Topic and Link Detection from Multilingual News

Huang Ruizhang

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

Supervised by

Prof. Lam Wai

June 2003

摘要

在這篇論文中，我們研究了兩個可以自動分析和挖掘有用數據的系統。第一個系統叫做事件與主題偵查系統(event and topic discovery system)。我們應用一個兩層的分層式聚類算法(two-level hierarchical unsupervised learning algorithm), 從中英文新聞數據中找到未知的主題和事件信息。在下層結構中，事件信息將從新聞信息中得到，新聞是按照年代順序排列的。我們應用信息提取(information extraction)的方法從新聞內容中自動獲取名字(named entity)和內容詞組(content term)。因爲新聞是時時刻刻都發生的，所以我們的系統支持增量聚類(incremental clustering)。在高層結構中，新聞的主題從事件中產生。我們用相關模型(relevance model)的方法來決定事件和主題之間的關系。

第二個系統是新聞相關系統(story link detection system)。這個系統的目的是判斷兩條新聞是否像關於同一個主題。新聞相關系統應用了自動主題類別判斷(automatica topic type categorization)的辦法把新聞劃分到一些事先規定好的主題類別中去。根據這些主題類別的信息，我們可以判斷新聞代表(story representation)中每個部分的重點。

我們進行了一系列試驗來測試事件主題偵查系統和新聞相關系統，並且對試驗的結果進行了分析和說明。

Abstract

In this thesis, we investigate two systems that can analyze news documents and find useful information automatically. The first system is called event and topic discovery system. A two level hierarchical unsupervised learning algorithm is employed to discover event and topic information not known to the system from news stories of different languages, including English and Chinese. At the lower level, events are discovered from incoming news stories in chronological order. Information extraction technique is used for automatically extracting useful terms such as named entities. Since news stories are coming around-the-clock, incremental clustering technique is used for the discovery task. At the higher level, topics are discovered from the generated event information. We use a relevance model to determine the relationship of an event and a topic.

The second system is the story link detection system. It aims at determining whether two stories are related to the same topic or not. The story link detection approach makes use of an automatic topic type categorization method to classify a story into some general topic types. After a story is automatically assigned to a set of topic types, different emphasis will be placed on different parts of story representation during the link detection process.

We have conducted experiments on the event and topic discovery system and the story link detection system with large-

scale real-word news corpora. The performance and effectiveness of two approaches are demonstrated and analyzed.

Acknowledgement

First of all, I would like to give my sincere gratitude and appreciation to my supervisor, Prof. Wai Lam. Two years ago, I was totally a stranger to the research. Without his kindly help, professional instructions and enthusiastic comments, the thesis could not be completed. In the past two years, I spent an ordered and instructive time with him. He not only taught me the research knowledge, but also directed me how to do the research. When I got confusion, when I met troubles, he always kindly helped me out. In addition, I want to thank my thesis committee, Prof. Jeffrey Yu and Prof. K.P. Lam, for their precious advice on thesis. I would also like to thank all the staffs of the Department of Systems Engineering and Engineering Management for their diligent help.

Finally, I would like to thank my parents and my friends. They are always back to me, and give me endless power, courage and support for me to face up any difficulties.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

The rapid growth of the Internet and the wide availability of electronic documents bring challenges to the traditional query-driven information retrieval (IR) technology. People are more likely to access news stories from diverse sources. The content of these news documents may come from multilingual languages. However, it is impossible for people to browse all or most of the news story content in the archive in order to know what events occurred or what happened. Query-based retrieval is useful only when you know precisely the nature of the facts you are seeking. Users will have difficulty formulating the "right query" or "right level of abstraction", or checking all the potentially relevant stories. Therefore it is very useful if a system can analyze these news documents and find the useful information automatically.

Consider, for example, a person has just returned from an extended vacation and needs to find out quickly what happened in the world during his absence. Reading the entire news collection is a daunting task. Intelligent assistance from the computer for discovering unseen topics and events is clearly very useful. The Topic Detection and Tracking (TDT) evaluation project, organized by Defense Advanced Research Projects Agency (DARPA) and National Institute of Standards and Technology (NIST), provides news

corpora for investigation and evaluation. One of the important tasks being evaluated by TDT is the topic detection task, which aims to detect and track topics not previously known to the system [29]. In addition to topics, we investigate an approach that can discover events. Another issue is that the news may come from different sources in different languages, in particular, Chinese and English. Another task investigated by TDT is the story link detection problem.

1.1 The Definition of Topic and Event

A "topic" is defined to be a seminal event or activity, along with all directly related events and activities [23]. An "event" is defined as a specific piece of incident or activity usually occurs in a short period of time. The relation between topic and event is shown in Figure 1.1. A topic usually contains a sequence of events. For instance, "Tony Blair Visits China in October, 1998" is a sample topic and it includes several events, such as the preparations for visit, his meeting with Chinese leaders, his interview on Chinese national television and so on. Each piece of event contains a set of news stories and is usually reported in different sources and in different languages.

1.2 Event and Topic Discovery

1.2.1 Problem Definition

The problem of event and topic discovery is defined to be the task of discovering events and topics not previously known to the system. The flow of the discovery problem is depicted in Figure 1.2. The input data are processed in chronological order, in Chinese or English. The discovery system needs to

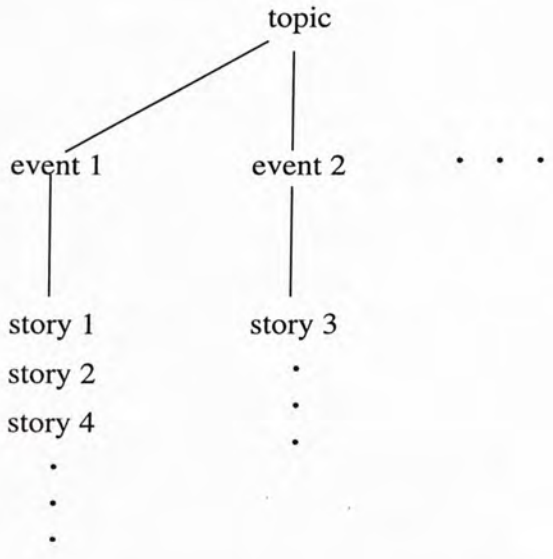

Figure 1.1: The relation between events and topics

make a judgment whether an incoming story belongs to a new event or an existing event that has been discovered previously. Another judgment is to determine whether the incoming story belongs to a new topic or an existing topic. After such processing, the system can identify a set of events and a set of topics.

1.2.2 Characteristics of the Discovery Problems

Several characteristics emerge in our discovery problem. The data we deal with comes from multilingual news sources. First, the news story cannot be used directly for the unsupervised learning. To capture the major idea of each news story, we need to extract semantics such as named entities and story content terms to represent the story. Unsupervised learning is performed on the extracted information. Second, news stories discussing the same event tend to be released closely. This suggests that 1considering the release time information during the discovery process will be helpful. Third, there is a

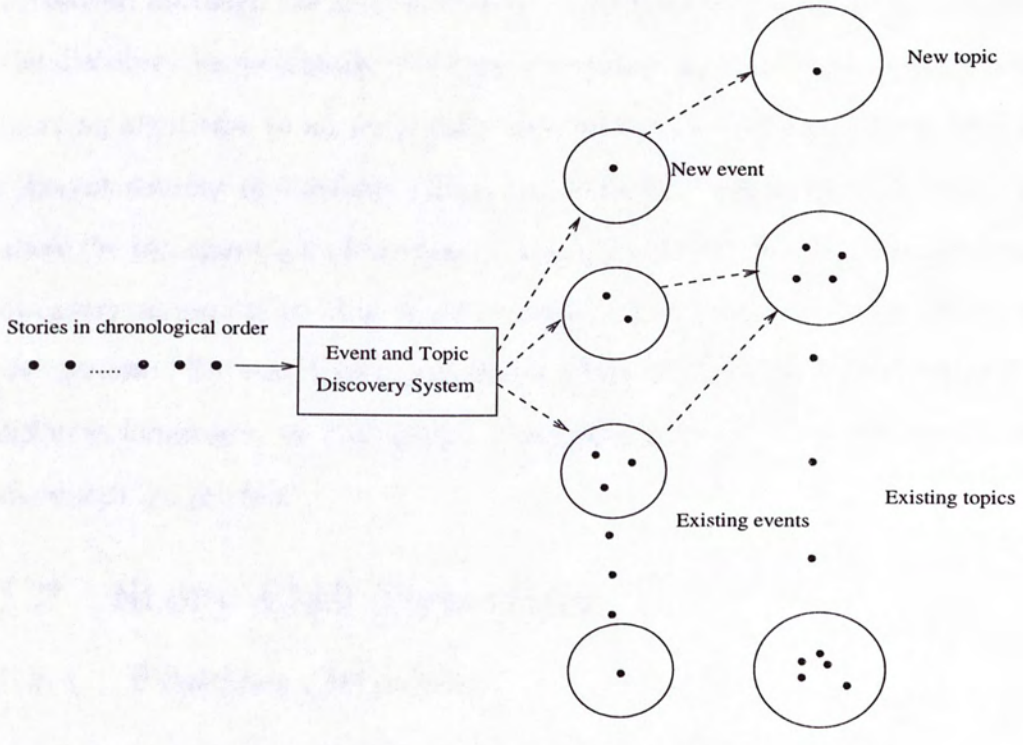

Figure 1.2: Event and topic discovery problem

significant distinction between different languages. Therefore, we need to take into account this language difference in our discovery approach.

1.2.3 Our Contributions

Most of existing approaches are designed to deal with the discovery of only topics or only events. They do not consider to discover them at the same time. Moreover, although the news stories arrive continuously, they do not conduct the discovery incrementally. We have developed a hierarchical unsupervised learning algorithm to automatically discover events and topics from news of different sources in different languages, namely, English and Chinese. To allow the incorporation of existing events or topics, we design an incremental discovery approach so that it can load the previously discovery result in the system. To take into account the effect of different distributions for different languages, we also design a language normalization scheme during the discovery process.

1.3 Story Link Detection

1.3.1 Problem Definition

Another important task being evaluated by TDT is the story link detection problem. The aim is to find out whether two given stories are related to the same topic or not. The stories may come from different sources. The definition of "topic" is described in Chapter 1.1. Story link detection is an important problem because it provides a fundamental tool for other intelligent tasks, such as topic detection and topic tracking. Topic tracking aims to associate incoming stories with topics that are known to the system. A solution for link detection can act as a kernel function from which these tasks

or other intelligent applications can be built.

The flow of the story link detection is depicted in Fig 1.3. The story link detection system can make a judgment whether an incoming story pairs belong to the same topic or not. It also gives a confidence score of the decision. The input data are story pairs which may come from different sources in different languages, namely, Chinese and English.

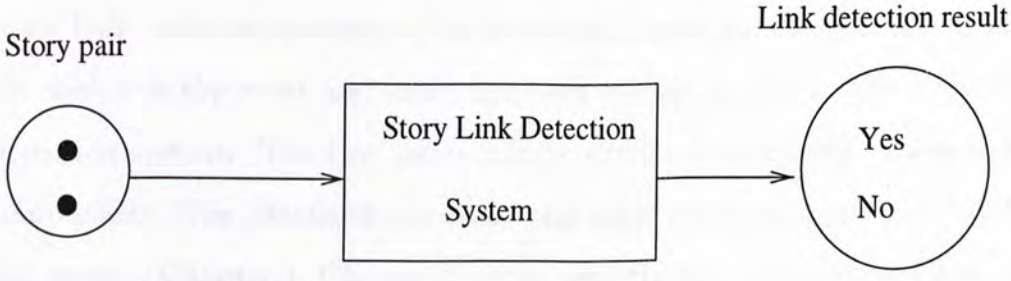

Figure 1.3: Story link detection task

1.3.2 Our Contributions

We investigate a link detection approach which employs an automatic topic type categorization model. Automatic topic type categorization models are trained for classifying a story to broad topic types. Examples of topic types are "Elections", "Scandal/Hearings", "Accident", and "Celebrity/Human Interest News", etc. We automatically extract key terms to represent the stories. The story representation consists of people names, geographical location names, organization names and content terms. A story related to "Accident" may emphasize more on the geographical location names while the people names may play a more important role in the stories related to "Celebrity/Human Interest News". Our link detection system uses an automatic topic type categorization method to decide how to distribute the

contribution on each kind of named entity and content term component in the story representation.

1.4 Thesis Organization

The remaining parts of the thesis are organized as follows: Chapter 2 will describe some related work on the event and topic discovery problem and the story link detection problem of other research groups. Chapter 3 will give the design of the event and topic discovery system as well as the story link detection system. The two systems have similar components except some components. The details of the event and topic discovery approach will be discussed in Chapter 4. Chapter 5 will present the experimental performance and analysis of the event and topic discovery component. In Chapter 6, we will discuss the story link detection approach supported by an automatic topic type categorization method and a language normalization method. Its experimental results will be given in Chapter 7. Conclusions and future directions will be presented in Chapter 8.

□ **End of chapter.**

Chapter 2

Literature Review

There are many research groups working on the topic detection problem of the Topic Detection and Tracking (TDT) evaluation project, including University of Massachusetts (UMass), BBN Technologies, IBM Research Center, Carnegie Mellon University (CMU) and National Taiwan University (NTU). Most existing approaches focus only on "topics". Moreover, several research groups have investigated the link detection problem such as CMU, UMass, and NTU. We will also discuss their methods in this chapter.

2.1 University of Massachusetts (UMass)

2.1.1 Topic Detection Approach

University of Massachusetts (UMass) employed a modification of a single pass clustering algorithm in their detection system in TDT1 [3]. They used feature extraction and selection techniques to build a query representation for the story's content. A threshold is estimated for each query which determines binary decisions. The new story is compared against earlier queries in memory. If a new story exceeds an existing query's threshold, the story is assumed to discuss the topic represented in the query, otherwise it contains

a new event. After the new story is made a decision, the existing queries will be rebuilt by adding the new story's query.

They applied the same threshold model to the TDT2 detection task [26]. They improved the time component by incorporating the number of days between stories, while in TDT1 the time component is based on a story sequence number.

In TDT3 [2], their detection system supports two models of comparing a story to previously seen material, namely, agglomerative centroid clustering and k-nearest neighbor comparison. Other important issues in their approach are the problem of determining the right similarity function and weighting of individual features that occur in the stories. They considered four similarity functions: cosine, weighted sum, language models, and Kullbach-Leiblar divergence. The feature weighting methods they employed are TF*IDF, TF and IDF, where TF refers to term frequency and IDF refers to inverse document frequency. Instead of developing new methods in TDT 2000, they spent a fair amount of time rearchitecting the code, learning to deal with its peculiarities and correct bugs detracted substantially from research [1]. Therefore, their TDT 2000 approach is very similar to those used in TDT3.

2.1.2 Story Link Detection Approach

UMass began their research on the Story Link Detection in TDT3 [2]. They tried several similarity measures and weighting schemes. Similarity measures sampled were cosine, weighted sum, language model, and Kullbach-Leiblar divergence. Weighting schemes sampled were TF*IDF, IDF and TF.

In TDT 2000, they did not report any novel results but some preliminary results that showed their improvements in link detection [1]. They explored

how a query expansion technique from IR could smooth the compared stories, and how score normalization depending on language mix can improve results.

