_D, syn_Q)$ using the GRH in order to select valid variations in dependence relationships for a given term pair (t_i, t_j) instead of the uniform distribution as follows:

$$\begin{aligned} P(syn_D|T_D, syn_Q) &= P(syn_D|t_i, t_j, syn_Q) \\ &= \frac{1}{|SYN^{Quasi}|} GRH_{syn_Q}(syn_D, t_i, t_j), \end{aligned} \tag{6.7}$$

where GRH is a binary function that evaluates valid variations in dependence relationships as we will describe in the next section. We assume that the validity of a specific syntactic relationship for a term pair (t_i, t_j) is independent from other terms. The evaluation of the syntactic relationship of term pairs in documents rely on the given term pairs (t_i, t_j) and its relationship in a query syn_Q.

Then, the binary function GRH in Eq. 6.7 is defined as follows:

$$GRH(syn_D, t_i, t_j) = \begin{cases} 1 & \frac{H_{\mathbb{R}}^{co-occur}(t_i, t_j, syn_D)}{H_{0}^{co-occur}(t_i, t_j, syn_D)} > 1 \\ 0 & otherwise. \end{cases} \tag{6.8}$$

In order to measure $H_{\mathbb{R}}$, we need true relevant documents for a specific information need. However, it is impossible to collect the set of relevant documents for submitted queries. Therefore, we use a machine learning method using pseudo-relevant documents to evaluate valid syntactic relationships.

6.3.1 Predicting Valid Variations of Dependence Relationships

When true relevance information is not available, pseudo relevance feedback can be used. Pseudo relevance feedback has been successfully used for query term expansion. Similarly, we use the top-k documents to measure the distribution of dependence relationships.

Pseudo relevance feedback adds and modifies the weights of expanded terms in addition to original query terms. Weights on expanded terms are typically much lower than the original query terms. On the other hand, the evaluation results of variations in dependence relationships are applied to the original query terms in Eq. 5.7. Using the top-k documents as pseudo relevant documents may not be reliable enough to evaluate valid variations in dependence relationships for certain term pairs, especially when a given term pair includes a rarely used term. Therefore, we use a machine learning method in which we generate training labels from true relevant documents. Then, we predict $GRH(syn_D, syn_Q|t_i, t_j)$ by using features extracted from pseudo relevant documents. Training data for modeling valid variations of dependence relationships is as follows:

$$(l_1, \{x_1, q_{x_1}, D_{x_1}\}), ..., (l_n, \{x_n, q_{x_n}, D_{x_n}\}),$$

in which n is the number of training instances that are the dependent term pairs with specific dependence relationships. l_i is the ith training label, which is the GRH value of Eq. 6.8 measured from true relevant documents. x_i and q_{x_i} represent the ith dependent term pairs and a query where x_i is used, respectively. D_{x_i} is the set of

top-k documents that will be used as pseudo relevant documents for x_i. From the top-k documents, we extract features for x_i. Features that we used are as follows:

- ***tf*** : Term frequencies of each term and co-occurrences of term pairs in top-k documents.
- ***GRH*** : The GRH values of term pairs in pseudo relevant documents.
- $\boldsymbol{H_R}$: H_R values of term pairs in pseudo relevant documents.
- $\boldsymbol{syn_Q}$: The dependence relationship of a term pair in a query.
- **Match**$(\boldsymbol{syn_Q}, \boldsymbol{syn_D})$: Whether syntactic relationships of a given term pairs are same in a query and a document.
- ***POS*** : The part-of-speech (POS) tags of term pairs. The POS tags are simplified into five categories:noun, verb, adjective, proper noun and others.
- $\boldsymbol{Phrase_{Noun}}$: This feature is a Boolean value based on whether a term pair is used in the same noun phrase or not.

When we use a smaller number of top-k documents, the data is sparse. On the other hand, the more pseudo relevant documents we use, the more noise our statistics will have. In order to compensate for this problem, we use the top 10, 50 and 100 documents to measure the feature values of tf, GRH and HR. We predict the valid variations in dependence relationships of Eq. 6.8 using a decision tree algorithm based on these features.

6.4 Experiments and Analysis

6.4.1 Experimental Settings

We evaluate the proposed methods using three test collections: news articles (Robust 2004), government web pages (Gov2) and web documents (ClueWeb-B). For preprocessing, we perform stemming using the Krovetz stemmer and remove all

stopwords. We use the standard list of stopwords (Allan et al., 2000) and eighteen TREC description stopwords such as *'describe'* and *'documents'*. After removing stopwords, the average length of queries are 7.5, 5.7 and 4.8 terms per query for the Robust 2004, Gov2 and ClueWeb-B, respectively. The queries for the Robust collection are somewhat longer than others.

We used Indri, an open-source search engine (Strohman et al., 2005), for indexing and retrieval. Dirichlet smoothing is used for smoothing. We set $\mu = 2,000$ for the Robust 2004 collection and $\mu = 3,500$ for the Gov2 and ClueWeb-B collection, for which the baseline SDM yields the best results for the baseline. For the ClueWeb-B collection, we applied a spam filter that filtered out approximately 40% of the documents (Cormack et al., 2011).

For the collection statistics of dependent term pairs, we randomly sampled another million documents from each of the Gov2 and ClueWeb-B collections. For predicting valid variations in the dependence model, we used the decision tree of Weka 3.0[1] (Hall et al., 2009). We train models for each dependence relationship syn_D separately. We used ten-fold cross validation for training models.

6.4.2 Retrieval Performance Evaluation

The quasi-synchronous model is developed for considering complex syntactic term dependencies in a parsing tree, so it is not appropriate for queries containing a single important keyword. In the previous chapter, the quasi-synchronous model was interpolated with the baselines, and four interpolation strategies were tested in order to compare the effectiveness of dependence relationships based on parsing trees to the existing dependence assumptions. Among the four interpolation strategies, we conducted experiments using the interpolation of the quasi-synchronous model and the SDM as follows:

[1]http://www.cs.waikato.ac.nz/ml/weka/

Table 6.1. Retrieval effectiveness comparison with all the baselines using the mean average precision and the R-Precision. Numbers in parentheses depict % improvement over the sequential dependence model.

Robust 04			
	MAP	**Prec@10**	**R-Pr**
QL	0.2528	0.4241	0.2912
SD	$0.2659^{\star}$	$0.4502^{\star}$	$0.3011^{\star}$
TS	$0.2816^{\star}_{\dagger}$ (5.90%)	$0.4647^{\star}$ (3.22%)	$0.3144^{\star}_{\dagger}$ (4.42%)
VD	$0.2869^{\star}_{\dagger}$ (7.46%)	$0.4847^{\star}_{\dagger}$ (7.66%)	$0.3182^{\star}_{\dagger}$ (5.44%)

Gov2			
	MAP	**Prec@10**	**R-Pr**
QL	0.2596	0.5047	0.3156
SD	$0.2876^{\star}$	$0.5456^{\star}$	$0.3487^{\star}$
TS	$0.2982^{\star}$ (3.58%)	$0.5725^{\star}_{\dagger}$ (4.93%)	$0.3487^{\star}$ (0.00%)
VD	$0.2992^{\star}$ (3.79%)	$0.5671^{\star}$ (3.93%)	$0.3493^{\star}$ (0.17%)

ClueWeb-B			
	MAP	**Prec@10**	**R-Pr**
QL	0.1298	0.2324	0.1737
SD	0.1434	0.2297	0.1750
TS	$0.1538^{\star}$ (7.25%)	$0.2648^{\star}_{\dagger}$ (15.28%)	$0.1916^{\star}$ (9.49%)
VD	$0.1600^{\star}_{\dagger}$ (10.79%)	$0.2779^{\star}_{\dagger}$ (20.98%)	$0.1974^{\star}_{\dagger}$ (11.69%)

Significance($p < 0.05$) shown compared to the QL($\star$) and SD($\dagger$).

$$\lambda_Q \cdot P_{SDM}(Q|D) + (1 - \lambda_Q) \cdot P_{Quasi}(Q|D), \tag{6.9}$$

in which P_{SDM} is the SDM. In the SDM, we used 0.85, 0.10 and 0.05 for the weights for the query-likelihood model, the ordered-window potential function and the unordered-window potential functions. The interpolation weight λ was estimated per query as in the original work. The interpolation of the quasi-synchronous model and the SDM showed competitive performance compared to other three interpolation strategies. This setting also requires one interpolation parameter. Therefore, we can minimize the effect of the predicted weights for interpolation and concentrate on the effectiveness of evaluating variations in dependence relationships.

We compare the performance of the quasi-synchronous model using the evaluation results of valid variations in dependence relationships. The query-likelihood model and the SDM are used as baselines. We also compare the effectiveness of evaluating variations in dependence relationships to the term selection method that was used for the quasi-synchronous model in order to remove harmful dependent terms (Park et al., 2011). We expect that the evaluation results of variations in dependence relationships also has a similar effect of the term selection method. If query terms in a given pair were not important, the evaluation results of all syntactic relationships for that term pair would be negative. In this way, term pairs containing useless terms will be indirectly removed.

Table 6.1 shows the experimental results. In the experiments, the following models are compared:

- Query-Likelihood model (**QL**)
- Sequential Dependence model (**SD**)
- The quasi-synchronous model using Term Selection (**TS**)

- The quasi-synchronous model based on evaluating valid Variations of Dependence relationships (**VD**)

Using both the term selection results and the evaluation results of valid variations of dependence relationships, the quasi-synchronous model shows significant improvements over the query-likelihood model for the three test collections. In the experiments, both versions of the quasi-synchronous model show significant improvements over the query-likelihood model for all three collections.

Compared to the SDM, the quasi-synchronous model with the term selection results shows significant improvement only for the Robust 2004 collection. On the other hand, the quasi-synchronous model refined by the evaluation results of valid variations of dependence relationships shows significant improvements for the Robust 2004 and ClueWeb-B collections.

With respect to mean average precision (MAP) and R-precision, the quasi-synchronous model using term selection shows significant improvements over the SD baseline only for the Robust 2004 collection. For ClueWeb-B, the SD baseline is worse than the QL baseline in precision at 10. Therefore, the quasi-synchronous model using both methods (TS and VD) show significant improvements in precision at 10 for Robust 2004 and ClueWeb-B.

For the Gov2 collection, using either the term selection results or the evaluation results of valid variations, the quasi-synchronous model fails to achieve significant improvements over the sequential dependence model. In the description queries of the Gov2 collection, there are less important dependent term pairs that cannot be captured by the SDM. Although the average length of the description queries for the Gov2 collection is longer than the topics for the ClueWeb-B collection, most of the important dependent terms can be captured by selecting adjacent term pairs. Therefore, the available improvement by evaluating valid variations in dependence relationships for the smaller number of important terms is also limited. Modeling variations in

Table 6.2. The effectiveness of evaluating the variations of dependence relationships according to statistical significance test approaches using true relevant documents. H and $\hat{H}$ are described in Eq. 6.3 and Eq. 6.4, respectively.

Robust 04			
	MAP	**Prec@10**	**R-Pr**
SD	0.2659	0.4502	0.3011
$H_{\mathbb{R}}/H_0$	0.3081	0.5213	0.3308
$H_{\mathbb{R}}^{co-occur}/H_0^{co-occur}$	0.3166	0.5373	0.3403

Gov2			
	MAP	**Prec@10**	**R-Pr**
SD	0.2876	0.5456	0.3487
$H_{\mathbb{R}}/H_0$	0.3093	0.6134	0.3613
$H_{\mathbb{R}}^{co-occur}/H_0^{co-occur}$	0.3197	0.6235	0.3676

ClueWeb-B			
	MAP	**Prec@10**	**R-Pr**
SD	0.1434	0.2297	0.1750
$H_{\mathbb{R}}/H_0$	0.2036	0.3228	0.2420
$H_{\mathbb{R}}^{co-occur}/H_0^{co-occur}$	0.2128	0.3469	0.2481

dependent relationships shows a marginal improvement over term selection for the Gov2 collection, but consistent improvements on the other collections.

6.4.3 Evaluating Statistical Significance Test Methods

In the experiments in Table 6.1, we use the significance test in Eq. 6.5 that evaluates the strength of dependence relationship of the predefined syntactic relationships over the dependence of co-occurrence. The quasi-synchronous model is interpolated with the SDM in which the ordered-window and the unordered-window potential functions are used to capture the dependencies of adjacent term pairs and co-occurrence of dependent terms.

We compare the effectiveness of statistical significance test settings $H_{\mathbb{R}}/H_0$ and $H_{\mathbb{R}}^{co-occur}/H_0^{co-occur}$ for evaluating valid variations in dependence relationships. In the interpolated model of the quasi-synchronous model and the SDM, predicted interpolation weights affect the final retrieval results. Errors in the machine learning models for evaluating valid variations in dependence relationships also affects on the results. By ruling out these factors, we try to focus on the effectiveness of statistical significant test settings. For this purpose, we conducted "cheating" experiments in which we use all the true relevant documents. We also assume that we know the optimal weight λ_Q of interpolation in Eq. 6.9.

