Transformational Tagging for Topic Tracking in Natural Language

葉俊華
IP Chun Wah Timmy

A Thesis
Submitted in Partial Fulfilment of the Requirements for the Degree of
Master of Philosophy
in
Systems Engineering and Engineering Management

June 2000

Abstract

Topic tracking is a new information trend in the research area of Information Retrieval (IR). The goal of the tracking task is to correctly classify all subsequent stories as to whether or not they are discussing the target topic.

This thesis proposes the use of named entities on topic tracking in Chinese text. We addressed three subproblems within the context of this task. First, we develop a part-of-speech tagger with a Transformation-based Error-Driven (TEL) approach, and the tagger is able to label named entities from Chinese documents. Second, we combine an n-gram technique with heuristics for detecting and identifying unknown words. This unknown word (OOV) identifier is capable of salvaging named entities lost due to tokenization errors and unknown word problems. Third, we integrated our TEL tagger and OOV identifier with a vector-space retrieval model. Named entities processed by the TEL tagger and OOV identifier will be used by vector-space information retrieval model to perform topic tracking. Our performance is measured in terms of evaluation costs and displayed as detection error trade-off (DET) curves. Our experimental results show that the use of named entities brought about approximately 28% evaluation cost improvement over the purely data-driven baseline approach.

摘要

主題追索是資訊檢索領域中的新興課題，其研究目的是根據訓練新聞的內容，對隨後的新聞作出適當的二元分類（屬於訓練新聞主題或是不屬主題）。

本論文提出了特定名詞在中文主題追索上的應用，其中包括三項題目。第一，我們引用了錯誤驅動的轉換式學習方法，發展了一個中文詞類標認器，用以在中文文件中標認出特定名詞。第二，我們揉合了N元模型的優點及中文語言的知識，發展了一個未知詞辨認器，用以偵察及辨認出未知詞，找出因斷詞錯誤及未知詞問題遺漏的特定名詞。第三，我們結合了以上的兩個系統，再加上向量資訊檢索模型，組成了一個主題追索系統：詞類標認器及未知詞辨認器首先找出了新聞內的特定名詞，再由向量資訊模型設定這些特徵詞的權重，最後在測試資料中進行二元分類的工作。

我們根據評估成本及偵察錯誤權衡曲線，評估了我們的實驗。結果顯示引用特定名詞的特徵抽取方法，與純粹數據的抽取方法比較，在評估成本上提升了大約28%的改善。

Acknowledgments

Someone says, "Doing research is a boring and arduous task". I agree with this statement if we do the research alone. Fortunately, I am not alone in this arduous road. There are many friends who support and encourage me. Their encouragement strengthen me to pass through this road and complete the degree.

I would like to thank my supervisor, Helen Meng, for her guidance and invaluable advice in my research. She also taught me the skill of problem-solving, and trained me to be a person who have confidence to face problems. I would also like to thank my thesis committee, Professors Wai Lam, Boon-Toh Low, Chun-Hung Cheng from the Department of Systems Engineering and Engineering Management, and Li-De Wu from the Fudan University for their precious comments and feedback.

I wish to thank my parents, who have given me concerns and support every time. They prepared a warm family for me. No matter how I was tired and anxious, I could regain my energy from this comfortable environment.

I would also thank everyone in the department of Systems Engineering and Engineering Management: Ada, "Ah-Fan", "Ah-Kin", Alex, Brenda, Carmen, Connie, "Fat-Yuk", Sally, Silvia, Tiffany, Timothy, Tony and "Wai-

Ip". Without these precious friends, my study in Master degree would be boring and lonely.

I would like to give my special thanks to my dearest Jesus Christ. Thanks to Him for giving me an opportunity for the study in Master degree. Whenever I feel hopeless in my research, He makes a new way for me. Whenever I feel helpless and afraid, He used His words to comfort me.

Finally, I would like to thank all of the brothers and sisters who pray for my study. Without their prayer, I may not complete my degree.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

News is one of the richest information sources for the human. With the explosive growth in the amount of news available on different media (newswire, radio, televisions, web sites), the significance of monitoring worldwide events from several sources of news is clear. For example, much useful information could be gleaned from various news sources, but no one has the time to read such large volumes of news carefully. Therefore, it would be very helpful to have a system which can filter for relevant news automatically. For example, segmenting the news stream into individual stories (the problem of story segmentation), determining which stories go with one another (the problem of topic tracking), and discovering when something new has happened (the problem of topic detection) [50][57].

In this thesis, we discuss and evaluate solutions to the topic tracking tasks in broadcast news. The goal of the tracking task is to correctly classify all subsequent stories as to whether or not they are discussing the target topic [51]. Our approach to the problem of topic tracking is related to the problem of "named entities", which often consist of key information identifying a topic. The key information includes time, name of person, name of organization, name of place, etc. One difficulty of using named entities for topic tracking is to accurately extract information concerning proper nouns. In this work, we use the Transformation-based Error-driven Learning (TEL) approach [22] for tagging Chinese text. Various kinds of named entities are captured in the tags and are fed to the topic tracker subsequently.

1.1 Topic Detection and Tracking

The Linguistic Data Consortium (LDC)[1] has prepared a news corpus named the Topic Detection and Tracking (TDT) corpus, which provides the test bed for research development of topic tracking technologies.

The TDT project was started in 1997 [57]. In the beginning, the TDT Pilot Project (TDT-1) only addressed the newswire (Reuters) and transcribed broadcast news (CNN). In the 1998 Topic Detection and Tracking (TDT-2) project, the LDC has prepared the TDT-2 corpus, a collection of newswire and transcribed broadcasts from 6 sources, including 2 newswires, 2 radio programs and 2 television [37]. In the 1999 Topic Detection and Tracking (TDT-3) project, the Mandarin Chinese News were added to the corpus [12]. All the data for TDT-3 were collected from broadcast and newswire sources covering October 2 through December 1998. Detail descriptions of the TDT corpus will be discussed in Chapter 6.

[1] http://www.ldc.upenn.edu/

1.1.1 What is a Topic?

The definition of "topic" was limited to be an "event" in the TDT pilot study. It refers to a unique event that happens at some point in time [34]. For example, "１９９８年１１月２５日ＮＢＡ資方拒絕談判" (transcribed as: Negotiations between the NBA players union and owners break down in 25, November, 1998) is considered as an event, whereas "ＮＢＡ勞資糾紛" (NBA labor disputes) is the general topic of this type of event.

In the TDT-2 project, a topic can be defined as other events and activities that are directly related to it. Here are the definitions of essential terms in the TDT-2 project [52]:

Topic A topic is an event or activity, along with all directly related events and activities.

Event An event is something that happens at some specific time and place, and the unavoidable consequences. Specific elections, accidents, crimes and natural disasters are examples of events.

Activity An activity is a connected set of actions that have a common focus or purpose. Specific campaigns, investigations, and disaster relief efforts are examples of activities.

The definition of topic is unchanged in TDT-3 project. Stories will be considered as "on topic" whenever it is directly related to a pre-specified event. For example, a story on the search for survivors of an airplane crash, or on the funeral of the crash victims, will be considered to be a story on the crash event [51].

1.1.2 What is Topic Tracking?

Topic Tracking is one of the five tasks[2] in TDT-3. The TDT topic tracking task is to correctly classify all subsequent stories as to whether or not they discuss the target topic which is defined by one or more stories. For example, the four stories about "Asian Games in Thailand" may be supplied by the user to train the tracking system. Then, the system needs to classify whether a subsequent news story is about the same topic of "Asian Games in Thailand". For example, a story about results of individual competitions is *on-topic* but a story about car bombs in Thailand is *off-topic.*

The topic tracking problem is similar to the information filtering and information routing problems. [35][62]. The main difference between topic tracking and information filtering/routing is the formulation of the user's query. The query in information filtering and routing is generally described by the user, while the query of topic tracking task is specified using a few stories provided by the user. Therefore, topic tracking can be viewed as a narrowly defined task of information filtering [35].

1.2 Research Contributions

There has been a number of recent efforts towards the task of topic tracking. Most of them are data-driven approaches. For example, BBN [72] used probabilistic approaches based on a topic spotting model, an information retrieval model and a relevance feedback models. Dragon Systems [44] used statistical

[2] The other four are Story Segmentation, Topic Detection, First-Story Detection and Story Link Detection.

approaches based on a beta-binomial model and a unigram language model. CMU [76] used three learning strategies based on a k-nearest-neighbor approach, a Rocchio approach and a language modeling approach.

In this thesis, we propose to make use of *named entities* for topic tracking. In our approach, we use named entity tagging to extract possibly useful named entities, e.g., person names, location and time. These named entities are then used classified by applying vector-space models. The main difference between our approach and the previous approaches is on feature selection. Previous approaches utilize mainly statistical information while our approach mainly utilize concepts which mainly consist of the named entities.

There are two research goals in this work. First, we demonstrate how the TEL approach can be applied to Chinese text to extract named entities (in the form of POS tags). Second, we evaluate the utility of the extracted named entities in the task of topic tracking. We have also applied two other techniques for extracting information attributes for the tracker. They are: (i) n-gram grouping [47], and (ii) formulation rules[3] for extracting named entities.

1.2.1 Named Entity Tagging

In this research, we applied the part-of-speech (POS) tagging technique to named entities labeling. We attempt to use POS tags to identify useful entities (e.g. time, person names, organization names, etc). We applied the Transformational-based Error-Driven Learning (TEL) approach to POS tag-

[3] The morphological rules used for extracting named entities [31].

ging for Chinese text. This is, to the best of our knowledge, the first attempt in applying the TEL technique for Chinese POS tagging. We have carefully analyzed the comparative contribution of the lexical and contextual rules from our TEL taggers, and benchmarked the performance of TEL against a stochastic tagger.

Our contributions in Chinese POS tagging are:

1. An initial attempt in using the TEL procedure for tagging Chinese text.

2. An analysis of the relative contribution of the lexical and contextual rules.

3. A comparison with a stochastic tagger and a demonstration of the effectiveness of TEL for Chinese.

1.2.2 Handling Unknown Words

Chinese word segmentation is our initial step in topic tracking. However, the unknown words may cause incorrect segmentations of named entities, which affect the performance of topic tracking. We tackle the out-of-vocabulary (OOV) problem by means of an n-gram grouping [47] [58] and formulation rules [45][31]. An OOV identifier is developed with the previous techniques for salvaging named entities lost due to tokenization errors and OOV problems.

Our research contributions towards OOV handling are:

1. The use of an n-gram grouping technique for detecting unknown words.

2. The use of entity tagging techniques together with heuristic rules for identifying the meaning (types of named entity) of unknown words.

1.2.3 Named-Entity Approach in Topic Tracking

We introduce the "Named-Entity" approach to topic tracking task. It transforms training stories into a series of concept (named entities) using the TEL tagger and OOV identifier. This extracted information will be used to formulate classifier in applying the vector-space information retrieval model to topic tracking. Evaluation is based on the *detection error tradeoff* (DET) curves [3] which measures the tradeoff between misses and false alarm. We will discuss the evaluation methodology in chapter 6.

Our contributions towards the problem of topic tracking is mainly on the use of named entities together with the vector-space information retrieval model to perform topic tracking.

1.3 Organization of Thesis

The rest of this thesis is organized as follows. Chapter 2 presents the previous work in topic tracking. The description of the problems of Chinese text processing and the background knowledge on Chinese part-of-speech Tagging, Unknown Word Identification and Information Retrieval Models will be provided in this chapter.

In Chapter 3, we present an overview of our topic tracking system. It includes a Chinese word segmenter, a TEL tagger, an OOV identifier and a topic tracker. These components of our system will be described in detail in the subsequent chapters. Chapter 4 introduces the TEL approach and its performance on Chinese POS tagging. Chapter 5 presents our proposed n-gram grouping method with formulation rules for OOV detection and identification. Chapter 6 presents our approach in using named entities for the topic tracking task. Conclusions and future directions are presented in Chapter 7.

Chapter 2

Background

Topic tracking is a new problem in the research area of Information Retrieval (IR). The earliest work on topic tracking was reported in 1998 [34], and several research sites such as BBN Technologies [32] and Dragon Systems [42] have developed their systems over the last three years. Work on topic tracking in Chinese text is more recent, with some initial work reported in the Topic Detection and Tracking Workshop 2000. Nevertheless, almost all of the developed systems are built upon English language (or translated English text from Mandarin). Work on Chinese topic tracking is relatively sparse.

In this chapter, we will give a brief overview of previous developments in Topic Tracking. Following that, we discuss our work related to topic tracking in Chinese. It includes POS tagging, OOV detection and identification, and information retrieval models.

2.1 Previous Developments in Topic Tracking

Topic tracking is a new and challenging research area. The problem of topic tracking is to give a few stories (4 stories is the default evaluation condition) about a particular topic to a system. Thereafter, the system needs to produce a score for each story from the incoming news streams whether this story is on-topic or off-topic. Since all the stories for topic tracking are collected from broadcast and newswire sources, we are faced in the challenge to accommodate errors produced by speech recognition for broadcast news audio.

Eight research sites participated in NIST's 1998 Topic Detection and Tracking (TDT-2) and the 1999 Topic Detection and Tracking Evaluation (TDT-3) [37]. In the following, we will give a brief descriptions for four leading systems according to the result of TDT-2 and TDT-3 participants.

2.1.1 BBN's Tracking System

The BBN tracking system [27] [32] [72] is based on three probabilistic models – Topic Spotting (TS), Information Retrieval (IR) and Relevance Feedback (RF). Assume S is the story, T is the group of stories on a topic, $SisR$ is the story S relevant to the topic. TS model is used to calculate the probability that S comes from the distribution of topic $P(T/S)$. The IR model is used to calculate the probability that any new story S is relevant given the topic model $P(SisR/T)$. The RF model is similar to the IR models except uses frequently occurring words in the training stories instead of all of the words in the training stories.

The scores of the three models are normalized by the following formula [72]:

$$score^{'}(D,T) = \frac{score(D,T) - \mu_{no}}{\sigma_{no}} \tag{2.1}$$

where μ_{no} is the mean score of all *no* documents, and σ_{no} is the standard deviation of *no* document scores.

Finally, logistic regression modeling is used to estimate the weights of the above normalized scores and the final score will be obtained after this score combination process.

2.1.2 CMU's Tracking System

Carnegie Mellon University (CMU)'s tracking system [36] [76] used decision trees (DT) and the k-Nearest Neighbor (kNN) approach in TDT-2 tracking task. In TDT-3, they tested the Rocchio approach and language modeling (LM) approach for topic tracking. Moreover, they combined the output of the previous models to form the resulting system, namely Best Overall Results Generator (BORG).

In the DT approach, the system uses the words feature and word co-occurrence statistics to formulate the classifier. In the kNN approach, the system uses the $tf \cdot idf$ document representation and selects the k nearest neighbors based on the cosine similarities between the input story and the training stories. In the Rocchio approach, the system uses a vector to present each class and a document, computed their similarity using the cosine value of these two vectors, and obtains a binary decision. In the LM approach, the system is based on the BBN Topic Spotting which we described in the previous section. In the BORG approach, the system combines the normalized (as same as BBN' s tracking system used) scores from kNN, Rocchio and LM approaches, re-normalized the combined score in the same way, and finally obtained the final score as well as binary decision.

2.1.3 Dragon's Tracking System

The Dragon's tracking system [43] [66] [67] [44] consists of two statistical models – the topic model built from the topic training stories, and the discriminator models built from available background material. The system is based on the following classifier [43]:

- Score an incoming story against a topic model built from the topic training stories.
- Score the story against a discriminator model built from the background data.
- Output the difference between these scores as a relevance value, and threshold this difference to generate a decision.

This means a document is considered relevant to a topic if it resembles the topic model (training stories) more than the discriminator model (background material). The topic and discriminator models are based on two statistical approaches – they are the unigram language model and the beta-binomial model. Moreover, linear interpolation is implemented to combine the scores generated from the topic models and discriminator models to improve the tracking effectiveness.

2.1.4 UPenn's Tracking System

The University of Pennsylvania (UPenn) developed a tracking system [40] [41] based on the vector-space approach. The model they used is a well-known idf-weighted cosine coefficient similarity metric.

This system used a $tf \cdot idf$ representation for classifiers and document, where the *idf* values are collected from the document frequencies in the TDT1 corpus. Surprisingly, the UPenn's system used the simple method of feature selection – without word stemming and without a score normalization, but achieve the competitive results with other tracking systems.

2.2 Topic Tracking in Chinese

Topic tracking in Chinese can be divided into two major parts. One is the feature extraction from training stories. The other is the classification between the training stories and the new subsequent stories. In order to tackle these problems, we should consider the linguistic characteristics of the Chinese language.

Unlike English text in which words are separated by spaces, Chinese text does not contain any explicit word boundaries. Hence, Chinese text needs to be segmented to form a sequence of words. For a given string of characters, there may exist multiple legitimate segmentations. Different segmentations lead to different word sequences and hence different meanings. For example, the corresponding character string "香港的士多" may be segmented into two legitimate word sequences:

香港 的士 多 (many taxis in Hong Kong)

香港 的 士多 (stores in Hong Kong)

Apart from the ambiguity caused by multiple segmentations, a given word may have multiple possible POS assignments (or meanings). For example, consider 白馬 and 馬步芳, where 馬 is a common noun (horse) in the former and a last name of person in the latter.

The OOV problem is another obstacle in a Chinese sentence. For western languages, POS tagging for the unknown words relies heavily on the morphological rules and related to word affixes [11]. For example, the word ends with "ing" is always tagged as verb. But the lexical structure of the Chinese word is very different compared to English. Inflectional forms are minimal, while morphology and word derivations abide to a different set of rules. A word may inherit the syntax and semantics of (some of) its compositional characters, for example, 紅 means red (a noun or an adjective), 色 means color (a noun), and 紅色 together means the color red (a noun) or simply red (an adjective). Alternatively, a word may take on totally different characteristics of its own, e.g. 東 means east (a noun or an adjective), 西 means west (a noun or an adjective), and 東西 together means thing (a noun). Yet another case is that the compositional characters of a word do not form independent lexical entries in isolation, e.g. the characters in 彷彿 (a verb) do not occur individually.

The above considerations lead to a number of questions related to feature extraction and story classification for topic tracking in Chinese language. In feature extraction, we will describe some existing Chinese word POS tagging approaches and unknown word identification in Section 2.3 and 2.4. In story classification, we will describe the information retrieval models in Section 2.5.

2.3 Part-of-Speech Tagging

POS tagging is an important linguistic problem which has garnered much research interest and effort over the years. The problem of POS tagging is assigning the appropriate syntactic category (POS tags[1]) to the word in a sentence. Figure 2.1 shows an example of POS tagging task.

To disambiguate the multi-tag words (會,這樣 and 的 in Figure 2.1) correctly is the major issue of POS tagging. In this section, we will give a brief introduction of tagging algorithms. In addition, we will present Brill's Transformation-based Error-Driven Learning approach in detail.

2.3.1 A Brief Overview of POS Tagging

A myriad of techniques have previously been used for automatic POS tagging, ranging from rule-based [8] [13] to data-driven approaches. The former tends to be hand-annotated by linguistic experts, while the latter tends to be relied on statistical information such as word tag co-occurrence frequencies in the training data.

[1] The tag set is provided in Appendix A.

pre-segmented sentence

你	會	聽到	這樣	的	話	，
(you)	(will)	(hear)	(this)		(saying)	，

⇓

finding the possible POS tags

你	會	聽到	這樣	的	話	，
rn	va	vgn	rp	usde	ng	，
		vc		rn	y	，
		ng				

⇓

selecting the correct POS tags

你	會	聽到	這樣	的	話	，
rn	va	vgn	rp	usde	ng	，

Figure 2.1: The Steps of POS Tagging.

Rule-based approaches are linguistically well-motivated, but expert handcrafting is often an expensive and tedious process. It requires human efforts to build and update the enormous size of rules. However, rule-based approach can provide general knowledge that can be reused in a different area without much modification. For example, the word ends with "ed" are always tagged as verb.

Typical rule-based approaches use lexical (or morphological) and contextual information to assign POS tags to unknown and ambiguous words. Here is an example of contextual rule used in Chinese POS tagging:

> If an word is followed by 西元 (AC), tag it as an time noun. [1]

Data-driven approaches attempt to ameliorate the tedium of rule writing by capturing relevant linguistic constraints from a corpus of annotated data. However, the linguistic constraints captured are encoded in a large body probabilities and statistics, which do not lend themselves well for exploratory linguistic analysis.

The simplest data-driven approach disambiguate words based solely on the probability that a word occurs with a particular tag. That means the tag encountered most frequently in the training set is the one assigned to an ambiguous instance of that word. Recently, more complex models were developed to utilize the statistical information automatically trained from the corpus. It includes stochastic n-grams [6], Hidden Markov Model [49], neural networks [33], trigger-pair predictions [16], genetic algorithms [56], Maximum Entropy [5], and the relaxation labeling method [70].

Brill [22] had previously proposed an alternative technique of transformation-based error-driven learning (TEL) for automatic POS tagging in English. This approach combines the merits of rule-based and data-driven techniques in an elegant manner. The algorithm may be initialized randomly or with some linguistically-motivated specifications. Machine learning then proceeds with an annotated corpus, and with the objective of maximizing tagging accuracies. Learning produces a compact rule set, which encodes the contextual and lexical constraints for tagging, and the rules easily interpretable by humans for studying the linguistic cues for POS tagging. We will present the TEL approach in the following.

2.3.2 Transformation-based Error-Driven Learning

The Transformation-based Error-Driven Learning (TEL) approach is a machine learning technique. It has been successfully applied to a number of natural language problems, including POS tagging [17] [21] [22] [23], prepositional phrase attachment disambiguation [24], syntactic parsing [18] [20] [69], spelling correction [55], Chinese word segmentation [39], and Chinese noun phrase extraction [75].

Figure 2.2 [19] is the flow chart explaining transformation-based error-driven learning process. First, the unannotated text is passed through an initial state annotator to give preliminary output to the text. This annotation is compared with the truth (or a reference). Rules for error correction are proposed according to the templates. The proposed rule which maximally reduces the number of errors is adopted in the ordered transformational rule set. The adopted transformation is then applied to the annotated corpus again until no proposed rules can reduce the minimum count of errors. Figure 2.3 [22] shows an example of learning transformational rules.

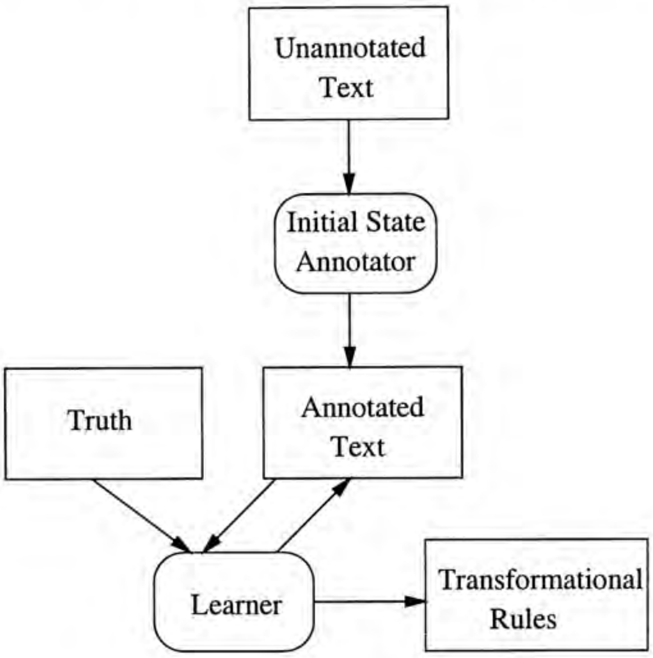


Figure 2.2: Learning Process of Transformation-Based Error-Driven Learning (TEL).