They used relevance modeling, a statistical language modeling technique related to query expansion, to the TDT 2001 link detection task [20]. The relevance modeling is used to enhance the topic model estimate associated with a news story, boosting the probability of words that are associated with the story even when they do not appear in the story. They used a modified form of Kullbach-Leiblar divergence as the similarity comparison method.

2.2 BBN Technologies

BBN developed their topic detection system in TDT2 and TDT3. BBN used an incremental k-means algorithm for clustering stories [28]. This algorithm processes stories one at a time and sequentially. For each story it undergoes a two-step process used two types of metrics, namely, selection metrics and thresholding metrics. The selection metric takes a story and outputs cluster scores such that the most similar cluster is found. They used the BBN topic spotting metric as the selection metric. The goal of a thresholding metric is to make a decision whether or not a story should be merged with a cluster. They utilized a hybrid of the BBN topic spotting metric with a more conventional cosine distance metric.

When determining whether a story should merge an existing topic cluster or create a new topic seed, they used a two-level normalization approach. First, they normalized the similarity score with respect to stories certain to be off-topic. Then, the score is normalized with respect to clusters of various sizes all of which are unlikely to be on the same topic as the test story.

In TDT3, BBN investigated a translation system so that their topic de-

tection system can deal with the multi-lingual news data, namely Chinese and English [21]. They segmented the original Mandarin document, looked up each Mandarin word in the bilingual dictionary. The quality of simple translation is not very good. Therefore, They used pinyin to extend the lookup translation and estimated non-uniform prior translation probabilities using the observations of aligned sentences in a parallel corpus. Additional, they devised an algorithm for iteratively improving a translation using co-occurrence statistics.

In addition to the TDT project, BBN also provided a topic discovery system which created topics from a collection of news stories and provided human understandable topic labels for each discovered topic instead of the list of stories related to the discovered topic [27]. The $OnTopic^{TM}$ system at BBN uses a Hidden Markov Model to model multiple topics in documents explicitly. The algorithm for finding topics in the corpus has two high-level steps. First, they found descriptive phrases in the corpus. Next, they determined an initial set of topics, based on the key-phrases that occur in each document. They then used the Estimate-Maximize algorithm to determine the full set of words and phrases that are statistically associated with each of topics.

2.3 IBM Research Center

IBM conducted a two-tiered clustering approach in TDT3 [13]: each cluster is composed of several microclusters. News stories are initially assigned to microclusters. Then, at the end of the deferral period, microclusters are grouped into the actual clusters. They used a symmetrized version of the Okapi formula to score the similarity of two stories. Each microcluster inter-

nally is the centroid of the stories contained in the microcluster. The score of a document with a microcluster becomes the mean of the scores of that document with the stories contained in the microcluster. Microcluster is assigned to a cluster by assigning it to the same cluster as the most similar microcluster, or starting a new cluster if none of the scores are sufficiently high.

The weighting scheme they adopted is both time-dependent and microcluster-dependent. The two-tiered clustering is used to reduce cohesion and make cluster assignments more dependent upon the topic and less dependent upon the source. Min-IDF technique is used to avoid different word statistics associated with each source. Min-IDF is to compute source-specific IDFs, and then, for scoring, choose the minimum, across all sources, of the source-specific IDFs. Therefore the topical effects and source-based effects caused by "disproportionately common" for each source will be lowered.

2.4 Carnegie Mellon University (CMU)

2.4.1 Topic Detection Approach

Carnegie Mellon University (CMU) developed their topic detection system from TDT1 [32]. They employed the conventional vector space model which uses the bag-of-terms representation. A story is represented using a vector of weighted terms. The normalized vector sum of documents in a cluster is used to represent the cluster, and called the prototype or centroid of the cluster. Terms in a story vector or a cluster prototype are statistically weighted using the term frequency (TF) and the Inverse Document Frequency (IDF) and are appropriately normalized. They used the standard cosine similarity to measure the similarity between the story and cluster prototype vectors. They

investigated two clustering method. The first method is an agglomerative (hierarchical) algorithm based on group-average clustering (GAC). The second method is a single-pass algorithm (INCR) which generates a non-hierarchical partition of the input collection. GAC is designed for batch processing, and is used for retrospective detection. INCR is designed for sequential processing, and is used for both retrospective and on-line detection.

The CMU topic detection system for TDT2 [7] was largely based on previous retrospective and online detection systems used in the TDT1. They used the vector space model to represent stories as weighted unigram models. They also used temporally-sensitive versions of their incremental and hierarchical GAC clustering algorithms to detect new topics within the 3 deferral periods.

In TDT3, they combined agglomerative clustering and single-pass clustering with different term-weighting schemes (TF-IDF and language-modeling based) [31]. The GACIncr system is a cosine-similarity based clustering system. When a new story in the deferral window is processed, a greedy agglomerative clustering algorithm can optionally be applied. If this option is turned off, only singleton clusters will be present in the look-ahead set. They call the system which works with only singleton-clusters in the deferral window set "Incr.VSM". Each cluster containing a story from the deferral window is compared to previously seen clusters. If a suitable match is found, the clusters will be merged.

The Incr.LM system works almost identically to the GAC Incr/Incr.VSM system, with the difference being the comparison criteria. The Incr.LM system does not allow the option of clustering within the deferral window. Clusters are represented using language models trained with EM on member sto-

ries. When hard decisions must be made, the likelihood of each cluster in the deferral window with the existing clusters will be computed. If a suitable match is found, those clusters will be merged.

The BORG.det system incorporates both the GACIncr and Incr.LM methods. They used a very simple voting scheme. Each method runs using its own distinct threshold. Whenever a hard decision of a cluster in the deferral window is requested, each method votes on whether to combine it with an existing member or create a new entry. If either method votes to combine with an existing member, the action is taken. Otherwise, a new member is added.

2.4.2 Story Link Detection Approach

CMU developed two link detection systems in TDT3 [6]. The first of their systems, identified as CMU-1, uses incremental TF*IDF weighted cosine similarity measures to determine whether or not two documents discuss the same topic. Documents are stop-worded, stemmed, and converted to binary term vectors. The second system, identified as CMU-2 in the evaluation, is also based on weighted cosine similarity measures, though with different weighting and thresholds. The logarithm of the term frequency is used and the TF*IDF statistics are derived solely from the test stories as they are processed, rather than having been initialized from the six-month training corpus.

2.5 National Taiwan University (NTU)

2.5.1 Topic Detection Approach

In TDT3, National Taiwan University (NTU) focused on Mandarin text [8]. They employed a Chinese named entity system used in MUC7 to identify

people names, organization names, locations names and some other named entities like date/time expressions and monetary and percentage expressions. At most 50 terms are selected to represent story and topic cluster. In the version used in TDT3, only named entities are used. They used a two-threshold scheme to determine the relationship between a news story and a topic cluster.

NTU presented their updated topic detection algorithms for Chinese and English-Chinese topic detection in [9]. Named entities, other nouns and verbs are cue patterns to relate news stories describing the same topic. They used lexical translation and name transliteration to translate Chinese story vector representation to English. The two-threshold detection algorithm is similar as before. They employed Top-N-weighted strategy and LRU+Weighting strategy to group a story vector representation to its related topic representation. The Top-N-weighted strategy selects N terms with larger weights from the two vectors. LRU+Weighting strategy is more complex. They kept a certain number of candidate terms for each topic and replaced the least-recently-used terms.

2.5.2 Story Link Detection Approach

NTU presented their link detection approach in [10]. Each story in a given pair is represented as a vector with TF*IDF weights. Then the cosine function is used to measure the similarity of two stories. They tried to use different lexical items to represent story, such as nouns & compound nouns, nouns & verbs & compound nouns, nouns & adjectives & compound nouns, nouns & verbs & adjectives & compound nouns. The story pairs whose similarity are higher than a predefined threshold are kept in a database for story

expansion. Stories are segmented into small passages according to the discussing topics and compute passage similarity instead of document similarity. For multilingual story pairs, they employed a simple approach to translate a Chinese story into an English one. If a Chinese word corresponds to more than one English word corresponds to more than one English word, these English words are all selected without disambiguating them.

□ **End of chapter.**

Chapter 3

System Overview

In this chapter, we discuss the framework of our event and topic discovery system and story link detection system. These two systems consist of six modules, namely, story preprocessing, information extraction, gloss translation, term frequency and term weight calculation, event and topic discovery component and story link detection component.

Figure 3.1 depicts the system overview of event and topic discovery system. A news story is first passed through the story preprocessing module. In this module, segmented sentence will be generated from the word token provided from the news sources. Next, useful story terms will be extracted to represent the news stories. The terms include named entities and story content terms. We make use of a transformation-based error-driven tagger to automatically extract people named entities, geographical location named entities and organization named entities from the news stories. Since we process news stories from multiple language sources, in particular English and Chinese, we translate the Chinese terms in the story representation into English by the gloss translation module so that the subsequent steps can be performed based on an uniform representation. In the term weight calculation module, we calculate the corresponding weight of each term and form

a four-dimensional vector space representation for each story. Finally, our event and topic discovery component will perform unsupervised learning on the story representation based on a relevance model technique. The discovery result on the event and topic can be generated.

The architecture of story link detection system is shown in Figure 3.2. The first four modules are the same as those in the event and topic discovery system. The multilingual news stories pass through story preprocessing module, information extraction module, gloss translation module and term weight calculation module and form uniform four-dimensional vector space representation. After that, the story link detection component can make decision on story pairs whether two news stories are related to the same topic or not.

3.1 News Sources

The corpora we used in our investigation are provided by Linguistic Data Consortium (LDC)[1]. The TDT2 corpus spans the first six months of 1998 and contains news data collected daily from 8 news sources in two languages (English and Mandarin Chinese). We used TDT2 corpus for training the "Relevance corpus" and "Topic Categorization Models" which will be used in the event and topic discovery component and the story link detection component respectively. The English sources of TDT2 corpus are:

- New York Times Newswire Service
- Associated Press Worldstream Service
- Cable News Network, "Headline News"