Table 6.2 shows the experimental results. Significance testing $H_{\mathbb{R}}/H_0$ for variations of dependence relationships over the independence assumption shows improvement over SDM for all the test collections. These results shows that selected variations of dependence relationships using $H_{\mathbb{R}}/H_0$ are still beneficial. However, when the valid variations of dependence relationships are selected based on $H_{\mathbb{R}}/H_0$, the quasi-synchronous model takes account of dependence relationships that may be covered by the co-occurrences of dependent terms. The quasi-synchronous model loses its advantage against the unordered-window potential function of the SDM. On the other hand, $H_{\mathbb{R}}^{co-occur}/H_0^{co-occur}$ is the relative strength of dependence relationships that indicate whether variations of dependence relationships for a given terms can be handled by the SDM or not. Therefore $H_{\mathbb{R}}^{co-occur}/H_0^{co-occur}$ achieves higher improvement than HR/H0 for all the test collections.

6.5 Summary

In this chapter, we proposed a method to evaluate the valid variations of dependence relationships for the quasi-synchronous model. The quasi-synchronous model uses the four syntactic configurations to extract dependent term pairs from queries and documents. In the previous chapter, a uniform distribution was used to cap-

ture the transformation of the dependence relationships of terms between queries and documents. However, dependent terms having different syntactic relationships do not always have the same meaning. The uniform distribution allows arbitrary combinations of dependence relationships in queries and documents.

In order to address this limitation, we use the evaluation results of variations in dependence relationships for given dependent terms using the GRH. The proposed method achieves significant improvements over strong baselines and also shows better results than the term selection method for the quasi-synchronous framework that is used to remove unnecessary dependent term pairs. The quasi-synchronous framework consists of multiple dependent assumptions. We measure the relative strengths of dependence relationships by comparing the relevance and null hypothesis of different dependent assumptions. Using these relative strengths of dependence relationships, we make the quasi-synchronous framework take account of only valid variations in dependence relationships. The experimental results using true relevant documents demonstrate that the relative strengths of different dependencies can maximize the advantages of modeling variations of dependence relationships.

In the next chapter, we propose a query term expansion method using a translation model in order to bridge lexical gaps between queries and documents as we match different syntactic relationships between queries and documents.

CHAPTER 7

CONTEXT-BASED TRANSLATION MODEL FOR QUERY TERM EXPANSION

7.1 Overview

Users express their information needs in queries in different ways from relevant information in documents. In Chapters 4 and 5, we describe methods to take account of variations in dependence relationships of terms between queries and documents. In the same way, we need to take account of lexical gap between queries and documents. It has been one of the fundamental problems for IR tasks to bridge the lexical gap between queries and documents. Query term expansion techniques (Qiu and Frei, 1993; Salton, 1980; Xu and Croft, 1996; Schütze and Pedersen, 1997; Metzler and Croft, 2007) have been intensively studied to solve this lexical mismatch problem.

In this chapter, we propose a method using a statistical translation model as a query term expansion method. A statistical translation model constructs a translation table that consists of the list of possible translations for a given source term. For query term expansion, the monolingual translations of original queries in the same language are treated as the expanded concepts (Karimzadehgan and Zhai, 2010; Surdeanu et al., 2011). In particular, we propose a context-based translation model in which a phrase-level translation model is modified to take account of users' information needs as the translation context when translating original query terms.

As collections of question-answer pairs from CQA services, such as Yahoo! Answers, have become available, they have been used to train translation models. Surdeanu et al. (2011) proposed phrase-level translation models trained in this way.

Q:	How do you get your *hair* to **grow** faster?
A:	Supposedly this works but never tried it. prenantal vitamins. they're just vitimans so they're not going to make u grow ...
Q:	How to **grow** *Columbine flowers*?
A:	Plant outside in sun or light shade, they will grow in both places. Scratch or loosen the soil lightly with a garden claw or rake. Sprinkle your seeds on and cover with the loose soil. You just cover with enough ...

Figure 7.1. The example questions with "*grow*" with different contexts "hair" and "flowers"

Instead of using higher-order versions of the IBM model (Brown et al., 1993), Surdeanu et al. used an approach in which they converted question answer pairs into a sequence of paired terms based on different types of phrasal information sources including n-grams, parsing results and semantic role labeling results. This approach suggests a solution for reflecting key concepts in the translation of question terms by converting questions to a sequence of question terms paired with key concepts.

Phrase-level translation models were introduced to capture the translations between dependent term pairs that can represent more specific meaning compared to the individual terms. For example, Surdeanu et al. (2011) describe an example of relevant translation from "*squeaky* $\rightarrow$ *door*" to "*spray* $\rightarrow$ *WD-40*". Similarly, Metzler and Croft (2007) proposed the latent concept expansion method in which the Markov random field model was used for modeling term dependencies during expansion. They aim to provide a framework for going beyond simple term occurrence that can be used to generate meaningful multi-term concepts for tasks such as query suggestion and reformulation.

We employ a phrase-based translation model in which we incorporate the key concepts of query terms for a translation model in order to improve the quality of

translation results. When translating a word, we need to consider the context around the word. With different contexts, the same word can be translated into different expressions. For example, Figure 7.1 shows an example questions with word "*grow*". The first question is about growing hair while the other question is about growing flowers. For these two queries, the translations of "*grow*" should be different. Based on this motivation, we propose a context-based translation model for the query term expansion. Our model identifies the key concepts of queries, and then the identified key concepts are used as the translation context for the rest of the query terms. For example. "*hair*" and "*Columbine flower*" in Figure 7.1 are the key concepts that can define the context for translating the word "grow".

Instead of the term pairs of questions and answers for training a phrase-level translation table, we extract term pairs from a question in which a term pair consists of an original question term and the key concept of the question. Therefore, the context-based translation model generates a translation table consisting of the translations of question terms accompanied with a certain key concept. We identify the key concept of a given question that is used as a context for translating original question terms based on the information need of the question.

In addition, we also used the identified key concepts for selectively applying a translation model for query term expansion. Lee et al. (2008) proposed a method for classifying question terms to selectively apply a translation model for expanding question terms. They use the TextRank algorithm to select important terms of questions to which they selectively applied a translation model. Compared to an approach to refine the quality of query term expansion results in the post query term expansion (Mandala et al., 1999), Lee at al. refined the original query terms before query term expansion. In this dissertation, we evaluate a machine learning method to identify question terms for which translations would be relevant to users' information need.

The rest of this chapter is organized as follows. Section 6.2 introduces a statistical translation model for IR tasks. Section 6.3 describes the context-based translation model. In Section 6.4, we describe a method to identify the key concepts of questions for a model using the key concepts of questions as the context for translating original query terms. In Section 6.5, we present the experimental results on the effectiveness of the proposed method using the CQA collection.

7.2 Statistical Translation Model

Statistical translation models have been used as query term expansion methods for a variety of retrieval tasks, for example, Berger and Lafferty (1999); Surdeanu et al. (2011); Xue et al. (2008); Jeon et al. (2005); Murdock and Croft (2005). For a given sentence T in a target language, a statistical translation model (Brown et al., 1993) seeks a sentence S in a source language which maximizes the probability that T is translated from S as follows:

$$P(S|T) = \frac{P(S)P(T|S)}{P(T)}, \tag{7.1}$$

in which $P(S)$ is a language model that measures the probability of a given source sentence S. $P(T|S)$ is a translation model that measure the probability that a target sentence T is generated from a source sentence S. In the simplest version of the IBM model 1, the translated target sentence T is generated from the source sentence word by word.

Translation models can be readily integrated into the language modeling framework for retrieval:

$$
\begin{aligned}
P(Q|D) &\approx \prod_{q_i \in Q} P(q_i|D) \\
&= \prod_{q_i \in Q} \sum_{t_j \in D} P(q_i|t_j) P(t_j|D),
\end{aligned}
\tag{7.2}
$$

where t_j is a term in a document D, $P(t_j|D)$ represents the probability that a term t_j is generated by a document D and $P(q_i|t_j)$ represents the translation probability that a document term t_j is translated into a query term q_i.

Self-translation probabilities have been shown to be a challenging issue in translation-based language models, Xue et al. (2008); Jeon et al. (2005); Murdock and Croft (2005). Murdock and Croft (2005) point out that underestimated self-translation probabilities reduce retrieval performance by assigning low weights to question terms, while overestimated self-translation probabilities remove the benefits of the translation model. We employ Xue and Croft's solution (Xue et al., 2008) to this problem as follows:

$$
\begin{aligned}
P_{mx}(q_i|D) = &(1-\beta) \cdot P(q_i|D) + \\
&\beta \cdot \sum_{t_j \in D, t_j \neq q_i} P(q_i|t_j) P(t_j|D),
\end{aligned}
\tag{7.3}
$$

where self-translation probability $P(q_i|D)$ is separated from the translation model. It allows us to systematically control the impact of the self-translation probability using β. Further, we adopt the bi-directional translation approach as follows:

$$
\begin{aligned}
P_{mx}(q_i|D) = &(1-\beta_1-\beta_2) \cdot P(q_i|D) + \\
&\beta_1 \cdot \sum_{t_j \in D, t_j \neq q_i} P(q_i|t_j) P(t_j|D) + \\
&\beta_2 \cdot \sum_{t_j \in D, t_j \neq q_i} P(t_j|q_i) P(t_j|D),
\end{aligned}
\tag{7.4}
$$

in which β_1 and β_2 are used to assign weights to self translation and translation probabilities in bi-directions. Using the bi-directional approach, if t_j is translated into

two query terms q_a and q_b with the same probability, it may have a different effect on each of the query terms. If q_a is translation of only t_j while q_b is a translation of many other terms, we can say the relationship of t_j and q_a is stronger than the relationship of t_j and q_b.

It may not be possible to match words in source and target sentences one-on-one. Brown et al. (1993) proposed *fertility* and *distortion*. They call the number of target words that a source word produces the fertility of a source word. The distortion probabilities are used to measure the probability that the ith source word generates l target words in the jth position. In the IBM model, Model 2 $\sim$ 5 were proposed to take into account other features such as the order of words, the length of aligned words in source and target expressions, etc., using fertility and distortion.

However, in ad-hoc retrieval, we do not need to consider the readability of queries because queries are composed by users. Furthermore, fertility and distortion is designed for alignment between paired sentences. Even in question answer pairs, the number of sentences in questions and answers are varying. For a question of one sentence, an answer can consist of several sentences. Therefore, fertility and distortion cannot be used for the settings of IR tasks. For query term expansion, we directly extract expanded concepts from a translation table that the most probable translations for a given term.

Therefore, instead of higher-order versions of the IBM model, Surdeanu et al. (2011) converted a question into the sequence of dependent term pairs for generating phrase-level translation tables. For example, they converted a sentence "*A helicopter gets its power from rotors or blades*" to "*(helicopter-gets) (gets-power) (power-rotors) (rotors-blades)*" to estimate the translation probabilities between adjacent term pairs. Without formulating translation models for specific types of linguistic features, Surdeanu et al. model translation between different types of text representations such

as a bag of words, n-gram, syntactically dependent pairs of terms and the predicate-argument pairs of semantic labels.

In the next section, we propose the context-based translation model based on this phrase-level translation model.

7.2.1 Context-based Translation Model

Similar to (Surdeanu et al., 2011), we convert questions into the sequence of term pairs. For a given question, we select one of its terms as the key concept of the question. We assume that every term in a question can be the key concept of a question. Therefore, we generate a set of term pair sequences in which we treat each term in a question as the key concept of the question. For example, consider the following question-answer pair:

Q: How do I remove candle wax from a polar fleece jacket?

A: I've heard the best thing to do is to try to pull off as much as you ...

We generate six questions in which one of question terms is used as the key concept of the question from this question "*How do I* ***remove candle wax*** *from a* ***polar fleece jacket****?*" as follows:

Table 7.1. The translation results of "*grow*" with different contexts (Bold-faced). ***PLANT*** represents a WordNet category.

(*grow*)	(**hair**−*grow*)	(**PLANT**−*grow*)
make	hair	plant
plant	growth	soil
healthy	biotin	cut
take	long	make
month	take	water
hair	healthy	garden
help	supplement	fruit
biotin	take	start
fast	grow	good
water	microgram	concrete

Q1: remove-***remove*** candle-***remove*** wax-***remove*** polar-***remove*** fleece-***remove*** jacket-***remove***

Q2: candle-***candle*** candle-***candle*** wax-***candle*** polar-***candle*** fleece-***candle*** jacket-***candle***

Q3: wax-***wax*** candle-***wax*** wax-***wax*** polar-***wax*** fleece-***wax*** jacket-***wax***

Q4: polar-***polar*** candle-***polar*** wax-***polar*** polar-***polar*** fleece-***polar*** jacket-***polar***

Q5: fleece-***fleece*** candle-***fleece*** wax-***fleece*** polar-***fleece*** fleece-***fleece*** jacket-***fleece***

Q6: jacket-***jacket*** candle-***jacket*** wax-***jacket*** polar-***jacket*** fleece-***jacket*** jacket-***jacket***

Using these six questions and the original answer, we generate six pairs from the example question-answer pair. From a question-answer pair having n question terms, we produce n question-answer pairs.

Then, we measure the translation probabilities from question term pairs to answer terms using the IBM model 1. Table 7.2.1 shows the top-10 translations of "*grow*" with different contexts. While the translation results of "*grow*" without considering

key concepts are a mixture of terms related to various topics, the translation results with key concepts consists of terms related to a specific topic.