In Figure 2.3, the unannotated corpus is passed through an initial state annotator, and 7,000 errors are found by comparing the truth. All possible transformations are then tested and found that the transformation *T1* results in the largest reduction of errors (3,500 errors are found), and so *T1* is adopted in the transformational rule set. The adopted transformation *T1* is then applied to the annotated corpus again, and learning continues. In this time, *T3* results in the largest reduction of errors (2,000 errors are found), and so *T3* is adopted in the transformational rule sets. The adopted transformation *T3* is then applied to the annotated corpus again, but no proposed rules can reduce the errors in this time, and so the learning stops. After the

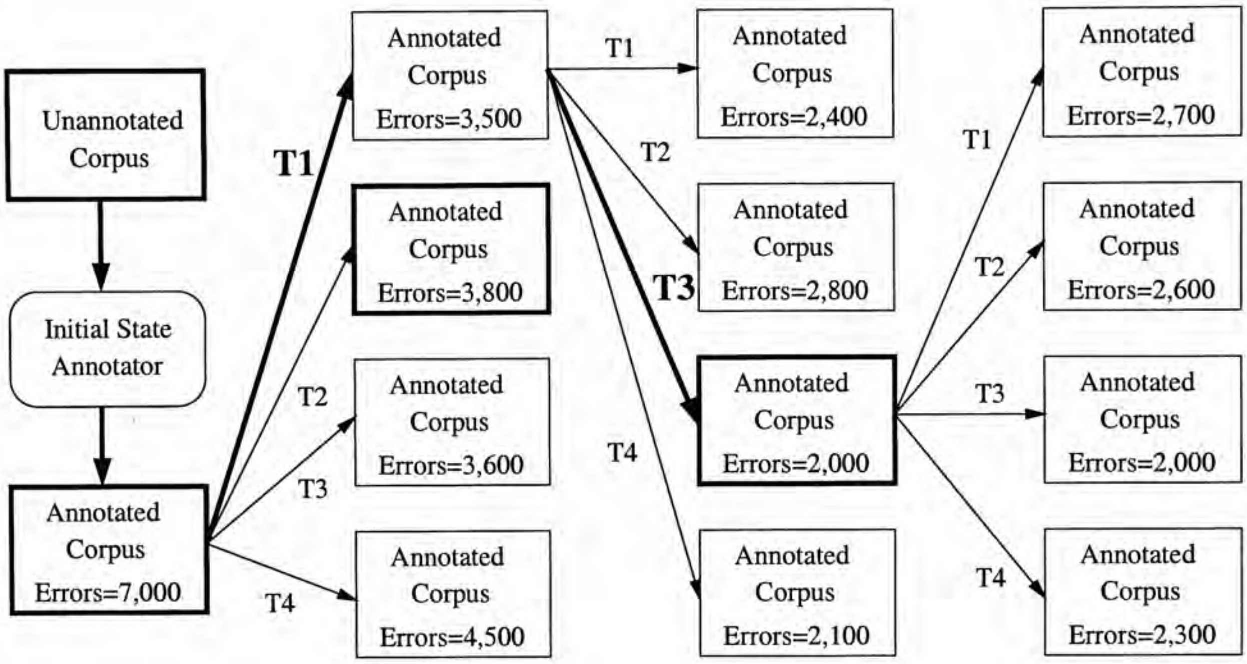

Unannotated Corpus
Initial State Annotator
Annotated Corpus Errors=7,000
T1
T2
T3
T4
Annotated Corpus Errors=3,500
Annotated Corpus Errors=3,800
Annotated Corpus Errors=3,600
Annotated Corpus Errors=4,500
T1
T2
T3
T4
Annotated Corpus Errors=2,400
Annotated Corpus Errors=2,800
Annotated Corpus Errors=2,000
Annotated Corpus Errors=2,100
T1
T2
T3
T4
Annotated Corpus Errors=2,700
Annotated Corpus Errors=2,600
Annotated Corpus Errors=2,000
Annotated Corpus Errors=2,300

Figure 2.3: An example of learning transformational rules.

learning process, we obtain the ordered transformational rules *T1*, *T3*.

To conclude, TEL provides several attractions in POS tagging. One is the automation for tagging. Two is the induction of interpretable rules in a concise and easy-to-understand form. Three is learning aimed at error-reduction. TEL was previously applied successfully on English by Eric Brill [22]. In this thesis, we try to apply it to Chinese text and we will discuss the details in Chapter 4.

2.4 Unknown Word Identification

Unknown words are defined as the words which are not in the lexicon (OOV). For example, inspection of the Academia Sinica Balance Corpus [1] shows a 3.51% of OOV on usage. It motivates us to develop a system to handle OOV since our approach of topic tracking is based on the extraction of named entities, and these named entities are mostly unknown words in Chinese text.

The task of unknown word identification can be divided into two steps [48]. The first step is to detect the existence of unknown words. Most of the unknown words are incorrectly segmented into sequences of single character word or shorter words. The second step is to determine the boundaries of unknown words. Figure 2.4 shows an example of unknown word identification.

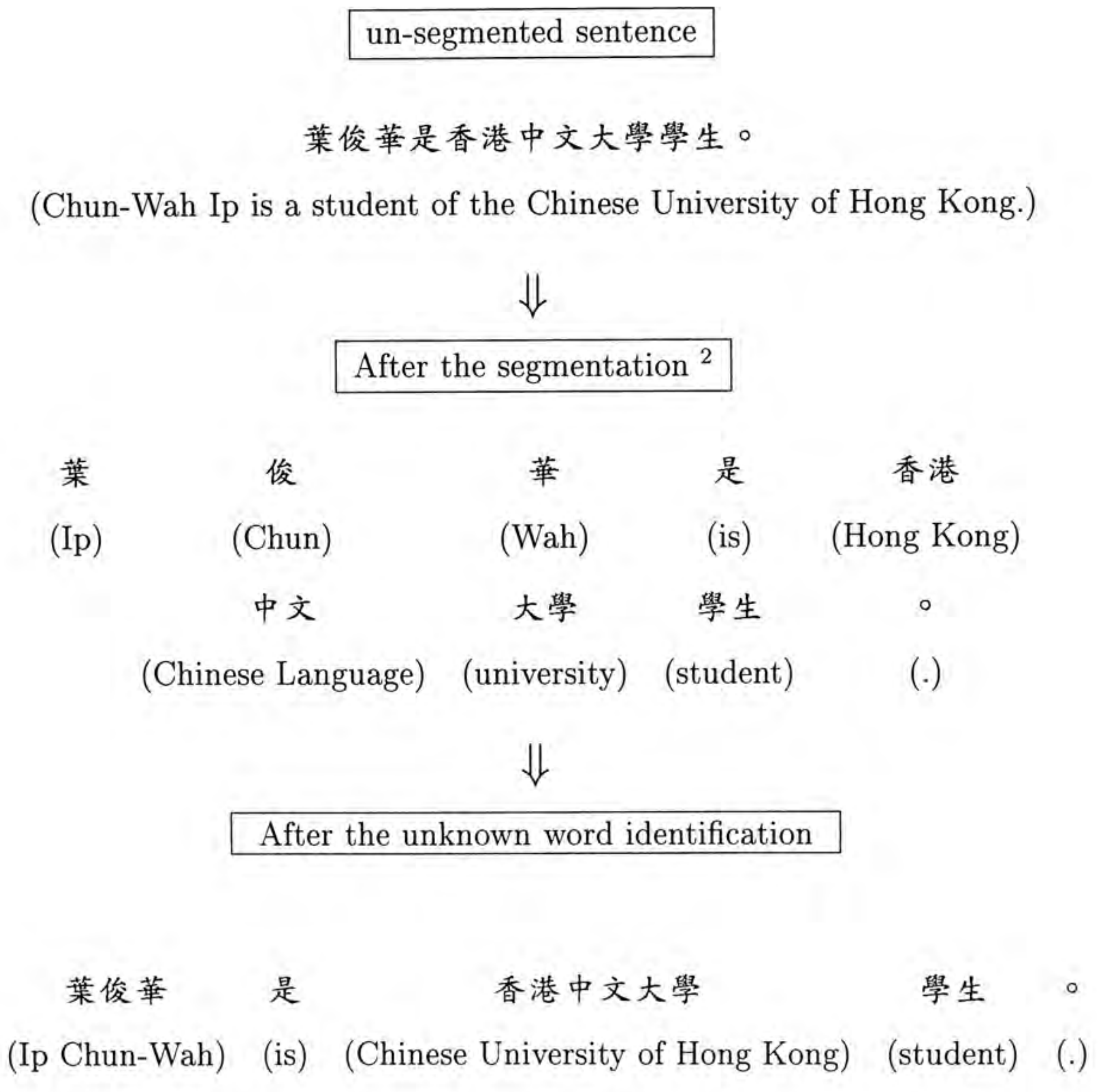
In what follows, we discuss the major approaches to unknown word identification. There have been three main groups of approaches in this task, they are rule-based approaches, statistical approaches and hybrid approaches.

2.4.1 Rule-based approaches

Rule-based approaches use morphological rules and contextual information to identify the unknown words. However, different kinds of unknown words have its different morphological structure, it is difficult to write simple rules to identify all kinds of unknown words. Therefore, different sets of rules has been developed by researchers to work on different type of problems.

The heuristic knowledge about word formation patterns plays an important role in rule-based approaches. Chang [54] used the titles and surname-driven rule to recognize unregistered names. Sun [2] claimed that around 90% of the family names are covered by 114 characters. Chen and Lee [30]

[2] This sentence is segmented by a segmenter provided by LDC, we will discuss this segmenter in chapter 3.



un-segmented sentence

葉俊華是香港中文大學學生。

(Chun-Wah Ip is a student of the Chinese University of Hong Kong.)

⇓

After the segmentation [2]

葉	俊	華	是	香港
(Ip)	(Chun)	(Wah)	(is)	(Hong Kong)
	中文	大學	學生	。
	(Chinese Language)	(university)	(student)	(.)

⇓

After the unknown word identification

葉俊華	是	香港中文大學	學生	。
(Ip Chun-Wah)	(is)	(Chinese University of Hong Kong)	(student)	(.)

Figure 2.4: The Steps of unknown words identification

used morphological rules and contextual information to identify the names of organizations. Moreover, Jin [74] used the duplication of characters to identify Iterative and fixed expressions. For example, "常常" (always) and "研究研究" (investigate).

However, rule-based approaches face difficulty in identifying the words with irregular morphological structures such as abbreviation[3]. Moreover, it is impossible for a human to write a complete set of rules. Therefore, statistical approaches are proposed to tackle the unknown words problem.

2.4.2 Statistical approaches

With the help of large annotated corpus, statistical approaches [71] [46] [65] are developed in handling OOV identification. This approach is based on statistical information such as word and character (co-)occurrence frequencies in the training data. The most commonly used algorithm is the n-gram [58] [47] [15] (especially for 2-gram or called bi-gram) technique. The principle of n-gram technique is to divide a sentence into many overlapping n-grams slices, where N is the number of characters in each slices. For example, "香港是你家" (Hong Kong is your family) can be divided into:

香港 港是 是你 你家, where N=2 (bigram)

香港是 港是你 是你家, where N=3 (trigram)

香港是你 港是你家, where N=4 (4-gram)

香港是你家, where N=5 (5-gram)

[3] e.g. 中電 (China-Electric)

Too many n-grams may be formed in the previous process and many of them are not legitimate words. Therefore, different schemes for selecting useful n-grams are proposed. For example, Nie et al. [61] used the word formation power (WFP) to find a set of function characters (e.g.的 and 在). Any n-grams involving these function characters will be eliminated. Shen and Sun [15] used the local frequency of n-gram (e.g. occurrence frequency of n-grams within 100 sentence) to recognize n-grams as a word.

Besides the n-gram technique, some statistical approaches are proposed in detecting and identifying unknown words. For example, the first-order Markov model [4], mutual information [46] and dice metrics [11].

Basically, all the above methods are relied heavily on the statistical information in the training data. Hence the statistical approach is sensitive to the application area, which reduces its reusability in other applications.

2.4.3 Hybrid approaches

Hybrid approaches [1] [60] [65] [47] combine the advantages of rule-based and statistical techniques. Rule-based approaches captures obvious regularies, e.g. the grouping of cardinal numbers (二, 十, 一) together naturally forms a larger number (二十一). Statistical approaches can automatically capture characteristic regularities from application-specific data.

Implementations of the hybrid approach vary from one system to another. In the most cases, the rule-based approach is considered as the background knowledge, and the statistical information is considered as the foreground knowledge. For example, Lin et al. [60] used 17 morphological rules to recognize regular components and a statistical model to handle the irregular unknown words. Chan et al. [1] applied the TEL approach to find the rules with statistical information. However, these rules are focused on identifying the tag of unknown words.

The hybrid approach seems more flexible and reliable than the rule-based and statistical approaches since it combines the merits of rule-based and statistical approaches. Therefore, we will apply the hybrid approach to handle the OOV problem in Chinese topic tracking. We will give a detail description of our approach in Chapter 5.

2.5 Information Retrieval Models

We are interested in information retrieval (IR) models because we will be adapting an IR model for the task of topic tracking. Information retrieval models are one of the most important issues in retrieving relevant documents from collection. The retrieving process begins with a query provided by the user. (Queries may be words, sentences or sample documents). The features (i.e. words, frequency of words, etc.) of the query is then transformed into a model representation. The presence or absence of query terms in the documents will be used to determine the degree of relevance of a document with respect to the query.

Different IR systems have been developed over the last two decades. In the following, we will describe two commonly used IR approaches (i) the vector-space model and (ii) the probabilistic model.

2.5.1 Vector-Space Model

The vector-space model is a well-known approach for IR. It was first proposed by Salton [28]. The vector-space model transforms the description of information (words or phrase) of queries and documents into vectors. A score is calculated using these vectors to determine the degree of relevance of a document with respect to a query. Here we will give the brief descriptions of this transformation and similarity calculation.

Assume there are n terms in a document d, the vector-space model will transform d into a vector of the form:

$$d = (w_{d1}, w_{d2}, \dots , w_{dn})$$

where w_{di} denotes the weight of term t_i in a document d.

A query q is transformed in a similar way as well:

$$q = (w_{q1}, w_{q2}, \dots , w_{qk})$$

where there are k terms in a query q and w_{qi} denotes the weight of term t_i in a query q.

When both query q and document d are represented by their corresponding vectors, the similarity (degree of relevance of the document to the query) between a query and a document is calculated by the inner product of their corresponding term weights (i.e. w_{di} and w_{qi}). This equation is shown as follows:

$$Sim(d, q) = \sum_{i=1}^{n} (w_{di} \times w_{qi}) \tag{2.2}$$

The vector-space model is widely applied in IR system. For example, InQuery retrieval system [64], PRISE [14] and the SMART system[29]. The main difference between these systems is their term weighting schemes. The variety of weighting schemes are calculated using: (i) term frequencies (e.g. frequency of term i in document d), (ii) collection frequencies (i.e. number of documents with term i in the collection) and (iii) a normalization procedure. The term frequency assigns higher weights to which occurs more frequently in a document. The collection frequency assigns lower weight to a term if it occurs frequently in a variety of documents. The normalization step normalizes the documents in the collection with varying vector length into equal lengths.

In topic tracking, the training stories can be represented by document vector while each testing story can be represented by query vector. In Chapter 6, we will give a detail description of applying vector-space model to topic tracking.

2.5.2 Probabilistic Model

The probabilistic model [72] [68] [63] is based on probabilistic information such as the the Bayes' Theorem. The probabilistic model calculates the probability of relevance between documents and queries. However, different systems apply different kind of probabilities calculation. For example, Elike et al. [25] used the transition probabilities in Hidden Markov Models (HMMs), Berthier et al. [7] used the conditional probabilities in Bayesian Belief Networks.

In our task of topic tracking, we only have four training stories to characterize the topic which we are trying to track. Hence the training data available tends to be sparse. This may affect probability estimation if we adopt the probabilistic approach. Hence in this thesis, we selected the vector-space approach instead of probabilistic approach.

2.6 Chapter Summary

In this chapter, we briefly described the previous developments in the task of topic tracking. We provide a sketch of 4 leading system from the TDT-2 and TDT-3 evaluations.

Topic tracking in Chinese is a challenging research area since Chinese text does not contain any explicit word boundaries. Our approach for topic tracking in Chinese is depend on named entity extraction. Two techniques for named entity extraction are discussed in this chapter: (i) the part-of-speech (POS) tagging and (ii) the unknown word (OOV) identification.

In POS tagging, we applied Brill's transformation-based error-driven learning (TEL) approach to Chinese. In OOV identification, we discuss three major approaches: (i) rule-based, (ii) statistical and (iii) hybrid.

The use of named entities to topic tracking is a crucial problem. Information retrieval (IR) models are proposed to tackle this problem. Two IR models are discussed in this chapter: (i) the vector-space model and (ii) the probabilistic model. Finally, we applied vector-space model to topic tracking.

Chapter 3

System Overview

In this chapter, we describe the system we have developed for topic tracking in Chinese. This system consists of four components: (i) the Chinese word segmenter, (ii) the transformation-based (TEL) tagger, (iii) the unknown words identifier and (iv) the topic tracker.

The process for topic tracking is depicted in Figure 3.1. The training stories are first passed to a Chinese segmenter to perform word tokenization. Then, an TEL tagger assigns the tag to each of segmented words. The words with selected tags[1] are passed to a topic tracker. At the same time, the tagged sentences will be passed to the unknown word identifier. Name entities found by unknown word identifier will be passed to topic tracker. Finally, a topic tracker will formulate a classifier and generate a feature table to store the information of features. This feature table is used by a classifier to compute the score of incoming stories and make the binary decision (relevant to training stories or non-relevant to training story) for each story.

[1] We will discuss this in later chapters.

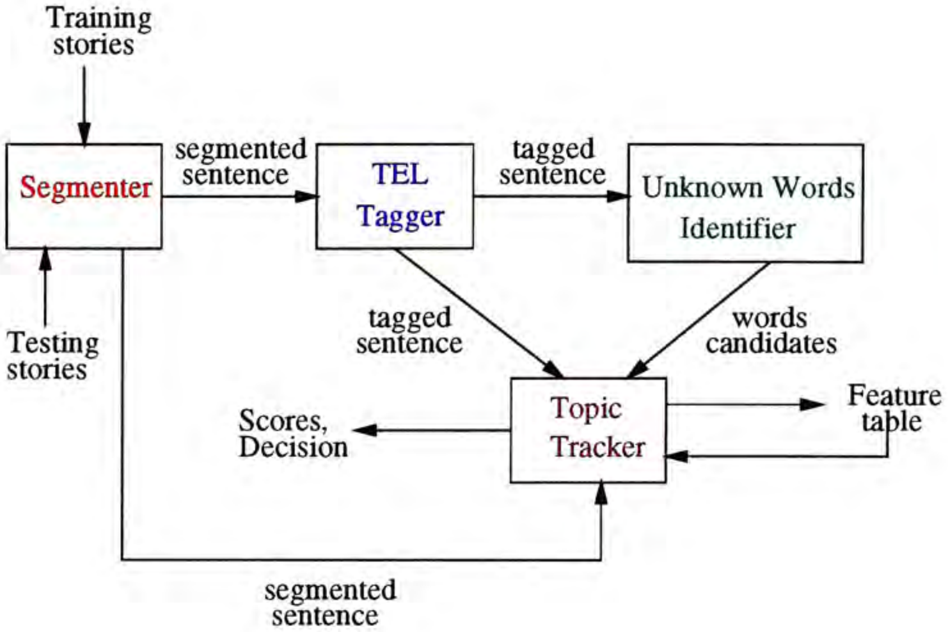


Figure 3.1: The Topic Tracking System.

In the following, we give a brief overview of the four sub-systems in Figure 3.1.

3.1 Segmenter

Our Chinese segmenter is provided by Linguistic Data Consortium (LDC).[2] This segmenter consists of a lexicon file in GB code. There are 44,405 entries in the lexicon. Each entry has three data members: (i) the word, (ii) the occurrence frequency of a word, and (iii) pronunciation of a word.

A dynamic programming approach is used in this segmenter. This approach finds the word segmentation path which has the highest multiple of word probabilities.

[2] http://morph.ldc.upenn.edu/Projects/Chinese

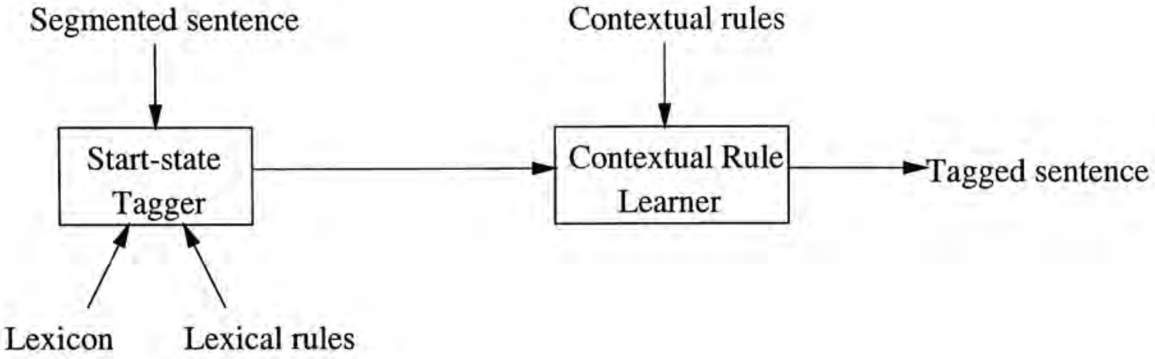


Figure 3.2: Components of the TEL tagger.

3.2 TEL Tagger

In the feature extraction process, we applied a Chinese POS tagging technique to extract named entities. This tagger is implemented by the transformation-based error-driven learning (TEL) approach which was described in Chapter 2.

The TEL tagger consists of two components: start-state tagger and final-state tagger. Three data files (lexicon, lexical rules file and contextual rules file) are used in this tagger. Figure 3.2 illustrates the process of tagging. First, the start-state tagger reads a tokenized word, then assign the most possible tags from a reference lexicon to each of the words. Second, the start-state tagger applies the lexical rules to tag the unknown words. Finally, the final-state tagger applies the contextual rules to tag the words.

A tagging lexicon contains 11,908 entries.[3] Each entry contains a word and its possible tags.[4] In two rules files, lexical rules file contains 138 lexical

[3] Some modifications has been made. We will discuss them in Chapter 6.

[4] The tag set is provided in Appendix A. We will discuss the details of the tags in Chapter 4.

rules and contextual rules file contains 246 contextual rules. All of these rules are learnt during training. We will give the detail description and explanation in Chapter 4.

3.3 Unknown Words Identifier

Word segmentation may be hampered by the occurrence of OOV. To tackle this problem, we developed an unknown word identifier which applied n-gram techniques together with heuristics to save the named entities "missed" during the segmentation process.

An unknown word identifier consists of two components: n-gram grouper and entity formulater. Besides these components, six data files are used in this identifier: (i) a word set of person name, (ii) a word set of time units, (iii) a word set of organization suffixes, (iv) a word set of location suffixes, (v) a word sets of stopwords, and (vi) a formulation rules file.

Figure 3.3 illustrates the processes of detecting and identifying OOV. First, a tagged sentence is passed to the n-gram grouper to extract the people names. The word set of stopwords helps the n-gram grouper to detect the boundaries of unknown words, and the word set of person name helps the n-gram grouper to determine person names. Second, the n-gram candidates (guessed person names) and tagged sentence will be passed to the entity formulator to extract organization names, location names and time by the formulation rules.

A brief description of each word set and the detail explanation of formulation rules will be discussed in Chapter 5.

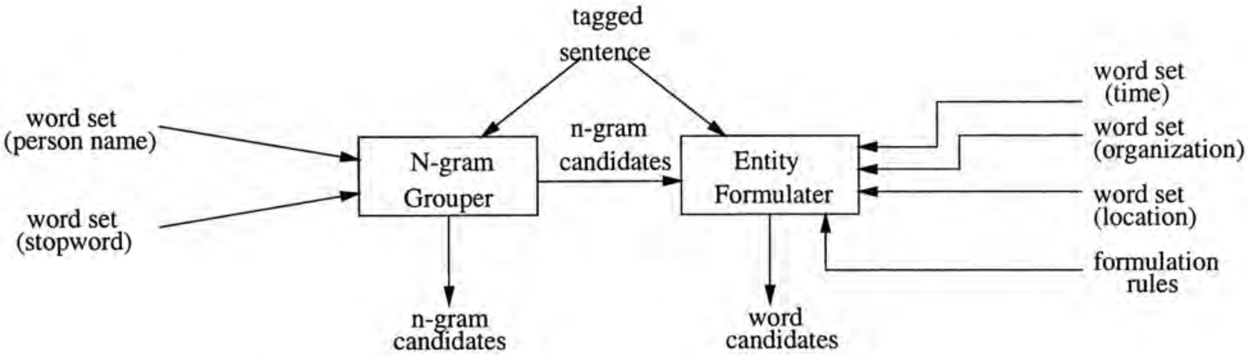


Figure 3.3: Components of unknown words identifier.

3.4 Topic Tracker

There are two tasks performed by the topic tracker: (i) using the training stories and named entities to formulate classifier for topic tracking, (ii) identify whether a testing story is on-topic (relevant to training topic) or off-topic (non-relevant to training topic).

An topic tracker consists of two components: (i) the trainer and (ii) the tracker. The trainer is used to generate a feature table which contains feature information representing the training stories. The tracker uses the feature table to compute the similarity between a testing story and a topic, and finally gives the binary decision (on-topic or off-topic) for each testing story. Furthermore, there are four data files used in topic tracker: (i) a stopword list, (ii) a search class file, (iii) an idf database which contains document frequency of the word in collection[5] and (iv) feature table where the features consist of keywords together with their weights.