[1]Linguistic Data Consortium URL: http://www.ldc.upenn.edu

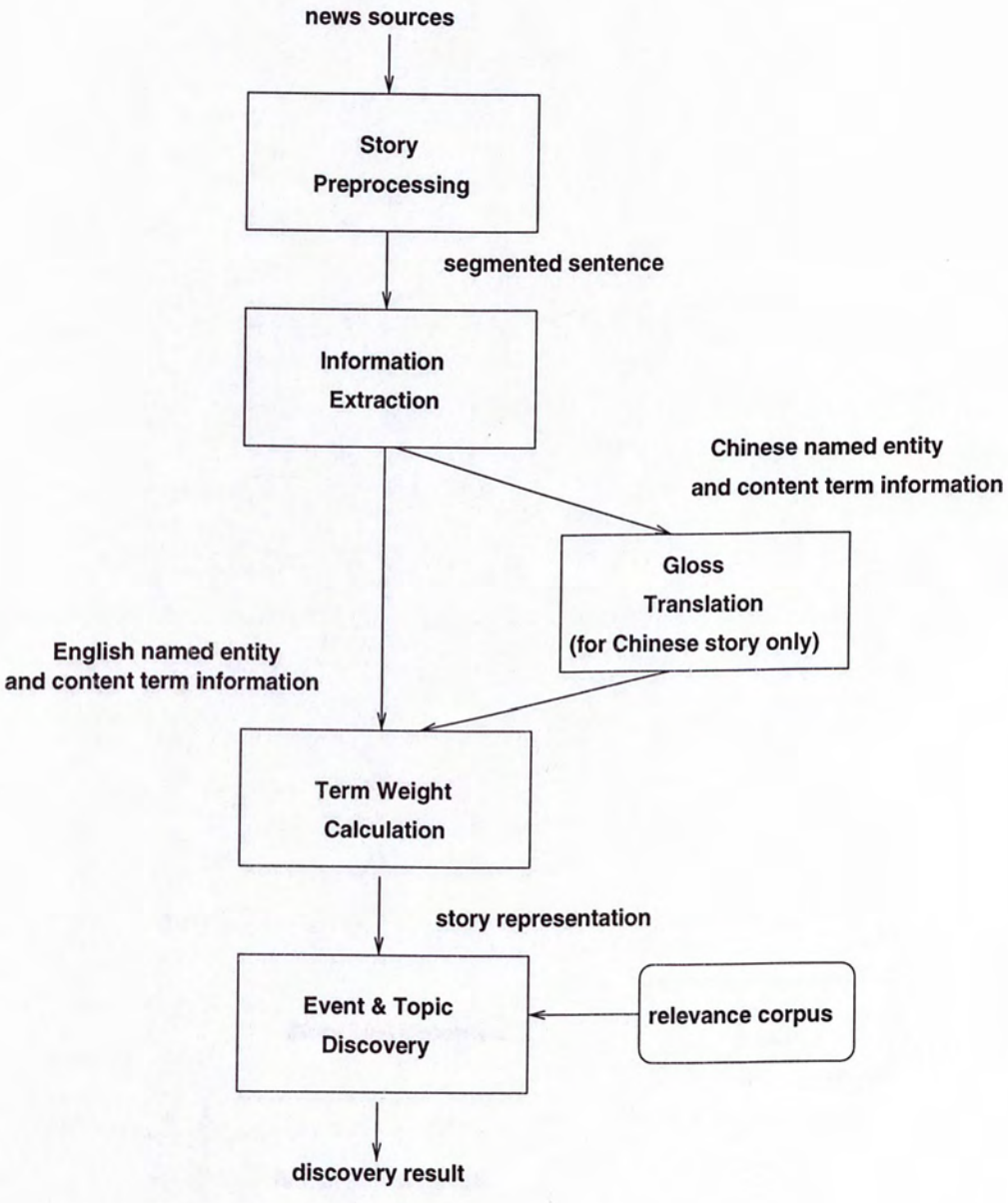

Figure 3.1: Overview of the event and topic discovery system

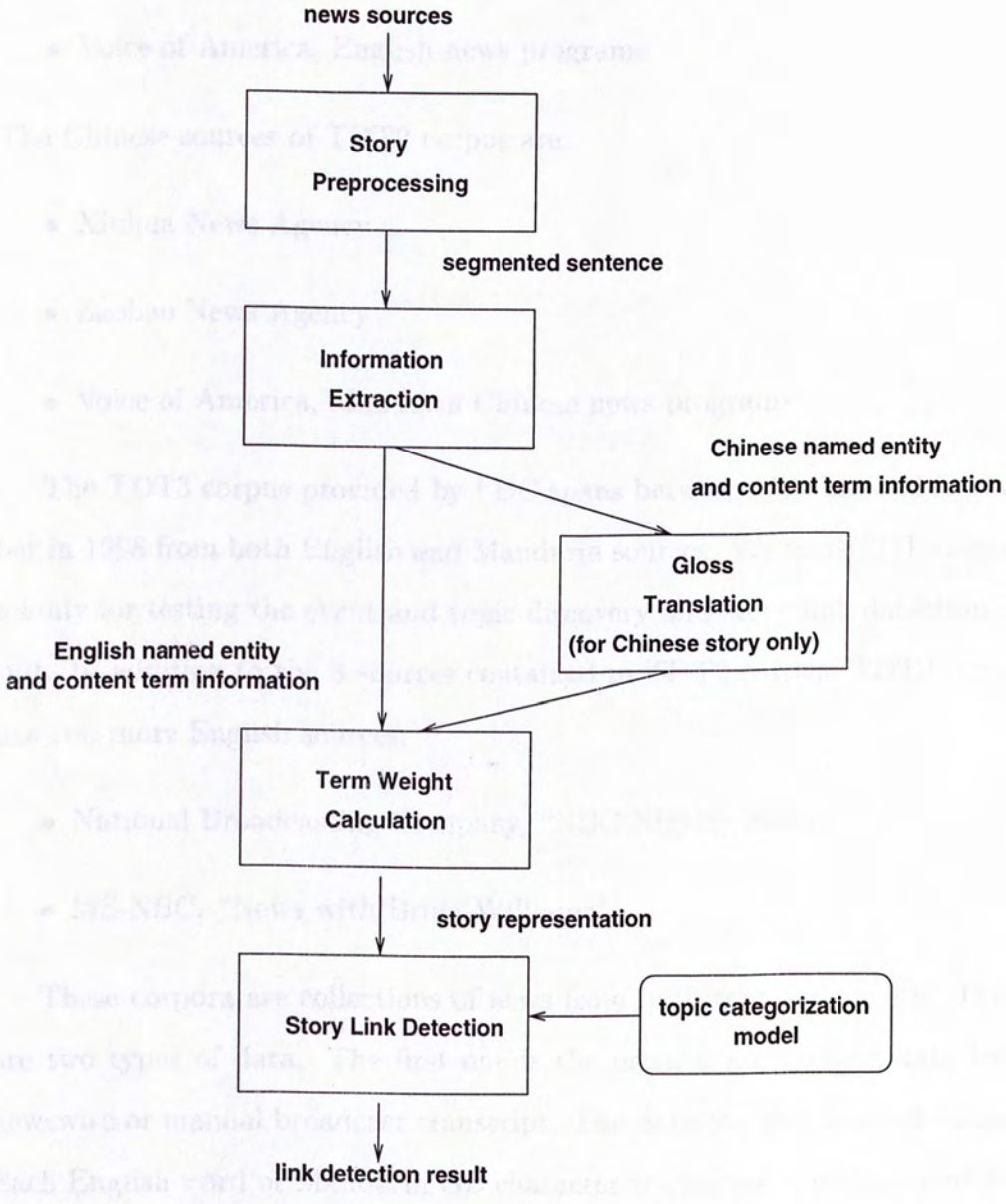

Figure 3.2: Overview of the story link detection system

- MS-NBC, "News with Brian Williams"
- Public Radio International, "The World"
- Voice of America, English news programs

The Chinese sources of TDT2 corpus are:

- Xinhua News Agency
- Zaobao News Agency
- Voice of America, Mandarin Chinese news programs

The TDT3 corpus provided by LDC spans between October and December in 1998 from both English and Mandarin sources. We used TDT3 corpus mainly for testing the event and topic discovery and story link detection result. In addition to the 8 sources contained in TDT2 corpus, TDT3 corpus has two more English sources:

- National Broadcasting Company, "NBC Nightly News"
- MS-NBC, "News with Brian Williams"

These corpora are collections of news from both text and speech. There are two types of data. The first one is the original source text data from newswire or manual broadcast transcript. The data are in tokenized format. Each English word or Mandarin GB character is assigned a unique identifier and presented on a separate line with a tag "recid". An example of the text source format is shown in Table 3.1.

The second kind is related to broadcast data. The Mandarin and English broadcast news data have been processed by speech recognition software

<W recid=1>	Two
<W recid=2>	months
<W recid=3>	after
<W recid=4>	a
<W recid=5>	hurricane
<W recid=6>	mauled
<W recid=7>	this
<W recid=8>	country,
<W recid=9>	it
<W recid=10>	is
<W recid=11>	still
<W recid=12>	unclear
<W recid=13>	how
<W recid=14>	many
<W recid=15>	Hondurans
<W recid=16>	died
<W recid=17>	in
<W recid=18>	the
<W recid=19>	storm.

Table 3.1: A sample of tokenized text data

<W recid=1 Bsec=15.98 Dur=0.30 Clust=1 Conf=0.76>	今天
<W recid=2 Bsec=16.28 Dur=0.23 Clust=1 Conf=0.91>	的
<W recid=3 Bsec=16.51 Dur=0.29 Clust=1 Conf=0.90>	主要
<W recid=4 Bsec=16.80 Dur=0.33 Clust=1 Conf=0.98>	内容
<W recid=5 Bsec=17.13 Dur=0.31 Clust=1 Conf=0.95>	有
<X Bsec=17.44 Dur=0.59 Conf=NA>	
<W recid=6 Bsec=18.03 Dur=0.31 Clust=1 Conf=0.98>	美国
<W recid=7 Bsec=18.34 Dur=0.38 Clust=1 Conf=0.94>	政府
<W recid=8 Bsec=18.72 Dur=0.35 Clust=1 Conf=0.95>	采取
<W recid=9 Bsec=19.07 Dur=0.32 Clust=1 Conf=0.90>	果断
<W recid=10 Bsec=19.39 Dur=0.42 Clust=1 Conf=0.96>	行动
<X Bsec=19.81 Dur=0.24 Conf=NA>	
<W recid=11 Bsec=20.05 Dur=0.34 Clust=1 Conf=0.90>	向
<W recid=12 Bsec=20.39 Dur=0.92 Clust=1 Conf=0.95>	国际货币基金
<W recid=13 Bsec=21.31 Dur=0.47 Clust=1 Conf=0.98>	组织
<W recid=14 Bsec=21.79 Dur=0.42 Clust=1 Conf=0.90>	提供
<W recid=15 Bsec=22.21 Dur=0.13 Clust=1 Conf=0.87>	的
<W recid=16 Bsec=22.34 Dur=0.30 Clust=1 Conf=0.76>	一百
<W recid=17 Bsec=22.64 Dur=0.30 Clust=1 Conf=0.87>	八十
<W recid=18 Bsec=22.94 Dur=0.10 Clust=1 Conf=0.90>	亿
<W recid=19 Bsec=23.04 Dur=0.41 Clust=1 Conf=0.94>	美元
<X Bsec=23.45 Dur=0.59 Conf=NA>	

Table 3.2: A sample of tokenized broadcast data

provided by Dragon Systems and the BBN Byblos respectively. The data is in token stream form without story boundary or punctuation. Each Mandarin word is assigned a unique "recid", with information on starting time and duration period. An example of the text source format is shown in Table 3.2.

作为新兴的旅游城市,
深圳日前评选出十十大旅游景点,
以吸引更多海内外游客前来游览观光。

Table 3.3: A sample of Chinese sentence segmentation

A new study indicates Americans are eating more vegetables .
It important because eating better helps reduce the risk of cancer .
But mom might not like this part so much .
Many of the vegetables aren 't green or leafy they 're deep fried .

Table 3.4: A sample of English sentence segmentation

3.2 Story Preprocessing

Since the news sources are in word token format, we need to group them into sentences for subsequent processing. Each sentence is stored in a separate line. A new sentence is generated if a delimiter punctuation such as question mark, comma, period is met. The sentence boundary of broadcast news is identified by the silence period of two tokens. If the silence period is longer than a predefined threshold, we decide that a new sentence starts from the next token. An example of a Chinese news story after the sentence segmentation process is shown in Table 3.3. Since the last few words of English broadcast a news always contain news source name and a reporter name which have nothing to do with the news content. The reporter name appears very near the news source name, for example, "JACK SMITH A. B. C. NEWS". These information may reduce the quality of content term extraction. Therefore, we remove the last few words corresponding to news source terms, such as "ABC", "CNN","PRI", from the English broadcast news stories. An example of an English news story after the sentence segmentation process is shown in Table 3.4.

作为 新兴 的 旅游 城市 ,
深圳 日前 评选 出十 十大 旅游 景点 ,
以 吸引 更多 海内外 游客 前来 游览 观光 。

Table 3.5: A sample of word segmentation generated from Table 3.3

After the Chinese news data has been sentence segmented. We employ a software provided by LDC which uses dynamic programming technique to perform word segmentation. The continuous singular terms will be combined into meaningful words. An example of word segmentation is shown in Table 3.5.

3.3 Information Extraction

In this step, we will extract useful information to represent news stories. A transformation-based error-driven linguistic tagger for each language, one for English and one for Chinese, is employed to perform this information extraction task [4]. There are two steps in this tagger: learning and tagging [5]. We first manually annotate a training corpus with part-of-speech and named entity information. The named entities that we consider include people names, geographical location names, and organization names. In the learning step, a compact set of rules including lexicons, contextual rules and lexical rules can be trained from the corpus. After the learning process, the set of learned rules can be used to conduct tagging for an unseen news story. A sample of tagged Chinese sentence is shown in Table 3.6. In this example, 深圳 is tagged as a geographical location name, others are tagged by their part-of-speech information.

The story key terms are extracted according to their tags. There are four kinds of story key terms, namely, people named entities, geographical location

作为/vgn 新兴/a 的/usde 旅游/ng 城市/ng ,/,
深圳/s 日前/ng 评选/ng 出十/vc 十大/ng 旅游/ng 景点/ng ,/,
以/p 吸引/vgn 更多/a 海内外/ng 游客/ng 前来/vv 游览/ng 观光/ng 。/。

Table 3.6: A sample of segmented Chinese sentence after tagging with part-of-speech and named entity information

named entities, organization named entities, and story content terms. Those terms not belonging to the three kinds of named entities are processed by stop word removal, and stemming. These terms are considered as story content terms. An example of story key terms extracted is shown in Table 3.7. The two columns represent the sentence number where the term appears and the term itself. For example, "2 深圳" means that 深圳 appears as a geographical location name in the second sentence of the story once.

3.4 Gloss Translation

The news stories that we process come from multiple languages. It is difficult to conduct comparison between Chinese and English stories directly. Our approach is to conduct gloss translation on the extracted story key terms of Chinese stories into English so that we can perform subsequent mining process based on a uniform representation [18]. Full-fledged translation is not necessary since our purpose is to conduct event and topic discovery as well as story link detection rather than machine translation.

The gloss translation is conducted sentence by sentence. For each Chinese term C in a sentence, we first look up C in an existing bilingual lexicon. The current bilingual lexicon used in our experiments was obtained from Linguistic Data Consortium (LDC). The lexicon returns a set of English terms $\{E_1, \ldots, E_m\}$ which are possible translations of C. The next step

...
<PLACE>
2 深圳
...
< /PLACE>
...
<TERM>
1 新兴
1 旅游
1 城市
2 日前
2 评选
2 出十
2 十大
2 旅游
2 景点
3 吸引
3 更多
3 多的
3 海内
3 游客
3 前来
3 游览
3 观光
...
< /TERM>

Table 3.7: A sample extracted story key term information

is to conduct term disambiguation so that appropriate English translation terms can be assigned a higher weight. Our term disambiguation algorithm makes use of a parallel corpus. We use the Hong Kong News parallel corpus obtained from LDC. It contains 18,146 aligned pair of parallel documents in English and Chinese. The documents are mainly government announcements and news. We performed automated sentence alignment for each pair of documents based on a length based alignment algorithm [14]. Then the sentences are indexed by an IR engine. Given the Chinese term C to be translated, we first retrieve the relevant Chinese sentences containing C. The next step is to collect the corresponding English sentence from the parallel corpus for each retrieved Chinese sentence. Let the set of collected English sentences be P_E. A score called usage factor is proposed to calculate the relative importance of the translation. The terms contained in these English sentences are used to compute the usage factor of each E_i in $\{E_1, \ldots, E_m\}$ as:

$$U(E_i) = \frac{f(P_E, E_i)Y(P_E, E_i)}{\sum_{i=1}^{m}(f(P_E, E_i)Y(P_E, E_i))} \tag{3.1}$$

where $f(P_E, E_i)$ represents the term frequency of E_i in P_E and $Y(P_E, E_i)$ represents the inverse sentence frequency of E_i in P_E. By changing the bilingual lexicon and parallel corpus accordingly, our gloss translation approach can be easily adapted to other language.

For example, the Chinese term 重要 can be translated as "importance" and "magnitude" as indicated in the bilingual lexicon. Suppose that, "magnitude" appears four times within two extracted sentences while "importance" appears six times within four extracted sentences. Therefore the usage factor

...	
<PLACE>	
2 Shenzhen	1.000000
...	
< /PLACE>	
...	
<TERM>	
1 develop	0.271186
1 newli	0.016950
1 emerg	0.686441
1 rise	0.016950
1 come	0.008475
1 tourist	0.066390
1 journei	0.004149
1 tour	0.551867
1 tourism	0.087137
1 travel	0.273859
1 trip	0.016598
...	
< /TERM>	

Table 3.8: A sample of English terms glossy translated from Chinese story key terms shown in Table 3.7

for each translation can be calculated as:

$$U(\text{“}magnitude\text{”}) = \frac{4 * 2}{4 * 2 + 6 * 4} = 0.25$$
$$U(\text{“}importance\text{”}) = \frac{4 * 2}{4 * 2 + 6 * 4} = 0.75$$

Table 3.8 is a sample of gloss translation of Chinese story key terms shown in Table 3.7. The first column is the sentence number that a translated key term appears. The second column is the English translation term. The third column is the usage factor.

3.5 Term Weight Calculation

We use a four-dimensional vector space representation for each news story. The representation comprises of four components, namely, people name component $R_p(S)$, geographical location name component $R_l(S)$, organization name component $R_o(S)$, and content term component $R_c(S)$. Each component is represented by a set of weighted terms shown as follows:

$$R_p(S) = (w(S, p_1), w(S, p_2), \ldots)$$
$$R_l(S) = (w(S, l_1), w(S, l_2), \ldots)$$
$$R_o(S) = (w(S, o_1), w(S, o_2), \ldots)$$
$$R_c(S) = (w(S, c_1), w(S, c_2), \ldots)$$

where $w(S, p_i)$, $w(S, l_i)$, $w(S, o_i)$, and $w(S, c_i)$ represent the weights of the corresponding people name p_i, geographical location name l_i, organization name o_i, and content term c_i in the story S respectively. Each component contains the story key terms we have extracted in the previous step.

The weight of each term is determined by several factors. One factor is the term frequency defined as the number of occurrence of a term in the story. The term frequency is also adjusted by the relative location of the term in the content of the story. Another factor is the incremental document frequency. Precisely, we calculate the term weight as given in Equation 3.2.

$$w(S, t) = f(S, t)I(t) \tag{3.2}$$

where $w(S, t)$ is the weight of the term t in the story S; $f(S, t)$ is the adjusted term frequency of the story term t in the story S; $I(t)$ represents the inverse document frequency of the term t.

The way to calculate $f(S,t)$ is different according the source language of the story. If the story is in English, $f(S,t)$ is calculated by Equation 3.3.

$$f(S,t) = \sum_i (1 - \alpha \frac{K_i}{L(S)}) \tag{3.3}$$

where K_i represents the sentence number of the i-th appearance of the term t in the story S; $L(S)$ represents the total number of sentences in the story S; α is a parameter for controlling the contribution of the relative location information.