Surdeanu et al. use WordNet senses and named-entity labels to solve the data sparseness problem. The general idea is to replace the terms by their categories so that we can have more samples per category and thus better estimations. For example, "*Columbine*" is used once in the test collection while we know its WordNet sense "***PLANT***". In addition to the sequence of original question term pairs, we also generate sequences in which we use WordNet senses or named-entity labels for a key concept, e.g. "PLANT-grow" for "*Columbine-grow*". The third column of Table 7.2.1 is the example of a translation table using WordNet senses or named-entity labels.

Using the phrase-level translation table, we derive a phrase-level translation model from Eq. 7.4 as follows:

$$\begin{aligned} P_{mx}(q_i|D) = &(1-\beta_1-\beta_2)\cdot P(q_i|D)+ \\ &\beta_1 \cdot \cdot \sum_{t_j \in D, t_j \neq q_i} P(q_i, \kappa_Q|t_j)P(t_j|D)+ \\ &\beta_2 \cdot \cdot \sum_{t_j \in D, t_j \neq q_i} P(t_j|q_i, \kappa_Q)P(t_j|D), \end{aligned} \tag{7.5}$$

in which κ_Q represents the key concept of a query Q. $P(t_j|q_i, \kappa_Q)$ and $P(q_i, \kappa_Q|t_j)$ represent the translation probability that a document term t_j is translated into a query term q_i for a given key concept κ_Q and vise versa. The context-based translation model can be interpreted in two ways: translating question terms based on the key concepts of questions or translating the key concepts of questions based on all other question terms.

Our method uses the key concepts of questions not only for translating question terms according to the context of questions but only for selectively applying the translation model to the secondary (key) parts of the question. For this purpose, we

apply a binary function to eq 7.5 for considering the secondary concepts of questions as follows:

$$\begin{aligned} P_{mx}(q_i|D) = &(1-\beta_1-\beta_2)\cdot P(q_i|D)+ \\ &\beta_1\cdot\varphi(q_i)\cdot\sum_{t_j\in D, t_j\neq q_i} P(q_i,\kappa_Q|t_j)P(t_j|D)+ \\ &\beta_2\cdot\varphi(q_i)\cdot\sum_{t_j\in D, t_j\neq q_i} P(t_j|q_i,\kappa_Q)P(t_j|D), \end{aligned} \tag{7.6}$$

in which $\varphi(q_i)$ is the binary function that is 1 when q_i is a secondary concept and 0 otherwise. In the next Section, we will explain a method to identify the key concepts of questions for κ_Q and $\varphi(q_i)$ in detail.

7.3 Identifying Key Concepts for the Context-based Translation Model

Identifying the key concepts of queries for more effective weighting of query terms has been studied (Bendersky and Croft, 2008; Park and Croft, 2010). In previous work, identified key concepts are used to assign higher weights to important terms in queries. The definition of key concepts can differ according to the purpose of detecting key concepts. Lee et al. (2008) proposed a method using the TextRank algorithm to selectively apply translation models to specific classes of terms.

We define the key concepts of questions as the most important terms representing the main topic of a question. Therefore, all terms in a question should be translated with the key concept of the question as the context of translation. We also use a second group of concepts, that we call secondary concepts, in order to selectively apply translation models to the more important concepts (Lee et al., 2008). We use a machine learning method to identify the key concepts and the secondary concepts of questions.

7.3.1 Identifying Key Concepts

In order to generate training data, Lee et al. use an algorithm in which the importance of terms is evaluated within a single document, which is represented as a graph for measuring the PageRank scores of terms. On the other hand, we select key concepts and secondary concepts of questions that maximize the effectiveness of translation models.

We separately estimate the training labels of key concepts and secondary concepts. For key concepts, we estimate the training label of a question term q_i based on the evaluation result when q_i is used for κ_Q in Eq. 7.6 in which we use the translation results of all question terms. The relative rank of an answer in an answer finding result is used as the training label of q_i for key concepts. We use the translation results of q_i without using κ for generating the training labels of a secondary concept for q_i. For a question Q, training data consists of triplets as follows:

$$(q_1, k_1, s_1), (q_2, k_2, s_2), \ldots, (q_n, k_n, s_n),$$

in which n is the number of question terms. k_i and s_i are the labels of key concept and secondary concept of a question term q_i, respectively.

We select only one key concept per question. There can be more than one term which can improve the effectiveness of the context-based translation model. Therefore, we use the Support Vector Regression (SVR) model (Joachims, 2002) to rank question terms for key concepts and select the best candidate.

7.3.2 Features

We use three types of features for identifying key concepts and secondary concepts: lexical features, syntactic features and semantic features. The aim is to estimate how likely a given term is to be a key concept or a secondary concept when having certain syntactic and semantic characteristics.

Lexical Features Lexical features are used to take account of the characteristic of an individual term.

- Is Capitalized: This feature is a Boolean indicator that is set to TRUE iff the first character is capitalized.
- All Capitalized: This feature is a Boolean indicator that is set to TRUE iff the entire characters are capitalized.
- Clarity score: This feature is the relative entropy between a query language model and the collection language model Cronen-Townsend and Croft (2002), which indicate the lack of ambiguity by measuring diversity of topics in documents related to the topics.
- OddsRatio: The odds ratio between a given term being used in a question and the term being used in an answer. This feature is motivated by the observation that clue terms of questions such as commonly-used verbs do not occurred in answers. Instead, expanded concepts corresponding to these clue terms are used. Using the odds ratio of terms between questions and answers, we measure how likely question terms will be observed in answers.
- Unseen: This feature is a Boolean indicator that is set to TRUE iff a term is not observed in the retrieved results of the top-15.

Syntactic Features The syntactic features are used to consider the role of a given term in a question.

- Phrase Label: This feature is the types of phrase where a given term is used.
- Part-of speech (POS) tags: We used Boolean features of four POS: noun, verb, adjective and proper noun.
- Depth in a parse tree: The distance from root node to a given term in the parse tree of a question.

Semantic Features We use the WordNet supersense classes and named-entity classes as semantic features (Ciaramita and Altun, 2006). These classes themselves are used as features. In addition, we also use the KL-divergences of these classes and key concepts as features.

7.4 Experiments and Analysis

7.4.1 Experimental Settings

We evaluate the context-based translation model using the Yahoo! Answers Comprehensive Questions and Answers version 1.0. We follow the same refinement process as Surdeanu et al. (2011) to choose questions which have reliable quality for training the translation model. Among 148,102 question-answer pairs, 30,761 question have their answers in the top 15 retrieval results[1]. We used 60%, 20%, 20% of these question-answer pairs as training data, development and test data, respectively.

Training data is used to estimate the translation probabilities and to train the machine learning method for identifying the key concepts and secondary concepts of questions. The development data is used to find optimal parameter settings of β_1, β_2 and λ in Eq 7.4. The estimated values of β_1, β_2 and λ are used for Eq. 7.5.

We indexed answers as documents. Then, we retrieved answers by submitting the questions as queries. We used the Galago toolkit (Croft et al., 2010) for indexing and retrieval. We used the Super Sense Tagging (SST) toolkit to annotate WordNet categories to question terms (Ciaramita and Altun, 2006).

7.4.2 Answer Retrieval

The bi-directional translation model showed better results than the relevance model for finding answers (Xue et al., 2008). Therefore, we compare the effectiveness

[1]We used the sequential dependence model as baseline. Therefore, our number is slightly different from previous work.

Table 7.2. The experimental results of answer retrieval using the CQA collection. **MRR** represents Mean Reciprocal Rank. Numbers in parenthesis are relative improvements over the baseline. Significant differences with *Baseline* and *Translation* are marked by † and ‡, respectively (statistical significance was measured using the two-tailed Wilcoxon test with $p < 0.05$).

	Prec@1	**Recall@5**	**MRR**
Baseline	0.476	0.774	0.612
Translation	0.492 (3.4%)	0.794 (2.6%)	0.628 (2.6%)
Secondary	$0.515^{\dagger}_{\ddagger}$ (8.2%)	$0.818^{\dagger}_{\ddagger}$ (5.7%)	$0.650^{\dagger}_{\ddagger}$ (6.2%)
Key+Secondary	$0.537^{\dagger}_{\ddagger}$ (12.8%)	$0.837^{\dagger}_{\ddagger}$ (8.1%)	$0.669^{\dagger}_{\ddagger}$ (9.3%)

of the key concepts and secondary concepts to the translation model for estimating the translation table (Surdeanu et al., 2011). Table 7.4.2 shows the experimental results for precision at rank 1, recall at rank 5 and the Mean Reciprocal Rank (MRR) of answers in retrieval results. We used the sequential dependence model as the baseline. Among 8,715 question-answer pairs in the test data, the baseline system retrieved the answers at the first rank for 4,150 (47.6%) of questions. The translation-based language model without using the key and secondary concepts (*Translation*) shows better results than the baseline. However, selectively using the key and secondary concepts significantly improved the effectiveness of the translation-based language model.

Secondary is the experimental results of the translation-based language model that applies the translation model only for the secondary concepts of question terms. By translating only secondary keywords in questions, we prevent the translation model from introducing non-relevant translation results. *Key+ Secondary* shows the results where we translated the secondary concepts of questions using the key concepts as context.

Table 7.3. The number of question-answer pairs of which retrieval results were unchanged, improved and deteriorated by using the translation-based language model.

	Unchanged	**Improved**	**Deteriorated**
Secondary	6,759	1,521	435
Key+Secondary	5,307	2,646	762

As we can see, considering key concepts as the context of translation improves the performance of the system. To analyze this further, we compare the experimental results of the baseline system and the translation-based language model for individual questions. Table 7.4.2 shows the number of question-answer pairs for which retrieval results are unchanged, improved and decreased by the translation-based language model with the key concepts and secondary concepts. The translation-based language model without the key concepts affected fewer retrieval results. The translation results without the key concepts consists of terms for all contexts. Therefore, an individual translation result has less effect on ranking.

Using predicted key concepts, the translation model generates more precise translation results. However, the ratio of questions for which results were decreased by the translation-based language model with key and secondary concepts is also higher than one with only the secondary concepts. This shows that, if the selection of key concepts is inaccurate, using them as context can have a negative impact on effectiveness.

7.5 Summary

In this chapter, we proposed the context-based translation model. We use the key concepts of questions as the context for translating other terms. Key concepts represent the most important part of queries expressing users' information needs. Based on the context defined by key concepts, we selectively apply the context-based translation model to the secondary parts of questions that are important clue terms

for finding answers. The key concepts improve the effectiveness of the translation results by constraining the translations of question terms within the contexts of questions. By considering secondary concepts, we can prevent the translation model from introducing non-relevant translation results. The context-based translation model significantly improved the effectiveness of the translation-based language model for finding answers from the CQA collection.

Because of the lack of training data, previous work on a translation model for ad-hoc retrieval has used pseudo data such as synthesized queries from documents (Berger and Lafferty, 1999). We use the CQA collection to train the context-based translation model for answer passage retrieval. However, the CQA collection is not enough to cover large-scale document collections for ad-hoc retrieval tasks such as web pages. The topics of the CQA collection may not able to cover topics that can be submitted in other IR tasks. In order to solve the limitation of the context-based translation model using the CQA collection, we propose the context-based translation model based on conditional mutual information in Chapter 8.

CHAPTER 8

ANSWER PASSAGE RETRIEVAL

8.1 Overview

Focused retrieval aims to provide more efficient access to retrieval results from the users' perspective. At the one end of the spectrum of focused retrieval, QA systems provide answers for a specific types of questions. Recent work on QA systems has expanded to non-factoid questions such as how-to questions that we describe in Chapter 7. At the other end of the spectrum, passage retrieval systems locate highly relevant positions in long documents. Using passage retrieval results, users can more efficiently decide whether corresponding documents are relevant. For example, passage retrieval results can be used to find entry points of documents (Arvola et al., 2011).

We propose an *answer passage* retrieval task that aims to compensate for the weak points of QA and passage retrieval systems. Although non-factoid QA systems can handle more general information needs, they are only feasible when there is a special data collection available, such as the Yahoo! Answer collection. Finding answers from unstructured, raw text has yet to be solved. Passage retrieval systems have focused on retrieving topically relevant text fragments instead of a ranked list of documents. Current passage retrieval results are reasonable for keyword queries representing general information needs. However, verbose queries are used to explain in detail the conditions of users' information needs. Although passage retrieval results might be helpful for users, there is no guarantee that users will find specific answers for their information needs represented by verbose queries. In Chapter 4, we describe the

different characteristics of answer passages compared to the relevant text fragments that have been used in previous work.

In this chapter, we propose a answer passage retrieval model in which we investigate two factors: a passage retrieval model with passages of multiple granularities and the context-based translation model. As shown in Table 4.1, the average length of answer passages is 45 words. This length is more reasonable than result lists or full documents for restricted search environments such as the screen of a smart phone. As an auxiliary feature, passage-level evidence can improve the relevance scores of long documents that contain comprehensive content including relevant information. In previous work, the optimal length of passages for the purpose of improving document ranking is longer than that of our answer passages. Therefore, we use two passage units. One of these units is used to evaluate the relevance scores of retrieval results themselves, while the other unit is used to measure the cohesion of relevant information.