Figure 3.4 depicted the whole tracking process. First, the tagged words

[5] There are 1,000 off-topic documents in the collection. We will discuss these collection in Chapter 6.

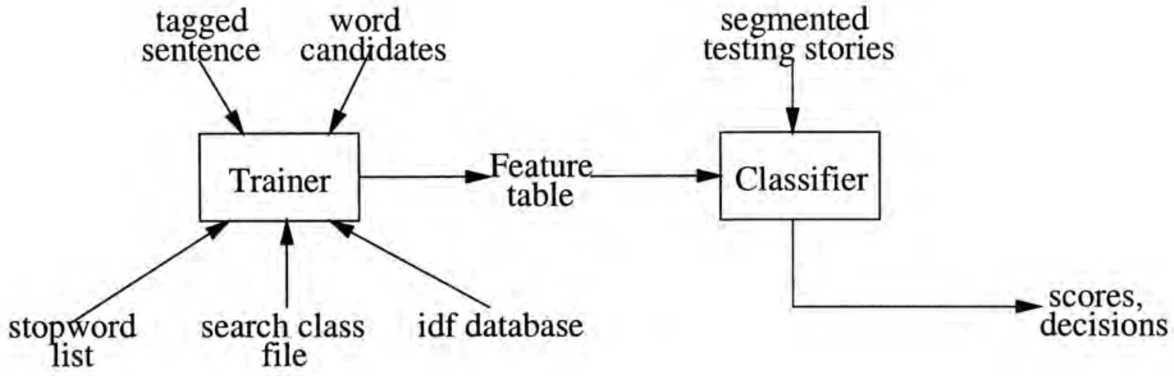


Figure 3.4: Components of topic tracker.

(tagged by TEL tagger) and word candidates (found by unknown words identifier) are passed to a trainer. Second, the trainer removes all the words which exist in stopword list. Third, the trainer extracts the named entities which are in search class file. Then, the trainer assigns the weight (with help of idf database) to each features and record it in feature table. Next, a classifier reads the feature table and then the testing stories are passed to a classifier. Finally, the classifier uses the information of feature table to output the decisions and scores for each testing story, regarding whether it is on-topic or off-topic.

The detail descriptions and explanations of topic tracker will be discussed in Chapter 6.

3.5 Chapter Summary

In this chapter, we briefly described tracking system in Chinese. This system consists of four sub-systems: (i) A Chinese segmenter is used to perform tokenization. (ii) A TEL tagger is used to extract named entities. (iii) An unknown words identifier is used to capture the named entities “missed” during the segmentation process. (iv) A topic tracker combines the named entities into a feature vector, and use it to make a binary decision (i.e. on-topic or off-topic) for tracking.

Aside from the Chinese segmenter, we have implemented all the subsystems, using different approaches. In the remainder of this thesis, we will present these approaches in detail.

Chapter 4

Named Entity Tagging

In this chapter, we applied POS techniques for entity tagging. It is our initial attempt in using the transformation-based error-driven learning (TEL) procedure for tagging Chinese text [59]. The goal of this task is to develop an automatic part of speech (POS) tagger to provide named entities for topic tracking.

We use the transformation-based error-driven learning (TEL) for POS tagging (or transformational tagging) of Chinese text. We are especially interested in the linguistic rules induced automatically by TEL for individual Chinese words, as well as across a sequence of multiple words. Chinese linguistic structures may be observed in such rules, including grammar, morphology and word derivations. TEL is applicable not only to in-vocabulary words, it is also designed to handle the occurrences of unknown words in corpora.

4.1 Experimental Data

This work is based on the pre-segmented and hand-tagged corpus from Tsinghua University [6]. This news corpus is from the People's Daily (Renmin Ribao) in the year 1993. Altogether there are 112 articles and 71,804 words of running text, distributed across five domains: computer, military, science, technology and general news. Unique vocabulary entries exceed 9,000. Information about the entire corpus is tabulated in Table 4.1, and the word count in the table refers to the length of running text. Table 4.2 displays some example sentences from each domain, which shows the word/tag pairs for each sentence. In this work, we only tackle the tagging problem – our tagger learns from pre-segmented and tagged training sets, and tests on a pre-segmented test sets.

Domain	**No. of Article**	**No. of Words (train)**	**No. of Words (test)**
Computing	10	5,479	509
Military	23	12,243	1,787
News	39	22,358	2,505
Science	20	12,922	1,391
Technology	20	11,383	1,228

Table 4.1: Distribution of Training and Testing Sets from the Tsinghua news corpus.

The original tag set found in the Tsinghua corpus consists of 108 unique labels. These were exhaustively enumerated in [56]. Out of this set, 25 are for punctuation, and the remaining ones draw fine distinctions for Chinese parts of speech. As an example, nouns are divided into 5 types: nf (last

name), npf (name of person), npu (name of organization), npr (other proper nouns) and ng (common noun). We added an extra tag, nvg, to represent words which can either be a noun or a verb, such as 運動 (exercise) or 表示 (express/expression). The reason is as follows: in the original tagged corpus, there are words like 運動 which are tagged as general verbs, e.g.
雖然/cf 經歷/vgn 了/utl 各/rn 種/qnk 運動/vg 變化/vg ，/，
where the tags are: cf (連詞前段), vgn (帶體賓動詞), utl (連詞 "了"), rn (體詞性代詞), qnk (種類量詞) and vg (一般動詞).
In this context, however, 運動 seems to play a role more similar to a noun, which motivated the design of the nvg [1] because 運動 in this case is not suitable to tag as verb.

One may wonder whether the full tag set is necessary for Chinese POS tagging.[2] A preliminary investigation of our entire corpus reveals that approximately 100 tags occurred, with the most frequent one being ng (common noun), which occurred about 25.5% of the time. The most frequent 18 tags (which include a few punctuation tags) covers 80% of our running text corpus, while the most frequent 32 tags already covers 90%. Nevertheless, we proceeded with the full set of 109.

The ambiguities found in the Tsinghua corpus is 1.88 tags per word (Please see Figure 4.1). Over 40% of the vocabulary can be tagged multiple ways. Out of this, the maximum number of tags per a word is 8. Table 4.3 lists the 8 POS tags of the word (表示) and their contexts.

[1] The idea of using nvg tags is attributed to Dr. Wenjie Li.

[2] We have found tag sets of approximately 50 entries of fewer in other literature.

Domain	Example Sentences
Computing	我們/rn 根據/p 這/rn 一/mx 漢化/vg 策略/ng 對/p DECnet-DOS/xch 進行/vgv 了/utl 分析/vgo 。/。
Military	反/vgn 機降/ng 成爲/vgn 戰鬥/vgo 的/usde 重要/a 内容/ng 。/。
Science	17世紀/t 英國/s 的/usde 醫學家/ng 哈維/npf ，/ ，
Technology	工作/ng 模式/ng 是/vy 當前/t 科技/ng 情報/ng 體制/ng 改革/nvg 中/f 的/usde 又/d 一/mx 熱點/ng 。/。
News	共同體/ng 將/va 参與/vgn 德國/s 統一/vg 問題/ng 的/usde 討論/nvg ，/ ，

Table 4.2: Example sentences from our corpus.

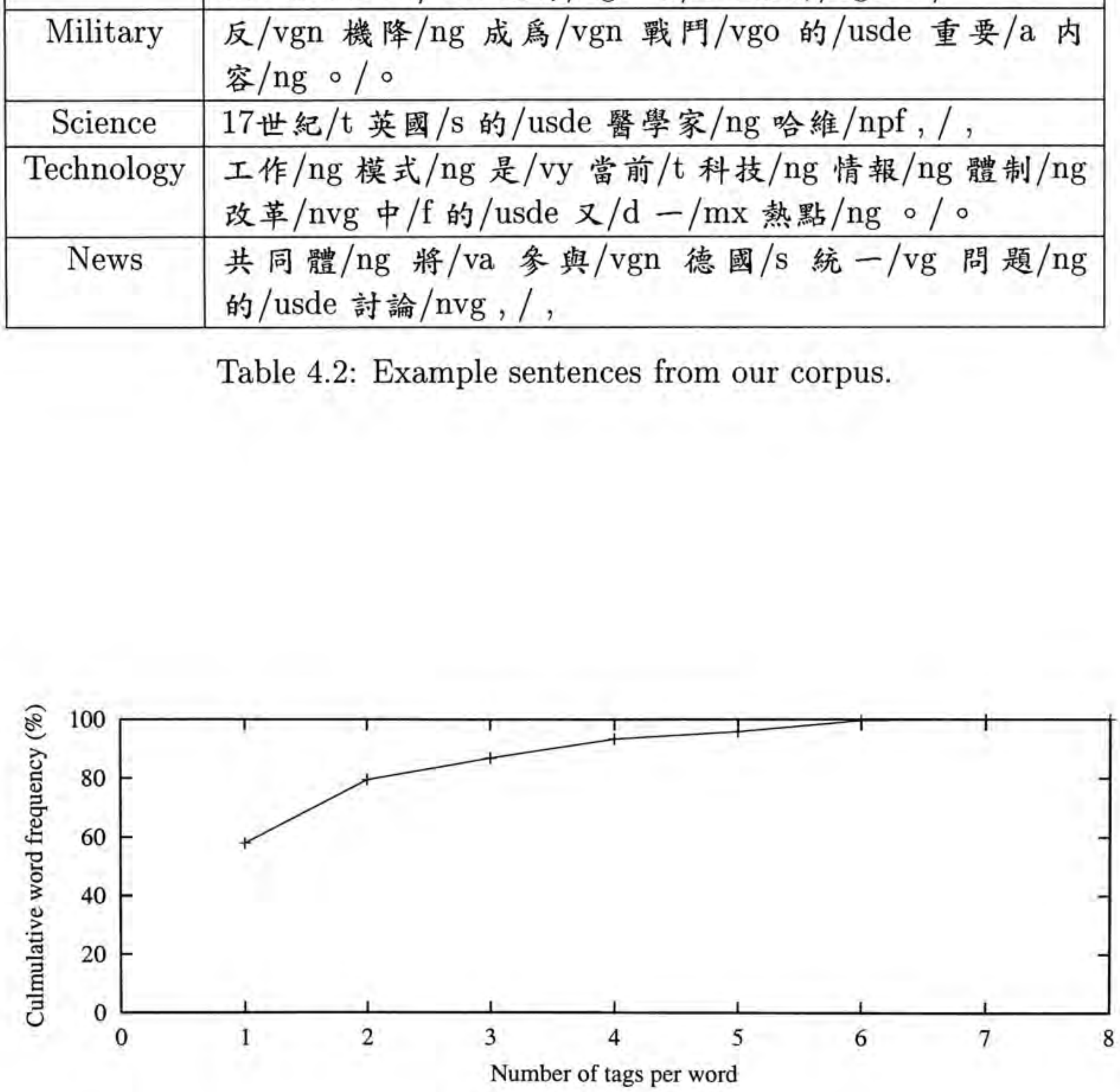


Figure 4.1: Cumulative distribution of words with single to multiple POS tags.

Tag	Example Sentences
vg (一般動詞)	這/rn 種/qnk 表示/vg 法/ng 與/p ＪＡＥ/xch 類似/a 。/。
vgo (不帶賓動詞)	文獻/ng 和/cpw 查詢/nvg 都/d 用/vgn 一/mx 組/qnc 正交基/ng 詞/ng 向量/ng 表示/vgo ，/，
vgn (帶體賓動詞)	我們/rn 采用/vgn Ｓｅｔ０/xch 表示/vgn 單/b 字節/ng 的/usde ＡＳＣＩＩ/xch 字符/ng ，/，
vgv (帶動賓動詞)	ＤＧ/xch 人士/ng 表示/vgv 將/d 爲/p 此/rn 繼續/vgv 做/vgv 出/vc 努力/vgo 。/。
vga (帶形賓動詞)	”/” 是/vy ”/” 可以/va 表示/vga 一樣/a 。/。
vgs (帶小句賓動詞)	無非/d 表示/vgs ：/：
ng (普通名詞)	情報/ng 檢索/vg 系統/ng 所/ng 儲存/ng 的/usde 是/vy 文獻/ng 的/usde 某/rn 種/qnk 表示/ng ，/，
nvg (動名詞)	一/mx 篇/qni 文獻/ng 的/usde 表示/nvg 中/f 所/ussu 使用/vgn 的/usde 標引詞/ng 的/usde 個數/ng ...

Table 4.3: Example sentences of the word ”表示” from our corpus.

4.2 Transformational Tagging

The transformation-based error-Driven Learning (TEL) approach is presented in detail in chapter 2. The tagger addresses its problem at both the lexical and contextual levels. Learning takes place in two phases. Lexical rules are learnt first, and are used during the subsequent learning of contextual rules. Here we will provide a procedural sketch.

4.2.1 Notations

For the sake of simplicity, we will adopt the following notations in describing our work:

- C^d_{type} , denotes a corpus C belonging to a specific domain d, and of a particular type - *training, testing, lexical* or *contextual*. The type is related to the transformational tagging procedure, and will be explained later.

- $T_i(C^d_{type})$, denotes a *tagged* corpus C. The variable i may adopt the instances *ref* (for the set of reference tags), *start* (for the tags resulting from the initialization of the tagger) or *final* (for the tags resulting from the final stage of the tagger, having applied all tagging rules). Details will be explained later. An example of a tagged sentence is:
  17世紀/t 英國/s 的/usde 醫學家/ng 哈維/npf , / ,

- $U(C^d_{type})$, denotes a *untagged* corpus C. A procedure may be applied to strip off all the tags, resulting in 17世紀 英國 的 醫生家 哈維, from the previous example.

- R^d_{type} , denotes a set of rules R. Rules may be of the type *lex* (lexical rules) or *context* (contextual rules). Example rules include:[3]

  Lexical rule : 了 goodleft vgn 135.820116353036

  Contextual rule : vgn vgo NEXT1OR2TAG STAART

- L_{type} , denotes a lexicon, which may be of type *lex* (the lexicon for training lexical rules only), or *all* (the lexicon containing all words in the training corpus).

4.2.2 Corpus Utilization

Figure 4.2 shows how the corpus is utilized. The entire corpus is first divided into a training set (90% of the size) and a test set (10%). The training set is in turn divided into two halves. One half is used to train *lexical* rules – these are rules applied in order to predict the tag of a word based on the intra-word characteristics. The other half is used to train *contextual* rules – these are rules applied to tag a word based on its neighboring word contexts.

4.2.3 Lexical Rules

These are used to tag unknown words. Learning lexical rules requires three word lists:

1. A list of all the words occurring in the untagged training corpus $U(C_{train})$, sorted by decreasing frequency of occurrences. The word list is used to find the most common prefixes and suffixes.

[3] The associated explanation is in the following section.

2. A list of triplets [word tag count] derived from $T_{ref}(C_{lex})$, e.g.

   是 vy 365

   和 cpw 358

   在 pzai 339

   The words with more than one tag will get different entries in the list. Besides the triplets [和 cpw 358], the list also contains three more triplets, [和 p 13], [和 cpc 1] and [和 cpw 1] . The count of the triplet is the frequency of the word tag pair in the tagged training corpus. The tagged words are used to calculate the weights of possible tags for a given word.

3. A list of word bigrams found in the untagged training corpus, $U(C_{train})$, e.g.

   是 利用

   都 採用

   心 還

   The bigrams list is used to calculate the weight of the tags to the preceding/following word.

The learning process for lexical rules is depicted in Figure 4.3. The learning process begins by giving the unknown word an initial tag. Such initialization can be done in a number of ways: The unknown word may be assigned **unk**, to denote its out-of-vocabulary nature. Alternatively, since unknown words are often common nouns, we may assign them with the tag **ng** upon initialization. In addition, we may utilize simple prior knowledge, e.g. assign **xch** (tag for non-chinese word) if English letters are encounted,

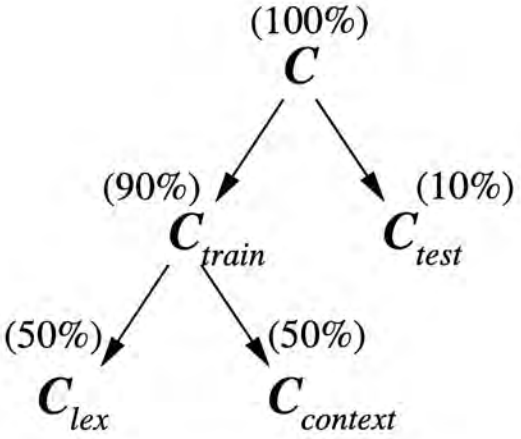


Figure 4.2: Corpus utilization in a particular domain.

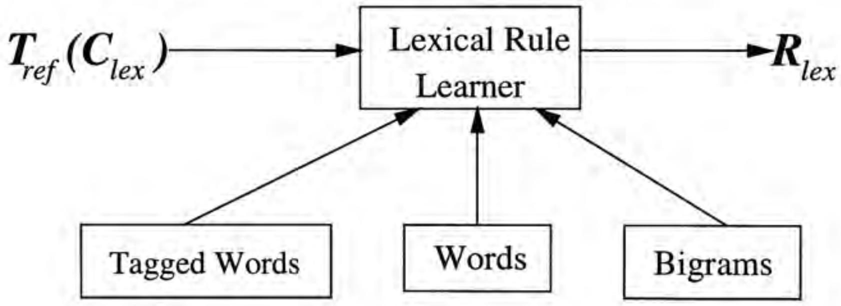


Figure 4.3: Learning Lexical Rules.

or **mx** (tag for numbers used in measurements).

Lexical rules are learnt according to some prescribed templates, so that they can utilize prefixes, suffixes, constitutent characters and bigram relationships to infer an appropriate tag for an unknown word. Some example templates include:

- **{x w fgoodright/fgoodleft y n}**, i.e. given the word in focus **wc** currently tagged as **x**, should the word **w** occur to its right/left, change its tag from **x** to **y**. A close variant of this template is {w goodright/goodleft y n}, which does not constrain the current tag of the word in focus. **n** reflects the relative frequency of rule application in the training set. Here is the equation for calculating n.

  $$n = \sum_{j=1}^{W} N\{word_j, tag_k\} - N\{word_j, tag_i\} \tag{4.1}$$

  where W is the number of words in the training set, tag_k is target tag to be changed, tag_i is current tag.

  $$N\{word_j, tag_k\} = \frac{word_j, tag_k}{\sum_{i=1}^{T} word_j, tag_i} \tag{4.2}$$

  where $word_j$ is a word in the training set, tag_k is a tag for the $word_j$, T is the number of tags for the $word_j$, $word_j tag_k$ is the number of frequency for the pair $word_j tag_k$ in the training set.

  Example of rule application:

  Rule: { ng 李 fgoodright npf 11 }

  Sentence: 年/ng 過/vgn 半百/mx 的/usde 煉鐵廠/ng 老/a 工人/ng 李/nf 傳杰/npf

Here 傳杰 is a unknown word, and the tagger assigns it with **ng** upon initialization. However, seeing the last name 李 towards its left (i.e. 李 is to the *left* of our current word) invokes the specified rule. 傳杰 is then correctly transformed as a **npf** (name of a person).

- **{x z fchar y n}**, i.e. given the word in focus **wc** currently tagged as **x**, should the character **z** occur in the word, change its tag from **x** to **y**. A close variant is {z char y n} which does not constrain the current tag of the word in focus. Example of rule application:

  Rule: { mx 年 fchar t 46 }

  Sentence: 1957年/t 7月/t 到/p 1958年/t 12月/t

  The unknown word 1957年 will be tagged as **mx** (number for measurements) upon initialization. This invokes the specified rule to change to the correct tag **t** (tag for time).

- **{x a fhassuf/fhaspref p y n}**, i.e. given the word in focus **wc** currently tagged as **x**, should it contain the **p** characters in its prefix or suffix a, change its tag from **x** to **y**. A close variant of this template is {a hassuf/haspref p y n}. Example of rule application:

  Rule: { 委員會 hassuf 3 npu 5 }

  Sentence: 聯合國常規軍備委員會/npu 曾/d 通過/vgn 決議/ng ，/ ，

  The unknown word 聯合國常規軍備委員會 will be initialized as **ng**. Owing to the occurrence of suffix 委員會 its tag will be changed to **npu** (name of organization). Therefore it can be seen that the lexical rules automatically learnt during this stage offers insight as to the lexical nature of the words, interpreted with the use of prefixes, suffixes, constitutent characters as well as bigram information.

4.2.4 Contextual Rules

The use of lexicons and lexical rules ensure that each and every word in the text is initialized with a tag. Contextual rules need to be learnt in order to correct any possible errors in the initialization. Hence these rules should be effective in disambiguating among the multiple tag assignments for a given word, using across-word contextual information. The learning process for contextual rules is depicted in Figure 4.4.

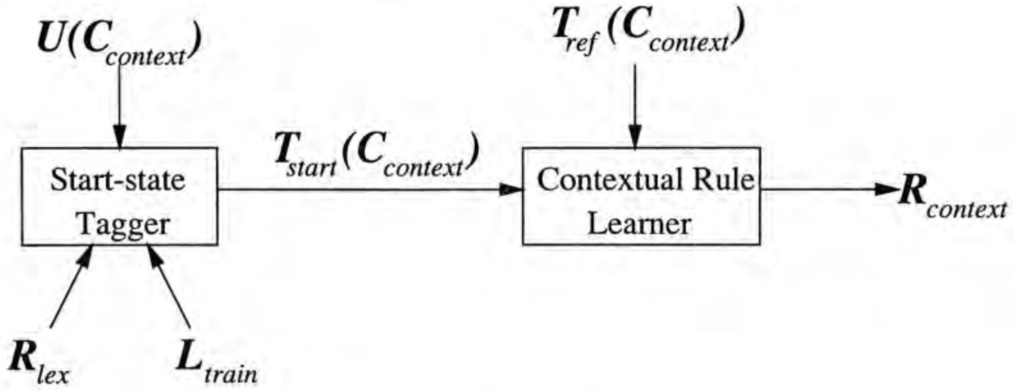


Figure 4.4: Flow chart showing the process of learning contextual rules.

The untagged corpus for learning contextual rules is first processed by the start-state tagger. This tagger references the training lexicon, L_{train}, to assign the most frequent tag to each of the words. Unknown words are tagged by applying the lexical rules. These procedures produce a set of start-state tags $T_{start}(C_{context})$ for the corpus. These are then compared with the reference tags, $T_{ref}(C_{context})$, in order to proceed with error-driven learning, which finally produces the set of contextual rules $R_{context}$. Error-driven learning of the contextual rules also follow a set of templates, which considers the across-word context in a seven-word window – between one to three words/tags to the left and right of the current word (word in focus). Examples of the templates include:

- **{x y next1or2tag staart}**, i.e. given that the current word **wc** is tagged as **x**, change the tag to **y** if the following one or two tags is the start/end of sentence symbol **(staart)**.

  Example of rule application:

  Rule: { usde y next1or2tag staart }

  Sentence: 全/a 過程/ng 中/f 是/vy 可/va 變/vgo 的/y 。/。

  的 is most commonly tagged as **usde**, and is initialized by the start-state tagger thusly. Application of our rule corrects the assignment from usde to y (語氣詞).

- **{x y prevwd w}**, i.e. given the current word wc is tagged as **x**, change the tag to **y** if the previous word is **w**. Example application:

  Rule: { vv f prevwd 年 }

  Sentence: 爲/vi 過去/t 15/mx 年/ng 來/f 的/usde 最/d 大/a 跌/vg 幅/ng 。/。

  The most frequent tag of 來 is **vv**, which becomes the initial assignment of the start-state tagger. However, the application of the rule corrects it to **f**(方位詞).

During the learning process, the start-state tags are compared with the reference tags for each sentence in $C_{context}$. Rules for error correction are proposed according to the templates. The proposed rule which maximally reduces the number of errors is adopted in the *ordered* transformational rule set. The adopted transformation is then applied to the entire training corpus, from left to right, and the transformation is invoked only after all matching contexts in the training set are identified. This constitutes one iteration in learning. Iteration continues until no proposed rules can reduce the minimum count of tagging errors. This minimum count threshold is therefore an experimental parameter.

The difference between the templates of lexical rules and contextual rules is that lexical rules only consider the lexical information of the words (such as prefix, suffix and characters in the word) and neighbouring words. For contextual rules, the considerations are contextual information (such as the previous/following tag of current word), lexical information (such as the previous/following words of current word) and combination of lexical and contextual information (such as the previous/following word and previous/following tag together).