For a Chinese story, we use its gloss translation to represent it. The adjust term frequency $f(S,t)$ of a translated English term t is calculated as:

$$f(S,t) = \sum_i (1 - \alpha \frac{K_i}{L(S)}) U(t_i) \tag{3.4}$$

where $U(t_i)$ is the i-th appearance of t in story S. Note that, t may be generated from different Chinese terms. For example, "beautiful" may come from the English translation of "美丽" and "很好". Then, we will compute $f(S, \text{"}beautiful\text{"})$ by combining all usage factor together, which may be generated from different Chinese terms. We choose those terms with weights larger than a threshold to represent the story.

Table 3.9 shows an example of a story representation. Each term in the each story component is unique. A term identification (TERM-ID) is assigned to each term as shown in the first column. The weight of each term is shown in the second column.

3.6 Event And Topic Discovery

For the event and topic discovery task, we aim at discovering events and topics automatically from a large set of news stories. Most of the existing

<PEOPLE>		
...		
< /PEOPLE>		
<PLACE>		
2708	0.334226	Shenzhen
...		
< /PLACE>		
<ORG>		
...		
< /ORG>		
<TERM>		
2724	0.153617	sight-seeing
2729	0.146114	sight-see
2756	0.135658	umbilicu
2755	0.135658	pegui
2751	0.135658	epicentrum
2754	0.135658	omphalo
2748	0.135658	centric
2768	0.132998	sightse
2767	0.132998	rubber-neck
574	0.132883	burg
...		
< /TERM>		

Table 3.9: A sample of story representation after gloss translation

approaches only focus on the topic level. We investigate the challenge for the event and topic discovery problem.

We develop a hierarchical clustering algorithm for the discovery task, which uses a modified agglomerative centroid clustering for grouping news stories for events and grouping events for topics. Event and topic have similar representation with the representation a of news story. The similarity between event and news story is computed by the cosine-similarity measure. Relevance model is employed to calculate the relationship confidence between events and topics. Because we deal with multiple languages, we investigate a language normalization approach to balance the difference between the clustering properties of different languages. A time adjustment scheme is also applied to control the relationship between a story and an event with respect to time. The details will be presented in Chapter 4.

3.7 Story Link Detection

We investigate a link detection approach which employs an automatic topic type categorization model. The automatic topic type categorization model is trained for classifying a story to broad topic types so that we can assign some topic type characteristic to the news story. Specifically, it controls how to distribute the contribution of each component of the story representation.

We use the cosine-similarity measure to compute the similarity between the representations of two stories. We make use of a language normalization approach for dealing with news stories from multiple languages. The details of the story link detection module is presented in Chapter 6.

□ End of chapter.

Chapter 4

Event And Topic Discovery

4.1 Overview of Event and Topic discovery

The event and topic discovery task aims at discovering events and topics not previously known to the system. The definitions of event and topic are described in Section 1.1. Unsupervised learning based on text clustering is employed to conduct the discovery task. There are some existing methods on text clustering. A clustering algorithm called CBC (Clustering By Committee) introduced in [25], produces good quality clusters in the document clustering task. It initially discovers a set of tight clusters scattered in the similarity space. The algorithm proceeds by assigning elements to their most similar committee. Inderjit et al. described a local search procedure called "first-variation" that refines a given clustering by incrementally moving data points between clusters, thus achieving a higher objective function value [11]. Liu et al. proposed a clustering method that strives to achieve a high accuracy of clustering and the capability of estimating the number of clusters in the document corpus [22]. They employed a richer feature set to represent each document and used the Gaussian Mixture Model (GMM) together with the Expectation-Maximization (EM) algorithm to conduct an initial cluster-

ing. A self-refinement process via discriminative feature identification and cluster label voting is iteratively applied until the convergence of clusters. The model selection capability is achieved by introducing randomness in the cluster initialization stage, and then discovering a value C for the number of clusters N by which running the clustering process for a fixed number of times yields sufficiently similar results. Ding et al. provided a method for clustering high dimensional data using adaptive dimension reductions [12].

We employ a two level hierarchical unsupervised learning component to conduct the event and topic discovery. It can be divided into two main parts: event discovery component and topic discovery component. Both of the discovery components are based on agglomerative clustering. The similarity between a news story and an event is computed by the cosine-similarity measure [16], while we use relevance model to measure the difference between an event and a topic. Time adjustment scheme and language normalization scheme are also designed in the event discovery component.

The overview of the event and topic discovery approach is shown in Figure 4.1. Since news stories are arriving around-the-clock, the discovery system is designed so that it can conduct learning incrementally. Previously discovery result will be loaded into the event and topic discovery system to initialize the system. Then, new discovery can be conducted based on the loaded data. The input news stories are processed in chronological order. Each time we deal with a batch of news stories stored in source files. Each source file consists of about 25 to 40 stories over a specific period of time from a certain source. For example, a file named 19981231_1553_1649_APW_ENG.tkn indicates that the file contains news stories from 15:53 to 16:49 on December 31th in 1998 reported by Associated Press Worldstream Service. The release

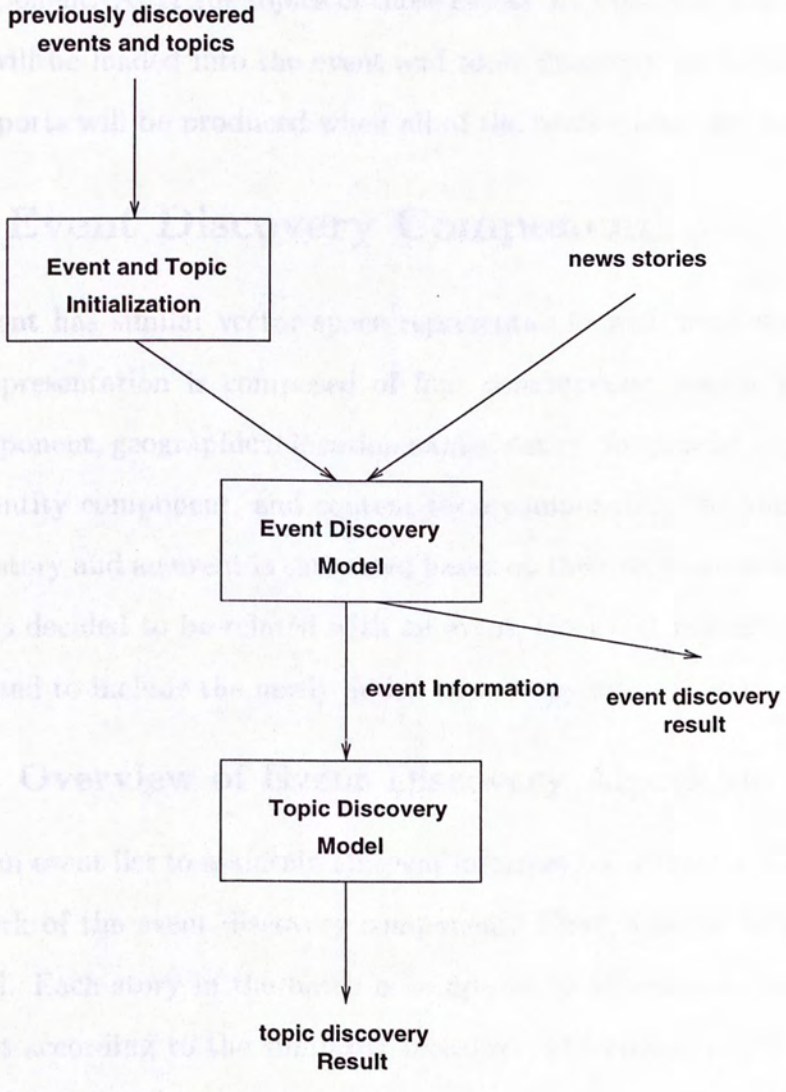

Figure 4.1: Overview of event and topic discovery component

time information of a news story will be used in the time adjustment scheme. The batch of stories will first be processed in the event discovery component. After that, the generated event information will be fed into the topic discovery component. After the topics of those events are obtained, a new batch of stories will be loaded into the event and topic discovery module. topic and event reports will be produced when all of the news stories are examined.

4.2 Event Discovery Component

Each event has similar vector space representation with news stories. The event representation is composed of four components: people named entity component, geographical location named entity component, organization named entity component, and content term component. The similarity between a story and an event is computed based on their representations. When a story is decided to be related with an event, the event representation will be updated to include the newly added news story information.

4.2.1 Overview of Event Discovery Algorithm

We use an event list to maintain all event information. Figure 4.2 depicts the framework of the event discovery component. First, a batch of news story is loaded. Each story in the batch is compared to all existing events in the event list according to the similarity measure. The closest event which has the highest similarity score with the story can be determined. If the final normalized similarity δ_f of the story to the closest event is larger than a user defined threshold θ_e, the story will be added to the event. In this case, the event representation, together with other event information such as the event release time and the event language indicator, will be updated. The event

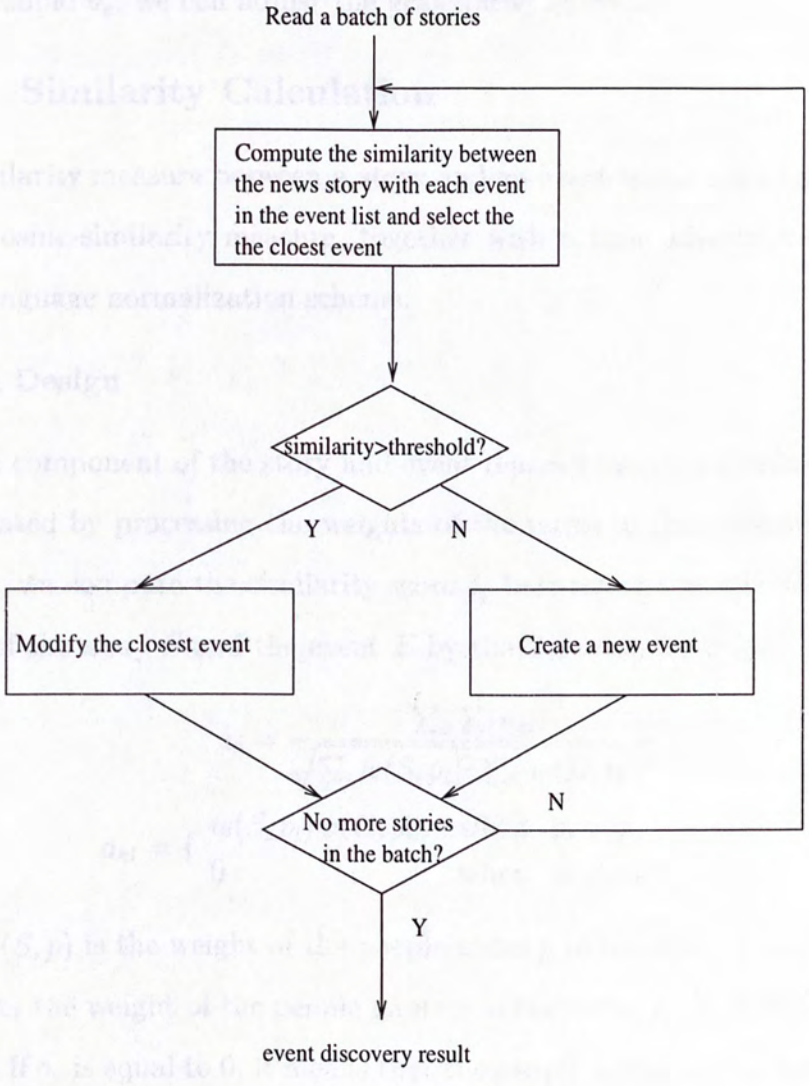

Figure 4.2: Design of the event discovery process

will be tagged as "updated" for the topic discovery process. Otherwise, the story will form a new cluster on its own representing a new event. The new event will be added to the event list and tagged as "updated". By changing this threshold θ_e, we can adjust the granularity of events.

4.2.2 Similarity Calculation

The similarity measure between a story and an event is calculated according to the cosine-similarity measure, together with a time adjustment scheme and a language normalization scheme.

Overall Design

For each component of the story and event representation, a similarity score is calculated by processing the weights of the terms in the component. For instance, we compute the similarity score δ_p between the people name component of the story S and the event E by the following formula:

$$\delta_p = \frac{\sum_k \sum_l a_{kl}}{\sqrt{\sum_l w(S,p_l)^2 \sum_k w(E,p_k)^2}} \tag{4.1}$$

$$a_{kl} = \begin{cases} w(S,p_l)w(E,p_k) & \text{when } p_l = p_k \\ 0 & \text{when } p_l \neq p_k \end{cases} \tag{4.2}$$

where $w(S,p)$ is the weight of the people name p in the story S and $w(E,p)$ represents the weight of the people name p in the event E. δ_p is in the range of [0, 1]. If δ_p is equal to 0, it means that the people named entity component of two representations are totally dissimilar. If δ_p is equal to 1, it means that the two components are the same.

We can compute the similarity score δ_l for the geographical location name component; δ_o for the organization name component; and δ_c for the content term component in a similar manner. The combined similarity, δ_a, is a

weighted sum of these similarity scores:

$$\delta_a = \delta_p W_p + \delta_l W_l + \delta_o W_o + \delta_c(1 - W_p - W_l - W_o) \tag{4.3}$$

where W_p, W_l, and W_o are the corresponding component weights. Note that W_p, W_l, and W_o are in range of [0,1]. $W_p + W_l + W_o$ is no more than 1. By adjusting these component weights, we can specify the relative contribution of each component to the final similarity. The higher the component weight, the more emphasis will be placed on the related component. For example, if we specify that W_p is equal to 0, it means that we do not consider people named entity component similarity at all. If we specify that W_p is equal to 1, while others are equal to 0, it means that we only consider people named entities.

Time Adjustment Scheme

It is common that stories reporting the same event are released in a short time window (e.g., several days). The number of the stories related to the event will drop drastically after a period of time. Therefore we introduce a time adjustment factor T to place more emphasis on the news stories that happen near the event release time. The time adjustment factor T is derived from the linear function as shown in Figure 4.3. Figure 4.3 gives the formula:

$$T = \begin{cases} 1.0 + \frac{|date_s - date_e|}{t}(L_p - 1) & \text{when } |date_s - date_e| < t \\ 0 & \text{when } |date_s - date_e| \geq t \end{cases} \tag{4.4}$$

where $date_s$ is the release date of a news story S; $date_e$ is the release time of an event E, which is calculated by the average release days of all stories belonging to the event; L_p is the time adjustment parameter. We assume that when a news story happens more than t days away from an event, it is not likely that the story is related to the event.