In addition, we use the context-based translation model for the answer passage retrieval task. In Chapter 7, we describe the context-based translation model for query term expansion. The context-based translation model shows significant improvement in finding answers from the question-answer collection. Although question-answer pairs is good source to train the translation table of the context-based translation model, its coverage is limited because the scale of the question-answer collection is small compared to the size of large-scale web collections. Therefore, the context-based translation model cannot provide useful expansion results for the topics of ad-hoc retrieval.

In order to overcome the limitation of the question-answer collection for training the context-based translation model, we propose a translation model using conditional mutual information. Karimzadehgan and Zhai (2010) proposed a translation model using the mutual information based on the co-occurrence of term pairs in documents.

We expand this method using conditional mutual information in which a conditional variable is used as the context of translation.

The rest of this chapter is organized as follows. Section 8.2 describes a passage retrieval model for answer passage retrieval. Section 8.3 describes query term expansion method using the context-based translation model based on conditional mutual information . In Section 8.4, we present the experimental results on the effectiveness of the context-based translation model and key concept identification results for answer passage retrieval.

8.2 Passage Retrieval Model

For passage retrieval, we split documents into the set of passages as follows:

$$PSG_D = \{psg_{(D,1)}, psg_{(D,2)}, \ldots, psg_{(D,K)}\}, \tag{8.1}$$

where K is the number of passages in D. The passage retrieval model evaluates psg to generate a ranked list of answer passages. The most common way to evaluate passages in retrieval models is to combine the score of the entire document with that of passages as follows:

$$P(Q, psg_{(D,k)}) = \gamma \cdot P(Q, psg_{(D,k)}) + (1 - \gamma) \cdot P(Q, D), \tag{8.2}$$

where γ is the interpolation weight of a passage retrieval model and a document retrieval model. The passage retrieval model $P(Q, psg_{(D,k)})$ treats a passage $psg_{(D,k)}$ as a document.

In Chapter 4, we guided annotators to tag one or two sentences as answer passages. In accordance with the guideline of answer passage annotation, we use sentences to define passages. Sentence-based passages begin the first passage of n sentences at the

beginning of a document. Then, we extract a new passage of length n sentences after $n/2$ sentences, where subsequent passages are half-overlapped with each other. When $n = 2$, the average length of sentence-based passages is 47 words that is commensurate with the average length of answer passage annotation results of 45 words. Moreover, we aim to generate retrieval results in which an individual passage can provide an independent and complete answer. A sentence is a grammatical unit that expresses an independent statement. On the other hand, users may not understand the complete meaning of a window-based passage when this passage starts from the middle of sentence. Therefore, we use sentence-based passages as the retrieval unit of our passage retrieval model.

However, the evaluate results of sentence-based passages might be too short to measure the cohesion of relevant information within a portion of short text. For example, for the TREC topic "*What information is available on the involvement of the North Korean Government in counterfeiting of US currency.*", there a relevant document about illegal activity in Asian countries. Because only one chapter of the document describes the illegal activities of North Korea, the overall relevance score of the document is lower than that of a document that dedicates to describe North Korea. In previous work, passage-level evidence is used to compensate for the lower relevance scores of long documents that contain comprehensive topics. Kaszkiel and Zobel (2001) compared the effectiveness of passage-level evidence according to the size of windows. Experimental results demonstrated that window-based passages are the most effective when the size of window is $350 \sim 400$ words—longer than the average length of sentence-based passages.

Therefore, we also use half-overlapped passages based on fixed-length windows (window-based passages) for the passage retrieval model. For window-based passages, we select passages based on the number of words instead of sentences. Window-based passages begin the first passage of N words at the beginning of a document.Then, we

extract a new passage of length N words after $N/2$ words, where subsequent passages are half-overlapped with each other. We add window-based passage retrieval model to the overall model as follows:

$$\begin{aligned} P(Q, psg_{(D,k)}) =& \gamma_1 \cdot P(Q, psg_{(D,k)}) + \gamma_2 \cdot P(Q, psg^N_{(D,k)}) \\ &+ (1 - \gamma_1 - \gamma_2) \cdot P(Q, D), \end{aligned} \tag{8.3}$$

where $psg^N_{(D,k)}$ is window-based passage that enclose the passage $psg_{(D,k)}$. When a sentence-based passage is split between two overlapped window-based passages, we select the one that encloses the larger portion of the sentence-based passage. While n in Eq 8.2 is selected considering a proper size of answer passage retrieval results, N is selected to measure the cohesion of relevant information.

8.3 Translation Model based on Conditional Mutual Information

In Chapter 7, we proposed the context-based translation model that use the key concepts of questions as translation contexts. Although the context-based translation model trained using the question-answer database shows good results, it is too limited for query term expansion in ad-hoc retrieval because the question-answer database is too small to cover large-scale document collections. In order to overcome this limitation of the context-based translation model, we use cMI to generate a translation table for ad-hoc retrieval.

Karimzadehgan and Zhai (2010) use the mutual information of document terms for a given query term to generate the translation table without paired sentences. Mutual information can estimate the relationships of two words by estimating the trans-information of two random variables. Karimzadehgan and Zhai use the trans-information between an original query s and a candidate expansion t to estimate the translation probability $P(t|s)$ using the mutual information as follows:

$$I(t;s) = \sum_{X_t=0,1} \sum_{X_s=0,1} p(X_t, X_s) log \frac{p(X_t, X_s)}{p(X_t)p(X_s)}, \tag{8.4}$$

in which X_s and X_t are the random variables of an original query s and a candidate expansion t, respectively. The probabilities of random variables are estimated using the co-occurrence statistics of t and s in a document collection as follows:

$$\begin{aligned}
p(X_a = 1) &= \frac{c(X_c = 1)}{N}, \\
p(X_a = 0) &= 1 - p(X_c = 1), \\
p(X_a = 1, X_b = 1) &= \frac{c(X_a = 1, X_b = 1)}{N}, \\
p(X_a = 1, X_b = 0) &= \frac{c(X_a = 1) - c(X_a = 1, X_b = 1)}{N}, \\
p(X_a = 0, X_b = 1) &= \frac{c(X_b = 1) - c(X_a = 1, X_b = 1)}{N}, \\
p(X_a = 0, X_b = 0) &= 1 - p(X_a = 1, X_b = 1) - p(X_a = 1, X_b = 0), \\
&\quad - p(X_a = 0, X_b = 1),
\end{aligned} \tag{8.5}$$

in which $c(X_a)$ is the number of documents containing term a. $c(X_a = 1, X_b = 1)$ is the number of documents containing a and b while $c(X_a = 1, X_b = 0)$ is the number of documents containing only a term a without b. We expand a translation model using conditional mutual information to generate a context-based translation table as follows:

$$\begin{aligned}
I(t;s|c) &= \sum_{X_k=0,1} P(X_k) \sum_{X_t=0,1} \sum_{X_s=0,1} P(X_t, X_s|X_k) log \frac{P(X_t, X_s|X_k)}{P(X_t|X_k)P(X_s|X_k)} \\
&= \sum_{X_k=0,1} \sum_{X_t=0,1} \sum_{X_s=0,1} P(X_t, X_s, X_k) log \frac{P(X_t, X_s, X_k)P(X_k)}{P(X_t, X_k)P(X_s, X_k)},
\end{aligned} \tag{8.6}$$

in which k is a key concept that is used as the context of translation from s to t. In addition to the probabilities in Eq 8.5, the joint probability of X_s, X_t and X_k is estimated as follows:

$$\begin{aligned}
p(X_a = 1, X_b = 1, X_c = 1) &= \frac{c(X_a = 1, X_b = 1, X_c = 1)}{N}, \\
p(X_a = 1, X_b = 1, X_c = 0) &= p(X_a = 1, X_b = 1) - p(X_a = 1, X_b = 1, X_c = 1), \\
p(X_a = 1, X_b = 0, X_c = 0) &= p(X_a = 1) - p(X_a = 1, X_b = 1) \\
&\quad - p(X_a = 1, X_c = 1) \\
&\quad + p(X_a = 1, X_b = 1, X_c = 1), \\
p(X_a = 0, X_b = 0, X_c = 0) &= 1 - p(X_a = 1, X_b = 0, X_c = 0) \\
&\quad - p(X_a = 1, X_b = 1, X_c = 1).
\end{aligned} \tag{8.7}$$

in which $c(X_a, X_b, X_c)$ is the number of co-occurrences of three random variables, X_a, X_b and X_c.

The translation table of the context-based translation model using cMI is used to generate query term expansions. Compared to question answer pairs, the conditional mutual information is estimated using only the set of documents. Therefore, we do not apply the bi-directional translation approach in Eq. 7.4 to cMI. The context-based translation model using cMI is derived from Eq. 7.3 as follows:

$$\begin{aligned}
P_{mx}(q_i|D) = &(1 - \beta) \cdot P(q_i|D) + \\
&\beta \cdot \varphi(q_i) \cdot \sum_{t_j \in D, t_j \neq q_i} P(q_i, \kappa_Q|t_j) P(t_j|D),
\end{aligned} \tag{8.8}$$

in which D can be either a document or a passage in Eq. 8.3. In this dissertation, we apply the context-based translation model to window-based passages and sentence-based passages.

8.4 Experiments and Analysis

8.4.1 Experimental Settings

We evaluate the answer passage retrieval model using the INEX Ad-Hoc track 2009/2010 and the Gov2 collection. Relevance judgments of the INEX collection are similar to the relevant text fragments of the Gov2 collection. We use the sentence segmentation capability of the Stanford dependency parser to extract sentence-based passages (Klein and Manning, 2003). We set $n = 2$ for sentence-based passages. In sentence segmentation, symbols such as the bullets of tables are classified as sentence boundaries. In order to avoid sentence-based passages from being affected by these sentences, we merged sentences shorter than three words into the next sentences.

We use Support Vector Regression (SVR) model for identifying key concepts and secondary concepts (Joachims, 2002). We use ten-fold cross validation for training SVR and selecting β for ths translation model.

We used Indri, an open-source search engine (Strohman et al., 2005), for indexing and retrieval. We first extract the top 50 documents that are used in the answer passage annotation. Then, we rerank passages in this top documents. The sequential dependence model is used to retrieved the top 50 documents. Dirichlet smoothing is used for smoothing. We set $\mu = 3,500$ for the document model $P(Q, D)$ in Eq 8.2 while $\mu = 50$ for the document model $P(Q, psg_{(D,k)})$ and $P(Q, psg^N_{(D,k)})$.

Similar to the tasks of the INEX 2010 Ad-hoc track, we setup two tasks for answer passage retrieval.

- **Per-Document Retrieval**: We assume that users read retrieval results document-by-document. This means that users start reading the document that contains the top ranked passage. Users will read retrieved passages in this documents up to 250 words. Therefore, we first retrieve the best passages of documents and select five documents corresponding to the top 5 passages. Then, we extract 100 words based on the ranked list of passages in a document.

- **Per-Passage Retrieval**: We assume that users read retrieval results passage-by-passage without considering documents. In this task, we evaluate retrieval results of top N passages.

8.4.2 Multi-Level Passage Retrieval Model

We first evaluate the effectiveness of the passage retrieval model with passages of multiple granularities, described by Eq 8.3. For the comparison of the parameter settings of multiple passage-level evidences, we evaluate the effectiveness of passage retrieval results using the Per-Passage task with the top 10 passages. In these experiments, we use normalized discounted cumulative gain in word (nDCGw) as the evaluation measure. Figure 8.1, 8.2 and 8.3 show the experimental results. Each graph in these figures shows the experimental results with window size $N = 200, 400$ and 800. X-axis and y-axis represent the interpolation weights γ_1 and γ_2, respectively. A bar next to each graph represents the range of nDCG values that the passage retrieval model shows for each collection. The more red the more effective the γ values for the interpolation of passage-level evidences.

Figure 8.1 and 8.3 show that using $N = 200$ for window-based passage retrieval model shows the best results for retrieving relevant text fragments and answer passages of the Gov2 collection. On the other hand, $N = 800$ shows the best result for the INEX collection.The average length of documents in the Gov2 collection (937.3 terms) is longer than that of the INEX collection (555.2 terms). However, Wikipedia documents in the INEX collection are more likely to focus on describing a single topic although the topic might be a more general concept of the information needs implied by queries. Window-based passages for measuring the cohesion of relevant information has less effect on overall retrieval results. Therefore, the optimal window size for the INEX collection is longer than the average length of documents in the INEX collection.