4.3 Experiment and Result

Our experiments are based on disjoint training and test sets, with a 9:1 divide. Each corpus domain is processed individually. We have also combined all the articles for all domains to form a large corpus (71,804 words). This is also divided into training and test sets of the same proportion, and used for experimentation. Figure 4.5 displays a couple of example sentences.

UNIX/xch	Pacific/xch	公司/ng	與/p	AT&T/xch	是/vy	
(UNIX)	(Pacific)	(company)	(and)	(AT&T)	(is)	
什麼/rn	關係/ng	?				
(what)	(relationship)	?				
它/rn	主要/d	是/vy	幹/vgn	什麼/rn	的/usde	?
(It)	(mainly)	(is)	(doing)	(what)		?

Figure 4.5: Examples from the training set, with both segmentation and tagging included. We also include a pseudo English translation in parentheses.

Since the training and test sets are disjoint, we see the occurrences of both *unknown words* as well as *unknown tags* in the test set. An "unknown tag" refers to the tagging of a (known) word in the test set, but the word/tag combination never appeared in the training set. For example, the single-character word 幹 was only seen with the tag **vgn** in the training set. However, it occurred in the test set with the tag **vgv**. Our tagger is bound to make mistakes with cases of unknown tags. The proportion of unknown words and unknown tags range from 8.95% to 33.20% across our domains. Details are shown in Table 4.4.

4.3.1 Lexical Tag Initialization

As mentioned in the previous section, there are multiple schemes for assigning the initial tag to an unknown lexical entity. We can either assign it as **unk** (unknown), **ng** (common noun, most frequently occurring tag for unknown words), or according to our *initial assignment rule*, which incorporates a small amount of prior knowledge:

```
If the word contains an English letter (A-Z / a-z),
    tag it as xch (non-chinese word)
else
    tag as ng (common noun).
```

Results comparing the three schemes are shown in Figure 4.6. Our initial assignment rule fares better than the straightforward unk or ng assignments. Hence we have decided to adopt it for our experiments.

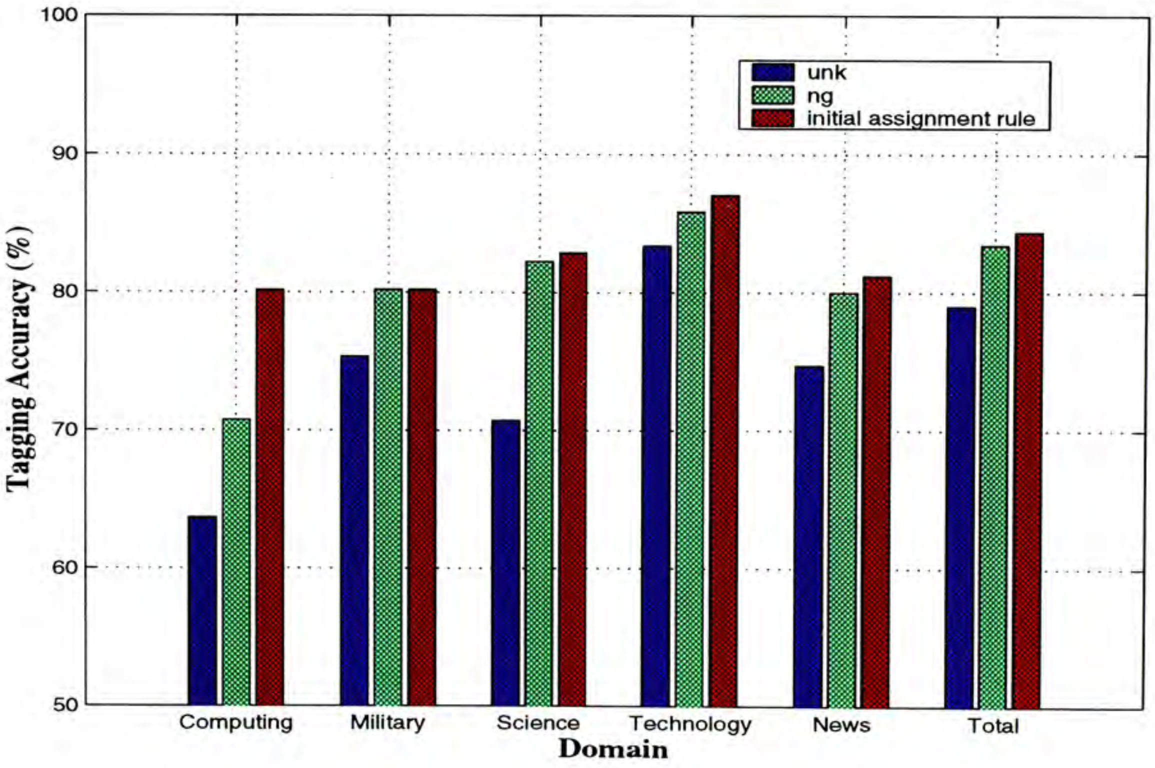


Figure 4.6: Test-set tagging accuracies (%) for the three different initial assignment schemes across the various domains.

4.3.2 Contribution of Lexical and Contextual Rules

Having acquired the initial stage assignments T_0, we proceeded with our experiments by applying first the lexical rules, and subsequently the contextual rules. At each point (T_{start} and T_{final}) we measured the tagging accuracy, in order to assess the respective contributions from the lexical and contextual rules. This procedure is illustrated in Figure 4.7. Experimental results on the test sets are shown in Figure 4.8.

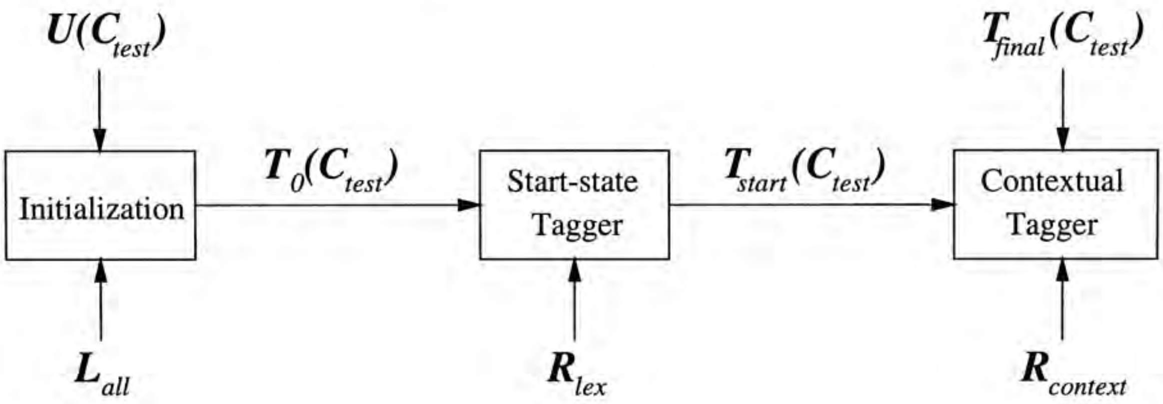


Figure 4.7: Illustration of experimental procedure.

Figure 4.8 shows that the lexical rules brought about a small but consistent improvement (from 0.08% to 1.06% across different domains) over the initial tag assignments across all the domains. However, the contextual rules led to a slight degradation in performance in three of the five domains. For the "Total" category, we believe that the relatively higher improvement is due to a greater amount of training data made available from gathering together 90% of the entire corpus and the co-operation between lexical rules and contextual rules. As an illustration of the co-operation between lexical rules and contextual rules, consider the example sentence:

Domain Proportion(%)	Unknown Words	Unknown Tags	Unknown Words & Tags
Computing	29.08	4.13	33.2
Military	13.26	4.31	17.57
Science	22.14	3.31	25.45
Technology	7.33	1.63	8.96
News	15.85	3.07	18.92
Total	10.00	2.99	12.99

Table 4.4: Distribution of unknown words and unknown tags in the test sets across domains.

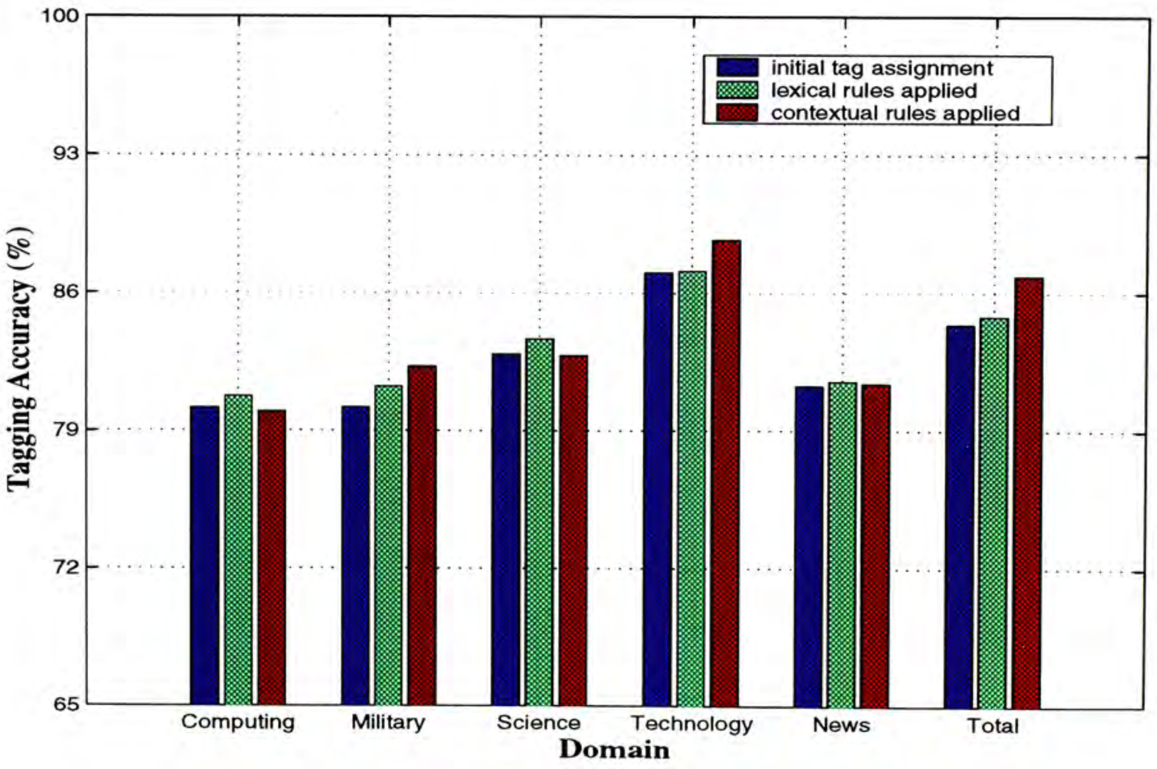


Figure 4.8: Tagging accuracies (%) on the test set.

Untagged Sentence: 各 個 崗位 上 的 各 族 青年 朋友 致以 節日 的 祝賀！/！

Reference Sentence: 各/rn 個/qng 崗位/ng 上/f 的/usde 各/rn 族/ng 青年/ng 朋友/ng 致以/vgn 節日/ng 的/usde 祝賀/nvg！/！

Since 致以 is an unknown word, which is tagged as ng by the start-state tagger. After the initial tag assignments and application of the lexical rule {以 hassuf 2 vgv}, the sentence is tagged as:
各/rn 個/qng 崗位/ng 上/f 的/usde 各/rn 族/ng 青年/ng 朋友/ng 致以/vgv 節日/ng 的/usde 祝賀/nvg！/！
Finally, the application of the contextual rule {vgv vgn SURROUNDTAG ng ng} corrects the tag for 致以 from vgv to vgn and it's the correct tag for 致以 in the sentence.

In order to further assess the contribution of the contextual rules, we examined their effects on the training corpus. Results are shown in Figure 4.9. Since the training corpus does not have unknown words, we only have two sets of tagging accuracies – one from the initial tag assignments, and the other from lexical rule application.

For the results in Figure 4.9, the initial tag assignments utilized the lexicon derived from the training set of the corresponding domain only. Compared to the test-set results, the contextual rules contributed to a more pronounced improvement, across the training sets in all the domains. The improvement did not carry over to the test sets, possibly due to over-fitting to the training sets.

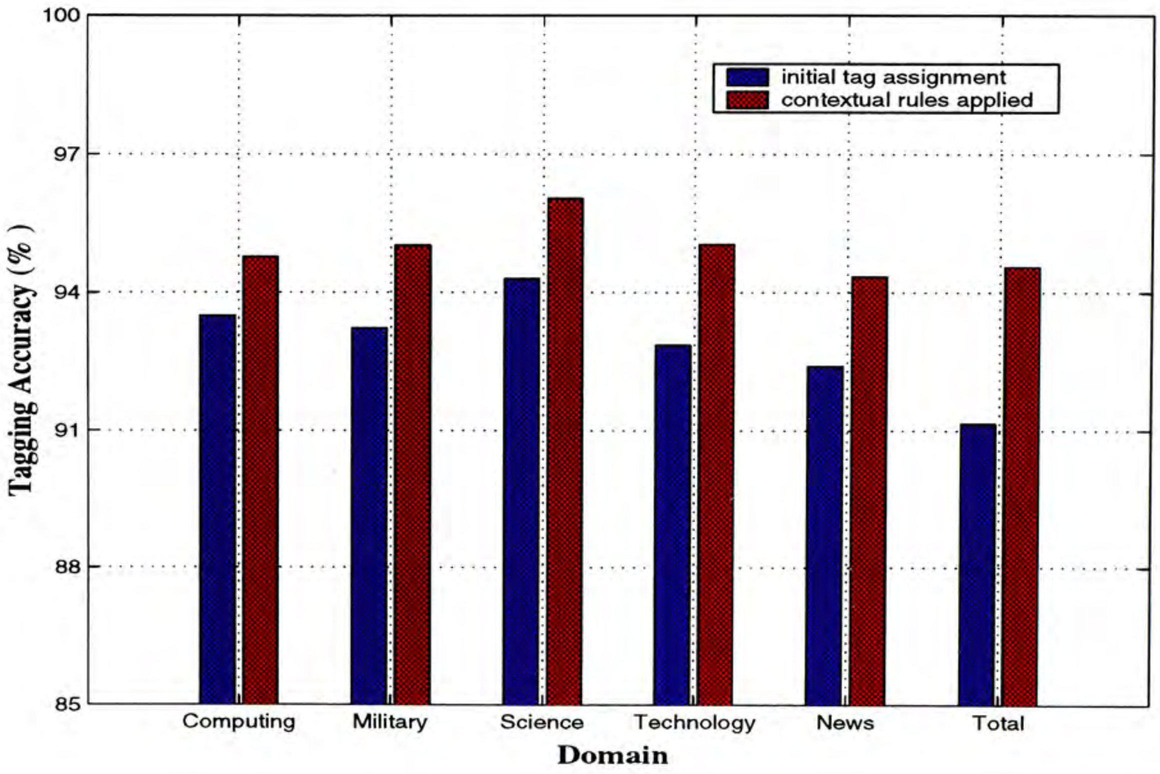


Figure 4.9: Tagging accuracies (%) on the training sets.

Test Sets	Computing	Military	Science	Technology	News	Total
Unknown word Performance	55.41	44.73	56.16	53.33	43.31	56.57

Table 4.5: Tagging accuracies(%) on the unknown words in the test sets.

4.3.3 Performance on Unknown Words

We have also examined our tagging performance on the unknown words and unknown tags in the test set. Performance accuracies on unknown words range between 40 to 50%, as shown in Table 4.5. Our experiments have also shown that the contextual rules learnt have not corrected any of the unknown tag errors in the test set. One reason is due to the propagation of errors – an errorful tag assignment to an unknown word may propagate via contextual rule applications to cause errors in subsequent tags. As an illustration of error propagation, consider the example sentence:

全 市 鄉鎮 企業 中 已 有 30 多 家 中 外 合資 合作 企業。

where 合資 is the unknown word, the tag **qni**(個體量詞) of 家 is the unknown tag. After the initial tag assignments and application of the lexical rules, the sentence is tagged as:

全/a 市/ng 鄉鎮/ng 企業/ng 中/f 已/d 有/vh 30/mx 多/mg 家/ng 中/f 外/f 合資/ng 合作/vg 企業/ng 。/。

The unknown word 合資 is tagged as **ng**.

Subsequent to this, application of the contextual rule {vg vgn prev1or2tag ng} transforms the tag for 合作 (from **vg** to **vgn**) since its left tag of word 合資 is **ng**. Therefore, the tag of 合作 is becomes an error. Now the sentence tags become:

全/a 市/ng 鄉鎮/ng 企業/ng 中/f 已/d 有/vh 30/mx 多/mg 家/ng 中/f 外/f 合資/ng 合作/vg 企業/ng 。/。

This is compared with the reference tags:

全/a 市/ng 鄉鎮/ng 企業/ng 中/f 已/d 有/vh 30/mx 多/mg 家/ng 中/f 外/f 合資/ng 合作/vg 企業/ng 。/。

We find five errors in the TEL tagging:

家/ng, 中/f, 外/f, 合資/ng, 合作/vg (hypothesized)

家/qni, 中/j, 外/j, 合資/d, 合作/vg (reference)

and among these three originated from unknown words and unknown tags (家, 合資, 合作)

4.3.4 A Possible Benchmark

We attempt to come up with an upper bound benchmark for our performance accuracies, by ameliorating the unknown word problem. To achieve this we included all the words in our *entire* corpus (L_{all}) for initial tag assignment. We have also used the entire training corpus for training the contextual rules (instead of divided it into the lexical and contextual portions, as mentioned previously). This experimental procedure is illustrated in Figure 4.10. Our experimental results suggest that possible upper bounds for tagging performance lies around 97% for training and 94% for testing in domain total. This compares with the previous performances of 94.56% in the training set (please see Figure 9 in textcolorredpp.55) and 86.87% in the testing set (please see Figure 8 in textcolorredpp.53).

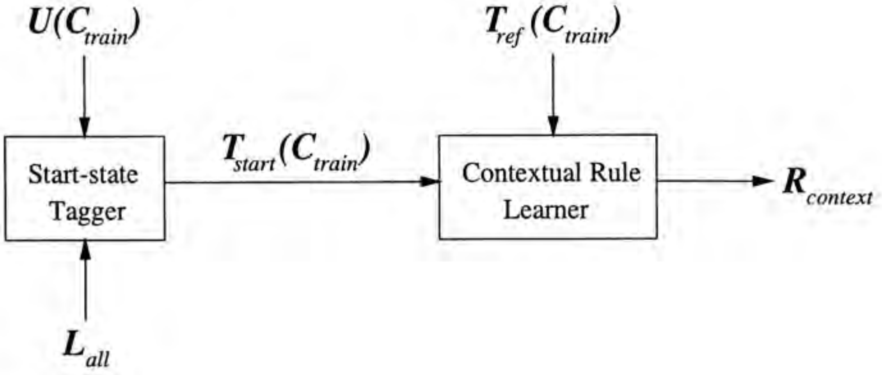


Figure 4.10: Training procedure which attempts to ameliorate the effect of unknown words.

Experimental results for both training and test sets are tabulated in Table 4.6.

Test Sets	Computing	Military	Science	Technology	News	Total
Training Accuracies	96.13	96.98	97.70	96.98	96.70	96.96
Testing Accuracies	94.10	92.05	94.18	92.51	92.73	93.88

Table 4.6: Tagging accuracies(%) for both training and test sets, under the condition with no unknown words.

4.3.5 Comparison between TEL Approach and the Stochastic Approach

We attempted to compare the TEL approach with a stochastic approach for POS tagging. Our stochastic tagger is provided by Tsinghua University. It utilizes a Markov model for POS tagging, i.e.

$$P(T's/W_s) = max_{T_1T_2...T_n}P(T_1/T_2)\prod_{i=2}^{n}P(T_i/T_i-1)P(W_i/T_i) \qquad (4.3)$$

and has been previously trained.[4] Therefore it was not straightforward for us to compare the two taggers based on identical training and testing sets. We divided each corpus into 10 partitions – 9 of them were used to train the TEL tagger and the remaining one for testing. This preserves the 9:1 divide between training and testing sets. These experiments are repeated 5 times by jackknifing the data sets, and the performance accuracies were averaged (see row 2, row 3 and column 7 of Table 7). We combined the average training and testing accuracies according to the formula:

$$OverallAccuracy(TEL) = 0.9 \times acc_{train} + 0.1 \times acc_{test} \qquad (4.4)$$

where acc_{train} is the average training accuracy and acc_{test} is the average testing accuracy.

The weights of the training and testing accuracies follow the proportion of the respective data sets. The Overall Accuracy (TEL), shown in the third row of Table 4.7 were compared with the corresponding values of the stochastic tagger, shown in the last row of the table. Our results suggest that the TEL and stochastic approaches produce comparable results.

4.4 Chapter Summary

This work is our initial attempt in using the transformation-based error-driven learning (TEL) procedure for tagging Chinese text. TEL has previ-

[4] Previous literature indicates that the training was based on 90% of the corpus.

Experimental Runs	1	2	3	4	5	Average (over 5 runs)
TEL tagger (Training Accuracy	95.20	95.17	95.16	95.00	95.17	95.14
TEL tagger (Testing Accuracy	88.33	87.60	87.46	88.40	87.26	87.80
TEL tagger (Overall Accuracy	94.50	94.35	94.33	94.39	94.41	94.38
Tsinghua tagger	91.59	91.59	91.59	91.59	91.59	91.59

Table 4.7: Tagging accuracies(%) for both training and test sets. Comparison between the TEL approach and stochastic approach.

ously been shown to be effective in POS tagging for English (achieving over 96% tagging accuracies in using the Brown and WSJ corpora) [Brill 1995]. It has several attractive properties: (i) it provides an automatic procedure for tagging, (ii) the lexical and contextual rules it learns often make intuitive sense for the Chinese language, and potential provides room for the incorporation of linguistic knowledge by a human, should there be sparse training data problems, (iii) the learning procedure aims to minimize errors to obtain maximum tagging accuracies.

We divided a Chinese news corpus of over 70,000 words into disjoint training and test sets of a 9:1 ratio, we achieved overall tagging accuracies of 94.56% (training) and 86.87% (testing). Across the different domains, the proportion of unknown words and unknown tags range between 8% to 33%, and tagging performance from 79.96% to 88.68%. In general, the higher the proportion of unknown words/tags, the lower the tagging performance. The baseline performance (without applying any rules) was 91.16% (training) and 84.39% (testing). Both the lexical and contextual rules were found to be contributive towards tagging performance. Performance accuracies are much improved upon the use of a comprehensive lexicon to ameliorate the unknown word problem, reaching 96.96% (training) and 93.88% (testing) respectively as a possible gauge of an upper bound performance for our experiment. While direct comparison with the work of others[5] is difficult due to uncertainties in training/testing data partitioning, our experimental results in comparison with a stochastic tagger suggests that TEL is equally effective and applicable for Chinese.

We believe that TEL tagging can capture many of the named entities. However, named entities which are unknown words tend to be incorrectly tokenized. In next chapter, we discuss our approach for handling this problem.

[5] e.g. Stochastic models in [6].

Chapter 5

Handling Unknown Words in Topic Tracking

A characteristic of Chinese is that a word may contain one or more characters but with no delimiters for word boundaries. Therefore, the first step toward topic tracking in Chinese is to tokenize a sequence of characters into words. However, tokenization is often ambiguous and may be hampered by the occurrence of OOV. For example, 萊溫斯基 (Lewinsky) is an OOV and segmented into four monosyllabic words (i.e. single-character words) 萊, 溫, 斯 and 基. Named entities are important for topic tracking and named entities are often OOV. Therefore, handling OOV is important for our task.

In this chapter, we discuss our attempt in using a hybrid approach for handling the unknown word (OOV) problem. The goal of this task is to formulate an unknown word identifier for salvaging the named entities "missed" during the segmentation process, and capturing these entities for topic tracking in Chinese text.

This work explores the use of n-gram techniques together with heuristic rules for detecting and identifying unknown words. We are especially interested in finding four types of entities (person names, organization names, location names and time) from a small set of documents since there are only four training stories for topic tracking. The relative contribution between the tagged named entities as well as unknown word identification is another focus in this study.

5.1 Overview

N-gram grouping is a commonly used technique in language processing [58] [58] [73] [47] [38] [15] [31]. Most of these tasks use large corpora (e.g. Nie et al. used 790,000 Chinese characters text [47]). However, for our task of topic tracking, the size of training data is just only 1,000 words on average. We are only given 4 training stories, which is just sufficient to characterize a topic. For this reason, we focus on the extraction of n-grams from a small amount of data.

Insufficient training data causes the pure n-gram technique to become unreliable. Therefore, the use of heuristic rules is introduced in this work. It is generally believed that heuristic rules are effective for words with regular structures, such as numbers and dates. As a result, the use of heuristic knowledge helps an n-gram technique to recognize the named entities, for example, the use of family name (surname) helps n-gram technique to identify the name of person.