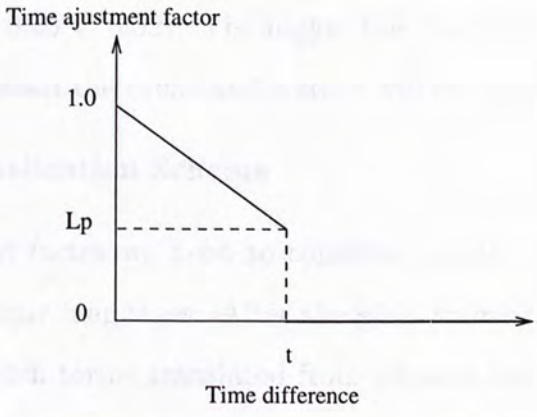

Figure 4.3: The linear function of time adjustment factor

As mentioned before, the release date of a news story can be obtained from the name of source file. For example, a story in a source file named 19981001_1553_1649_APW_ENG.tkn happened in the 271th (9*30 + 1) day of the year 1998. Suppose an event contains 3 member stories, which are released on the 262th, 264th, 266th day of the year 1998 respectively. Then the release time of the event is:

$$\begin{aligned} date_e &= \frac{262+264+266}{3} \\ &= 264 \end{aligned}$$

Therefore this event is assumed to be released on the 264th day of year 1998. By setting the time adjustment parameter, L_p to 0.5 and t to 10, we can calculate the time adjustment factor T as:

$$\begin{aligned} T &= 1.0 + \frac{|date_s - date_e|}{10}(L_p - 1) \\ &= 1.0 + \frac{|271-264|}{10}(0.5 - 1) \\ &= 0.65 \end{aligned}$$

Assume the combined cosine-similarity δ_a between the story and the event, is equal to 0.8. Then this δ_a will be modulated by the time adjustment factor T. It means that the similarity will be discounted by 0.65 and equals

to 0.52 (i.e. 0.8 * 0.65 = 0.52). The higher the time adjustment factor, the less similarity between the event and a story will be discounted.

Language Normalization Scheme

Another important factor we need to consider is that the news stories are coming from multiple languages. After the gloss translation process, distribution of the English terms translated from Chinese may be different from that of the English terms coming from English stories. Therefore, we employ a language normalization scheme to adjust the difference.

In order to compare the language difference between stories and events. We employ a language indicator l_s for each story. l_s is set to 1 for an English news story and 0 for a Chinese news story. The language indicator for an event is computed by the mean of the language indicators of all member stories.

From the event language indicator, we may know the language distribution of news stories related to the event. The normalization scheme employed to adjust the language difference is given in Equation 4.5:

$$\ell = \begin{cases} (1 - l_s) * g_c + (l_s - l_e) * g_m + l_e * g_e & \text{when } l_e < l_s \\ (1 - l_e) * g_c + (l_e - l_s) * g_m + l_s * g_e & \text{when } l_e \geq l_s \end{cases} \tag{4.5}$$

where l_s is the language indicator of news story S; l_e is the language indicator of event E; g_c is the Chinese normalization factor. It indicates the similarity discount when the news story comes from Chinese sources, and all of the stories related to the event are also from Chinese. g_e is the English normalization factor. It indicates the similarity discount when the news story comes from English sources, and all of the stories related to the event are also from English. g_m is the multilingual normalization factor. It indicates the similarity discount when the news story comes from Chinese sources and

all of the stories related to the event are from English, or when the news story comes from English sources and all of the stories related to the event are from Chinese.

For example, suppose that g_c is equal to 0.8; g_e is equal to 0.9; g_m is equal to 1, the language normalization factor ℓ between an English story S and a event E that contains two English stories and one Chinese story can be calculated as:

$$\begin{aligned} l_e &= \tfrac{1+1+1+0+0}{5} = 0.667 \\ \ell &= (1 - l_s) * g_c + (l_s - l_e) * g_m + l_e * g_e \\ &= (1 - 1) * 0.8 + (1 - 0.667) * 1 + 0.667 * 0.9 \\ &= 0.9333 \end{aligned}$$

In addition to the time adjust factor T, the combined similarity score δ_a will also be modified by the language normalization factor ℓ. The final similarity score δ_f is computed as given in Equation 4.6:

$$\delta_f = \delta_a * T * \ell \tag{4.6}$$

4.2.3 Story and Event Combination

When a story is decided to be related to an existing event, the story information needs to be included into the event. The event information needs to be updated to incorporate the event representation, the event release time and the event language indicator.

As mentioned above, an event has similar representation as a news story. The content of the event representation will be modified when a news story is added to the event cluster. The news story representation may contain some new terms to the event and some terms that the event representation has already covered. New terms, for instance t_i, will be inserted into the event

representation. The weight of the newly inserted term becomes $w(S, t_i)/(n+1)$, where n is the number of news stories that already exist in the event and $w(S, t_i)$ is the weight of term t_i in story S. The weight of existing term, say t_j, will be updated as $(w(E, t_j) * n + w(S, t_i))/(n+1)$, where $w(E, t_j)$ is the weight of term t_j in the event E before updating. The weight of other terms in the event, say t_k, will be $n * w(E, t_k)/(n+1)$.

The event release time is computed as the mean of the release time of all stories related to the event. The updated release time is $(date_e * n + date_s)/(n+1)$. The event language indicator is calculated by a similar method. The updated language indicator is $(l_e * n + l_s)/(n+1)$.

4.2.4 Event Discovery Output

We follow the topic evaluation method described in TDT2002 project to evaluate the topic and event result. The format of the event report and topic report is presented according to the evaluation requirement. A sample of the event output report is shown in Table 4.1.

The report has five columns, the first column is a unique index integer starts from one to indicate the event. The second column is the name of the source file to which the news story belongs. The third column is the starting "recid" of the story in the source file. From the source file name and the "recid", we can easily locate the news story from the corpus. The event discovery decision should be either YES or NO as described in the fourth column. YES means that the news story is related to the event. NO indicates that the news story is not related to the target event. The fifth column indicates the decision confidence which is obtained from the similarity score of the news story and the event.

	detection_system	YES	1	RECID
1	tkn/19981001_0023_1310_XIN_MAN.tkn	1	YES	0.02967
1	tkn/19981001_0023_1310_XIN_MAN.tkn	390	YES	0.02967
2	tkn/19981001_0023_1310_XIN_MAN.tkn	735	YES	0.06745
3	tkn/19981001_0023_1310_XIN_MAN.tkn	1128	YES	0.04077
3	tkn/19981001_0023_1310_XIN_MAN.tkn	1871	YES	0.04077
3	tkn/19981001_0023_1310_XIN_MAN.tkn	3006	YES	0.05925
1	tkn/19981001_0023_1310_XIN_MAN.tkn	3529	YES	0.05905
4	tkn/19981001_0023_1310_XIN_MAN.tkn	3793	YES	0.05346
5	tkn/19981001_0023_1310_XIN_MAN.tkn	4328	YES	0.05099
6	tkn/19981001_0023_1310_XIN_MAN.tkn	5021	YES	0.05144
7	tkn/19981001_0023_1310_XIN_MAN.tkn	5351	YES	0.02215
8	tkn/19981001_0023_1310_XIN_MAN.tkn	5933	YES	0.04395
9	tkn/19981001_0023_1310_XIN_MAN.tkn	6417	YES	0.02895
10	tkn/19981001_0023_1310_XIN_MAN.tkn	6740	YES	0.05963
3	tkn/19981001_0023_1310_XIN_MAN.tkn	7070	YES	0.03589
11	tkn/19981001_0023_1310_XIN_MAN.tkn	7893	YES	0.01732
8	tkn/19981001_0023_1310_XIN_MAN.tkn	8770	YES	0.04968
...				

Table 4.1: A sample of event report

4.3 Topic Discovery Component

After one batch of news stories have been processed, the event information will be fed to the topic discovery module for further learning the relationship between the events. We use the concept of topics to group related events. The topic representation is similar to the story and event representation mentioned above. An event is compared to a topic according to relevance models. The relevance model is derived from [20]. It attempts to expand the terms characterizing an event or a topic.

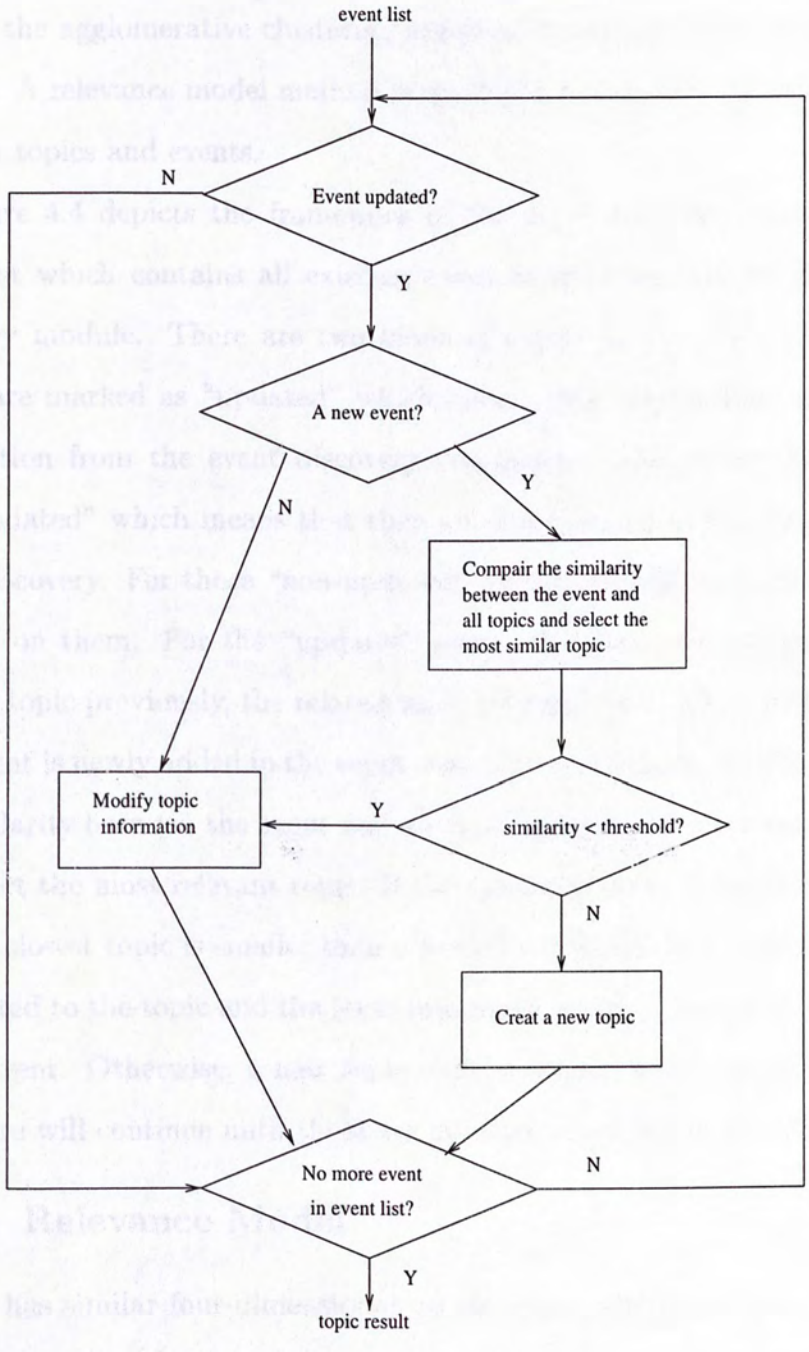

Figure 4.4: Design of the topic discovery process

4.3.1 Overview of Topic Discovery Algorithm

We use the agglomerative clustering approach to design the topic discovery process. A relevance model method is employed to compare the relationship between topics and events.

Figure 4.4 depicts the framework of the topic discovery module. The event list which contains all existing event information will be fed to the discovery module. There are two kinds of events in the event list. Some events are marked as "updated" which means that they contain new story information from the event discovery component. Others are marked as "non-updated" which means that they are not changed in the last batch of event discovery. For those "non-updated" events, we will keep the previous decision on them. For the "updated" event, if it has been assigned to an existing topic previously, the related topic information needs to be modified. If an event is newly added in the event discovery component, we will compute the similarity between the event and all existing topics by a relevance model and select the most relevant topic. If the similarity score between the event and the closest topic is smaller than a predefined threshold θ_t, the event will be inserted to the topic and the topic information will be modified according to the event. Otherwise, a new topic will be created in the topic list. The procedure will continue until there are no more event left in the event list.

4.3.2 Relevance Model

A topic has similar four-dimensional vector space representation as a story or an event. Because a topic involve a seminal event together with all related events, it contains a large amount of information. Therefore, it will not be enough just using the terms in the vector representation to compare

the relationship of an event and a topic. We use relevance model, a statistical language modeling technique related to query expansion, to enlarge the representation of topics and events. Important terms that associated with the topic and event are found even thought they do not appear in the vector representations.

Corpus Preparation

A background corpus containing a large number of news documents is first prepared and indexed by an IR engine. We used TDT2 corpus as described in Section 3.1 to build the background corpus. Since the quality of newswire news stories is better than that of the broadcast news stories, we choose the story key terms of the newswire stories instead of the whole corpus. The story key terms are extracted by the Information Extraction module described in Section 3.3. After that, key terms from the same news story will be combined. Table 4.2 shows a portion of the relevance corpus we prepared. Note that the terms are stemmed.

Using Relevance Model Retrieval

For each event, we use the terms in its vector representation as a query $(q_1, \ldots, q_k)$ for the background corpus. The news documents in the background corpus are ranked by $P(D|q_1, \ldots, q_k)$, where D is a document in the background corpus. But $P(D|q_1, \ldots, q_k)$ cannot be used directly because it is too close to zero. Consider the following formula:

$$P(D|q_1, \ldots, q_k) = \frac{P(q_1, \ldots, q_k|D)P(D)}{P(q_1, \ldots, q_k)} \tag{4.7}$$

Because $P(q_1, \ldots, q_k)$ and $P(D)$ are constants across the queries, so we make use of $P(D|q_1, \ldots, q_k)$ to determine the rank. Assume that $q_1, \ldots, q_k$ are

independent, we have

$$P(q_1, \ldots, q_k|D) = \prod_{i=1}^{k} P(q_i|D) \tag{4.8}$$

$$\tag{4.9}$$

where $\prod_{i=1}^{k} P(q_i|D)$ is very small because $P(q_i|D)$ is a small number less than 1. It will be difficult to extract any related document because the score cannot be comparable. To solve this problem, we use $\prod_{i=1}^{k} P(q_i|D)^{1/k}$ to represent $P(D|q_1, \ldots, q_k)$, because:

$$\prod_{i=1}^{k} P(q_i|D) \sim \prod_{i=1}^{k} P(q_i|D)^{1/k} \tag{4.10}$$

After the documents in the background corpus are ranked, we choose the top n documents and extract the terms in these n documents to form the relevance model M. There are a large amount of terms in these n documents, we select the top k terms with the highest related score to the relevance model $P(w|M)$, where $P(w|M) = \sum_{D \in M} P(w|D)P(D|M)$. $P(D|M)$ is approximately computed by $P(D|q_1, \ldots, q_k)$, which is estimated by $\prod_{i=1}^{k} P(q_i|D)^{1/k}$ instead. $P(w|D)$ is calculated using maximum likelihood estimate as shown in Equation 4.11

$$P(w|D) = \frac{f_{w,D}}{|D|} \tag{4.11}$$

where $f_{w,D}$ is the number of times w appears in document D, $|D|$ is the total number of terms of document D. The relevance model of topic is formed in a similar way.