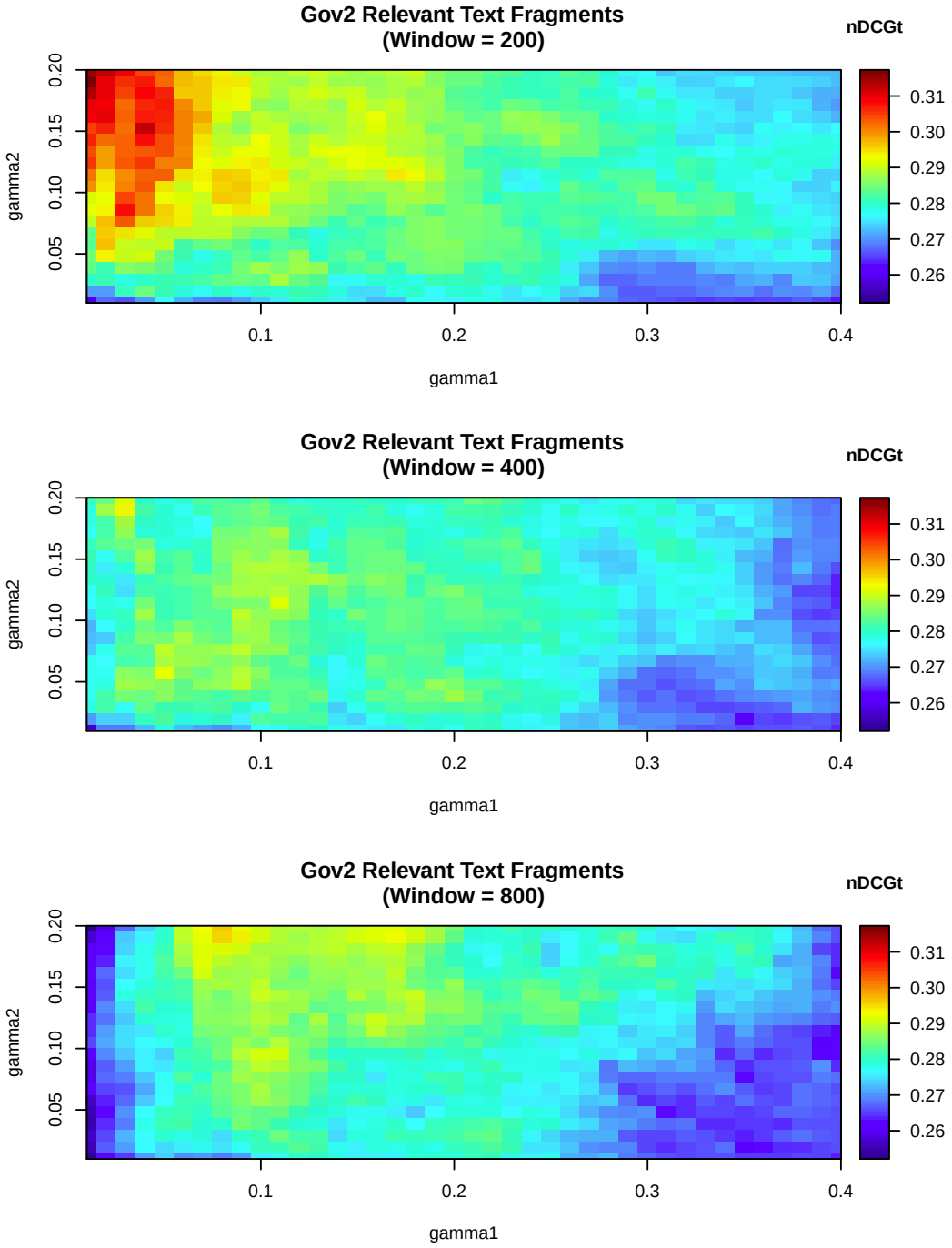


Figure 8.1. The effectiveness comparison of relevant text fragment retrieval of the Gov2 collection according to the interpolation weights. nDCG is measured at the top 5 passages.

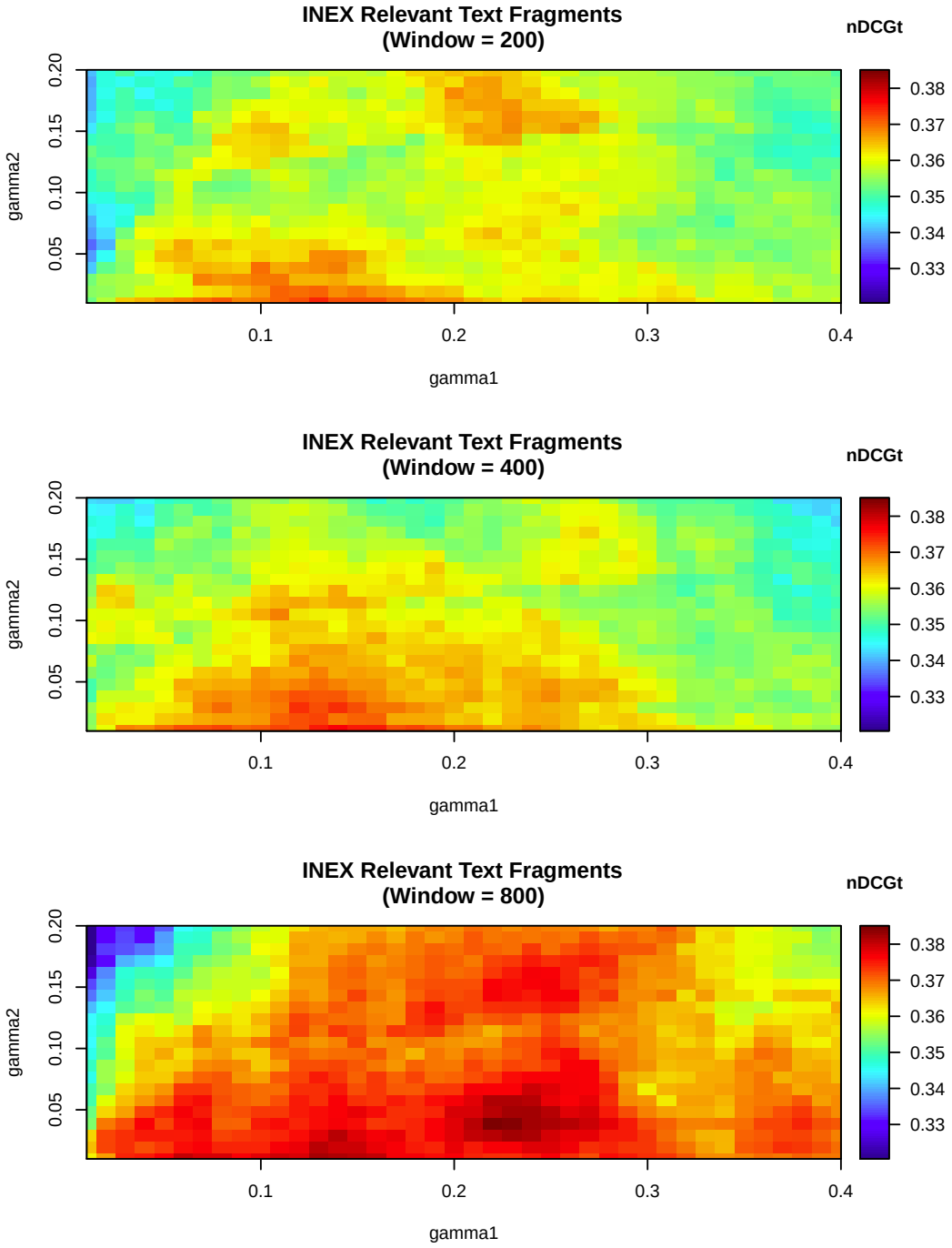


Figure 8.2. The effectiveness comparison of relevant text fragment retrieval of the INEX collection according to the interpolation weights. nDCG is measured at the top 5 passages.

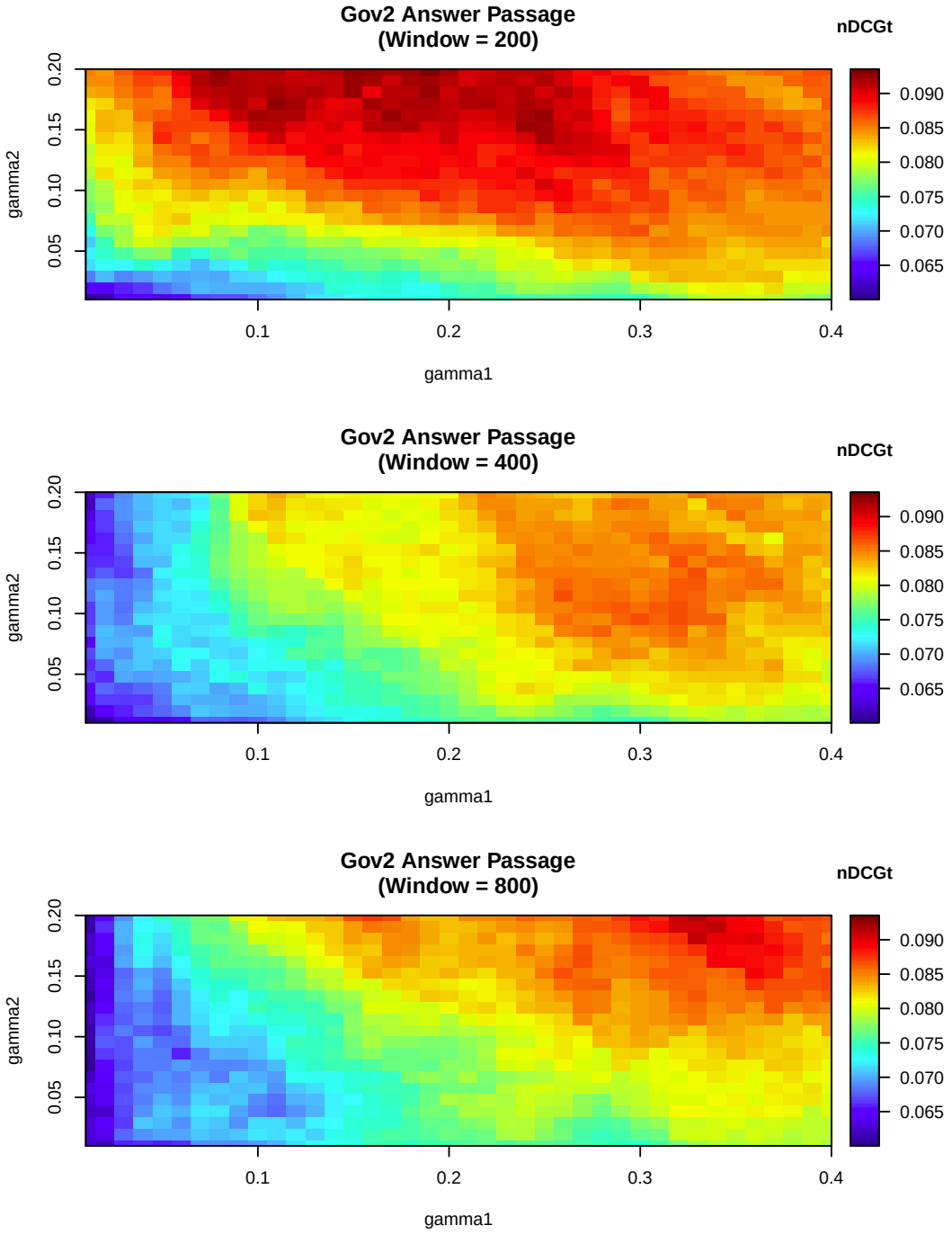


Figure 8.3. The effectiveness comparison of answer passage retrieval of the Gov2 collection according to the interpolation weights. nDCG is measured at the top 5 passages.

For the Gov2 collection, the optimal values of γ_1 and γ_2 are 0.01 and 0.19 for relevant text fragments of which the nDCG value is 0.3173. The optimal values for answer passage retrieval are 0.19 and 0.20 with the nDCG value 0.0935. In these experimental results, the optimal values of γ_2 for relevant text fragments and answer passage retrieval are similar. On the other hand, the higher weight on sentence-based passage retrieval model for retrieving relevant text fragment decrease the effectiveness of the retrieval model. Therefore, sentence-based passages are too short to measure the relevance scores of relevant text fragments.

For the INEX collection, $\gamma_1 = 0.22$ and $\gamma_2 = 0.03$ show the best results, nDCG 0.3751. The experimental results of the INEX collection demonstrate that sentence-based passages can be important evidence for determining the relevance scores, while window-based passages have little effect on the effectiveness of the overall retrieval results.

8.4.3 Evaluation of Query Term Expansion

We now evaluate the the effectiveness of the context-based translation model for answer passage retrieval. In these experiments, we set $N = 200$ for the Gov2 collection and $N = 400$ for the INEX collection, which shows the best results for each collection. We compare query term expansion results of the context-based translation model with the relevance model (Lavrenko and Croft, 2001). Expanded concepts are applied to the passage retrieval models $P(Q, psg_{(D,k)})$ and $P(Q, psg^N_{(D,k)})$ in Eq 8.3. If the value of γ were too small, the expanded concepts would not affect the experimental results. Therefore, we use 0.2 for γ_1 and γ_2 although these values do not show the best results in the previous section.

Table 8.1 shows the experimental result of the Per-Document task. *Baseline* is the interpolated passage retrieval model in Eq 8.3. *RM* is the experimental results with the expanded concepts of the relevance model. *Translation* is the translation

Table 8.1. The effectiveness of query term expansions for the Per-Document task with top 5 documents. Mean average term precision (MAtP) and normalized discounted cumulative gain (nDCGt) are measured in word.

	Relevant (Gov2)		Relevant (INEX)		Answer (Gov2)	
	MAtP	nDCGt	MAtP	nDCGt	MAtP	nDCGt
Baseline	0.0195	0.4095	0.0315	0.4245	0.0306	0.1838
RM	0.0197	0.4029	0.0328	0.4359	0.0306	0.1835
Translation	**0.0209**	**0.4154**	0.0317	0.4294	**0.0320**	0.1877
Secondary	0.0203	0.4115	**0.0328**	**0.4376**	0.0312	**0.1881**
Key+ Secondary	0.0187	0.4012	0.0311	0.4339	0.0292	0.1813

Table 8.2. The effectiveness of query term expansions for the Per-Passage task with top 10 passages. Mean average term precision (MAtP) and normalized discounted cumulative gain (nDCGt) are measured in word.