In the following, we introduce a hybrid approach to extract four kinds of entities: person names, organization names, location names and times.

5.2 Person Names

Names of people in Chinese text can be divided into two kinds: Chinese names and foreign names [31]. The composition of these names are different, as described in the following.

Chinese names consists of two elements: family names and given names. Most family names consist of a single character, such as 王 (Wang), 陳 (Chen) and 李 (Li). A few of family names have two characters, for example, 歐陽 (Ouyang) and 司馬 (Sima). In given names, most of them have two characters but some have only a single character. The properties of family names and given names restrict the length of Chinese names to between 2 and 4 characters.

There are millions to billions Chinese names in the world. However, the number of family names are limited to a finite set of characters. According to the report of "姓氏人名用字分析" (Analysis of person names)[1] in 1990, the most frequent 200 family names cover 96% of the sample of 174,900 persons. In 2000, Tsai's "Frequency Counts of Chinese Names"[2] stated that the most frequent 100 family names cover 97% of the sample of 217,913 Chinese name tokens. In Figure 5.1, we show the frequency distribution of the top 10 family names in Tsai's report.

Foreign names are different from Chinese names. Most foreign names

[1] http://zhongwen.com/x/xing.htm

[2] http://www.geocities.com/hao510/namefreq

#	Family Names	Frequency
1	陳 (Chen)	24589
2	林 (Lin)	18628
3	黃 (Huang)	13813
4	李 (Li)	11460
5	張 (Zhang)	11333
6	吳 (Wu)	9063
7	王 (Wang)	8952
8	蔡 (Cai)	6739
9	劉 (Liu)	6692
10	楊 (Yang)	6057

Table 5.1: The frequency distribution of first 10 family names in Tsai's statistic[10].

are transliterated phonetically. A complete foreign name is composed of the first name, middle name and last name, such as 阿卜(first name)・巴塞特(middle name)・阿里(middle name)・賽格拉西(last name)[3], where the first, middle and last names are separated by a ‘・’.

Unlike Chinese names, the length of foreign names are not constrained to 2 to 4 characters. An example is 內塔尼亞胡 (Netanyahu). Unlike Chinese name, foreigners name do not have fixed morphological structure. However, there is a constrained character set for transliterated names [31], for example, 阿 (a), 埃 (ai) and 奧 (ao).

[3] This example is drawn from [31].

5.2.1 Forming possible named entities from OOV by grouping n-grams

There are three considerations in formulating n-gram candidates: (i) detecting the existence of possible OOV words. (ii) determining the number of tokens to group (i.e. the value for *n* in n-gram grouping). (iii) constraining the groupings to give person names. In the following, we will discuss these considerations.

The monosyllabic word is a major cue in the detect of unknown words. For example, inspection of the Sinica corpus [48] shows that 4572 out of 4632 unknown words were incorrectly segmented into a sequence of shorter words, and each sequence contained at least one monosyllabic word. Therefore, monosyllabic words in segmented text may suggest the occurrence of a named entity. As an example, 斯塔爾 (Starr) is segmented into three monosyllabic words 斯, 塔 and 爾. In addition, the segmenter tends to tokenize a person name into a sequence of monosyllabic words since most of the given names do not contain legitimate words. Therefore, we decided to create n-grams from contiguous isolated character in the segmentation.

As mentioned, Chinese names generally range from 2 to 4 characters. Although the length of foreign names are not restricted to certain value, we observe that the 5-gram is sufficient to include the necessary words for topic tracking. Therefore, in this study, value of n is set to 5 (i.e. a penta-gram). Consider an example:

- 美國 (U.S.A.) 國務卿 (Secretary) | 奧 爾 布 賴 特 | (Albright) 說 (said) ...

First, a monosyllabic word "奧" triggers the n-gram grouping process. Thereafter, the n-grams 奧爾 (bi-gram), 奧爾布 (tri-gram), 奧爾布賴 (4-gram), 奧爾布賴特 (5-gram) are created independently for n=1,2 ... 5. Since the maximum n is set to 5, therefore, the n-gram grouping process is stopped after the 5-gram "奧爾布賴特" has been created. On the other hand, a monosyllabic word "説" will terminate because it is a stopword. We will explain this in detail later.

Furthermore, if we observe an n-gram which begins in a character from the list of Chinese surnames, the probability that this candidate belongs to person name tends to be high. Therefore, we create a word set containing the most frequent family names and transliterated characters for foreign names. We hope this word list can help us select the n-grams which belong to the person name. The formulation of this word set is based on the following sources:

1. Lexicons used in segmenter and POS tagger
2. 姓氏人名用字分析
3. Frequency Counts of Chinese Names
4. Erik's Chinese Named Extraction System[4]

From (1) to (4) above, we created a person-name list[5] with 396 words. It includes 286 family names (283 single-character surnames and 3 two-characters surnames) and 110 common characters used for foreign names.

[4] http://www.mandarintools.com/segmenter.html

[5] A complete person-name list is provided in Appendix B.

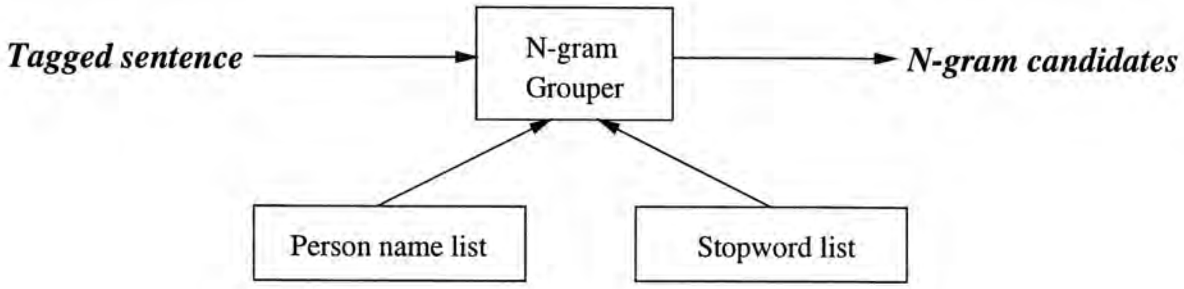


Figure 5.1: Formulating n-grams.

Furthermore, if we observe an n-gram which involve a character (e.g. 是, 會) from the the list of stopwords, the probability that this candidate belongs to noise tends to be high. For these, we created a stopword list[6] with 32 punctuation marks and 18 single characters based on our POS lexicon and the Erik's system.

After we formulate the words sets, we apply the n-gram grouping technique to capture the possible word candidates. This procedure is described in Figure 5.2. Illustration of n-gram grouping system is shown in Figure 5.1.

The process for formulating n-grams begins by finding a word which exists in the person-name list (first step in Figure 5.2). In step 2, we constrain the maximum length of n-gram formed. In step 3, we restrict all the elements in n-gram to be the isolated single character. In step 4, we end the forming process with possible ending characters or symbols obtained from stopword list. The forming process will be continued until it reaches the maximum size or violate the constraints.

Some examples of grouped n-grams are shown in Figure 5.3. All of these samples are extracted from four training stories (about 1,000 words) in the

[6] A complete stopword list is provided in Appendix C.

```
1. If the current word exists in family names and transliterated words set,
       go to step 5.
   else, exit.
2. If the length of n-gram candidate is more than 5 characters, exit.
3. If the length of current word is more than 2 characters, exit.
4. If the current word exists in stopword set, exit.
5. Formulate n-gram candidate with current word.
6. Go to next word (current word=next word).
7. Go to step 2.
```

Figure 5.2: Procedure of n-gram Grouping.

training set of topic tracking.

5.2.2 Overlapping

Since n-grams are created independently for each n, a person name in a text may be counted as several n-grams with different values of n. For example, 克林頓 (Clinton) in a text may be counted for bi-gram "克林" and tri-gram "克林頓". That means the appearance of person name 克林頓 is also counted for its substring 克林 which are not words. We refer to this as *overlapping between n-grams.*

We favor longer n-grams than shorter ones. Therefore, we use a simple equation[7] to eliminate the possibly overlapped n-grams.

Suppose the n-gram X is included in a longer n-gram Y, then,

$$freq(X) = freq(X) - freq(Y) \tag{5.1}$$

[7] This equation is drawn from [47].

n-gram	frequency	N-gram	frequency
克林頓 (name)	14	萊溫斯基 (name)	7
斯塔爾 (name)	10	阿爾弗來德 (name)	3
德貝內 (part of name)	5	馬科斯 (name)	3
伊梅爾達 (name)	2	單閣號 (name of balloon)	4
多帶了 (bring more)	3	福塞特 (name)	10
里特 (part of name)	5	斯科特 (part of name)	2
特委 (abbreviation)	12	華航 (abbreviation)	3
華行 (abbreviation)	1	多起(-)	3
伯勒 (part of name)	3	和一 (-)	2
高難 (-)	2	關穎珊 (name)	9
關穎珊以 (-)	1	郭政新 (name)	8
明尼阿波利 (name)	3	普魯申科 (name)	2
申雪 (-)	4	趙宏博 (name)	2
奧爾布賴特 (name)	3	內塔尼亞胡 (name)	9

Figure 5.3: Sample of n-gram word candidates.

For example, a bigram 克林 is included in a trigram 克林頓, then

$$\mathit{freq}(\text{克林}) = \mathit{freq}(\text{克林}) - \mathit{freq}(\text{克林頓})$$

Nie et al. [47] has proven this method is an efficient way to eliminate the overlapping n-grams with a particular frequency threshold from a large corpus. If the frequency of an n-gram formed in a corpus is lower than the frequency threshold, this n-gram will be counted as "noise". In our study, we apply method to eliminate overlapped n-grams in the task of topic tracking. Since the extracted entities are provided for topic tracking, a frequency threshold is used to optimize the performance of topic tracking. Therefore, the method for setting an appropriate value for threshold will be discussed in Chapter 6.

1.	Org. Name	=	Org. Name (npu)	+	Suffix (osuf)
2.	Org. Name	=	Location Name (s)	+	Suffix (osuf)
3.	Org. Name	=	Person Name (npf)	+	Suffix (osuf)
4.	Org. Name	=	Proper Noun (npr)	+	Suffix (osuf)
5.	Org. Name	=	Person Name (npf)	+	Org. Name (npu)
6.	Org. Name	=	Location Name (s)	+	Org. Name (npu)
7.	Org. Name	=	N-gram Candidate (Ngm)	+	Org. Name (npu)

Table 5.2: Formulation rules for extracting organization name.

5.3 Organization Names

There are seven formulation rules covering six categories of elements for extracting organization names (see Table 5.2[8]). Referring to the rules, each formulated organization name is composed of two elements, which may be one of the following components: (i) organization name, (ii) location name, (iii) person name, (iv) proper noun, (v) n-gram candidate and (vi) suffix of an organization.

Before we apply the formulation rules, the training stories have been tagged by our TEL tagger. Hence, the organization names (npu), location names (s), person names (npf) and proper nouns (npr) are found by their corresponding tags.[9] Moreover, the n-gram candidates found by n-gram grouping will be used to formulate the possible organization names.

There are 51 organization suffixes used in the formulation rules. The suffixes of organization name are collected from 3 sources: (i) the lexicon of TEL tagger, (ii) the Erik's Chinese Named Extraction System [26], and

[8] We decide the tag *osuf* to denote a suffix of organization. All of the other tags are listed in Appendix A.

[9] All the tags and meanings are shown in Appendix A.

(iii) the tagging set of Tsinghua University described in Chapter 4. Here are some example suffixes:[10]

部, 會, 組織 , 基金會, 廣播公司

The process for formulating organization names is depicted in Figure 5.4. The tagged sentence for formulating organization names is first processed by the component checker. This checker references the formulation rules, to check whether the current word (**wc**) is one of the component (npu, s, npf, npr, s or Ngm) in the rules. If **wc** is a component of the formulation rules, formulating process progress done in two ways: The tagged sentence will be passed to a suffix checker when **wc** is one of the component in rule 1 to 4 (Table 5.2. Otherwise, the sentence will be passed to entity formulater. In the suffix checker, the words following **wc** checked to whether they are one of the suffixes in the list of organization-suffixes. If these words can be grouped into one of the organization suffixes, they will be grouped into a suffix and passed to entity formulater. In entity formulater, the words will be grouped into organization names based on the formulation rules. Here is an example of rule applications:

- 哥倫比亞(s) + 廣播(ng) 公司(ng)
  → 哥倫比亞(s) + 廣播公司(osuf)
  → 哥倫比亞廣播公司 (Rule 2 in Table 5.2)

First, a location name "哥倫比亞" triggers the formulation process. Next, "廣播" and "公司" will be passed to suffixes checker. In suffixes

[10] A complete listing is provided in Appendix D.

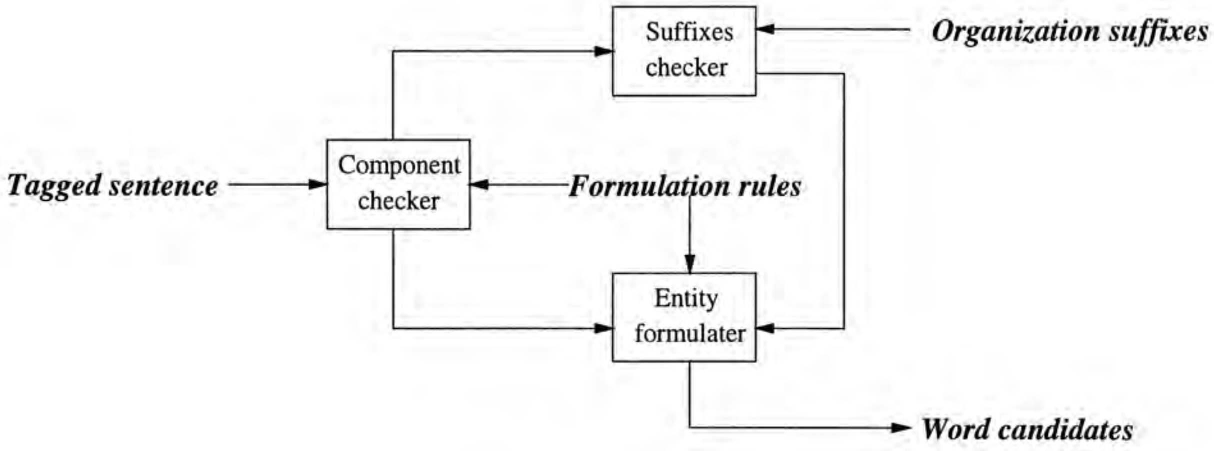


Figure 5.4: Formulating organization names.

checker, a complete suffix “廣播公司” will be grouped from “廣播” and “公司’. Finally, “哥倫比亞” and “廣播公司” will be merged to form a company name “哥倫比亞廣播公司” in entity formulater based on rule 2.

5.4 Location Names

The formulating process of location names is similar to that for organization names. In Table 5.3, we show three formulation rules for extracting location names.

1.	Location Name	=	Person Name (npf)	+	Suffix (lsuf)
2.	Location Name	=	Location Name (s)	+	Suffix (lsuf)
3.	Location Name	=	N-gram Candidate (Ngm)	+	Suffix (lsuf)

Table 5.3: Formulation rules for extracting location name.

There are 32 location suffixes used in the formulation rules. The suffixes of location names are collected from the same sources of organization names.

Here are some example suffixes[11]:

州, 山, 商場, 機場, 中心

In addition, we show an example of rule application:

- 阿肯色(s) + 州(suf) → 阿肯色州 (Rule 2 in 5.3)

Some location names may overlap with organization names. For example, 印尼大學 (University of Malaysia) may denote an organization or a location. However, since our goal is to extract the key named entities, the overlap does not affect us and we assign the same tag to all of formulated organization names and location names.

5.5 Dates and Times

The morphological structures of dates and times are similar. All of these entities are composed of two elements: numbers and units. Table 5.4 shows the formulation rules for extracting these two kinds of entities.

1.	Date	=	Number (mx)	+	Suffix (tsuf)
2.	Time	=	Number (mx)	+	Suffix (tsuf)
3.	Date	=	Date (t)	+	Date (t)
4.	Time	=	Time (t)	+	Time (t)

Table 5.4: Formulation rules for extracting date and time.

There are two kinds of numerical expressions – Arabic numbers, e.g.. *1,2,* and Chinese numbers, e.g. 一, 二 and 壹, 貳 etc. A number sequence is

[11] A complete listing is provided in Appendix E.

grouped into larger number unit. For example, two adjacent tokens 一二三四 and 五 will be transformed to 一二三四五 after the grouping process.

In this task, we have created a word set containing 15 basic time units. Here are the time units:

年, 月, 日, 時, 分 ,秒, 點, 年初, 年底, 年前, 年中, 月初, 月底, 月前, 月中

Basically, date and time entities are used to represent the concept "time" with different measuring units. Therefore, we simply assign a tag *t* to all of formulated date and times entities. Here is an example of rule application:

- 1997 (mx) 年 (qnz) 1 (mx) 月 (qnz)
  → 1997年 (t) + 1月 (t) (Rule 1 in Table 5.4)
  → 1997年1月 (t) (Rule 3 in Table 5.4)

In the previous example, two date entities (1997年 and 1月) are formed in first step by rule 1 in Table 5.4. Next, according to rule 3, two date entities will be grouped to form a larger date entity. Therefore, the final entity (1997年1月) will be formed after the rule application.

5.6 Chapter Summary

Extracting named entities is the important step for topic tracking. However, tokenization ambiguities and unknown words are two obstacles for finding these entities. In this chapter, we propose two techniques for handling these problems: one is n-gram grouping (a data-driven approach), the other is the use of formulation rules (a heuristic approach). We applied these two techniques to the output of a Chinese tokenizer and our TEL tagger. In addition, we describe how our approach extracting the named entities lost due to tokenization errors and the unknown word problem.

We believe that capturing named entities is key to the task of topic tracking. In chapter 6, we will explore this hypothesis.

Chapter 6

Topic Tracking in Chinese

In chapter 4 and 5, we described two systems (TEL taggers and OOV identifier) we have developed for named entities extraction. In this chapter, we attempt to use these extracted named entities for topic tracking in Chinese. The goal of this task is to formulate a classifier in applying the vector-space model, for classifying the subsequent story whether it is on-topic or off-topic.

This work explores the use of "Named Entity" approach for topic tracking in Chinese text. The previous approaches for topic tracking are mostly data-driven which cause the performance to vary from one domain to another. In "Named Entity" approach, we transform a story into a series of concepts (named entities). We believe these named entities are generally appeared in different domain, for example, person names, location names and time. Moreover, these named entities are also plausible terms to represent the content of a news story.

This chapter describes the features selection process, and how the features are classified as on-topic or off-topic for topic tracking. We evaluated our experimental results in two official evaluation methods [51] of TDT-3 project: (i) cost function and (ii) detection error tradeoff (DET) graph.

6.1 Introduction of Topic Tracking

Topic tracking begins with extracting named entities from the training stories. We apply the TEL tagger and OOV identifier to extract named entities. Then, we use the vector-space model to transform these entities into a *document* vector. Next, the subsequent testing story is transformed into a *query* vector by the vector-space model. The similarity score between document and query vector is then calculated with four weighting schemes in the SMART's system. Finally, the tracker outputs the score and make the binary decision (testing story is on-topic or off-topic).

Four parameters are used in this experiment: (i) The length of a document vector; (ii) the minimum occurrence frequency for feature selection; (iii) the weighting schemes for calculating the similarity score between the document and query vectors. (iv) the similarity score threshold which used to decide the story is on-topic or off-topic.

Our experiments involve a training phase and a testing phase. In the training phase, we tune all the four parameters to optimize the performance of our tracker. Tuning includes deciding the length of a document feature vector, the minimum occurrence frequency as a threshold for feature selection, the weights associated to each feature, and a single similarity threshold which optimizes the overall tracking performance. These "optimized" parameters are then used during the testing phase to evaluate our tracker based on the test set.

6.2 Experimental Data

This work is based on the TDT-3 corpus from the Linguistic Data Consortium [12]. This news corpus were collected from broadcast and newswire sources between January 1 and June 30, 1998. The data sources of Mandarin language include 2 newswire services (Xinhua and Zaobao) and 1 radio broadcasters (Voice of Mandarin). The news stories collected from TDT-3 is divided into two sets: (i) training set (January 1 - April 30, 1998) and (ii) development set (May 1 - June 30, 1998). Altogether there are 911 files which contain 18,721 news stories, distributed across 20 specific topics. In this thesis, we use the designated training set in TDT-3, and the designated development set to be our test set. Information about the TDT-3 corpus is tabulated in Table 6.1:

Corpus	**Date**	**# of files**	**# of stories**	**# of topics**
TDT3-Train	Jan. 1, 1998 - Apr. 30, 1998	504	11224	16
TDT3-Test	May 1, 1998 - June 30, 1998	407	7497	15
TDT3-Total	Jan. 1, 1998 - June 30, 1998	911	18721	20

Table 6.1: Information about the TDT-3 Corpus.

For topic annotation, 20 topics of Mandarin data are selected from the original 100 TDT-2[1] topics. Each story in the corpus will be judged to be 3 cases: (i) non-relevant to the topic (off-topic), (ii) relevant to the topic

[1] History of TDT is discussed in chapter 1.

Corpus	# of topics	Total stories	# of non-relevant	# of relevant	# of briefs
TDT3-Train	16	11224	9410	1239	575
TDT3-Test	15	7497	6106	902	489
TDT3-Total	20	18721	15516	2141	1064

Table 6.2: Topic Information about the TDT-3 corpus.

Corpus	Total topics	Available topics	Overlapped topics	Unique topics
TDT3-Train	16	14	7	7
TDT3-Test	15	12	7	5

Table 6.3: Distribution of topics in training and test sets.

(on-topic), or (iii) belief in a story that included a short mention of the topic (brief).

The topics used in the TDT-3 corpus are summarized in Table 6.2. There are 16 topics in the training set and 15 topics in the test set. In topic tracking, each topic is defined by 4 training stories. But 2 out of 16 topics in training set and 3 out of 15 topics in the train set has less than 4 stories in its topic. As a result, only 14 topics in training set and 12 topics in the test set are used for topic tracking.

Table 6.3 shows the distribution of topics in training and test sets. Seven topics are overlapped in both sets. Seven topics are included in test set but not in test set. Five topics are included in test set but not in training set.

There are hundreds to thousands of non-relevant stories. This non-relevant stories provide the background information to topic tracking. We will discuss it in Section 6.4.

6.3 Evaluation Methodology

The task of topic tracking is evaluated in terms of the ability to detect which stories are on-topic (i.e. relevant to the topic) and which are off-topic (i.e. non-relevant to the topic). Each topic is treated individually. For each topic, a tracking system should output a score and a binary decision (YES: judging the testing story which is on-topic, NO: judging the testing story which is off-topic). All of these outputs are used to find the probability of miss and false alarm errors which are the evaluation factors in TDT-3 evaluation.

In topic tracking, a miss occurs if the system declares a story to be off-topic but in reality it is on-topic, and a *false alarm* error occurs when the system declares a story to be on-topic but in reality it is off-topic. Consider the following table:

	retrieved	**not retrieved**
relevant	a	b
non-relevant	c	d

The probability of miss and false alarm errors are:

- *Probability of Miss Error* $= \frac{b}{a+b}$
- *Probability of False Alarm Error* $= \frac{c}{c+d}$

These error probabilities are used in two evaluation methodologies: (i) cost function and (ii) DET curve.

6.3.1 Cost Function

In TDT-3, a cost function is used to analyze the effectiveness of topic tracking. Here is the cost function:

$$C_{track} = C_{Miss} \cdot P_{Miss} \cdot P_{target} + C_{FA} \cdot P_{FA} \cdot P_{non-target} \tag{6.1}$$

where

C_{Miss} and C_{FA} are the costs of miss and false alarm errors

P_{Miss} and P_{FA} are the probabilities of miss and false alarm errors

P_{target} and $P_{non-target}$ are the priori target probabilities

C_{miss}	1.0
C_{FA}	0.1
P_{target}	0.02
$P_{non-target}$	0.98

Table 6.4: Topic tracking evaluation cost parameters.