Relevance Score Calculation

Armed with the relevance model, we can compute the similarity Δ between the relevance model of event M_E and the relevance model of each topic

M_A using Kullback-Leibler (KL) divergence measure. The KL divergence measure is a measure of the dissimilarity of two distributions. The smaller the value, the more similar the two relevance models are. The KL divergence measure is given below:

$$D(M_1||M_2) = \sum_w P(w|M_1)\frac{P(w|M_1)}{P(w|M_2)} \tag{4.12}$$

Because the KL divergence is asymmetric, we calculate Δ as follows:

$$\Delta = D(M_A||M_E) + D(M_E||M_A) \tag{4.13}$$

4.3.3 Event and Topic Combination

When an event is decided to be related to a topic, its representation information need to be combined into the topic. The topic representation will be used to form the relevance module for the similarity calculation. Each topic is represented by a centroid, which is an average of the vector representations of the events in the topic. Similar with the merging of events and stories, the event representation may contain some new terms and some terms that the topic representation has already covered. New terms, for instance t_i, will be inserted into the topic representation. The weight of the newly inserted term becomes $w(E, t_i)/(n+1)$, where n is the number of events that already exist in the topic and $w(E, t_i)$ is the weight of term t_i in event E. The weight of existing term, say t_j, will be updated as $(w(T, t_j) * n + w(E, t_i))/(n+1)$, where $w(T, t_j)$ is the weight of term t_j in the topic T before updating. For other terms, say t_k, in the topic T, the weight will be $n * w(T, t_k)/(n+1)$.

4.3.4 Topic Discovery Output

The topic report has similar format as the event report. A sample of a topic report is shown in Table 4.3. The first column of topic report is the event

indicator. The event indicator is a unique index integer starts from one to indicate the topic. Other columns are the same with those in the event topic.

□ **End of chapter.**

.I 1 APW19980104.0002
.W
Pol Pot Thailand Khmer Rouge Cambodia mysteri surround deepen sundai foreign minist claim leader fled China earlier chines diplomat deni ...
I 2 APW19980104.0012
.W
Seven Selkirk Mountains skier kill person miss avalanch hit separ ski parti southeast british columbia polic saturdai Kokanee Glacier bodi ...
.I 3 APW19980104.0020
.W
Pat Rafter Sweden Thomas Enqvist Australia U.S. play time open win beat 6-3 1-6 7-5 sundai hopman cup mix team tenni tournam Rafter Annabel ...
.I 4 APW19980104.0021
.W
China Xinjiang member gang kill peopl crime spree peopl execut murder robberi charg remot northwestern region offici newspap Xinjiang Daily ...
.I 5 APW19980104.0035
.W
polic rare sumatran tiger wild villag trap fear kill peopl year report sundai Sumatra Jakarta news resid villag fajar bulan island kilomet ...
...

Table 4.2: A portion of the background corpus for the relevance model

	detection_system	YES	1	RECID
1	tkn/19981101_0008_1121_XIN_MAN.tkn	1	YES	1.000000
2	tkn/19981101_0008_1121_XIN_MAN.tkn	464	YES	0.258833
3	tkn/19981101_0008_1121_XIN_MAN.tkn	858	YES	0.344798
4	tkn/19981101_0008_1121_XIN_MAN.tkn	2361	YES	0.435078
4	tkn/19981101_0008_1121_XIN_MAN.tkn	2861	YES	0.435078
3	tkn/19981101_0008_1121_XIN_MAN.tkn	3394	YES	0.344798
5	tkn/19981101_0008_1121_XIN_MAN.tkn	4580	YES	0.330094
6	tkn/19981101_0008_1121_XIN_MAN.tkn	4771	YES	1.000000
4	tkn/19981101_0008_1121_XIN_MAN.tkn	5238	YES	0.300382
7	tkn/19981101_0008_1121_XIN_MAN.tkn	5664	YES	0.218491
8	tkn/19981101_0008_1121_XIN_MAN.tkn	5908	YES	0.399329
9	tkn/19981101_0008_1121_XIN_MAN.tkn	6261	YES	0.256721
10	tkn/19981101_0008_1121_XIN_MAN.tkn	6588	YES	0.237190
8	tkn/19981101_0008_1121_XIN_MAN.tkn	7146	YES	0.399329
11	tkn/19981101_0008_1121_XIN_MAN.tkn	7394	YES	0.285169
12	tkn/19981101_0008_1121_XIN_MAN.tkn	7722	YES	0.258100
13	tkn/19981101_0008_1121_XIN_MAN.tkn	8185	YES	0.469067
14	tkn/19981101_0008_1121_XIN_MAN.tkn	8500	YES	0.332590
15	tkn/19981101_0008_1121_XIN_MAN.tkn	9037	YES	0.221480
...				

Table 4.3: A sample of topic report

Chapter 5

Event And Topic Discovery Experimental Results

We have conducted experiments on event and topic discovery. In this chapter, we present the evaluation methodology, parameter tuning process, and the discovery result.

5.1 Testing Corpus

We used TDT3 corpus to conduct the experiments on the event and topic discovery system. We have discussed the TDT3 corpus in Chapter 3.1. It contains 1,956 files. Each file includes about 40 stories. There are 43,612 stories in total covering three months data from October to December in 1998. The stories are arranged in chronological order. The topic discovery experiments were conducted on the whole TDT3 data. We selected news stories in November 1998 from the corpus containing 15,260 stories to evaluate the event discovery performance.

120 topics are annotated by NIST for topic discovery evaluation purpose. Appendix A gives the description of all topics. To evaluate the performance of event discovery, we manually collected sample events from the corpus. We

selected seven topics which contain 160 stories in total in November 1998. The seven topics are shown as follows:

- Anti-Doping Proposals

  The International Olympic Committee adopts a package of drug sanctions, and announces the formation of an anti-doping agency in 1998.

- North Korean Food Shortages

  Food crisis and famine in North Korea from winter 1995 to 1998.

- SwissAir111 Crash

  SwissAir Flight 111 crashes. The crash occurred on 9/2/98; the investigation continued through the fall of 1998.

- Michigan Prosecutes Kevorkian

  Dr. Jack Kevorkian is arrested and charged with murder, assisted suicide and delivery of a controlled substance in late November and early December, 1998.

- Taipei Mayoral Elections

  Taiwan's Nationalist Party claims victory in Taipei mayoral race in 1998.

- Shuttle Endeavor Mission for Space Station

  Space Shuttle Endeavor is sent into space on a mission to start assembling the international space station from 12/4/98 to 12/16/98.

- China Closes GITIC Bank

The People's Bank shuts down the Guangdong International Trust and Investment Corp. (GITIC) in October 6, 1998.

For each selected topic, we read all the November news stories and classified them into events. We obtained a list of 64 sample events. Some sample events related to the topic "Anti-Doping Proposals" are given as follows:

- 一名国际反兴奋剂专家戴·科文在11月 10号警告说,假如在悉尼奥运会之前不采用血液 检查的话,反兴奋剂的战争将面临失败的危险。
- French lawmakers adopted a bill that gave France one of the world 's toughest anti-doping laws on November 19, 1998.
- The fight against doping in sports got a million-dollar boost from the White House on November 25, 1998.
- All Olympic sports, except for soccer, tennis and cycling, agreed to a package of measures aimed at unifying the fight against banned drugs on November 27, 1998.

The full set of events are given in Appendix B.

5.2 Evaluation Methodology

We follow the topic detection evaluation method described in TDT2002 project [23] to evaluate the event and topic discovery results. Event discovery performance is evaluated by measuring the discovery performance separately for each event, in terms of "miss" and "false alarm". "Miss" means that a story is determined as not related to a certain event but actually it is. "False alarm" means that a story is decided to be related to an event but actually

it is not. Performance will be evaluated on a set of predefined sample events known as event keys. Each event key is mapped to an appropriate system output event to which it matches the best. The best matching system event is defined to be the one that produces the lowest detection cost. After this matching process, the event discovery performance is measured by the detection cost, which is a combination of the error probabilities. The detection cost is defined in Equation 5.1:

$$C_{cost} = C_m P_m P_{target} + C_f P_f P_{non-target} \tag{5.1}$$

where C_m and C_f are the cost of a miss and a false alarm respectively. P_m and P_f are the conditional probability of a miss and a false alarm respectively. P_{target} and $P_{non-target}$ are a priori target probabilities (Noted that $P_{target} = 1 - P_{non-target}$). C_{cost} is the lower bound of the discovery performance. Since the value of C_{cost} varies with the application, C_{cost} should be normalized so that C_{norm} can be no less than one without extracting information from the source data. The normalization formula is given as follows:

$$C_{norm} = \frac{C_{cost}}{\min(C_m P_{target}, C_f P_{non-target})} \tag{5.2}$$

C_{norm} is a cost metric. The lower the value, the better is the performance. The TDT2002 project specifies C_m as 1, C_f as 0.1, and P_{target} as 0.02. Hence the range of C_{norm} is in (0, 5.9).

There are two possible methods for estimating error probabilities, called story-weighted scheme and event-weighted scheme. For the story-weighted method, equal weight is assigned to each decision for each story, and errors are accumulated over all events. The event-weighted method accumulates errors separately for each event and then averages the error probabilities over events, with equal weight assigned to each event. The event-weighted

method is chosen for evaluation because the high variability in the number of stories per event. It is important to reduce the contribution of event variance by equalizing the contribution of different events.

The topic discovery performance can be evaluated using similar evaluation method as event discovery. Therefore, the detection cost is the same as in Equation 5.1. The only difference is that sample topics known as topic keys should be used instead of event keys. Similarly, the C_{cost} is normalized by Equation 5.2. Topic-weighted scheme is employed to estimate the detection error probabilities. Topic-weighted method is similar with the event-weighted method. It accumulates error separately for each topic and then averages the error probabilities with topics.

5.3 Experimental Results on Event Discovery

In this section, we present experimental results on event discovery. First, we have conducted a tuning process using a small amount of data to determine the language normalization factors, time adjustment parameter L_p, and similarity threshold θ_e. After that, we conducted event discovery experiment using the tuned parameters. We also compared our event discovery system with the one that does not use time adjustment scheme.

5.3.1 Parameter Tuning

We conducted the parameter tuning using the first ten days of news from November 1st to November 10th 1998. It contains 5,099 news stories. 1,638 of them are Chinese stories and others are English stories.

Language normalization factor tuning

Table 5.1 shows the language normalization factor tuning process. We ran 32 sets of language normalization factors. First we set multilingual language normalization factor g_m as 1.0, and varied Chinese language normalization factor g_c and English language normalization factor g_e with 0.4, 0.6, 0.8 and 1.0. The best performance, which corresponds to 0.0795, is achieved when g_c is 0.6 and g_e is 0.8. Then we set g_m as 0.8 and conducted the similar sets of runs. After all the runs were conducted, the best performance achieved when g_m is 1.0, g_c is 0.6, and g_e is 0.8.

Time adjustment parameter tuning

Table 5.2 shows the time adjustment parameter L_p tuning process. Note that the smaller the L_p, the more discount will be put to the similarity because of time difference. From Table 5.2, the best performance is 0.0656 when L_p is set to 0.2. We also tested the event discovery system that does not use the time adjustment scheme. All the other parameters are kept the same values. The performance is 0.1268 which is not as good as the result using the time adjustment scheme.

Similarity threshold tuning

We varied the similarity threshold θ_e in the tuning process. The performance is shown in Table 5.3. When θ_e equals to 0.2, it has the best performance, which is equal to 0.0656.

5.3.2 Event Discovery Result

After the parameters have been tuned, we have conducted event discovery on news stories from November 1998. The tuned parameters are as follows:

Language Normalization Factor (Chinese, English, Multilingual)			
(1.0,1.0,1.0)	(1.0,0.8,1.0)	(1.0,0.6,1.0)	(1.0,0.4,1.0)
0.0889	0.0839	0.0886	0.0935
(0.8,1.0,1.0)	(0.8,0.8,1.0)	(0.8,0.6,1.0)	(0.8,0.4,1.0)
0.0844	0.0837	0.0835	0.0859
(0.6,1.0,1.0)	(0.6,0.8,1.0)	(0.6,0.6,1.0)	(0.6,0.4,1.0)
0.0811	0.0795	0.0832	0.0856
(0.4,1.0,1.0)	(0.4,0.8,1.0)	(0.4,0.6,1.0)	(0.4,0.4,1.0)
0.1457	0.1234	0.1256	0.1344
(1.0,1.0,0.8)	(1.0,0.8,0.8)	(1.0,0.6,0.8)	(1.0,0.4,0.8)
0.0935	0.0887	0.0895	0.1023
(0.8,1.0,0.8)	(0.8,0.8,0.8)	(0.8,0.6,0.8)	(0.8,0.4,0.8)
0.0998	0.0877	0.0910	0.0932
(0.6,1.0,0.8)	(0.6,0.8,0.8)	(0.6,0.6,0.8)	(0.6,0.4,0.8)
0.1034	0.0995	0.1003	0.1098
(0.4,1.0,0.8)	(0.4,0.8,0.8)	(0.4,0.6,0.8)	(0.4,0.4,0.8)
0.1332	0.1118	0.1228	0.1359

Table 5.1: Performance measured by C_{norm} on event discovery under different language normalization factor sets in the tuning process. Note that the lower the cost, the better is the performance.

Chinese normalization factor g_c is set to 0.6; g_e is set to 0.8; g_m is set to 1.0; L_p is set to 0.2; θ_e is set to 0.2. The event discovery result as measured by C_{norm} is 0.0986.

We further investigate the effect of θ_e on the discovery performance as shown in Table 5.4 and Figure 5.1.

In Figure 5.1, the value of P_f is enlarged 100 times so that it is in the same range with P_m and C_{norm}. With the increasing of threshold θ_e, the granularity of event is cut down. P_f decreases and P_m increases generally.

Time adjustment parameter L_p	Discovery Performance
0.1	0.0663
0.2	0.0656
0.3	0.0795
0.4	0.0891
0.5	0.0920
0.6	0.0918
0.7	0.0928

Table 5.2: Performance measured by C_{norm} on event discovery under different time adjustment parameter in the tuning process. Note that the lower the cost, the better is the performance.

Similarity Threshold	Discovery Performance Cnorm
0.10	0.1120
0.15	0.1270
0.20	0.0656
0.25	0.0853
0.30	0.0877
0.35	0.0910
0.40	0.0972

Table 5.3: Performance measured by C_{norm} on event discovery under different similarity threshold θ_e in the tuning process. Note that the lower the cost, the better is the performance.

Similarity Threshold θ_e	P_m	P_f	Discovery Performance C_{norm} (Standard Deviation)
0.10	0.0904	0.0017	0.0987 (0.733)
0.15	0.1052	0.0011	0.1107 (0.213)
0.20	0.0944	0.0009	0.0986 (0.206)
0.25	0.1072	0.0006	0.1102 (0.217)
0.30	0.1235	0.0004	0.1252 (0.238)

Table 5.4: Performance measured by C_{norm} on event discovery under different similarity threshold θ_e, Note that the lower the cost, the better is the performance. The standard deviation of C_{norm} of is shown in the bracket.

When θ_e is equal to 0.2, it has the best performance.

We further conducted an experiment to compare our event discovery system with the one that does not use the time adjustment scheme. We conducted the investigation by also varying the language normalization factors. The results of the comparison is shown in Table 5.5. The event discovery system with time adjustment scheme overall performs better than the system that does not use the time adjustment scheme.