	Relevant (Gov2)		Relevant (INEX)		Answer (Gov2)	
	MAtP	nDCGt	MAtP	nDCGt	MAtP	nDCGt
Baseline	0.0151	0.2850	0.0247	0.3687	0.0181	0.0923
RM	**0.0181**	**0.2973**	**0.0298**	**0.3860**	**0.0198**	0.0888
Translation	0.0142	0.2888	0.0252	0.3721	0.0158	**0.0939**
Secondary	0.0140	0.2900	0.0227	0.3612	0.0150	0.0883
Key+ Secondary	0.0152	0.2828	0.0225	0.3708	0.0179	0.0924

model using mutual information without considering secondary concepts. *Secondary* shows the experimental results when we selectively apply the translation model. *Key+ Secondary* shows the results where we translated the secondary concepts of questions using the key concepts as context.

The translation model without using key concepts and secondary concepts shows good results for retrieving relevant text fragments and answer passages of the Gov2 collection. For the INEX collection, the translation model using secondary concepts shows the best results. However, the experimental results of all query term expansion methods failed to show significant improvement over the baseline. The context-based

Table 8.3. Experimental results of answer passage retrieval in sentence-level precision at N. **Entire** represents that we treat a retrieval result as correct answer if the entire retrieval result overlapped answer passages. **Partial** assumes that a retrieval result was correct if more than ten percent of the retrieval result overlapped.

	N=5		N=10		N=100	
	Entire	Partial	Entire	Partial	Entire	Partial
Baseline	0.0964	0.1927	0.0891	0.1855	0.0460	0.0958
RM	**0.1036**	**0.2091**	0.0864	0.1800	**0.0484**	**0.0981**
Translation	0.0982	0.2000	0.0882	0.1864	0.0449	0.0931
Secondary	0.0993	0.2082	**0.0932**	0.1921	0.0464	0.0949
Key+ Secondary	0.0973	0.2075	0.0930	**0.1935**	0.0466	0.0952

translation model using the key concepts of queries shows worse results than the baseline in most cases.

The experimental results in Table 8.2 of the Per-Passage task shows similar results. In these experiments, the relevance model shows the best results except for answer passage retrieval in the Gov2 collection. However, the relevance model failed to show significant improvement over the baseline, neither.

In the answer passage retrieval task, we aim to retrieve text fragments that can independently provide answers for users information needs. Instead of the ratio of terms retrieved from relevant text fragments or answer passages, it is also important to evaluate whether a retrieved passage cover an entire answer passage. Therefore, we evaluate answer passage retrieval results in sentence-level precision.

Table 8.3 shows the experimental results for the top 5, 10 and 100 passages. **Entire** represents that we treated a retrieved passage as a correct answer passage when the retrieved passage cover an entire answer passage. On the other hand, **Partial** treats a retrieved passage as a correct answer passage when more than ten percent of a retrieved passage overlapped annotated answer passages. The experimental results of the top 5 and 10 passages show that one out of ten retrieved passages was

Table 8.4. The experimental results using the true key concepts of queries for the context-based translation model.

Per-Doc	Relevant (Gov2)		Relevant (INEX)		Answer (Gov2)	
	MAP-w	nDCG-w	MAP-w	nDCG-w	MAP-w	nDCG-w
Baseline	0.0195	0.4095	0.0315	0.4245	0.0306	0.1838
True Key + Secondary	0.0213	0.4116	0.0338	0.4418	0.0341	0.1897

Per-Passage	Relevant (Gov2)		Relevant (INEX)		Answer (Gov2)	
	MAP-w	nDCG-w	MAP-w	nDCG-w	MAP-w	nDCG-w
Baseline	0.0151	0.2850	0.0247	0.3687	0.0181	0.0923
True Key + Secondary	0.0157	0.2891	0.0234	0.3742	0.0179	0.0956

an answer passage. One out of five retrieved passages was an answer passage or the part of an answer passage. The proportion of correct answer passages in the top 100 passages was 4.8% for **Entire** and 9.8% for **Partial**.

Table 8.4 shows the experimental results when we know the true key concepts of queries for the context-based translation model. This shows there is little margin for improvement by the context-based translation model using the key concepts of queries.

For query term expansion, the source for measuring the scores of expanded concepts are important. For example, Xu and Croft (1996) proposed a query term expansion method using the best passages instead of whole documents. We use query term expansion results for passage retrieval. Therefore, the relevance model based on the top ranked documents measures the scores of expanded concepts using inappropriate units.

Similarly, conditional mutual information for training the translation table of the context-based translation model use the co-occurrence information of an original

query term and an expanded term. However, the expanded concepts of the context-based translation model is used to evaluate the window-based passages and sentence-based passages. Therefore, the co-occurrence information of an original query term and an expanded term should be measured within passage levels instead of documents.

8.5 Summary

In this chapter, we proposed a passage retrieval model. In this model, we use two types of passages: a sentence-based passage and a windows-based passage. A sentence-based passage is used to measure the relevance score of an answer passage while a window-based passage is used evaluate the cohesion of relevant information around the answer passage. Experimental results show that the combination of different types of passages can improve the effectiveness of the overall passage retrieval model. However, the size of window-based passages and the weights for interpolating retrieval models in different granularities can differ according to document collections and tasks.

We also proposed the context-based translation model based on conditional mutual information for passage retrieval. We use conditional mutual information to train the translation table of the context-based translation model without using a paired sentence collection. However, the context-based translation model failed to show significant improvement. One of the reasons why the expanded concepts of the context-based translation model failed may be error in predicting the key concepts and secondary concepts of queries. In particular, considering the baseline query term expansion methods also failed to show significant improvements, a more serious problem is that we train the context-based translation model using inappropriate units of text.

In the next and final chapter of this dissertation, we will summarize the findings of this dissertation and discuss potential directions for future work.

CHAPTER 9

SUMMARY AND FUTURE WORK

In this chapter, we conclude the dissertation and provide a broad perspective on our work. We first summarize the dissertation by discussing the main results and our contributions. Then, we discuss the limitations of the current work and suggest potential directions for future research.

9.1 Conclusion and Contributions

The goal of the research in this dissertation is to investigate methods for utilizing the semantic and syntactic features implied by *verbose natural language queries* to improve the effectiveness of information retrieval models. In order to maximize the effectiveness of verbose natural language queries, we measure the importance of concepts in queries and evaluate the dependencies between concepts.

In Chapter 5, we proposed the quasi synchronous framework. Terms in verbose queries are used together to express information needs. Syntactic structures of natural language expressions reveal the dependence relationships between terms. In the quasi synchronous framework, we aim to solve two limitations in term dependence models based on the head-modifier relation in the syntactic parsing results.

First, although the head-modifier relation can cover dependence relationships over a longer distance, it will exclude important term dependencies compared to the term dependence model based on term proximity. Furthermore, important dependent term pairs are more likely not to have the head-modifier relation in verbose natural language queries. The quasi synchronous framework captures these dependence relationships

between term pairs by adopting more flexible syntactic configurations from the quasi synchronous stochastic approaches of machine translation.

Second, dependent term pairs may not be used in the same dependent relationship in queries and relevant documents. The current state-of-the-art term dependence model allows variations in the distance and order of dependent term pairs between queries and documents. In the quasi synchronous framework, we take account of variations in dependence relationships using inexact matching between different syntactic configurations.

By capturing dependent term pairs more flexibly and allowing inexact matching between queries and document, the quasi synchronous framework can significantly improve the effectiveness of term dependence model over the current state-of-the-art term dependence model, the sequential dependence model.

In Chapter 6, we investigate valid variations in dependence relationships. Although dependent term pairs of queries may not have the same syntactic relationship in relevant documents, this does not mean that terms co-occurring in documents have the same meaning as queries. According to the users' information needs, valid variations in the dependence relationships of term pairs can differ. Therefore, we evaluate the validity of variations in dependence relationships of term pairs using the Generative Relevance Hypothesis (GRH) and apply the evaluation results to the inexact matching process of the quasi synchronous framework.

We also compare the settings of the Generative Relevance Hypothesis for considering different characteristics of various dependence assumption in the quasi synchronous framework. Experimental results show that the validity of variations in dependence relationships should be evaluated corresponding with other term dependence assumptions in the overall term dependence model.

In Chapter 7, we propose the context-based translation model for query term expansion. Statistical translation models are used for query term expansions by

translating query terms for IR. Translations for an expression will differ when the expression is used in different contexts. The context-based translation model treats the key concepts of queries as the context of translations representing users' information needs. In addition, there are as more important terms in verbose queries and, therefore, the translation results of important terms should be treated more important. We use the evaluation result of query terms to selectively apply the context-based translation model for query term expansion. The context-based translation model shows significant improvement for finding answers from the question-answer collection.

We also suggest the important new task of focused retrieval of answer passages. In Chapter 4, we describe the process of annotating answer passages to construct the test collection for the task. In Chapter 8, we propose the passage retrieval model incorporating varying granularities of passages. Experimental results demonstrate that different types of passage-level evidences can be used together to improve the effectiveness of passage retrieval models. We also apply the context-based translation model using conditional mutual information of term pairs. In our approach, it turns out that we used inappropriate units for estimating conditional mutual information to train the probabilities of the translation table of the context-based translation model. Therefore, the context-based translation model failed to show improvement for answer passage retrieval. Study on the source for estimating conditional mutual information is future work.

9.2 Future Work

Although we tackled many critical issues, several challenges remain. We now briefly describe some of these challenges and suggest potential directions for future research.

- **Predefined Syntactic Configurations.** We use four predefined syntactic configurations adopted from the quasi synchronous stochastic process. These

syntactic configurations are defined in the tree form of the dependence parsing results of queries and documents. We can define more specific syntactic relationships to identify the dependence relationships of term pairs. For example, Maxwell et al. (2013) use the dependence path to evaluate the dependence relationship between a term pair. Cui et al. (2005) propose a method to measure the dependence path matching score to count the different dependence paths that term pairs can have documents. The quasi synchronous framework can be used for dependence relationships other than the four predefined syntactic configurations. Using different syntactic configurations and applying the GRH to these syntactic configurations, the quasi synchronous framework can more accurately consider term dependencies.

- **Integrating Two-phase Optimal Parameter Estimation.** In the quasi synchronous framework, we predict optimal weights in two phases. First, we estimate the optimal weight for individual term pairs using the GRH. Then, we estimate the optimal interpolation weights of the query likelihood model, the sequential dependence model and the quasi synchronous model. We can merge the independence assumption and the sequential dependence assumption into the set of predefined syntactic configurations to capture term dependencies in query. The ordered and unordered window term pairs also can be treated as predefined syntactic configurations. Then, we can merge two-phase parameter estimation processes. This will reduce the error propagation problem for the optimal parameter estimation.

- **Query Term Expansion for the Quasi-Synchronous Framework.** While we consider the dependence relationships between query terms, query term expansion results are used separately from original query terms. However, as query terms are used together to express users' information needs, expanded

concepts should be used together with original query terms to express relevant information. Therefore, we can configure the quasi synchronous framework to take account of an original query term and an expanded term as a dependent term pair. We can also use the GRH to evaluate the valid variations in dependent relationships of expanded term pairs.

- **Expanding Answer Passage Annotation** As we pointed out, the current answer passage annotation result does not tag all the text that can be good answers for users' information needs. Therefore we need to automatically augment the current answer passage annotation result. Carterette and Allan (1996) proposed a method for constructing sets of relevance judgments in which they intelligently select documents to be judged. For the answer passage annotation, we can use similarity measure such as the phase-level Recall-Oriented Understudy for Gisting Evaluation (ROUGE) measure in a similar way. This would mean extracting candidate text fragments from relevant documents and, then, evaluate the relevance scores of these candidates to find additional answer passages from relevant documents.

- **Training Source of the Context-based Translation Model** We estimate conditional mutual information using the co-occurrence frequency of an original query term and a candidate expansion in documents. However, as focused retrieval systems assume that the entire content of a document is not related to a single topic, the co-occurrence of term pairs in a document may not mean that these term are related in terms of users' information needs. Therefore, estimating the conditional mutual information of a term pair not in documents but in smaller units such as sentences could more accurately evaluate the relationship of a original query term and its expansions.

BIBLIOGRAPHY

Elif Aktolga, James Allan, and David A Smith. Passage reranking for question answering using syntactic structures and answer types. In *Advances in Information Retrieval*, pages 617–628. Springer, 2011.

James Allan. Hard track overview in TREC 2004 high accuracy retrieval from documents. *Computer Science Department Faculty Publication Series*, page 117, 2004.

James Allan, Margaret E. Connell, W. Bruce Croft, Fang-Fang Feng, David Fisher, , and Xiaoyan Li. Inquery and TREC-9. *Proceedings of 9th Text REtrieval Conference (TREC-9)*, pages 551–562, 2000.

Giambattista Amati, Claudio Carpineto, and Giovanni Romano. Query difficulty, robustness, and selective application of query expansion. In *Advances in information retrieval*, pages 127–137. Springer, 2004.