The evaluation cost parameters are shown in Table 6.4.[2] The evaluation cost calculated by these parameters is used to explore the trade-off between miss and false alarm errors. Besides the parameters, in TDT-3, the cost function is normalized in the following equation [41]:

$$C_{norm} = C_{track}/min(C_{miss} \cdot P_{target}, C_{FA} \cdot P_{non-target}) \tag{6.2}$$

There are two evaluation methods: (i) story-weighted and (ii) topic-weighted methods. The story-weighted method assigns equal weight to each decision for each story and accumulates errors over all topics. The topic-

[2] These are the official values used in the TDT-3 evaluation [51].

weighted method accumulates errors separately for each topic and then averages the error probabilities over topics, with equal weight assigned to each topic [51]. In this work, we use topic-weighted method to be our evaluation. Here are two reasons of using the topic-weighted method:

1. In the story-weight method, the evaluation result will be biased to the topic which has more news stories. It is because the story-weighted method accumulates the error probabilities over all topics. However, the evaluation result of the topic-weight method is the average of error probabilities over all topics. It avoids the bias to the topic which has more news stories.

2. The topic-weight method is an official evaluation method in the TDT-3.

6.3.2 DET Curve

The *detection error tradeoff* (DET) [3] curve uses the same evaluation concepts (miss and false alarm) of cost function mentioned as Section 6.3.1. The main difference is that DET curve presents the evaluation results in visual form while cost function represents the evaluation results into a single figure.

An example of a DET graph is shown in Figure 6.1. During topic tracking, each testing story possesses a similarity score related to a given document vector. This similarity score is used to determine if the testing story is on-topic or off-topic (using a similarity threshold) in relation to the document vector. All testing stories are then sorted based on their similarity scores. The values of miss and false alarm errors are computed for a given similarity threshold. In the curve, the points closer to the origin indicate a better

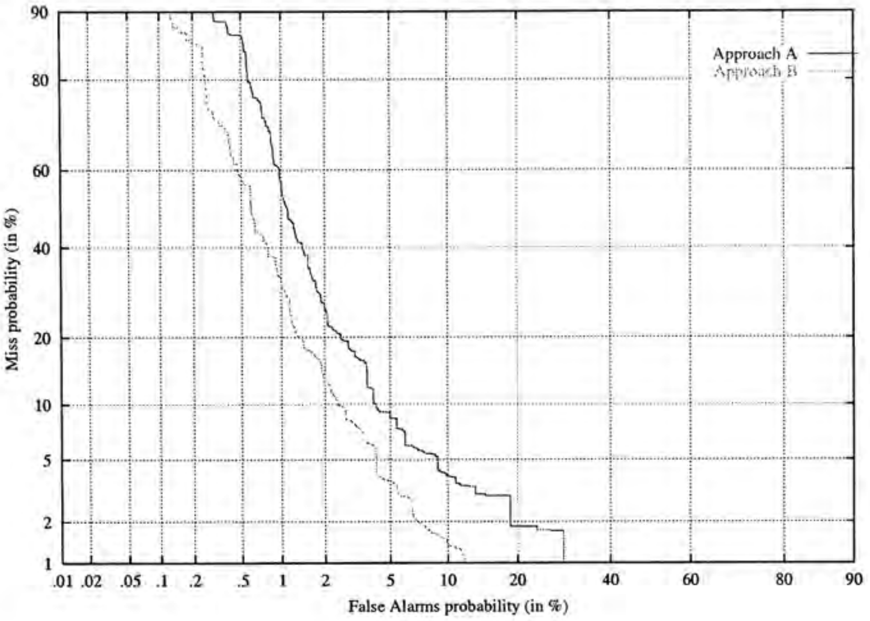


Figure 6.1: An example of DET graph.

overall performance. Referring to Figure 6.1, the tracking performance of approach B is better than approach A since all the points on DET curve of approach B are below the points on DET curve of approach A. The performance of topic tracking is bounded between 0% to 100% miss and false alarms probability. A low miss probability will bring the high false alarms probability.

Similar to the case of cost functions, there are two evaluation methods (story-weighted and topic-weighted) of DET curves. As mentioned in the last section, the topic-weighted method is better method in evaluation. Therefore, we used the topic-weighted DET curves to be our evaluation methods.

6.4 The Named Entity Approach

Time, people, and place are three important entities in a news story. These information attributes are what the readers or listeners are expected and interested in. These entities often form the key attributes in a news story. Hence, we use named entities to be our major features in deciding whether an input story is on-topic or off-topic.

6.4.1 Designing the Named Entities Set for Topic Tracking

In the message understanding conference (MUC)[3], named entities cover person names, company names, percentage, organization names, location, dates, times and currency. In this work, we choose all of the above named entities except currency to be our extracted entities. The reason of eliminating entity "currency" is that currency is not general to all news stories.

Besides MUC, we also consider the tag set[4] output by our TEL tagger. In this tag set, nouns are divided into 6 types: last name (nf), name of person (npf), name of organization (npu), other proper noun (npr), common noun (ng) and 動名詞 (nvg[5]). Considering both this tag set and MUC, we add two more named entries, npr and nvg into the "named-entities-extracting list". We ignore the common noun from named entities list because many words have this tag.[6]

[3] http://www.muc.saic.com

[4] A complete tag set is provided in Appendix A.

[5] Description of nvg is shown in Section 4.1.

[6] Referring to Section 4.1, about 25.5% of the words are tagged with common noun.

Tag	Meaning	Example Word	Extracted by
nf	last name	王 (Wang)	TEL tagger
npf	name of person	拉莫斯	TEL tagger
npu	name of organization	聯合國 (the United Nation)	TEL tagger
npr	other proper noun	房地產 (real estate)	TEL tagger
nvg	動名詞	決策 (decision)	TEL tagger
s	name of location	中國 (China)	TEL tagger
t	time	5月 (May), 5月11日 (11, May)	TEL tagger OOV identifier
ngm	name of person	萊溫斯基 (Lewinsky)	OOV identifier
ogs	name of location name of organization	阿肯色州 台灣民航局	OOV identifier OOV identifier
oth	others	參加 (join)	- - -

Table 6.5: Kinds of named entities used in topic tracking.

As a result, we have seven named entities. Figure 6.5 shows examples of these named entities. All of the tags in Figure 6.5 is based on the tag set used by TEL tagger. In addition, two tags: ngm (person names) and ogs (location names or organization names) are designed to denote the named entities extracted by unknown words (or OOV) identifier. Moreover, the words which are not named entities will be tagged as "oth".

6.4.2 Feature Selection

Named entities are major target for feature selection. Thus far the named entities have been output by our TEL tagger and OOV identifier (as mentioned in Chapter 5). Ten possible tags (see Table 6.5) which cover eight kinds of named entities are produced. The selection process then proceeds to choose the named entities to enter into the document vector.

Figure ?? illustrates the algorithm of selecting features. First, all the words in training stories are tagged by the TEL tagger and OOV identifier. In step 2, we eliminate all common words with a stopwords list which are described in section 6.5. In step 3 and 4, we classify a words into two kinds: named entities or other words, with its corresponding tag described as Section 6.3.1. We produce 2 word lists called "named entities list" and "other list" .

The two word lists are formed, we sort the named entities list in descending order according to occurrence frequencies in the training stories. Then, we check the occurrence frequencies of these words. If the frequency of a word is higher than the threshold, this word will be inserted a features list, which will be used by our tracker. This selection process is continued until all the words in named entities list are checked or when our features list reaches the maximum length.

If all the words in the named entities list has been entered as features, then we proceed to select features from the "other list". The process of selecting "other" words to formulate classifier is as same as the named entities used. When all the words in other list are checked or feature list is full, the selection process stops and the feature list will be used to represent the topic for tracking.

6.4.3 Integrated with Vector-Space Model

In this thesis, we use the vector-space model for topic tracking. This model has been described in Chapter 2. Here we give a brief description of the vector-space model used in this thesis.

```
1.  Tag the training stories. (TEL)
2.  Eliminate all stopwords in training stories.
3.  Extract all named entities words by its corresponding tag mentioned as Section
     6.4.1 in the "named entities list".
4.  Extract all other words (tagged as "oth") in "other list".
5.  Sort "named entities list" according to its frequency in training stories.
6.  Set the first element of named entities list to be the current word.
7.  If the frequency of the current word exceeds frequency-threshold,
        go to step 8.
    else, go to step 13.
8.  Put current word into feature list.
9.  If the number of features in feature list reaches maximum,
        go to step 22.
10. If no more words in named entities list,
        go to step 13.
11. Set current word to the next word of it.
12. Go to step 7.
13. If no words tagged with "oth",
        go to step 22.
14. Sort "other list" according to its frequency in training stories.
15. Set the first element of other list to be current word.
16. If the frequency of current word exceeds frequency-threshold,
        go to step 17.
    else, go to step 22.
17. Put current word into feature list.
18. If the number of features in feature list reaches maximum,
        go to step 22.
19. If no more words in other list,
        go to step 22.
20. Set current word to next word of it.
21. Go to step 16.
22. Formulate classifier.
```

As mentioned as Section 2.5.1, a story will be transformed into a vector. In the topic tracking task, two vectors are formed during processing: (i) document vector relate to tracking and (ii) query vector relate to tracking. Referring to Section 6.4.3, features of training stories[7] are extracted in Figure ??. These extracted features are then transformed into a document vector. After a document vector is formed, The *incoming* testing story will be transformed into a query vector.

SMART is a widely used information retrieval system. In this thesis, we used SMART's weighting schemes to be our weighting scheme. Table 6.6 illustrates the formulas used in SMART system. These schemes are in term of three components (term frequency, collection frequency and normalization). Each components are denoted by its symbol, for example, term frequency is represented in one of *b, n, a or l*. SMART's weighting schemes are represented a pair of 3-tuples: [tuple for document terms] $\cdot$ [tuples for query term]. For example, $atc \cdot atc$ means the system used augmented normalized term frequency × inverse document frequency for document (first tuple) and query term (second tuple) weights in cosine normalization.

Four of the weighting schemes in SMART system will be applied to this work. These four schemes are: (i) $ntn \cdot ann$, (ii) $ntc \cdot anc$, (iii) $atn \cdot atn$ and (iv) $atc \cdot atc$. All document vector and query vector are represented by term frequency and collection frequency. However, there are only four training stories in topic tracking. In order to utilize the use of inverse document frequency, we estimate the inverse document frequency from 1,000 off-topic

[7] In TDT-3, 4 training stories are provided to tracking task. In this work, we will use the same number of training stories.

Term Frequency		
b	1.0 or 0.0	Term weight equal to 1 if the term present in a vector. Otherwise, term weight equal to 0.
n	*tf*	Term weight equal to term frequency (*tf*) in query.
a	$0.5+0.5\frac{tf}{max\_tf}$	Term weight lies between 0.5 and 1, where *max_tf* is the maximum term frequency in document. This t erm weight also called augmented normalized term frequency.
l	$\ln tf+1.0$	Logarithm term frequency.
Document Frequency		
n	1.0	Document weight equal to 1.
t	$\ln\frac{N}{n}$	Inverse document frequency where N is the total number of documents and n is the number of documents with assigned term. This term weight is also called inverse document frequency.
Normalization		
n	1.0	No normalization.
c	$\frac{1}{\sqrt{\sum_{vector} w_i^2}}$	Cosine normalization.

Table 6.6: Weighting schemes in SMART system [38].

stories which randomly chosen from the training set data.

A similarity value is calculated when comparing document vector (the training stories) to query vector (a testing story). These calculation is based on the four weighting schemes which we mentioned before. For example, $ntc \cdot anc$ is defined as

$$sim(d,q) = \frac{\sum_{i=1}^{n} w_{di} \cdot w_{qi}}{\sqrt{\sum_{i=1}^{n} w_{di}^2} \cdot \sqrt{\sum_{i=1}^{n} w_{qi}^2}} \tag{6.3}$$

where

$w_{di} = \frac{N}{df(i)}$, where mentioned as Table 6.6

$w_{qi} = 0.5 + 0.5 \cdot \frac{tf}{max_t f}$, where mentioned as Table 6.6

6.5 Experimental Results and Analysis

Our experiments are based on disjoint training set and test sets as mentioned in Section 6.2. An official software NIST TDT3 Evaluation Software Version 1.7 is provided by TDT-3 project to evaluate the performance of topic tracking. Each topic has its index file which generated by this software. Each index file includes 4 training stories file index and testing stories file index. A classifier is formed by these training stories and tested in the remainder of testing stories. All training and testing stories are in chronological order.

In topic tracking, each topic is processed individually. After all topics in training set is tracked, the performance will be evaluated on cost function and DET curve which mentioned as Section 6.3.

6.5.1 Notations

For the sake of simplicity, we will adopt the following notations in describing our work:

- W_{scheme}, denotes the weighting schemes used in experiment. The type may adopt the instances $ntn \cdot ann$, $ntc \cdot anc$, $atn \cdot atn$ or $atc \cdot atc$.

- D_n^t, denotes a document vector representing the training stories. The variable t denotes the occurrence frequency threshold for features selected. The variable n denotes the maximum length of a document vector (number of features selected).

- $S(W_{scheme}, D_n^t)$, denotes the overall setting of the experiment.

- θ, denote the threshold of score. If output score of a testing story is greater than θ, this story will be classified as on-topic. Otherwise, this story will be classified as off-topic.

6.5.2 Stopword Elimination

In Chinese text, some of the words are not meaningful (stopwords). For example, 是 (is), 的 (of), 和 (and), 總是 (always) and 爲 (because). However, these words often have high frequencies across different topics. In order to eliminate these stopwords for topic tracking, we create a stopword list based on the following sources:

1. Stopword in the SMART system [53].

2. Tsai 's "Character Frequency Statistic" [9]

3. 1,000 off-topics stories in training set.

Feature selection from the four training stories is by means of the "frequency method". We first eliminate stopwords. Then, we select the n most frequent words to be our representative features in the document vector.

Table 6.7 shows the result of using different stopwords list with setting $S(W_{atn\cdot atn}, D_{20}^2)$ in topic tracking, where T denotes Tsai's "Character Frequency Statistic" and O denotes 1,000 off-topic stories in training set data.

In the beginning, we remove all punctuation marks. Then, we get the evaluation cost 0.3250 by the calculation of cost function mentioned in Section 6.2.1. Next, we add 229 stopwords into SMART's stopword list and get the cost down to 0.1864. In order to improve the coverage of stopwords, we insert the top 10, 20, 50, 100, 200, 300, 500 and 1000 single characters from Tsai's "Character Frequency Statistic[8]" into SMART's stopwords list. The evaluation costs are then improved to 0.1546 and 0.1550 for adding top 200 and 300 single characters into the list.

Moreover, we also tried to eliminate the common words across the off-topic stories in TDT-3. We randomly select 1,000 out of 9,410 off-topic stories from training set data. These 1,000 stories are used to extract common words based on its occurrence frequencies. Five thresholds are set in this work, they are 500, 450, 300, 250 and 200. If the document frequency of a word is higher than a threshold, this word will be counted as stopwords. As a result, we get five sets of word which include 15, 25, 42, 58 and 83 common words (e.g. 今天 and 時) for threshold 500, 400, 300, 250 and 200. Finally, the evaluation

[8] This report is based on the Chinese characters appeared on Usenet newsgroups which consists of 171,882,493 characters during 1993-1994.

	Symbol	Components of word set	# of words	Cost
1.	p	punctuation marks	61	0.3250
2.	s	p + SMART	290	0.1864
3.	s-200	s + top 200 words from T	413	0.1546
4.	s-300	s + top 300 words from T	493	0.1550
5.	s-300-58	s-300 + 58 words from O	518	0.1495

Table 6.7: Evaluation cost for different stopwords lists in the train set with $S(W_{atn\cdot atn}, D^2_{20})$ setting.

Setting	$W_{ntn\cdot ann}, D^4_{10}$	$W_{atn\cdot atn}, D^3_{20}$	$W_{atn\cdot atn}, D^2_{30}$	$W_{atn\cdot atn}, D^2_{40}$
Training	0.1905	0.1444	0.1539	0.1919
Testing	0.3080	0.2968	0.3034	0.2971

Table 6.8: Evaluation cost for both training and test sets, under the condition with optimal setting in different length of document vector.

cost is further improved to 0.1495.

After this stopword list is formed, we performed topic tracking with the frequency method on the training set. Different combinations of $S(W_{atn\cdot atn}, D^2_{20})$ are tested in our experiments. It includes four weighting schemes, four frequency threshold (2,3,4 and 5) and also four possible values of n vectors (10, 20, 30, 40). Optimization based on the setting of $S(W_{atn\cdot atn}, D^2_{30})$ which gives the minimum evaluation cost 0.1444 with θ=3.2. Experimental results for both training and test sets with optimal frequency threshold and document length are tabulated in Table 6.8. These results are the baselines of our following experiments.

Modification	No. of entries in lexicon
At the beginning	8839
Insert punctuation marks and symbols	8878
Insert last names of people	9037
Insert words from word-frequency lexicon	11908

Table 6.9: Number of entries after modification.

6.5.3 TEL Tagging

Preparation of TEL Lexicon

Having acquired the stopword list, we proceeded with our experiments by extracting named entities from TEL tagger. Before we start our experiment, we consider the inconsistency of lexicons used in a Chinese segmenter and our TEL tagger. The lexicon used in segmentation process has 44,405 entries while the lexicon used in TEL tagging only has 8,839 entries. To tackle this problem, we insert the "named-entities-word" from a "segmentation-lexicon" into our "tagging-lexicon". This insertion process ensures all named-entities in "segmentation-lexicon" are in "tagging-lexicon".

We insert a set of tags for all punctuation marks into our word-tag lexicon. This ensures that the punctuation symbols will not be wrongly defaulted to the *ng* tag. To facilitate the tagging of person names, we insert the most popular family names mentioned as Chapter 5 into our TEL tagger. In Figure 6.9, we show the number of entries in "tagging-lexicon" during the above modifications. Hence we adopt this modified lexicon for our experiments.

Results

We have examined our performance on topic tracking with our extracted named entities. Named entities are extracted as described previously. Our proposed method for feature selection is applied to generate a document vector. Figure 6.2 shows the DET curves for the results in test sets. The DET curve with applying TEL tagger outperforms the other curves (with and without stopwords elimination). The shape of the DET curves "without stopword elimination" is in concave. It may caused by the scores of off-topic stories which are concentrated in particular range. When the similarity score threshold θ is out of this range, the miss or false alarm probabilities will be increased/decreased significantly.

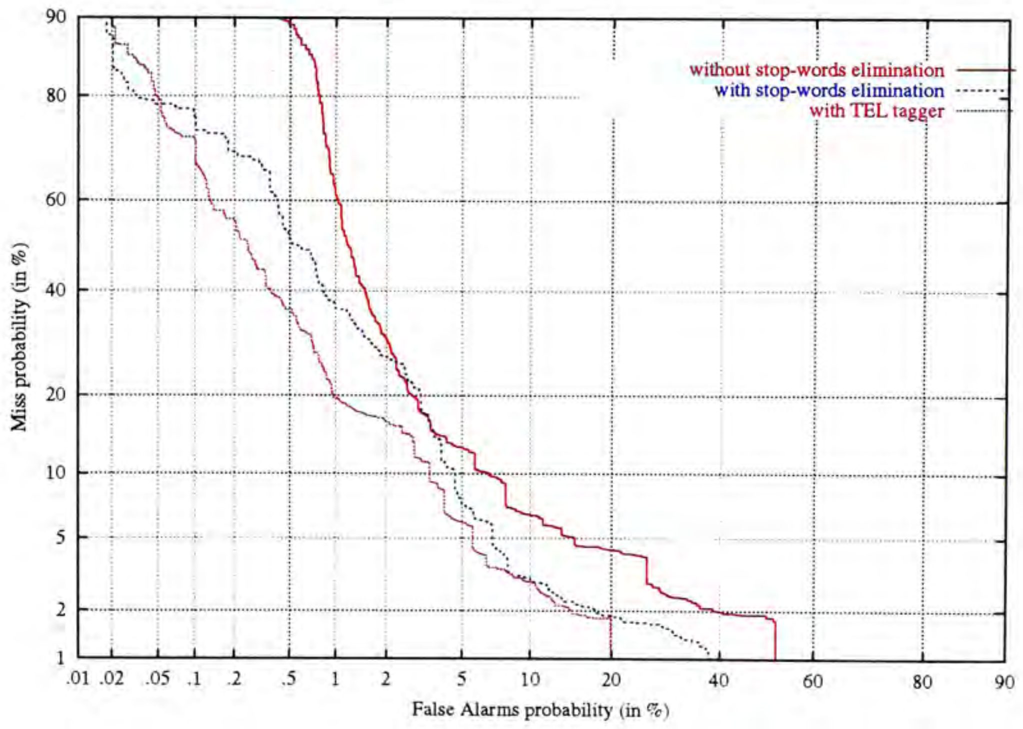


Figure 6.2: DET curves for applying TEL tagger in test set.

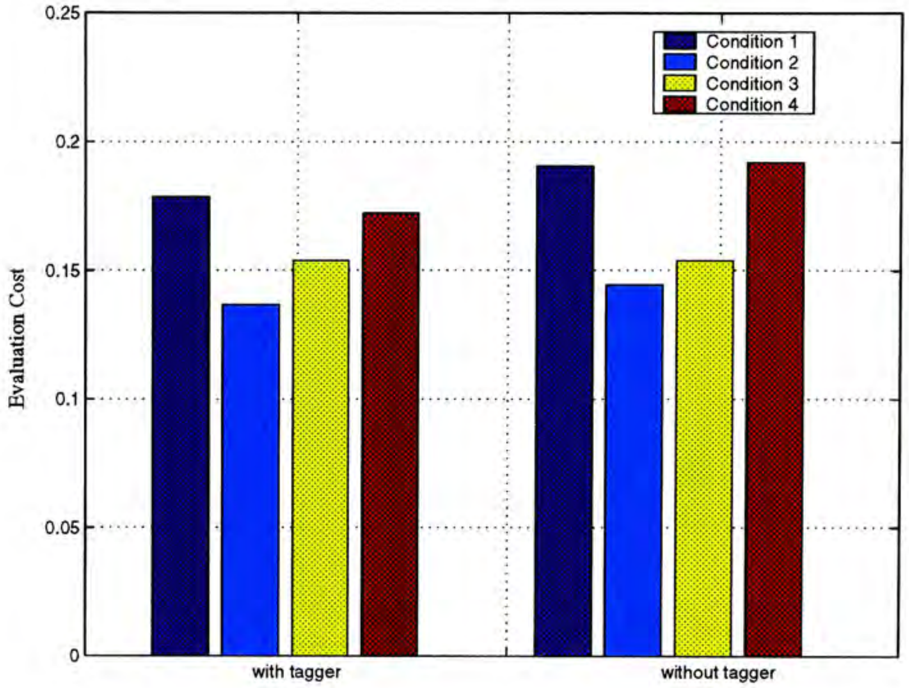


Figure 6.3: Evaluation cost for training set (with and without tagger), under the optimal condition. (Condition 1: $W_{atn\cdot atn}, D^4_{10}$, Condition 2: $W_{atn\cdot atn}, D^3_{20}$, Condition 3: $W_{atn\cdot atn}, D^2_{30}$, Condition 3: $W_{atn\cdot atn}, D^2_{40}$)

Figure 6.3 and 6.4 show the evaluation costs for training and test sets. The score θ used in test set is the optimal score threshold obtained from training set. The performance of using TEL tagger gives the best result (0.2419) in condition $W_{atn\cdot atn}, D^3_{20}$. However, the evaluation cost is increased when the length of a document vector (the number of features) increased. One reason is due to the "dilution" effect of "noise" words. The longer the document vector, the more words included. Some less useful words may be inserted into a document vector. The scores obtained from these less useful words will dilute the importance of the keywords. For example, the features extracted from training stories in topic 20071 with $W_{atn\cdot atn}, D^3_{20}$ are:

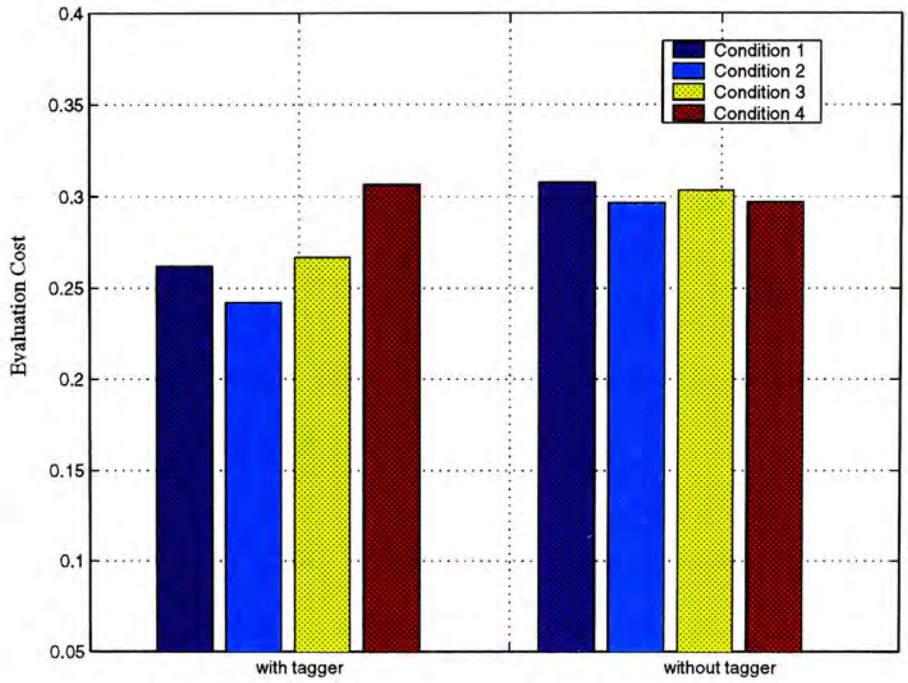


Figure 6.4: Evaluation cost for test set (with and without tagger), under the optimal condition. (Condition 1: $W_{atn \cdot atn}, D_{10}^{4}$, Condition 2: $W_{atn \cdot atn}, D_{20}^{3}$, Condition 3: $W_{atn \cdot atn}, D_{30}^{2}$, Condition 3: $W_{atn \cdot atn}, D_{40}^{2}$)

Person names (2)	:	羅斯, 阿拉法特
Location names (7)	:	倫敦, 美國, 中東, 華盛頓, 以色列, 約旦河 and 巴勒斯坦
Others (11)	:	塔, 尼, 亞, 胡, 會談, 舉行, 進程, 總理, 西岸, 和平 and 奧爾布萊特(wrongly tagged)

When we change the condition from D^3_{20} to D^2_{30}, ten more words are included in a document vector:

方, 頓, 林, 階段, 實施, 宣布, 會晤, 議會, 萊特 and 國務卿

Some of the inserted words seem common to other topics, such as 方, 頓 and 林. It may increase the false alarm error in topic tracking. Here is the comparison between D^3_{20} and D^3_{30} in topic 20071:

Condition	P_{Miss}	P_{FA}	C_{norm}
D^3_{20}	0.0000	0.0101	0.0497
D^2_{30}	0.0000	0.0405	0.1986

Both of the conditions keep the miss rate equals to zero. However, false alarm error in D^3_{30} is higher than D^3_{20} (0.0101 and 0.0405). The increased rate of error may caused by the nine inserted words which are more common to other off-topic stories.