5.4 Experimental Results on Topic Discovery

In our topic discovery method, besides the parameters that exist in the event discovery component, there is another parameter needed to be determined, namely, the relevance similarity threshold θ_t. We have conducted a parameter tuning process using the first five days data of TDT3 corpus from October 1st to October 5th 1998. It contains 2,245 news stories. 717 of them are Chinese stories and others are English stories.

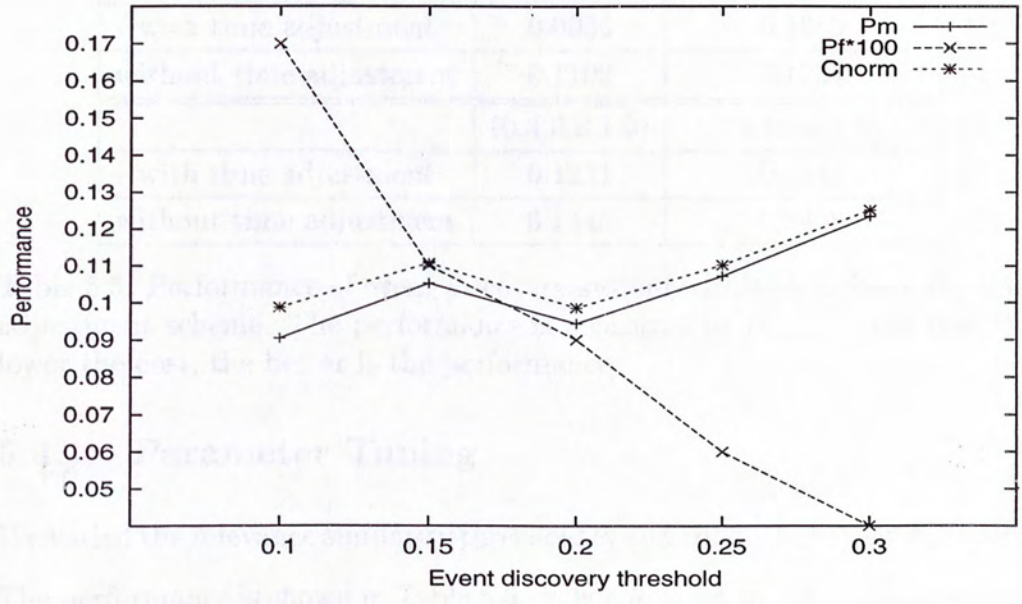

Figure 5.1: Performance measured by C_{norm} on event discovery under different similarity threshold θ_e. Note that the lower the cost, the better is the performance.

With or Without time adjustment	Language Normalization Factor (Chinese, English, Multilingual)	
	(1.0,0.8,1.0)	(1.0,0.6,1.0)
with time adjustment	0.1097	0.1299
without time adjustment	0.1155	0.1258
	(0.8,0.8,1.0)	(0.8,0.6,1.0)
with time adjustment	0.0993	0.1096
without time adjustment	0.1109	0.1254
	(0.6,0.8,1.0)	(0.6,0.6,1.0)
with time adjustment	0.0986	0.1059
without time adjustment	0.1102	0.1234
	(0.4,0.8,1.0)	(0.4,0.6,1.0)
with time adjustment	0.1231	0.1245
without time adjustment	0.1445	0.1498

Table 5.5: Performance of event discovery system with and without the time adjustment scheme. The performance is measured by C_{norm}. Note that the lower the cost, the better is the performance.

5.4.1 Parameter Tuning

We varied the relevance similarity threshold θ_t and conducted event discovery. The performance is shown in Table 5.6. When θ_e is set to 0.2, C_{norm} performs best, which is equal to 0.4299.

5.4.2 Topic Discovery Results

After the parameters have been tuned, we conduct event discovery on news stories for the whole TDT3 corpus. The tuned parameters are as follows: Chinese normalization factor g_c is set to 0.6; g_e is set to 0.8; g_m is set to 1.0; L_p is set to 0.2; θ_e is set to 0.2; and θ_t is set to 0.2. The topic discovery result as measured by C_{norm} is 0.6238.

Similarity Threshold θ_t	Discovery Performance
0.15	0.4301
0.20	0.4299
0.25	0.4533
0.30	0.4795
0.35	0.5037

Table 5.6: Performance measured by C_{norm} on topic discovery under different similarity threshold θ_t in the tuning process. Note that the lower the cost, the better is the performance.

Similarity Threshold θ_t	P_m	P_f	Discovery Performance C_{norm}
0.10	0.6007	0.0062	0.6311
0.15	0.6012	0.0048	0.6247
0.20	0.6027	0.0043	0.6238
0.25	0.6102	0.0045	0.6323

Table 5.7: Performance measured by C_{norm} on topic discovery under different similarity threshold θ_t, Note that the lower the cost, the better is the performance

We further investigate the effect of θ_t on the discovery performance as shown in Table 5.7 and Figure 5.2.

In Figure 5.2, the value of P_f is enlarged 100 times so that it is in the same range with P_m and C_{norm}. With the decreasing of threshold θ_t, the granularity of event is cut down. P_f decreases and P_m increases generally. When θ_t is equal to 0.2, it has the best performance.

□ End of chapter.

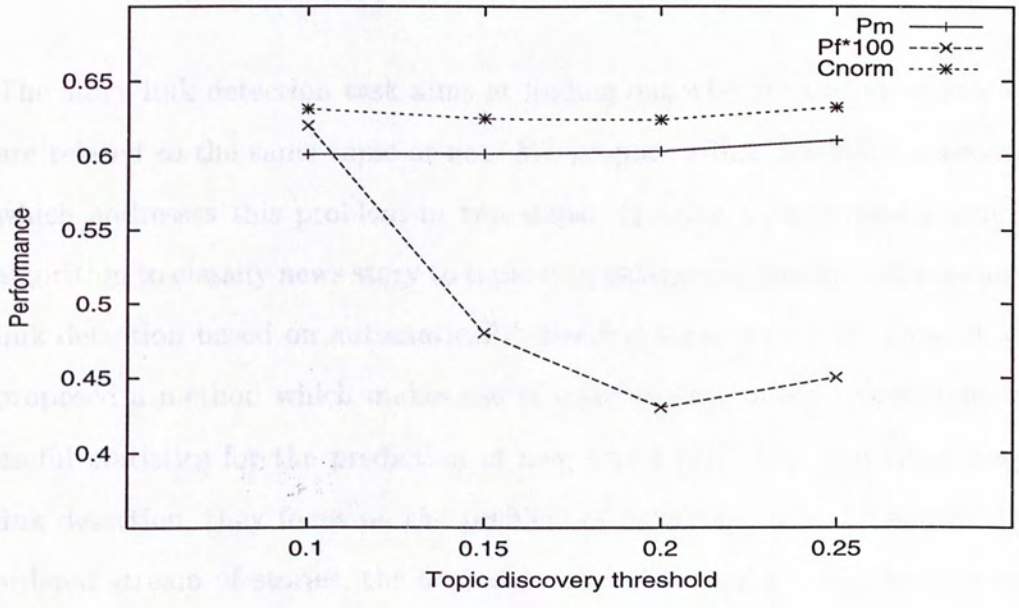

Figure 5.2: Performance measured by C_{norm} on topic discovery under different similarity threshold θ_t. Note that the lower the cost, the better is the performance.

Chapter 6

Story Link Detection

The story link detection task aims at finding out whether two given stories are related to the same topic or not. We propose a link detection approach which addresses this problem in two steps: 1) using a supervised learning algorithm to classify news story to topic type categories, and 2) perform story link detection based on automatically classified topic type [15]. Yang et al. proposed a method which makes use of training data of old topics to learn useful statistics for the prediction of new topics [33]. Different from story link detection, they focus on the problem of detecting, in a chronologically ordered stream of stories, the first story that discusses a topic. Stories are automatically routed to the corresponding topic by the classifier at the first level before they are sent to the second level for novelty detection. In this chapter, we are investigating the story link detection problem.

6.1 Topic Types

The TDT 2002 Annotation Guide [24] provides twelve broad topic types with corresponding rules of interpretation in order to give a guideline on what constitutes "related" topics. These twelve topic types are like some

general topic categories. Topics generally fall into these general categories. The twelve topic types and some sample topics under each topic type are illustrated in Table 6.1.

The stories related to the same topic type share some characteristics, especially on how different kinds of named entities should be emphasized. For example, a story related to "Accidents" may emphasize more on the geographical location named entity while the people named entity may play a more important role in the stories related to "Celebrity/Human Interest News". Our link detection approach uses an automatic topic type categorization method to decide how to distribute the weight on each kind of named entity and story term component in the story representation. We use the cosine similarity measure to compare two stories. Similarity will be calculated for each component of the story representation. The final similarity score is a weighted sum of each component similarity. Each component weight is determined based on the result of automatic topic type categorization. The higher the component weight, the more emphasis is to the corresponding part of representation.

6.2 Overview of Link Detection Component

Figure 6.1 shows the framework of our story link detection component. First, each story is automatically classified into 12 topic types. Topic type categorization scores, which indicate the degree that the story is related to a topic type, are produced. After that, component weights are calculated according to topic type categorization scores. Categorization model of each topic type for each language, is learned from a training corpus via a supervised learning method beforehand. The supervised learning algorithm that we em-

Topic Type	Sample Topic Under a Topic Type
Elections	Taipei Mayoral Elections: Taiwan's Nationalist Party claims victory in Taipei mayoral race in December, 1998.
Scandals /Hearings	Olympic Bribery Scandal: Bribery is admitted in Salt Lake City's bid to host the 2002 Olympic Games in December, 1998
Legal/Criminal Cases	Pinochet Trial: Pinochet, who ruled Chile from 1973 -1990, is arrested on charges of genocide and torture during his reign, 1998.
Natural Disasters	Hurricane Mitch: Hurricane Mitch, which forms in late September 1998, forms over warm ocean waters, killing thousands and causing millions of dollars in damage.
Accidents	Nigerian Gas Line Fire: An explosion and fire erupted in a damaged government owned gasoline pipeline, killing over 1000 people in October 17, 1998.
Acts of Violence or War	Indonesia/East Timor Conflict: Pro-independence groups in East Timor clash with Indonesian military forces in November and December, 1998
Science and Discovery News	AIDS Vaccine Testing Begins: The first full-scale human trials of AIDS vaccine, Aidsvax, 1998
Financial News	Euro Introduced: The Euro, a new common currency of Europe, is introduced on January 1, 1999
New Laws	Anti-Doping Proposals: The International Olympic Committee adopts a package of drug sanctions, and announces the formation of an anti-doping agency in November, 1998
Sports News	ATP Tennis Tournament: 1998 Shanghai Open
Political and Diplomatic Meetings	Tony Blair Visits China: Tony Blair visits mainland China and Hong Kong from October. 6th to 10th, 1998
Celebrity/Human Interest News	Joe DiMaggio Illness: Dimaggio spends 99 days in the hospital for lung cancer and pneumonia treatments from October 21st, 1998 to January 18th, 1999

Table 6.1: Topic types and sample topics under each topic type

ploy is Support Vector Machines (SVM) due to its good performance on text categorization [17].

The similarity of the story pair is computed by augmenting the cosine similarity measure by the component weights on the two story representations. Since we process news stories from multilingual news stories, we make use of a language normalization scheme to deal with the difference in similarity measure for different language pairs. If the final normalized similarity is larger than a user-defined threshold θ_l, the story pair is decided to be on the same topic, otherwise, they are not related. By changing this link detection threshold θ_l, we can adjust the sensitive of our link detection system. After all the story pairs have been processed, a link detection report will be generated.

6.3 Automatic Topic Type Categorization

An automatic text categorization technique is used for classifying each story to a set of topic types. For each topic type, there are two categorization models, which are trained from two training corpora produced from two language sources, English and Chinese.

6.3.1 Training Data Preparation

We used TDT2 corpus provided by DARPA and NIST to prepare the training data for topic type categorization. As described in Section 3.1, TDT2 corpus contains news articles collected daily from six English sources and three Mandarin sources in the period of January 1998 through June 1998. There are 100 topics manually identified by NIST. Each topic contains a set of stories in the corpus and is classified to one of twelve topic types. We

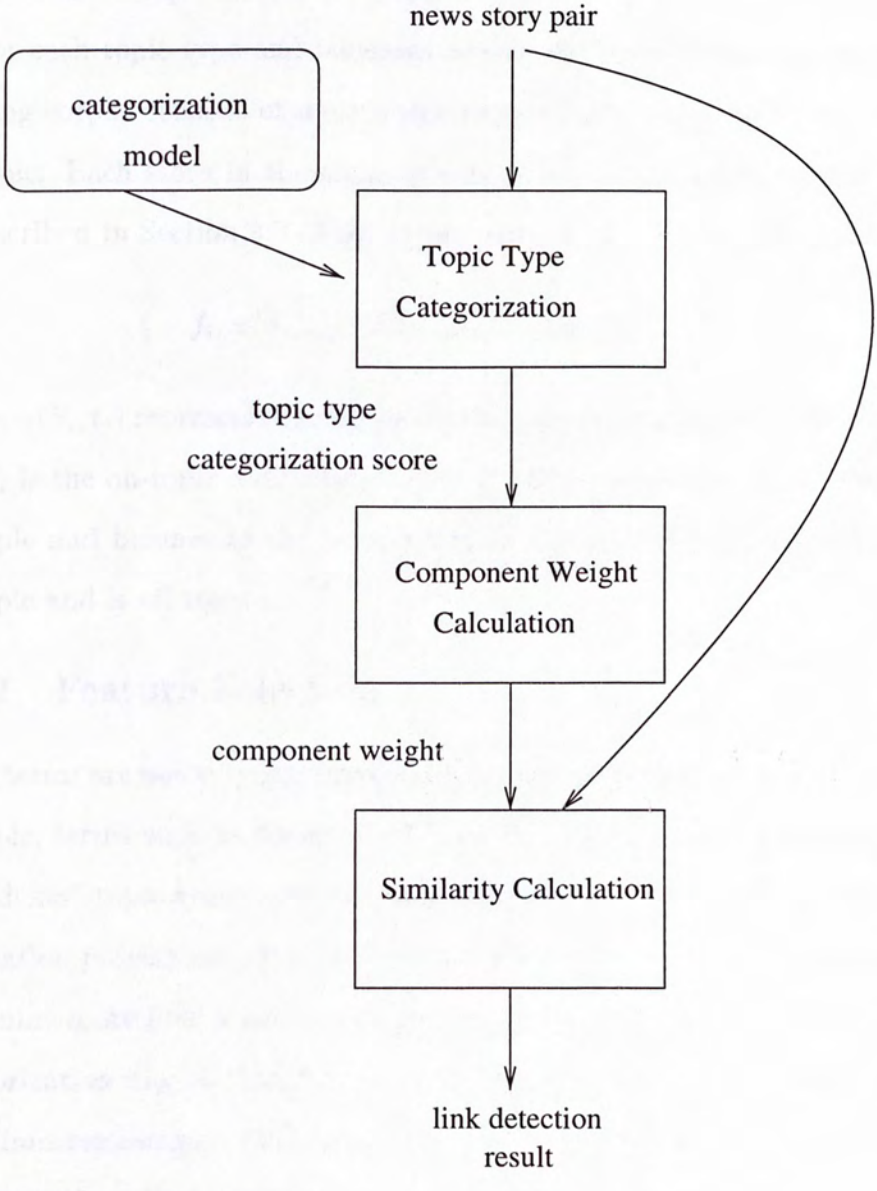

Figure 6.1: Overview of the story link detection component

collected those stories in the topics that are related to a topic type and form a set of on-topic stories for that topic type. Other stories in the corpus are regarded as off-topic for the topic type.