Paavo Arvola, Jaana Kekäläinen, and Marko Junkkari. Focused access to sparsely and densely relevant documents. In *Proceeding of the international SIGIR conference on Research and development in information retrieval*, pages 781–782. ACM, 2010.

Paavo Arvola, Shlomo Geva, Jaap Kamps, Ralf Schenkel, Andrew Trotman, and Johanna Vainio. Overview of the INEX 2010 ad hoc track. In *Comparative Evaluation of Focused Retrieval*, pages 1–32. Springer, 2011.

Niranjan Balasubramanian and James Allan. Syntactic query models for restatement retrieval. In *String Processing and Information Retrieval*, pages 143–155. Springer, 2009.

Niranjan Balasubramanian, Giridhar Kumaran, and Vitor R Carvalho. Exploring reductions for long web queries. In *Proceedings of the 33rd international ACM SIGIR conference on Research and development in information retrieval*, pages 571–578. ACM, 2010a.

Niranjan Balasubramanian, Giridhar Kumaran, and Vitor R Carvalho. Predicting query performance on the web. In *Proceedings of the 33rd international ACM SIGIR conference on Research and development in information retrieval*, pages 785–786. ACM, 2010b.

Michael Bendersky and W Bruce Croft. Discovering key concepts in verbose queries. In *Proceedings of SIGIR*, pages 491–498. ACM, 2008.

Michael Bendersky and W Bruce Croft. Analysis of long queries in a large scale search log. In *Proceedings of the workshop on Web Search Click Data*, pages 8–14. ACM, 2009.

Michael Bendersky and W Bruce Croft. Modeling higher-order term dependencies in information retrieval using query hypergraphs. In *Proceedings of the 35th international ACM SIGIR conference on Research and development in information retrieval*, pages 941–950, 2012.

Michael Bendersky and Oren Kurland. Utilizing passage-based language models for document retrieval. *Advances in Information Retrieval*, pages 162–174, 2008.

Michael Bendersky, W Bruce Croft, and David A Smith. Two-stage query segmentation for information retrieval. In *Proceedings of the 32nd international ACM SIGIR conference on Research and development in information retrieval*, pages 810–811. ACM, 2009.

Michael Bendersky, Donald Metzler, and W Bruce Croft. Learning concept importance using a weighted dependence model. In *Proceedings of the international conference on Web search and data mining*, pages 31–40. ACM, 2010.

Adam Berger and John Lafferty. Information retrieval as statistical translation. In *Proceedings of SIGIR*, pages 222–229. ACM, 1999.

Shane Bergsma and Qin Iris Wang. Learning noun phrase query segmentation. In *Proceedings of the 2007 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning (EMNLP-CoNLL)*, pages 819–826, 2007.

Delphine Bernhard and Iryna Gurevych. Combining lexical semantic resources with question & answer archives for translation-based answer finding. In *Proceedings of the Joint Conference of the 47th Annual Meeting of the ACL and the 4th International Joint Conference on Natural Language Processing of the AFNLP: Volume 2-Volume 2*, pages 728–736. Association for Computational Linguistics, 2009.

Roi Blanco and Christina Lioma. Graph-based term weighting for information retrieval. *Information retrieval*, 15(1):54–92, 2012.

Donna Bogatin. Yahoo: Searches more sophisticated and specific. http://www.zdnet.com/blog/micro-markets/yahoo-searches-more-sophisticated-and-specific/27, May 2006.

Peter F Brown, Vincent J Della Pietra, Stephen A Della Pietra, and Robert L Mercer. The mathematics of statistical machine translation: Parameter estimation. *Computational linguistics*, 19(2):263–311, 1993.

Emanuele Di Buccio, Massimo Melucci, and Federica Moro. Detecting verbose queries and improving information retrieval. *Information Processing & Management*, 2013.

James P. Callan. Passage-level evidence in document retrieval. In *Proceedings of the 17th annual international ACM SIGIR conference on Research and development in information retrieval*, pages 302–310, 1994.

Chih-Chung Chang and Chih-Jen Lin. *LIBSVM: a library for support vector machines*, 2001.

Hsinchun Chen and Tobun Dorbin Ng. An algorithmic approach to concept exploration in a large knowledge network (automatic thesaurus consultation): symbolic branch-and-bound search vs. connectionist hopfield net activation. 1995.

Massimiliano Ciaramita and Yasemin Altun. Broad-coverage sense disambiguation and information extraction with a supersense sequence tagger. In *Proceedings of the conference of EMNLP*, pages 594–602, 2006.

Massimiliano Ciaramita and Mark Johnson. Supersense tagging of unknown nouns in wordnet. In *Proceedings of the 2003 conference on Empirical methods in natural language processing*, pages 168–175. Association for Computational Linguistics, 2003.

Gao Cong, Long Wang, Chin-Yew Lin, Young-In Song, and Yueheng Sun. Finding question-answer pairs from online forums. In *Proceedings of the 31st annual international ACM SIGIR conference on Research and development in information retrieval*, pages 467–474. ACM, 2008.

William S Cooper. Some inconsistencies and misidentified modeling assumptions in probabilistic information retrieval. *ACM Transactions on Information Systems (TOIS)*, 13(1):100–111, 1995.

Gordon V Cormack, Mark D Smucker, and Charles LA Clarke. Efficient and effective spam filtering and re-ranking for large web datasets. *Information retrieval*, 14(5): 441–465, 2011.

W Bruce Croft and David J Harper. Using probabilistic models of document retrieval without relevance information. *Journal of documentation*, 35(4):285–295, 1979.

W. Bruce Croft, Howard R. Turtle, and David D. Lewis. The use of phrases and structured queries in information retrieval. In *Proceedings of the 14th ACM SIGIR conference*, pages 32–45. ACM, 1991.

W Bruce Croft, Donald Metzler, and Trevor Strohman. *Search engines: Information retrieval in practice*. Addison-Wesley Reading, 2010.

Steve Cronen-Townsend and W. Bruce Croft. Quantifying query ambiguity. In *Proceedings of the conference of HLT*, pages 104–109, 2002.

Steve Cronen-Townsend, Yun Zhou, and W Bruce Croft. Predicting query performance. In *Proceedings of the 25th annual international ACM SIGIR conference on Research and development in information retrieval*, pages 299–306. ACM, 2002.

Hang Cui, Renxu Sun, Keya Li, Min-Yen Kan, and Tat-Seng Chua. Question answering passage retrieval using dependency relations. In *Proceedings of the 28th annual international ACM SIGIR conference on Research and development in information retrieval*, pages 400–407, 2005.

Hoa Trang Dang, Diane Kelly, and Jimmy J Lin. Overview of the TREC 2007 question answering track. In *TREC*, volume 7, page 63. Citeseer, 2007.

Dipanjan Das and Noah A Smith. Paraphrase identification as probabilistic quasi-synchronous recognition. In *Proceedings of the Joint Conference of the 47th Annual Meeting of the ACL and the 4th International Joint Conference on Natural Language Processing of the AFNLP: Volume 1-Volume 1*, pages 468–476. Association for Computational Linguistics, 2009.

Marie-Catherine De Marneffe, Bill MacCartney, Christopher D Manning, et al. Generating typed dependency parses from phrase structure parses. In *Proceedings of LREC*, volume 6, pages 449–454, 2006.

Yuan Ding and Martha Palmer. Machine translation using probabilistic synchronous dependency insertion grammars. In *Proceedings of the 43rd Annual Meeting on Association for Computational Linguistics*, pages 541–548. Association for Computational Linguistics, 2005.

Jason Eisner. Learning non-isomorphic tree mappings for machine translation. In *Proceedings of the 41st Annual Meeting on Association for Computational Linguistics-Volume 2*, pages 205–208. Association for Computational Linguistics, 2003.

Christiane Fellbaum. Wordnet: An electronic lexical database. 1998. 2010. URL `http://www.cogsci.princeton.edu/wn`.

Norbert Fuhr, Jaap Kamps, Mounia Lalmas, Saadia Malik, and Andrew Trotman. Overview of the INEX 2007 Ad Hoc track. In *Focused Access to XML Documents*, pages 1–23. Springer, 2008.

Jianfeng Gao, Jian-Yun Nie, Guangyuan Wu, and Guihong Cao. Dependence language model for information retrieval. In *Proceedings of the 27th annual international ACM SIGIR conference on Research and development in information retrieval*, pages 170–177. ACM, 2004.

Jianfeng Gao, Xiaodong He, and Jian-Yun Nie. Clickthrough-based translation models for web search: from word models to phrase models. In *Proceedings of the 19th ACM international conference on Information and knowledge management*, pages 1139–1148. ACM, 2010.

Shlomo Geva, Jaap Kamps, Miro Lethonen, Ralf Schenkel, James A Thom, and Andrew Trotman. Overview of the INEX 2009 Ad Hoc track. In *Focused retrieval and evaluation*, pages 4–25. Springer, 2010.

Daniel Gildea. Loosely tree-based alignment for machine translation. In *Proceedings of the 41st Annual Meeting on Association for Computational Linguistics-Volume 1*, pages 80–87. Association for Computational Linguistics, 2003.

Norbert Gövert and Gabriella Kazai. Overview of the initiative for the evaluation of XML retrieval (INEX) 2002. In *INEX Workshop*, pages 1–17. Citeseer, 2002.

Gregory Grefenstette. Use of syntactic context to produce term association lists for text retrieval. In *Proceedings of the 15th annual international ACM SIGIR conference on Research and development in information retrieval*, pages 89–97. ACM, 1992.

Deepa Gupta and Niladri Chatterjee. Study of divergence for example based english-hindi machine translation. *STRANS-2001, IIT Kanpur*, pages 43–51, 2001.

Kadri Hacioglu. Semantic role labeling using dependency trees. In *Proceedings of the 20th international conference on Computational Linguistics*, page 1273. Association for Computational Linguistics, 2004.

Liliane Haegeman. *Introduction to government and binding theory*. 1991.

Mark Hall, Eibe Frank, Geoffrey Holmes, Bernhard Pfahringer, Peter Reutemann, and Ian H Witten. The weka data mining software: an update. *ACM SIGKDD explorations newsletter*, 11(1):10–18, 2009.

Donna Harman. Overview of the trec 2002 novelty track. In *TREC*, 2002.

Claudia Hauff, Vanessa Murdock, and Ricardo Baeza-Yates. Improved query difficulty prediction for the web. In *Proceedings of the 17th ACM conference on Information and knowledge management*, pages 439–448. ACM, 2008.

Ben He and Iadh Ounis. Inferring query performance using pre-retrieval predictors. In *String Processing and Information Retrieval*, pages 43–54. Springer, 2004.

Marti A Hearst and Christian Plaunt. Subtopic structuring for full-length document access. In *Proceedings of the 16th annual international ACM SIGIR conference on Research and development in information retrieval*, pages 59–68. ACM, 1993.

Samuel Huston and W Bruce Croft. Evaluating verbose query processing techniques. In *Proceedings of the 33rd international ACM SIGIR conference on Research and development in information retrieval*, pages 291–298. ACM, 2010.

Kalervo Järvelin and Jaana Kekäläinen. Using graded relevance assessments in IR evaluation. *Journal of the American Society for Information Science and Technology*, 53(13):1120–1129, 2002.

Jiwoon Jeon, W Bruce Croft, and Joon Ho Lee. Finding similar questions in large question and answer archives. In *Proceedings of the 14th ACM international conference on Information and knowledge management*, pages 84–90. ACM, 2005.

Yufeng Jing and W Bruce Croft. An association thesaurus for information retrieval. In *Proceedings of RIAO*, volume 94, pages 146–160. Citeseer, 1994.

Thorsten Joachims. *Learning to classify text using support vector machines: Methods, theory and algorithms*. Kluwer Academic Publishers, 2002.

K Sparck Jones and David M Jackson. The use of automatically-obtained keyword classifications for information retrieval. *Information Storage and Retrieval*, 5(4): 175–201, 1970.

K Sparck Jones and Roger M Needham. Automatic term classifications and retrieval. *Information Storage and Retrieval*, 4(2):91–100, 1968.

Maryam Karimzadehgan and ChengXiang Zhai. Estimation of statistical translation models based on mutual information for ad hoc information retrieval. In *Proceedings of SIGIR*, pages 323–330. ACM, 2010.

Marcin Kaszkiel and Justin Zobel. Passage retrieval revisited. In *ACM SIGIR Forum*, volume 31, pages 178–185. ACM, 1997.

Marcin Kaszkiel and Justin Zobel. Effective ranking with arbitrary passages. *Journal of the American Society for Information Science and Technology*, 52(4):344–364, 2001.

Boris Katz and Jimmy Lin. Selectively using relations to improve precision in question answering. In *Proceedings of the workshop on Natural Language Processing for Question Answering (EACL 2003)*, pages 43–50, 2003.

Dan Klein and Christopher D Manning. Fast exact inference with a factored model for natural language parsing. *Advances in neural information processing systems*, pages 3–10, 2003.

Philipp Koehn, Franz Josef Och, and Daniel Marcu. Statistical phrase-based translation. In *Proceedings of the 2003 Conference of the North American Chapter of the Association for Computational Linguistics on Human Language Technology-Volume 1*, pages 48–54. Association for Computational Linguistics, 2003.