Adjusting to an "optimal" length of document vector is difficult task since each topic has its optimal number of representative words. However, we believe most of the important information are existed in the scope of named entities. In this example, our TEL tagger captures ten named entities with D^3_{20}. All of these named entities seem useful to represent the topic.

Some of the words in D^3_{20} seem to be little use, such as the four single characters “塔”, “尼”,“ 亞”and “胡”. In a real case, these words combined with an eliminated stopwords “內” is a person name “內塔尼亞胡”. Unfortunately, this person name is an OOV. A Chinese segmenter tokenizes this person name into a sequence of characters. To tackle this problem, we develop an unknown word identifier to capture these “missed ” named entities. In next section, we discuss the performance of applying our unknown word identifier in topic tracking.

6.5.4 Unknown Word Identifier

As mentioned in the previous section, named entities in the TDT-3 corpus may be OOV. To tackle this problem, we combined our unknown words identifier with TEL tagger to perform topic tracking. Figure 6.5 shows the DET curves for the result in the test set. The evaluation cost for the training and test sets are shown in Figure 6.6 and Figure 6.7. The combination of OOV identifier and TEL tagger brought about a small improvement in test set (from 0.2620 to 0.2205) with $W_{ntn\cdot ann}, D^4_{10}$. The evaluation cost is increased when the number of features is increased. This may caused by the “dilution” effect which we described in last section.

- **Threshold of Frequency**

One point we consider in this experiment is the threshold of occurrence frequency. Since our named entities extracted by the OOV identifier is dependent on the occurrence frequency threshold. For example, when we set

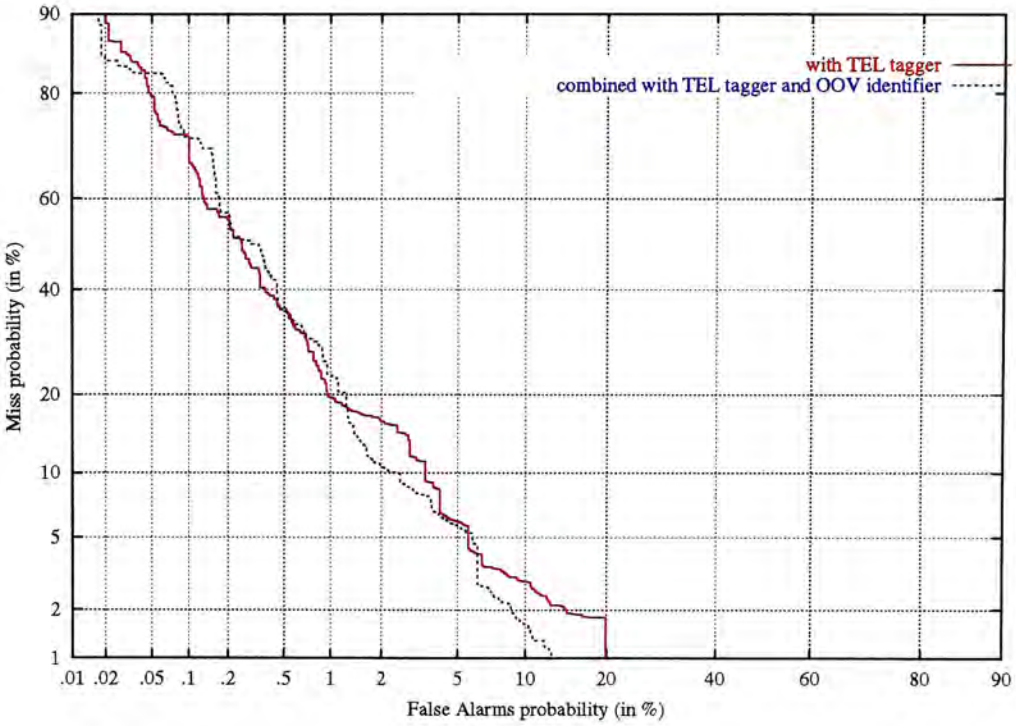


Figure 6.5: DET curves for combining OOV identifier with TEL tagger in test set.

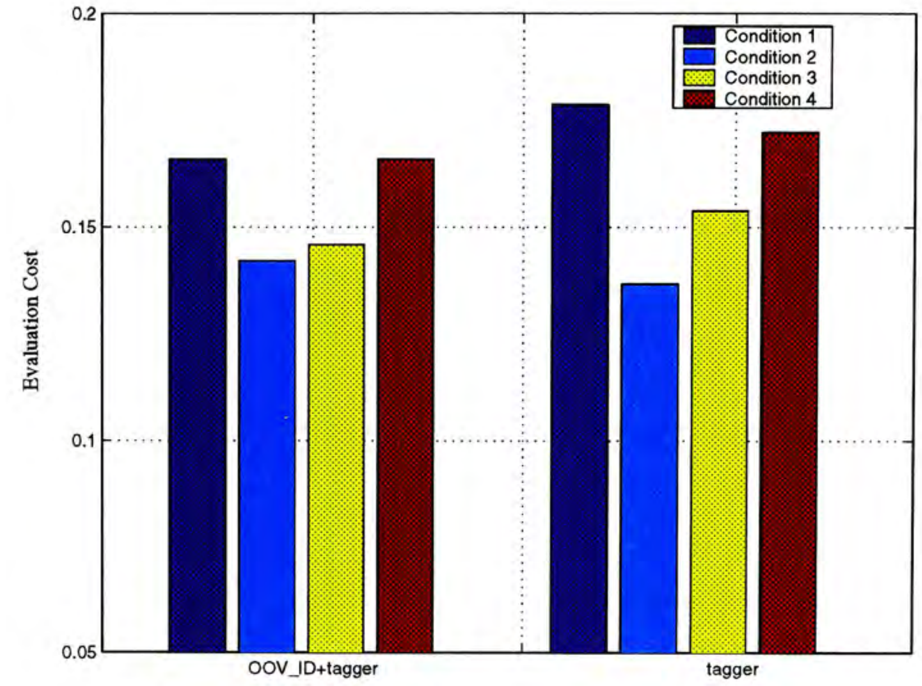


Figure 6.6: Evaluation cost for training set. Comparison between the TEL tagger and the combination of OOV identifier and TEL tagger. (Condition 1: $W_{ntn\cdot ann}, D_{10}^{4}$, Condition 2: $W_{atn\cdot atn}, D_{20}^{3}$, Condition 3: $W_{atn\cdot atn}, D_{30}^{2}$, Condition 4: $W_{atn\cdot atn}, D_{40}^{2}$)

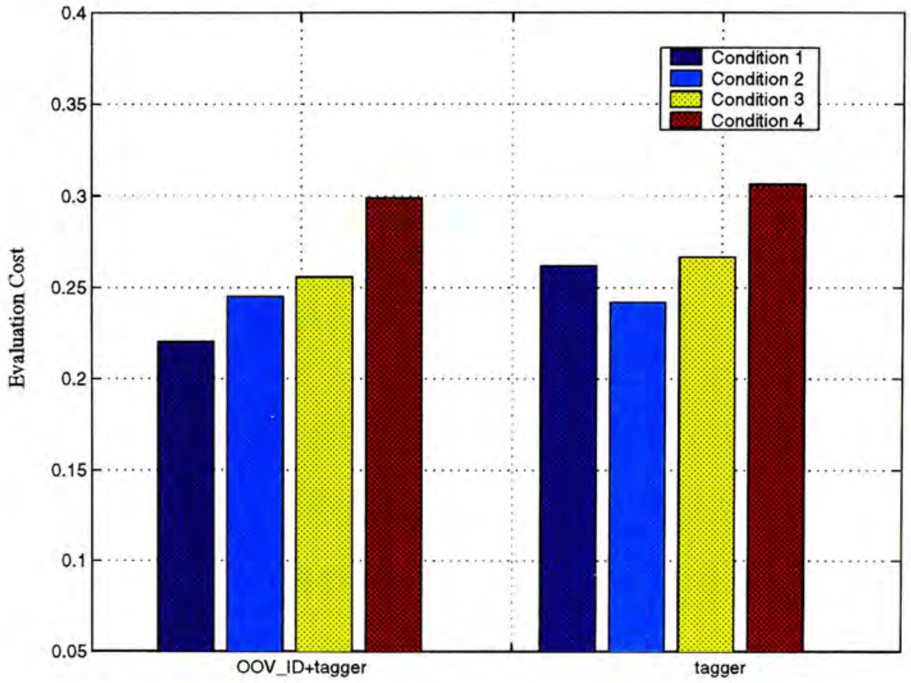


Figure 6.7: Evaluation cost for test set. Comparison between the TEL tagger and the combination of OOV identifier and TEL tagger. (Condition 1: $W_{ntn \cdot ann}, D_{10}^{4}$, Condition 2: $W_{atn \cdot atn}, D_{20}^{3}$, Condition 3: $W_{atn \cdot atn}, D_{30}^{2}$, Condition 4: $W_{atn \cdot atn}, D_{40}^{2}$)

Frequency	2	3	4	5
D^n_{10}	0.3178	0.1874	*0.1660*	0.1676
D^n_{20}	0.1587	*0.1422*	0.2167	0.1837
D^n_{30}	*0.1539*	0.1759	0.2050	0.1891
D^n_{40}	*0.1660*	0.1826	0.2310	0.1891

Table 6.10: Evaluation cost varies with frequency tuned.

the frequency threshold to 2, seven named entities are suggested by OOV identifier in topic 20057:

- 高難(2), 關穎珊(6), 關穎珊曾獲(2), 明尼阿波利(3), 普魯申科(3), 申雪(4) and 趙宏博(2)

"高難" and "關穎珊曾獲" are the "noisy" entities. Both have the occurrences of frequency equal 2 in training stories. When we set the occurrence frequency threshold to 3, these two entities will be eliminated. However, a named entities "趙宏博" (name of person) with occurrence 2 will be eliminated at the same time. Therefore, the determination of frequency threshold is a tradeoff between the named entities extraction and noise elimination.

In topic tracking, we use the overall performance to decide the value of occurrence frequency threshold. Table 6.10 shows the performance of difference frequency thresholds with different length of document vector. These results suggest that higher frequency threshold should be applied in a shorter document vector especially for D^n_{10}. However, the lower frequency threshold is better to apply in a longer document vector.

- **Effect of Feature Selection**

In the problem of unknown words, a Chinese segmenter will tokenize an OOV into a sequence of characters. A word may be decomposed into several characters. If the frequencies of decomposed word is higher than frequency threshold, all of these words may be selected into a document vector. For example, the features (with condition D_{10}^4) extracted from training stories (Topic : 20002) are shown in the follows:

Person names (1)	:	瓊斯
Location names (2)	:	白宮, 美國
動名詞 (1)	:	調查
Others (6)	:	頓, 基, 萊, 林, 塔 and 溫

It seems that some of the words are decomposed into the six single characters ("頓", "基", "萊", "林", "塔" and "溫"). "溫" and "林" are the last names of Chinese. "基" and "頓" may be combined to form a foreign name "基頓".

When we apply our OOV identifier in this topic, the features extracted from training stories will be change to:

Person names (1)	:	瓊斯
Location names (2)	:	白宮, 美國
動名詞 (1)	:	調查
N-gram candidates (3)	:	克林頓, 萊溫斯基 and 斯塔爾
Others (3)	:	供, 總統 and 醜聞

Our OOV identifier recovers three "missed" named entities ("克林頓", "萊溫斯基" and "斯塔爾") in the previous example. Table 6.11 shows the

Condition	θ	miss rate	false alarm rate	cost
TEL	2.5	0.2500	0.0290	0.3923
tagger	*3.8	0.2500	0.0062	0.2806
combined with	2.4	0.0000	0.0342	0.1677
OOV identifier	*3.2	0.0000	0.0117	0.0574

Table 6.11: Performance in Topic 20002. * denotes optimal θ.

performance for with and without OOV identifier. It is clear that our OOV identifier brings the evaluation cost decreases significantly in a overall score threshold θ.

Moreover, we tune the optimal θ (please see Figure 6.11) in each case to observe the optimal result from different features. In applying OOV identifier, the evaluation cost is dropped from 0.1677 to 0.0574 when θ changed from 2.4 to 3.2. The evaluation cost for no OOV identifier is improved from 0.3923 to 0.2806. However, its miss rate are stopped in 0.25. One reason may the single segmented words (頓, 基, 萊, 林, 塔 and 溫) are common words to other topic. It makes the task of discrimination between on-topic and off-topic becomes difficult. In our approach, we group these "noise" words into the useful concept (named entities). That not only eliminate the noise, it also utilize the possible data in training stories.

6.5.5 Error Analysis

There are two possible sources of errors for topic tracking in Chinese. (i) synonym or phrases with the same meaning and (ii) a "broadly" or "vaguely" defined topic. In the following, we discuss it in detail.

- **Variation in word used**

In Chinese text, different words may denote same meaning. For example, Indonesia can be represented in "印度尼西亞" or "印尼", financial crisis can be represented in "金融危機" or "經濟危機". The variation of word brings the difficulty to feature representation since the usage of words are varied from time to time.

In topic tracking, this problems may led the extracted keywords become useless. For example, "印尼"(Indonesia) is a extracted word from training stories (topic 20076). Unfortunately, some of the on-topic stories in testing set used "印度尼西亞" to denote Indonesia. Our tracker cannot accommodate this variation even though the feature selection process extracted the appropriate features, the tracker cannot make good use of them. The absence of this term lower the output score of some on-topic stories. It may cause the on-topic story judged to be off-topic owing to the decreased output score, especially for a document vector which has the short length (i.e. W_{10}^{n}).

- **"Broadly defined" topics**

In TDT-3 corpus, some of the topic may be broadly defined. For example, topic 20001 talks about "financial crisis in Asia" which has a large scope. For example, "the financial crisis in Indonesia" and "the financial reveal in Korea" are all classified in this category. We make errors on the topic above. If we exclude this topic from our data and observe the overall results, the evaluation cost is improved significantly. Table 6.12 shows the evaluation costs after topic 20001 is eliminated. There are 13 topics in training set

Condition	#of topics	θ	miss rate	false alarm rate	cost
Training	13	2.5	0.0649	0.0132	0.1294
Testing	11	–	0.0496	0.0279	0.1865

Table 6.12: Evaluation costs for both training and test sets, under the condition without topic 20001.

and 11 topics in test set after the exclusion process. The "optimal score" is then tuned in training data and set it from 0.24 to 0.25. As a result, the performance of the evaluation costs improved significantly from 0.1660 to 0.1294 for training. In the test set, the evaluation cost is improved from 0.2205 to 0.1865.

Our experiment suggests that the definition of topic is important for topic tracking. If the scope of topic is too general, the performance of topic tracking suffers degradation.

6.6 Chapter Summary

This work explores the use of a "Named Entity" approach for topic tracking in Chinese text. Our approach is unique in that we focus on the use of named entities, because we believe named entities are the key to the task of topic tracking.

The TDT-3 corpus is used in this work. This corpus has designated training and test sets. Two official evaluation methods are employed based on miss and false alarms. In addition, we used four of the weighting schemes in SMART system to be our method to calculate similarity between a testing story and the training stories.

In the beginning, we generate a stopwords list which gives the best performance in "frequency selection" method. The performances for this approach achieved 0.1444 and 0.2968 evaluation cost (C_{norm}) with $W_{atn\cdot atn}, D^3_{20}$ for training and testing. Using the TEL tagger, we extract named entities to formulate a document vector. This proposed approach improve the evaluation cost to 0.1368 and 0.2419 with $W_{atn\cdot atn}, D^3_{20}$ for training and testing. Finally, we combined the unknown word identifier with TEL tagger, this approach further improves the result to 0.2205 with $W_{ntn\cdot ann}, D^4_{10}$ for testing set.

Furthermore, we discuss two possible source errors in topic tracking. One is the variation of words used in the stories (i.e. synonyms or phrases with the same meaning). Two is the "broadly defined" topic. We ran an experiment to show that the evaluation cost is improved to 0.1294 for training and 0.1865 for testing if a "broad-defined" topic is eliminated.

To be concluded, our experiment results suggest that the "Name Entity" approach is applicable for topic tracking in Chinese.

Chapter 7

Conclusions and Future Work

7.1 Conclusions

In this thesis we have described our "Named Entity" approach to topic tracking in Chinese. We transform a topic into a series of concept (named entities). Then using these named entities to perform topic tracking.

There are two research goals in this research. (i) The extraction of named entities. (ii) The use of these named entities for task of topic tracking. In this thesis, we have developed different techniques to achieve the previous goals.

For named entities extraction, we proposed the TEL techniques to capture useful named entities. We applied the Transformation-based Error-Driven Learning (TEL) approach to POS tagging for Chinese text. In Chapter 4, we demonstrate how the TEL approach can be applied to POS tagging in Chinese text. Using a Chinese news corpus of over 70,000 words, divided into disjoint training and test sets of a 9:1 ratio, we achieved overall tagging accuracies of 94.56% (training) and 86.87% (testing). We compared our work with a stochastic tagger and the experimental results suggest that TEL is equally effective and applicable for Chinese.

Tokenization ambiguities and unknown words are two problems one encounters in finding named entities. In Chapter 5, we combined an n-gram grouping together with heuristics for detecting and identifying unknown words. We described how our approach extracting the named entities lost due to tokenization errors and the unknown word problems.

Once the named entities has been extracted, we used them for the task of topic tracking. In Chapter 6, we integrated the selected named entities extraction with vector space model. The TDT-3 corpus and two evaluation methodologies (cost function and DET graph) are used in experiments. Varied with four weighting schemes in SMART system, we used the simple "frequency selection" method which achieves 0.1444 and 0.2968 evaluation cost for training and testing (This result is our baseline performance). Applying our TEL tagger with feature selection, we improve the evaluation cost to 0.1368 and 0.2419 for training and testing. Combined with our unknown word identifier, we further improve the evaluation cost to 0.2205 for the testing data. The experimental results suggest that our "Named Entity" approach is applicable for topic tracking in Chinese.

7.2 Future Work

There are a number of possible directions where in extending this research:

1. Our results suggest that named entities are the important features to

determine the topic. However, the method for assigning weight to named entities are depended on term frequency and inverse document frequency. It is as same as other "non-named entities". We propose to add the higher weight to named entities. The weight assignment between named entities and other words may improve the performance of topic tracking.

2. In our work, we consider single words of named entity to be our features. However, the relationship between different named entities (co-occurrence) may provide important information to topic tracking. For example, <克林頓 (Clinton), 萊溫斯基 (Lewinsky)>, <萊溫斯基 (Lewinsky), 斯塔爾 (Starr)>, <巴勒斯坦 (Palestine),以色列 (Israel)>. It would be useful if we could merge these co-occurrence words into the score calculation.

3. Besides the vector-space model, we can also use other information retrieval models (e.g. HMM models) for topic tracking. The performance of the system may be more stable and accurate based on a combined decision from multiple models. Also the combined decision may reduce the effects on data sensitivity.

Bibliography

[1] 白明弘, 陳超然, and 陳克健. 以語境判定未知詞詞類的方法. In *Proceedings of ROCLING XI*, 1998.

[2] 孫茂松, 黄昌寧, 高海燕, and 方捷. 中文姓名的自動辨識. In *Communications of COLIPS, Vol. 4, No.2*, 143-149.

[3] A.Martin, G.Doddington, T.Kamm, M.Ordowski, and M.Przybocki. The DET Curve in Assessment of Detection Task Performance. In *EuroSpeech '97, Vol. 4*, pages 1895–1898. European Speech Communication Association (ESCA), 1997.

[4] A.Qin and W.S.Wong. ACCESS: Automatic Segmentation and Part of Speech Tagging of Chinese Text. In *Technical report, the Chinese University of Hong Kong*, 1998.

[5] A.Ratnaparkhi. A Maximum Entropy Model for Part-Of-Speech tagging. In *Proceedings of the Empirical Methods in Natural Language Processing Conference*, pages 131–137, University of Pennsylvania,U.S.A., 1996.

[6] Bai, S.H.Y., Huang, and C.N. Automatic Part of Speech Tagging System for Chinese. In *Technical Report, Tsinghua University*, Beijing, China, 1992.

[7] B.A.N.Ribeiro and R.Muntz. A Belief network Model for IR. In *Proceedings of the 19th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, 1996.

[8] B.B.Greene and G.M. Rubin. Automated Grammatical Tagging of English. In *Department of Linguistics, Brown University, Providence, Rhode Island*, 1971.

[9] C.H.Tsai. Frequency and Stroke Counts of Chinese Characters. 1996.

[10] C.H.Tsai. Frequency Counts of Chinese Names. 2000.

[11] C.J.Chen, M.H.Bai, and K.J.Chen. Category Guessing for Chinese Unknown Words. In *Proceedings of the Natural Language Processing Pacific Rim Symposium*, pages 35–40, Thailand, 1997.

[12] D.Graff, C.Cieri, S.Strassel, and N.Martey. The TDT-3 Text and Speech Corpus. In *Proceedings of Topic Detection and Tracking Workshop*, 2000.

[13] D.Hindle. Acquiring Disambiguation Rules from Text. In *Proceedings of the 27th Annual Meeting of the Association for Computational Linguistics*, pages 118–125, 1989.

[14] D.W.Oard. Topic Tracking with the PRISE Information Retrieval System. In *Proceedings of the DARPA Broadcast News Workshop*, 1999.

[15] D.Y.Shen and M.S.Sun. The application and implementation of local statistics in Chinese unknown word identification. In *Communications of COLIP8*, pages 119–128, 1998.

[16] E.Black, A.Finch, and H.Kashioka. Trigger-Pair Predictors in Parsing and Tagging. In *Proceedings of the International Conference on Computational Linguistics*, pages 131–137, 1998.

[17] E.Brill. A simple rule-based part of speech tagger. In *Proceedings of the Third Conference on Applied Natural Language Processing*, 1992.

[18] E.Brill. Automatic grammar induction and parsing free text: Atransformation-based approach. In *Proceedings of the 31st Meeting of the Association of Computational Linguistics*, Columbus,Oh., 1993.

[19] E.Brill. A Corpus-Based Approach to Language Learning. In *Ph.D. thesis,Department of Computer and Information Science, University of Pennsylvania*, 1993.

[20] E.Brill. Transformation-based error-driven parsing. In *Proceedings of the Third International Workshop on Parsing Technologies*, Tilburg, the Netherlands, 1993.

[21] E.Brill. Some advances in rule-based part of speech tagging. In *Proceedings of the Twelfth National Conference on Artificial Intelligence (AAAI-94)*, Seattle,Wa, 1994.

[22] E.Brill. Transformation-based Error-driven Learning and Natural Language Processing: A Case Study in Part of Speech Tagging. In *Computational Linguistics, Vol.21*, 1995.

[23] E.Brill. Unsupervised Learning of Disambiguation Rules for Part of Speech Tagging. In *Natural Language Processing Using Very Large Corpora Kluwer Academic Press.*, 1997.

[24] E.Brill and R.Philip. A transformation-based approach to prepositional phrase attachment disambiguation. In *Proceedings of the Fifteenth International Conference on Computational Linguistics (COLING-1994)*, Kyoto, Japan, 1994.

[25] E.Mittendorf and P.Schauble. Document and Passage Retrieval Based on Hidden Markov Models. In *Proceedings of the Seventeenth Annual International ACM-SIGIR Conference on Research and Development in Information Retrieval*, 1994.

[26] E.Peterson. A chinese named entity extraction system. 1997.

[27] F.Walls, H.Jin, S.Sista, and R.Schwartz. Probabilistic Models for Topic Detection and Tracking. In *ICASSP99*, 1999.

[28] G.Salton, A.Wong, and C.S.Yang. A Vector Space Model for Automatic Indexing. In *Communication of ACM, Vol. 18, No. 11*, pages 613–620, 1975.

[29] G.Salton and M.J.McGill. In *Introduction to Modern Information Retrieval*, New York, 1983. McGraw-Hill.

[30] H.H.Chen and J.C.Lee. The Identification of Organization Names in Chinese Texts. In *Communication of COLIPS,4.2*, pages 131–142, 1994.

[31] H.H.Chen, Y.W.Ding, and S.C.Tsai. Named Entity Extraction for Information Retrieval. In *Computer Processing of Oriental Languages, Vol. 12, No.1*, 1998.

[32] H.Jin, R.Schwartz, S.Sista, and F.Walls. Topic Tracking for Radio TV Broadcast and Newswire. In *Proceedings of the DARPA Broadcast News Workshop*, 1999.

[33] H.Schmid. Part-of-speech tagging with neural networks. In *Proceedings of the International Conference on Computational Linguistics*, pages 172–176, 1994.

[34] J.Allan, J.Carbonell, G.Doddington, J.Yamron, and Y.Yang. Topic Detection & Tracking Pilot Study: Final Report. In *Proceedings of DARPA Broadcast News Transcription and Understanding Workshop*, 1998.

[35] J.Allan, V.Lavrenko, and R.Papka. Event Tracking. In *CIIR technical report, UMASS Computer Science Department*, 1999.

[36] J.Carbonell, Y.Yang, J.Lafferty, R.Brown, T.Pierce, and L.Xiu. Cmu Report on TDT2: Segmentation Detection and Tracking. In *Proceedings of the DARPA Broadcast News Workshop*, 1999.

[37] J.Fiscus, G.Doddington, J.Garofolo, and A.Martin. Nist's 1998 Topic Detection and Tracking Evaluation (tdt2). In *Proceedings of the DARPA Broadcast News Workshop*, 1999.

[38] J.H.Lee and J.S.Ahn. Using n-Grams for korean text retrieval. In *Proceedings of the 19th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 216–224, 1996.

[39] J.Hockenmaier and C.Brew. Error-Driven Learning of Chinese Word Segmentation. In *12th Pacific Conference on Language and Information*, 1998.

[40] J.M.Schdultz and M.Liberman. Topic Detection and Tracking using idf-Weighted Cosine Coefficient. In *Proceedings of DARPA Broadcast News Workshop*, 1999.

[41] J.M.Schultz and M.Liberman. Cross-Lingual Topic Tracking using idf-Weighted Cosine Coefficient. In *Proceedings of Topic Detection and Tracking Workshop*, 2000.

[42] J.P.Yamron, I.Carp, L.Gillick, S.Lowe, and P.van Mulbregt. Topic Tracking in a News Stream. In *Proceedings of DARPA Broadcast News Workshop*, 1999.

[43] J.P.Yamron, I.Carp, L.Gillick, S.Lowe, and P.van Mulbrget. Topic tracking in a new stream. In *Proceedings of the DARPA Broadcast News Workshop*, 1999.

[44] J.P.Yamron, L.Gillick, S.Knecht, S.Lowe, and P.van Mulbregt. Statistical Models for Tracking and Detection. In *Proceedings of Topic Detection and Tracking Workshop*, 2000.

[45] J.S.Chang and et al. Recognizing unregistered names for Mandarin word identification. In *Proceedings of 14th International Conference on Computational Linguistics*, 1992.

[46] J.S.Chang, S.D.Chen, S.J.Ker, Y.Chen, and J.Liu. A Multiple-Corpus Approach to Recognition of Proper Names in Chinese Texts. In *Communications of COLIPS, 4.2*, pages 75–85, 1994.

[47] J.Y.Nie, M.L.Hannan, and W.Jin. Unknown Word Detection and Segmentation of Chinese using Statistical and Heuristic Knowledge. In *Communications of COLIPS*, 1995.

[48] M.H.Bai K.J.Chen. Unknown Word Detection for Chinese by a Corpus-based Learning Method. In *Computational Linguistics and Chinese Language Processing, Vol.3, No.1*, pages 27–44, February 1998.

[49] J. Kupiec. Robust Part-of-Speech Tagging using a Hidden Markov Model. In *Computer Speech and Language*, pages 6:226–242, 1992.

[50] LDC. Topic Detection and Tracking - Phase 3 (TDT-3) Overview. 1998.

[51] LDC. The 1999 Topic Detection and Tracking (TDT3) Task Definition and Evaluation Plan. 1999.

[52] LDC. TDT3 Annotation Guide. 1999.

[53] L.F.Chien and H.T.Pu. Important Issues on Chinese Information Retrieval. In *Computational Linguistics and Chinese Language Processing, Vol.1, No.1*, pages 205–221, 1996.

[54] L.J.Wang, W.C.Li, and C.H.Chang. Recognizing unregistered names for Mandarin word identification. In *Proceedings of 14th International Conference on Computational Linguistics*, pages 1239–1243, 1992.

[55] L.Mangu and E.Brill. Automatic Rule Acquisition for Spelling Correction. In *ICML 97*, 1997.

[56] K.T. Lua. Part of Speech Tagging of Chinese Sentences using Genetic Algorithm. In *International Conference on Chinese Computing*, pages 45–49, 1996.

[57] Charles L.Wayne. Topic Detection & Tracking (tdt) Overview & Perspective. In *Proceedings of DARPA Broadcast News Transcription and Understanding Workshop*, 1998.

[58] M.C.Wang, K.J.Chen, and C.R.Huang. The identification and classification of unknown words in Chinese: A N-Gram approach. In *Proceedings of PAcFocol2*, pages 17–31, 1994.

[59] M.L.Meng and C.W.Ip. An Analytical Study of Transformational Tagging for Chinese Text. In *Research on Computational Linguistics Conference XII (ROCLING XII)*, Taipei, Taiwan ROC, 1999.

[60] M.Y.Lin, T.H.Chiang, and K.Y.Su. A Preliminary Study on Unknown Word Problem in Chinese Word Segmentation. In *Proceedings of ROCLING VI*, pages 31–48, 1993.

[61] J.Y. Nie and Martin Brisebois. On Chinese Text Retrieval. In *Proceedings of the 19th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 225–233, Zurich, Switzerland, 1996.

[62] D.W. Oard and G.Marchionini. A Conceptual Framework for Text Filtering. In *Journal User Modeling and User-Adapted Interaction*, 1997.

[63] C.J.van Rijsbergen. In *Information Retrieval.* Butterworths, 1979.

[64] R.Papka. On-Line New Event Detection, Clustering, and Tracking. In *Ph D Dissertation of the University of Massachusetts Amherst*, 1999.

[65] R.Sproat, C.Shih, W.Gale, and N.Chang. A Stochastic Finite-State Word-Segmentation Algorithm for Chinese. In *Computational Linguistics,22,3*, pages 131–145, 1996.

[66] S.A.Lowe. The Beta-Binomial Mixture Model and Its Application toTDT Tracking and Detection. In *Proceedings of the DARPA Broadcast News Workshop*, 1999.

[67] S.A.Lowe. The Beta-binominal Mixture Model for Word Frequencies in Documents with Applications to Information Retrieval. In *Proceedings of Eurospeech'99*, Budapest, 1999.

[68] S.E.Robertson, van Rijsbergen, and M.F.Porter. Probabilistic model of indexing and searching. In *Information retrieval research*, 1981.

[69] S.Giorgio and E.Brill. Efficient Transformation-Based Parsing. In *ACL 1996*, 1996.

[70] S.H.Liu, K.J.Chen, L.P.Chang, and Y.H.Chin. Automatic Part-of-Speech Tagging for Chinese Corpora. In *Computer Processing of Chinese and Oriental Languages,vol. 9, no. 1, June*, pages 31–47, 1995.

[71] T.H.Chiang and et. al. Statistical models for segmentation and unknown word resolution. In *5th R.O.C. Computational Linguistics Conference*, pages 123–146, 1992.

[72] T.Leek, H.Jin, S.Sista, and R.Schwartz. The BBN Crosslingual Topic Detection and Tracking System. In *Proceedings of Topic Detection and Tracking Workshop*, pages 123–127, 2000.

[73] W.B.Cavnar. n-gram-based text filtering for trec-2. In *Proceedings of the Second Text REtrieval Conference (TREC-2)*, pages 171–179, 1994.

[74] W.Jin and L.Chen. Identify Unknown Words in Chinese Corpus. In *Proceedings of the Third Natural Language Processing Pacific-Rim Symposium, Vol.1*, pages 234–239, Seoul, Korea, 1995.

[75] K.F. Wong, K.C.Chan, and C.H. Cheng. An Investigation on Transformation-based Error-driven Learning Algorithm for Chinese Noun Phrase Extraction. In *2000 International Conference on Chinese Language Computing*, Chicago, July,2000.

[76] Y.Yang, T.Pierce, T.Ault, and J.Carbonell. Combining multiple learning strategies to improve tracking and detection performance. In *Proceedings of Topic Detection and Tracking Workshop*, 2000.

Appendix A

The POS Tags

1.	nf	姓氏	38.	qng	名量詞"個"	75.	e	嘆詞
2.	npf	人名	39.	qnm	度量詞	76.	hm	數詞前綴
3.	npu	機構名	40.	qns	不定量詞	77.	hn	名詞前綴
4.	npr	其它專名	41.	qnv	容器量詞	78.	k	后綴
5.	nvg	動名詞	42.	qnf	成形量詞	79.	i	成語
6.	ng	普通名詞	43.	qnt	臨時量詞	80.	j	簡稱語
7.	t	時間詞	44.	qnz	准量詞	81.	l	習用語
8.	s	處所詞	45.	qvp	專用動量詞	82.	x	其他
9.	f	方位詞	46.	qvn	名動量詞	83.	xch	非漢字
10.	vg	一般動詞	47.	rn	體詞性代詞	84.	xfl	數學公式
11.	vgo	不帶賓	48.	rp	謂詞性代詞	85.	、	、
12.	vgn	帶體賓	49.	rd	副詞性代詞	86.	，	，
13.	vgv	帶動賓	50.	p	介詞	87.	。	。
14.	vga	帶形賓	51.	pba	把(將)	88.	；	；
15.	vgs	帶小句賓	52.	pbei	被(讓,叫)	89.	：	：
16.	vgd	帶雙賓	53.	pzai	在	90.	！	！
17.	vgj	帶兼語賓	54.	d	副詞	91.	？	？
18.	va	助動詞	55.	cf	連詞前段	92.	（	（
19.	vc	補語動詞	56.	cpw	幷連詞	93.	）	）
20.	vi	系動詞	57.	cpc	幷連分句	94.	“	“
21.	vy	動詞"是"	58.	cps	幷連句子	95.	”	”
22.	vh	動詞"有"	59.	cbc	分句詞語間	96.	〔	〔

23.	vv	來、去連謂	60.	cbs	句子間	97.	〕	〕
24.	a	形容詞	61.	usde	”的”	98.	《	《
25.	z	狀態詞	62.	uszh	”之”	99.	》	》
26.	b	區別詞	63.	ussi	”似的”	100.	〈	〈
27.	mx	系數詞	64.	usdi	”地”	101.	〉	〉
28.	mw	位數詞	65.	usdf	”得”	102.	……	……
29.	mg	概數詞	66.	ussu	”所”	103.	‘	‘
30.	mf	分數詞	67.	ussb	”不”	104.	’	’
31.	mb	倍數詞	68.	utl	”了”	105.	——	——
32.	mm	數量詞	69.	utz	”著”	106.	／	／
33.	mh	數詞”半”	70.	utg	”過”	107.	～	～
34.	mo	數詞”零”	71.	upb	被	108.	.	.
35.	qni	個體量詞	72.	upg	給	109.	@	@
36.	qnc	集合量詞	73.	y	語氣詞			
37.	qnk	種類量詞	74.	o	象聲詞			

Appendix B

Surnames and transliterated characters

1.	阿	81.	根	161.	烈	241.	森	321.	仰
2.	埃	82.	耿	162.	林	242.	沙	322.	姚
3.	艾	83.	龔	163.	凌	243.	山	323.	耶
4.	愛	84.	古	164.	劉	244.	尚	324.	葉
5.	安	85.	顧	165.	柳	245.	邵	325.	伊
6.	奥	86.	關	166.	龍	246.	申	326.	易
7.	澳	87.	管	167.	婁	247.	沈	327.	殷
8.	巴	88.	廣	168.	盧	248.	盛	328.	尹
9.	白	89.	郭	169.	魯	249.	施	329.	印
10.	柏	90.	國	170.	路	250.	石	330.	應
11.	班	91.	哈	171.	陸	251.	時	331.	雍
12.	包	92.	韓	172.	呂	252.	什	332.	尤
13.	鮑	93.	杭	173.	倫	253.	史	333.	游
14.	貝	94.	郝	174.	羅	254.	舒	334.	于
15.	比	95.	和	175.	洛	255.	水	335.	虞
16.	畢	96.	何	176.	駱	256.	斯	336.	余
17.	卞	97.	合	177.	麻	257.	司	337.	俞
18.	賓	98.	赫	178.	馬	258.	司馬	338.	禹
19.	波	99.	賀	179.	麥	259.	司徒	339.	郁
20.	博	100.	洪	180.	邁	260.	松	340.	喻

21.	伯	101.	胡	181.	曼	261.	宋	341.	元
22.	卜	102.	花	182.	茅	262.	蘇	342.	袁
23.	布	103.	華	183.	毛	263.	孫	343.	云
24.	蔡	104.	滑	184.	梅	264.	索	344.	澤
25.	曹	105.	桓	185.	蒙	265.	塔	345.	曾
26.	查	106.	黃	186.	孟	266.	泰	346.	詹
27.	柴	107.	惠	187.	糜	267.	譚	347.	湛
28.	昌	108.	霍	188.	米	268.	談	348.	章
29.	常	109.	基	189.	苗	269.	坦	349.	張
30.	陳	110.	稽	190.	明	270.	湯	350.	趙
31.	成	111.	吉	191.	摩	271.	唐	351.	甄
32.	程	112.	籍	192.	莫	272.	陶	352.	鄭
33.	仇	113.	及	193.	墨	273.	特	353.	支
34.	儲	114.	汲	194.	姆	274.	田	354.	鐘
35.	茨	115.	季	195.	穆	275.	童	355.	周
36.	崔	116.	計	196.	那	276.	圖	356.	朱
37.	達	117.	紀	197.	納	277.	土	357.	諸
38.	大	118.	家	198.	乃	278.	托	358.	祝
39.	戴	119.	加	199.	內	279.	瓦	359.	卓
40.	單	120.	賈	200.	倪	280.	萬	360.	茲
41.	德	121.	姜	201.	尼	281.	汪	361.	宗
42.	登	122.	江	202.	聶	282.	王	362.	鄒
43.	鄧	123.	蔣	203.	鈕	283.	危	363.	左
44.	迪	124.	杰	204.	努	284.	韋	364.	佟
45.	狄	125.	解	205.	諾	285.	維	365.	鄔
46.	翟	126.	金	206.	歐	286.	魏	366.	邴
47.	蒂	127.	靳	207.	歐陽	287.	衛	367.	邸
48.	刁	128.	荊	208.	毆	288.	溫	368.	酈
49.	丁	129.	經	209.	帕	289.	文	369.	酆
50.	董	130.	井	210.	潘	290.	聞	370.	芮
51.	都	131.	喀	211.	龐	291.	翁	371.	荀
52.	杜	132.	卡	212.	裴	292.	沃	372.	奚
53.	段	133.	凱	213.	佩	293.	巫	373.	岑
54.	多	134.	康	214.	彭	294.	烏	374.	庾
55.	俄	135.	柯	215.	蓬	295.	吳	375.	閔

56.	爾	136.	科	216.	皮	296.	伍	376.	闞
57.	法	137.	可	217.	匹	297.	西	377.	繆
58.	樊	138.	克	218.	平	298.	希	378.	璩
59.	范	139.	孔	219.	普	299.	席	379.	臧
60.	方	140.	庫	220.	戚	300.	夏	380.	昝
61.	房	141.	拉	221.	奇	301.	項	381.	晁
62.	費	142.	萊	222.	齊	302.	蕭	382.	晏
63.	芬	143.	來	223.	祁	303.	謝	383.	賁
64.	封	144.	賴	224.	錢	304.	邢	384.	腓
65.	馮	145.	藍	225.	強	305.	熊	385.	滕
66.	鳳	146.	蘭	226.	喬	306.	休	386.	於
67.	夫	147.	郎	227.	切	307.	徐	387.	賓
68.	伏	148.	勞	228.	秦	308.	許	388.	褚
69.	福	149.	勒	229.	邱	309.	宣	389.	胥
70.	弗	150.	樂	230.	屈	310.	薛	390.	竺
71.	傅	151.	雷	231.	全	311.	遜	391.	裘
72.	蓋	152.	累	232.	冉	312.	雅	392.	羿
73.	干	153.	黎	233.	饒	313.	亞	393.	麴
74.	甘	154.	李	234.	任	314.	嚴	394.	訾
75.	岡	155.	里	235.	戎	315.	延	395.	瞿
76.	高	156.	利	236.	榮	316.	顏	396.	穌
77.	哥	157.	連	237.	阮	317.	閻		
78.	戈	158.	廉	238.	薩	318.	燕		
79.	葛	159.	梁	239.	塞	319.	楊		
80.	格	160.	廖	240.	桑	320.	羊		

Appendix C

Stopword List for Person Name

1.	被	18.	於	36.	「
2.	的	19.	、	37.	」
3.	對	20.	。	38.	【
4.	和	21.	．	39.	】
5.	會	22.	—	40.	⊙
6.	將	23.	——	41.	★
7.	那	24.	～	42.	！
8.	是	25.	……	43.	（
9.	説	26.	‘	44.	）
10.	所	27.	’	45.	，
11.	他	28.	“	46.	－－
12.	爲	29.	”	47.	／
13.	有	30.	〔	48.	：
14.	與	31.	〕	49.	；
15.	在	32.	〈	50.	？
16.	這	33.	〉		
17.	最	34.	《		

Appendix D

Organization suffixes

1.	安理會	19.	會	37.	外交部
2.	安全部	20.	基金會	38.	委員會
3.	辦公室	21.	機構	39.	衛生部
4.	辦事處	22.	局	40.	協會
5.	部	23.	理事會	41.	學會
6.	部隊	24.	聯合會	42.	學院
7.	参議院	25.	聯盟	43.	研究所
8.	大藏	26.	民主黨	44.	醫院
9.	黨	27.	内務部	45.	銀行
10.	隊	28.	人事部	46.	院
11.	府	29.	人壽	47.	眾議院
12.	工會	30.	社	48.	總公司
13.	公司	31.	署	49.	總會
14.	廣播電台	32.	司法部	50.	總署
15.	廣播公司	33.	堂	51.	組織
16.	國防部	34.	特別委員會		
17.	國會	35.	廳		
18.	國務院	36.	通訊社		

Appendix E

Location suffixes

1.	辦公室	12.	河	24.	洋
2.	辦事處	13.	湖	25.	醫院
3.	川	14.	機場	26.	銀行
4.	大飯店	15.	角	27.	幼稚園
5.	大使館	16.	區	28.	院
6.	大學	17.	人民醫院	29.	中心
7.	島	18.	山	30.	中學
8.	法院	19.	商場	31.	州
9.	府	20.	商店	32.	州府
10.	公立醫院	21.	市		
11.	海	22.	小學		

Appendix F

Examples of Feature Table (Train set with condition D_{10}^{4})

Word	Tag	Term Frequency
貶值	oth	6
韓國	s	4
金融	npr	7
經濟	oth	9
率	oth	6
日本	s	7
台幣	npr	8
台灣	s	4
泰國	s	9
形勢	oth	4

Table F.1: Feature table of topic 20001.

Word	Tag	Term Frequency
白宮	s	9
調查	oth	7
關系	oth	5
檢察官	oth	5
克林頓	Ngm	7
萊溫斯基	Ngm	7
美國	s	5
斯塔爾	Ngm	10
指控	oth	5
總統	oth	12

Table F.2: Feature table of topic 20002.

Word	Tag	Term Frequency
德貝內	Ngm	5
菲律賓	s	14
副總統	oth	8
候選人	oth	23
競選	oth	12
拉莫斯	npf	6
斯特拉	npf	6
選舉	oth	7
執政黨	npr	10
總統	oth	14

Table F.3: Feature table of topic 20005.

Word	Tag	Term Frequency
單闖號	Ngm	4
飛行	oth	23
福塞特	Ngm	10
福斯特	s	4
駕駛	oth	13
利比亞	s	7
路易斯	npf	4
美國	s	5
氣球	oth	17
熱氣球	oth	18

Table F.4: Feature table of topic 20007.

Word	Tag	Term Frequency
奧	oth	14
奧運會	npr	5
冬	oth	14
冬季	oth	5
日本	s	12
委	oth	5
雪	oth	8
野	oth	16
運動員	oth	5
組	oth	5

Table F.5: Feature table of topic 20013.

Word	Tag	Term Frequency
安理會	npr	9
核	oth	19
進行	oth	6
里特	Ngm	5
聯合國	npu	19
美國	s	6
特委會	Ogs	12
武器	oth	10
伊拉克	s	20
中國	s	16

Table F.6: Feature table of topic 20015.

Word	Tag	Term Frequency
飛機	oth	10
機場	oth	10
空難	oth	14
美國	s	4
莫斯科	s	4
台北	s	7
台灣	s	22
香港	s	7
造成	oth	7
中國	s	12

Table F.7: Feature table of topic 20020.

Word	Tag	Term Frequency
阿爾及利亞	s	14
安全部	oth	4
分子	oth	5
恐怖	oth	8
平民	oth	9
事件	oth	10
死亡	oth	4
屠殺	oth	11
無辜	oth	7
現場	oth	4

Table F.8: Feature table of topic 20023.

Word	Tag	Term Frequency
大選	oth	8
舉行	oth	5
聯合	oth	6
人民	oth	4
投票	oth	8
印度	s	9
陣線	oth	7

Table F.9: Feature table of topic 20039.

Word	Tag	Term Frequency
阿肯色	s	12
開槍	oth	8
美國	s	6
年齡	oth	5
瓊斯	npf	5
少年	oth	14
事件	oth	16
學生	oth	7
學校	oth	6
州	oth	13

Table F.10: Feature table of topic 20048.

Word	Tag	Term Frequency
比賽	oth	24
關穎珊	Ngm	6
郭政新	Ngm	8
滑	oth	10
獲得	oth	13
節目	oth	13
美國	s	11
申雪	Ngm	4
選手	oth	20
中國	s	12

Table F.11: Feature table of topic 20057.

Word	Tag	Term Frequency
阿拉法特	npf	7
巴勒斯坦	s	5
華盛頓	s	7
倫敦	s	5
羅斯	npf	8
美國	s	9
内塔尼亞胡	Ngm	6
以色列	s	7
約旦河	s	6
中東	s	13

Table F.12: Feature table of topic 20071.

Word	Tag	Term Frequency
活動	oth	6
進行	oth	5
警察	oth	5
警方	oth	12
舉行	oth	7
抗議	oth	6
學生	oth	7
雅加達	s	5
印度尼西亞	s	5
印尼	s	12

Table F.13: Feature table of topic 20076.

Word	**Tag**	**Term Frequency**
暴力	oth	5
官員	oth	4
觀察	oth	4
華人	oth	6
活動	oth	5
人權	oth	6
亞洲	s	4
印度尼西亞	s	7
印尼	s	15
政府	oth	14

Table F.14: Feature table of topic 20088.