Giridhar Kumaran and James Allan. A case for shorter queries, and helping users create them. In *Proceedings of NAACL HLT*, pages 220–227, 2007.

Giridhar Kumaran and Vitor R. Carvalho. Reducing long queries using query quality predictors. In *Proceedings of the 32nd international ACM SIGIR conference on Research and development in information retrieval*, pages 564–571. ACM, 2009.

Victor Lavrenko. *A Generative Theory of Relevance*, volume 26. Springer, 2009.

Victor Lavrenko and W Bruce Croft. Relevance based language models. In *Proceedings of the 24th annual international ACM SIGIR conference on Research and development in information retrieval*, pages 120–127. ACM, 2001.

Changki Lee, Gary Geunbae Lee, and Myung-Gil Jang. Dependency structure applied to language modeling for information retrieval. *ETRI journal*, 28(3):337–346, 2006.

Chia-Jung Lee, Ruey-Cheng Chen, Shao-Hang Kao, and Pu-Jen Cheng. A term dependency-based approach for query terms ranking. In *Proceedings of the 18th ACM conference on Information and knowledge management*, pages 1267–1276. ACM, 2009.

Jung-Tae Lee, Sang-Bum Kim, Young-In Song, and Hae-Chang Rim. Bridging lexical gaps between queries and questions on large online q&a collections with compact translation models. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing*, pages 410–418. Association for Computational Linguistics, 2008.

Xiaoyong Liu and W. Bruce Croft. Passage retrieval based on language models. In *Proceedings of the eleventh international conference on Information and knowledge management*, pages 375–382. ACM, 2002.

Rachel Tsz-Wai Lo, Ben He, and Iadh Ounis. Automatically building a stopword list for an information retrieval system. In *Journal on Digital Information Management: Special Issue on the 5th Dutch-Belgian Information Retrieval Workshop (DIR)*, volume 5, pages 17–24, 2005.

Robert M Losee. Term dependence: truncating the Bahadur Lazarsfeld expansion. *Information processing & management*, 30(2):293–303, 1994.

Wen-Hsiang Lu, Lee-Feng Chien, and Hsi-Jian Lee. Translation of web queries using anchor text mining. *ACM Transactions on Asian Language Information Processing (TALIP)*, 1(2):159–172, 2002.

Yuanhua Lv and ChengXiang Zhai. Positional language models for information retrieval. In *Proceedings of the 32nd international ACM SIGIR conference on Research and development in information retrieval*, pages 299–306. ACM, 2009.

Loic Maisonnasse, Eric Gaussier, and Jean-Pierre Chevallet. Revisiting the dependence language model for information retrieval. In *Proceedings of the 30th annual international ACM SIGIR conference on Research and development in information retrieval*, pages 695–696. ACM, 2007.

Rila Mandala, Takenobu Tokunaga, and Hozumi Tanaka. Combining multiple evidence from different types of thesaurus for query expansion. In *Proceedings of the 22nd annual international ACM SIGIR conference on Research and development in information retrieval*, pages 191–197. ACM, 1999.

Christopher D. Manning and Hinrich Schütze. *Foundations of statistical natural language processing*. MIT press, 1999.

K. Tamsin Maxwell, Jon Oberlander, and W. Bruce Croft. Feature-based selection of dependency paths in ad hoc information retrieval. In *Proceedings of the 43rd Annual Meeting on Association for Computational Linguistics*. Association for Computational Linguistics, 2013.

I Dan Melamed, Giorgio Satta, and Benjamin Wellington. Generalized multitext grammars. In *Proceedings of the 42nd Annual Meeting on Association for Computational Linguistics*, page 661. Association for Computational Linguistics, 2004.

Donald Metzler and W Bruce Croft. A Markov random field model for term dependencies. In *Proceedings of SIGIR*, pages 472–479. ACM, 2005.

Donald Metzler and W Brucel Croft. Latent concept expansion using Markov random fields. In *Proceedings of the 30th annual international ACM SIGIR conference on Research and development in information retrieval*, pages 311–318. ACM, 2007.

Donald A Metzler. Automatic feature selection in the Markov random field model for information retrieval. In *Proceedings of the sixteenth ACM conference on Conference on information and knowledge management*, pages 253–262. ACM, 2007.

Vanessa Murdock and W Bruce Croft. A translation model for sentence retrieval. In *Proceedings of HLT-EMNLP*, pages 684–691. Association for Computational Linguistics, 2005.

Jae Hyun Park and W. Bruce Croft. Query term ranking based on dependency parsing of verbose queries. In *Proceeding of the 33rd international ACM SIGIR conference on Research and development in information retrieval*, pages 829–830, 2010.

Jae Hyun Park, W Bruce Croft, and David A Smith. A quasi-synchronous dependence model for information retrieval. In *Proceedings of the 20th ACM international CIKM*, pages 17–26. ACM, 2011.

Jie Peng, Craig Macdonald, Ben He, Vassilis Plachouras, and Iadh Ounis. Incorporating term dependency in the DFR framework. In *Proceedings of the 30th annual international ACM SIGIR conference on Research and development in information retrieval*, pages 843–844. ACM, 2007.

Jay M Ponte and W Bruce Croft. Text segmentation by topic. In *Research and Advanced Technology for Digital Libraries*, pages 113–125. Springer, 1997.

Jay M. Ponte and W. Bruce Croft. A language modeling approach to information retrieval. In *Proceedings of the 21st annual international ACM SIGIR conference on Research and development in information retrieval*, pages 275–281. ACM, 1998.

Yonggang Qiu and Hans-Peter Frei. Concept based query expansion. In *Proceedings of the 16th annual international ACM SIGIR conference on Research and development in information retrieval*, pages 160–169. ACM, 1993.

Dragomir R Radev, Hong Qi, Zhiping Zheng, Sasha Blair-Goldensohn, Zhu Zhang, Weiguo Fan, and John Prager. Mining the web for answers to natural language questions. In *Proceedings of the tenth international conference on Information and knowledge management*, pages 143–150. ACM, 2001.

Yves Rasolofo and Jacques Savoy. Term proximity scoring for keyword-based retrieval systems. In *Advances in Information Retrieval*, pages 207–218. Springer, 2003.

Stefan Riezler, Alexander Vasserman, Ioannis Tsochantaridis, Vibhu Mittal, and Yi Liu. Statistical machine translation for query expansion in answer retrieval. In *ANNUAL MEETING-ASSOCIATION FOR COMPUTATIONAL LINGUISTICS*, volume 45, page 464, 2007.

Stefan Riezler, Yi Liu, and Alexander Vasserman. Translating queries into snippets for improved query expansion. In *Proceedings of the 22nd International Conference on Computational Linguistics-Volume 1*, pages 737–744. Association for Computational Linguistics, 2008.

Ian Roberts and Robert Gaizauskas. Evaluating passage retrieval approaches for question answering. *Advances in Information Retrieval*, pages 72–84, 2004.

Stephen E Robertson and K Sparck Jones. Relevance weighting of search terms. *Journal of the American Society for Information science*, 27(3):129–146, 1976.

Stephen E Robertson and Steve Walker. Some simple effective approximations to the 2-poisson model for probabilistic weighted retrieval. In *Proceedings of the 17th annual international ACM SIGIR conference on Research and development in information retrieval*, pages 232–241. Springer-Verlag New York, Inc., 1994.

Tetsuya Sakai, Makoto P Kato, and Young In Song. Overview of NTCIR-9 1click. *Proceedings of NTCIR-9*, pages 180–201, 2011.

Gerard Salton. Automatic term class construction using relevance – summary of work in automatic pseudoclassification. *Information Processing & Management*, 16(1): 1–15, 1980.

Gerard Salton and Christopher Buckley. Term-weighting approaches in automatic text retrieval. *Information processing & management*, 24(5):513–523, 1988.

Gerard Salton, Anita Wong, and Chung-Shu Yang. A vector space model for automatic indexing. *Communications of the ACM*, 18(11):613–620, 1975.

Gerard Salton, James Allan, and Chris Buckley. Approaches to passage retrieval in full text information systems. In *Proceedings of the 16th annual international ACM SIGIR conference on Research and development in information retrieval*, pages 49–58. ACM, 1993.

Gerard Salton, James Allan, and Amit Singhal. Automatic text decomposition and structuring. *Information Processing & Management*, 32(2):127–138, 1996.

Hinrich Schütze and Jan O Pedersen. A cooccurrence-based thesaurus and two applications to information retrieval. *Information Processing & Management*, 33(3): 307–318, 1997.

Stuart M Shieber and Yves Schabes. Synchronous tree-adjoining grammars. In *Proceedings of the 13th conference on Computational linguistics-Volume 3*, pages 253–258, 1990.

David A Smith and Jason Eisner. Quasi-synchronous grammars: Alignment by soft projection of syntactic dependencies. In *Proceedings of the Workshop on Statistical Machine Translation*, pages 23–30. Association for Computational Linguistics, 2006.

Mark D Smucker, James Allan, and Ben Carterette. A comparison of statistical significance tests for information retrieval evaluation. In *Proceedings of the sixteenth ACM conference on Conference on information and knowledge management*, pages 623–632. ACM, 2007.

Y.I. Song, K.S. Han, S.B. Kim, S.Y. Park, and H.C. Rim. A novel retrieval approach reflecting variability of syntactic phrase representation. *Journal of Intelligent Information Systems*, 31(3):265–286, 2008. ISSN 0925-9902.

Munirathnam Srikanth and Rohini Srihari. Biterm language models for document retrieval. In *Proceedings of the 25th annual international ACM SIGIR conference on Research and development in information retrieval*, pages 425–426. ACM, 2002.

Trevor Strohman, Donald Metzler, Howard Turtle, and W Bruce Croft. Indri: A language model-based search engine for complex queries. In *Proceedings of the International Conference on Intelligent Analysis*, volume 2, pages 2–6. Citeseer, 2005.

Mihai Surdeanu, Massimiliano Ciaramita, and Hugo Zaragoza. Learning to rank answers to non-factoid questions from web collections. *Computational Linguistics*, 37(2):351–383, 2011.

Tao Tao and ChengXiang Zhai. An exploration of proximity measures in information retrieval. In *Proceedings of the 30th annual international ACM SIGIR conference on Research and development in information retrieval*, pages 295–302. ACM, 2007.

Andrew Trotman and Shlomo Geva. Passage retrieval and other XML-retrieval tasks. In *Proceedings of the SIGIR 2006 Workshop on XML Element Retrieval Methodology*, pages 43–50. Department of Computer Science, University of Otago, 2006.

Ellen M Voorhees. Query expansion using lexical-semantic relations. pages 61–69. Springer, 1994.

Ellen M. Voorhees. Overview of the TREC-9 question answering track. In *TREC*, 2000.

Ellen M Voorhees. Overview of the TREC 2001 question answering track. In *Proceedings of TREC*, 2001a.

Ellen M Voorhees. Question answering in TREC. In *Proceedings of the CIKM*, pages 535–537. ACM, 2001b.

Ellen M Voorhees. Overview of the TREC 2003 question answering track. In *Proceedings of TREC*, pages 54–68, 2004.

Ellen M Voorhees. Overview of the TREC 2004 question answering track. In *Proceedings of TREC*, pages 52–62, 2005.

Mengqiu Wang, Noah A. Smith, and Teruko Mitamura. What is the Jeopardy model? a quasi-synchronous grammar for QA. In *Proceedings of the 2007 Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning*, pages 22–32, 2007.

Jinxi Xu and W. Bruce Croft. Query expansion using local and global document analysis. In *Proceedings of the 19th annual international ACM SIGIR conference on Research and development in information retrieval*, pages 4–11. ACM, 1996.

Xiaobing Xue and W Bruce Croft. Modeling subset distributions for verbose queries. In *Proceedings of the 34th international ACM SIGIR conference on Research and development in Information Retrieval*, pages 1133–1134. ACM, 2011.

Xiaobing Xue, Jiwoon Jeon, and W Bruce Croft. Retrieval models for question and answer archives. In *Proceedings of SIRIR*, pages 475–482. ACM, 2008.

Xiaobing Xue, Samuel Huston, and W Bruce Croft. Improving verbose queries using subset distribution. In *Proceedings of the 19th ACM international conference on Information and knowledge management*, pages 1059–1068. ACM, 2010.

Kenji Yamada and Kevin Knight. A syntax-based statistical translation model. In *Proceedings of the 39th Annual Meeting on Association for Computational Linguistics*, pages 523–530. Association for Computational Linguistics, 2001.

C. T. Yu, Chris Buckley, K. Lam, and Gerard Salton. A generalized term dependence model in information retrieval. Technical report, Cornell University, 1983.

ChengXiang Zhai and John Lafferty. Two-stage language models for information retrieval. In *Proceedings of the 25th annual international ACM SIGIR conference on Research and development in information retrieval*, pages 49–56. ACM, 2002.

Guangyou Zhou, Li Cai, Jun Zhao, and Kang Liu. Phrase-based translation model for question retrieval in community question answer archives. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies-Volume 1*, pages 653–662. Association for Computational Linguistics, 2011.

Guangyou Zhou, Yang Liu, Fang Liu, Daojian Zeng, and Jun Zhao. Improving question retrieval in community question answering using world knowledge. In *Proceedings of the Twenty-Third international joint conference on Artificial Intelligence*, pages 2239–2245. AAAI Press, 2013.