For each topic type and language source, we build a training corpus. A training corpus consists of sample stories which are known to be on-topic or off-topic. Each story in the training corpus is represented in a similar way as described in Section 3.5. The corpus representation is as follows:

$$(\quad f_i, w(S_i, t_1), w(S_i, t_2), ..., w(S_i, t_m) \quad) \tag{6.1}$$

where $w(S_i, t_j)$ represents the weight of the corresponding term t_j in story S_i, and f_i is the on-topic indicator of story S_i. If f_i equals to 1, S_i is a positive example and belongs to the topic type. If f_i equals to -1, S_i is a negative example and is off-topic.

6.3.2 Feature Selection

Some terms are not very informative to the categorization of topic types. For example, terms such as "Senate", "Congress" are not very indicative to the "Accidents" topic type. Also the high dimension of data makes the text categorization process not effective. We employ a feature selection algorithm to determine more informative terms and select those useful terms to build the categorization model. The feature selection algorithm used in our approach is the information gain (IG) measure [30], which measures the number of bits of information obtained for category prediction by knowing the presence or absence of a term in a document. The IG is calculated as:

$$\begin{aligned} G(t) = \; & -P_r(h) log P_r(h) \\ & +P_r(t) P_r(h|t) log P_r(h|t) \\ & +P_r(\bar{t}) P_r(h|\bar{t}) log P_r(h|\bar{t}) \end{aligned} \tag{6.2}$$

where t is a unique term that appears in the corpus. h represents the on-topic story in the training corpus. $P_r(h)$ represents the probability of the on-topic story in the training corpus. $P_r(h|t)$ represents the probability of on-topic story in the stories that contain t. $P_r(t)$ is the probability of the term t appears in the whole training corpus. $P_r(\bar{t})$ is the probability of term t that does not appear in the training corpus. $P_r(h|\bar{t})$ is the probability of on-topic story in the stories not containing t.

With the information gain method, we obtain a feature score for each term for each topic type and language. After that, we rank the terms in descending order according to the scores. Those terms with high feature scores will be extracted to build the training data.

6.3.3 Training and Tuning Categorization Model

Support Vector Machines (SVM) is used to train a categorization model. There is a useful parameter in SVM training known as cost factor. This cost factor controls the degree by which training errors on positive examples outweigh errors on negative examples. We explore a suitable value for this cost factor via a parameter tuning process.

In order to conduct parameter tuning, we divide each training corpus into two parts. The first part is used for training a model. The second part is used for evaluating the classification performance of the trained model so as to facilitate the tuning process. To train a model, a value is chosen for the cost factor and the training process is invoked. The classification performance of the trained model is measured by F-measure, which is defined as:

$$F = \frac{2(Precision)(Recall)}{Precision + Recall} \tag{6.3}$$

where F denotes F-measure; $Precision$ is the percentage of the stories clas-

sified positive that are really on-topic; *Recall* measures the percentage of the on-topic stories that are correctly classified. We varied the cost factor and conducted the training process repeatedly. We chose the value of cost factor that has the best classification performance and used it to train the final classifier with the whole set of training data.

6.4 Link Detection Algorithm

6.4.1 Story Component Weight

The topic type categorization will be incorporated into the link detection method. Specifically, the story component weights are calculated taking into account of the topic type categorization scores. Given a pair of stories S_1 and S_2. The steps of computing the story component weight are:

1. Categorize each story to the set of twelve topic types and get the score m_{ij} $(i = 1, 2; j = 1, 2, \ldots, 12)$, which stands for the degree of the story belonging to a particular story type. i specifies a story in a story pair S_1 and S_2. j stands for a twelve story type. If m_{ij} is negative which means that the story i is totally not related to topic type j, we assign m_{ij} as zero.

2. Normalize each m_{ij} with a normalization score. We test each topic type category with TDT2 news stories and use the average of the scores related to each topic type category as the normalization constant. m_{ij} is divided by this normalization constant.

3. Combine the topic type scores of two stories by taking the average.

The combined score m_k $(k = 1, 2, \ldots, 12)$ is calculated as follows.

$$m_i = \frac{m_{1i} + m_{2i}}{2} \tag{6.4}$$

4. For each topic type category, we define a set of component weight W_{p_k},W_{l_k}, W_{o_k}, W_{c_k} $(k = 1, 2, \ldots, 12)$ for each component of representation according to the nature of the topic type. W_{p_k},W_{l_k}, W_{o_k}, W_{c_k} are all in the range of [0,1] and satisfy:

$$W_{p_k} + W_{l_k} + W_{o_k} + W_{c_k} = 1 \quad (k = 1, 2, \ldots, 12) \tag{6.5}$$

The higher the component weight, the more emphasis will be place on the corresponding representation component.

5. The weight on the each story representation component is then allocated according to the combined score m_k and W_{p_k}, W_{l_k}, W_{o_k}, W_{c_k}, $(k = 1, 2, \ldots, 12)$ as follows:

$$W_p = \begin{cases} \frac{\Sigma_k m_k W_{p_k}}{\Sigma_q m_q} & \text{when } \sum_q m_q > 0 \\ W_{pd} & \text{when } \sum_q m_q = 0 \end{cases} \tag{6.6}$$

where m_k is the combined topic type score of the story pair. W_{pd} is the predefined component weight for the stories which are related to none of the topic type categories. W_p is the combined weight for the people name components of the story pairs. Similarly, we can get the combined weight of the geographical location name components W_l, the combined weight of organization name component W_o and the combined weight of content term component W_c. W_{pd} is calculated as the mean of W_{p_k} for each topic type as follows:

$$W_p = \frac{\Sigma_k W_{p_k}}{12} \tag{6.7}$$

6.4.2 Story Link Similarity Calculation

Our link detection system uses the cosine similarity score to measure the similarity between two stories [19]. The similarity formula is similar to Equation 4.2 described in Section 4.2.2. We make use of a language normalization scheme to deal with the difference in similarity measure for different language pairs.

We design three language normalization factors. The final similarity δ_f is given as follows:

$$\delta_f = \delta_b * N \tag{6.8}$$

where δ_f is the final similarity score of the two stories; N is the language normalization factor. We examine the nature of stories in the story pair. Then we set the value of N as N_c, N_e or N_m according to the nature of the stories. The three language normalization factors are:

- Chinese Normalization Factor N_c

  It is used to compute the final similarity when both of the stories in the story pair are Chinese stories.

- English Normalization Factor N_e

  It is used to compute the final similarity when both of the stories in the story pair are English stories.

- Multilingual Normalization Factor N_m

  It is used to compute the final similarity when one of the story in the story pair is English story and the other story in the story pair is Chinese story.

An example of the language normalization is shown in Figure 6.2. Suppose that N_c is 0.6; N_e is 0.8; and N_m is 1.0. There are four stories. Two of them are Chinese and the other two are English. These four stories form four story pairs. The similarity of the Chinese story pair is 0.8. The similarity of the English story pair is 0.9. The similarity of the two multilingual story pair is 0.7 and 0.6. Then the final similarities are 0.48, 0.72, 0.7, and 0.6 respectively. If we set the threshold θ_l as 0.65, the English story pair and the multilingual story pair which contains Chinese Story 1 and English Story 2 are considered as on-topic. The other two story pairs are not.

6.5 Story Link Detection Output

After all the story pairs have been processed, we evaluate the story link detection result by the TDT 2002 evaluation method. The format of the story link detection report is shown in Figure 6.2.

The first two columns are the story pointer to the two stories, which correspond to the file name and the story name. The third column is the decision of whether the two stories are related. YES means that the two stories are related. NO indicates that they are not related. The fourth column is the confidence score of the decision. The higher the confidence score, the more likely the decision is correct.

□ End of chapter.

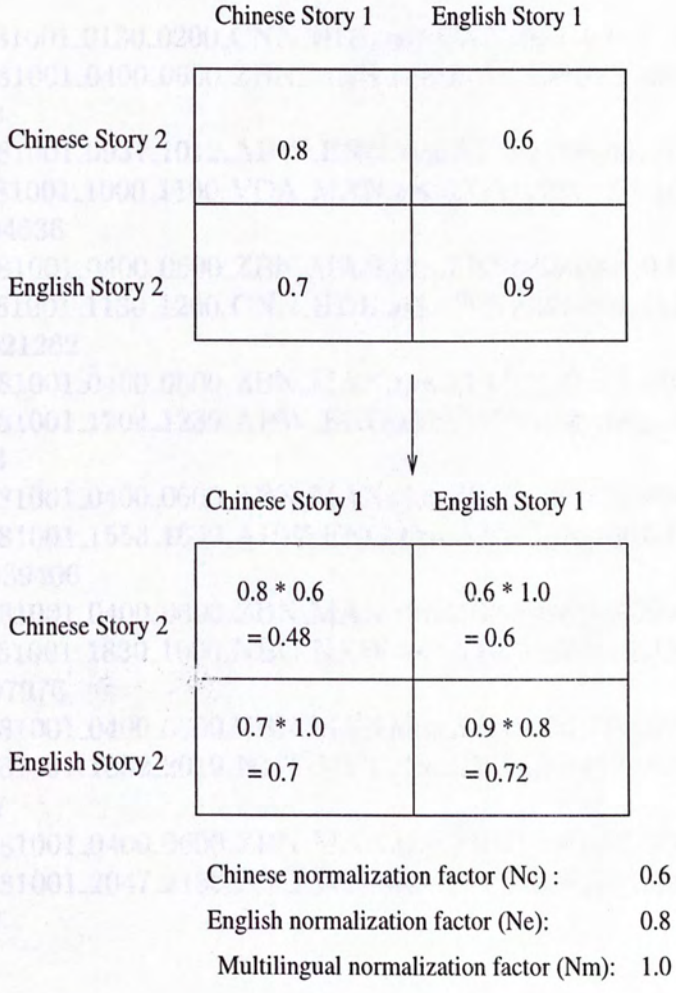

Figure 6.2: An example of language normalization scheme

as1/19981001_0130_0200_CNN_HDL.as1:CNN19981001.0130.0433
tkn/19981001_0400_0600_ZBN_MAN.tkn:ZBN19981001.0001 NO
0.002203
tkn/19981001_0931_1012_APW_ENG.tkn:APW19981001.0598
as0/19981001_1000_1100_VOA_MAN.as0:VOM19981001.1000.1974
NO 0.004536
tkn/19981001_0400_0600_ZBN_MAN.tkn:ZBN19981001.0001
as1/19981001_1130_1200_CNN_HDL.as1:CNN19981001.1130.1030
YES 0.021262
tkn/19981001_0400_0600_ZBN_MAN.tkn:ZBN19981001.0001
tkn/19981001_1204_1239_APW_ENG.tkn:APW19981001.0837 NO
0.002263
tkn/19981001_0400_0600_ZBN_MAN.tkn:ZBN19981001.0001
tkn/19981001_1553_1649_APW_ENG.tkn:APW19981001.1184
YES 0.039406
tkn/19981001_0400_0600_ZBN_MAN.tkn:ZBN19981001.0001
as1/19981001_1830_1900_NBC_NNW.as1:NBC19981001.1830.0438
NO 0.007976
tkn/19981001_0400_0600_ZBN_MAN.tkn:ZBN19981001.0001
tkn/19981001_1832_2019_NYT_NYT.tkn:NYT19981001.0321 NO
0.008507
tkn/19981001_0400_0600_ZBN_MAN.tkn:ZBN19981001.0001
tkn/19981001_2047_2156_NYT_NYT.tkn:NYT19981001.0424 NO
0.013525
...

Table 6.2: A sample of the story link detection report

Chapter 7

Link Detection Experimental Results

In order to evaluate our story link detection result, we have conducted experiment on the TDT3 corpus. In this chapter, we introduce the evaluation method. We also describe our experiment settings and analyze the link detection result in the following sections.

7.1 Testing Corpus

We used TDT3 corpus to test the story link detection approach. The TDT3 corpus has been described in Chapter 3.1. It contains the news stories collected daily from October to December in 1998 from 11 news sources. In our experiment, we used the native language newswire text and the audio broadcast news. The news sources come from multiple languages, including English and Chinese. There are a total of 43,612 stories in the TDT3 corpus. 12,336 are in Chinese and 31,278 are in English. 23,205 news stories are newswire text and 20,407 are broadcast news stories. Evaluation is done on a set of story pairs sufficient to provide reliable estimates of error probability. 13,613 pairs of stories are selected from TDT3 corpus to test the link detec-

tion performance. 4,440 of them are Mandarin Chinese story pairs; 4,604 of them are English story pairs; and the rest are multilingual story pairs. The correct answer is prepared by NIST for evaluating purpose.

7.2 Topic Type Categorization Result

Recall that there is an automaitc topic type categorization task in the link detection system. In this section, we will report the categorization performance. The link detection performance will be described in subsequent sections.

The performance of the topic type categorization for English News is shown in Table 7.1. The best performance of the topic type categorization is 0.9767 corresponding to the topic type "Accidents". In general, the categorization performance is very good.

Topic Type	F-measure
Elections	0.9036
Scandals/Hearings	0.9493
Legal/Criminal Cases	0.9752
Natural Disasters	0.9589
Accidents	0.9767
Acts of Violence or War	0.9248
Science and Discovery News	0.9456
Financial News	0.8851
New Laws	0.7272
Sports News	0.9598
Political and Diplomatic Meetings	0.9294
Celebrity/Human Interest News	0.8654

Table 7.1: F-measure performance of topic type categorization after the tuning process for English news.

Table 7.2 depicts the topic type categorization performance for Mandarin

news. Some topic types have very good categorization performance such as "Elections", "Financial News", and "Sports News". Some topic types, such as "Legal/Criminal Cases" and "Celebrity/Human Interest News" are not as good due to insufficient positive examples in the training corpus. In fact, three topic types, namely, "Science and Discovery News", "New Laws" and "Political and Diplomatic Meetings" have no positive examples in the training data. Hence they are not used in the training process.

Topic Type	F-measure
Elections	0.9670
Scandals/Hearings	0.8235
Legal/Criminal Cases	0.4444
Natural Disasters	0.8750
Accidents	0.8235
Acts of Violence or War	0.9157
Financial News	0.9538
Sports News	0.9848
Celebrity/Human Interest News	0.6667

Table 7.2: F-measure performance of topic type categorization after the tuning process for Mandarin news.

7.3 Link Detection Evaluation Methodology

We follow the link detection evaluation method described in TDT2002 project [23]. The performance is evaluated in terms of their ability to determine whether specified pairs of stories discuss the same topic. The evaluation method for link detection is measured by the detection cost which is similar with the one for event and topic discovery. The detection cost is given in Equation 5.1. Then it will be normalized by Equation 5.2. Topic-weighted scheme is employed to estimate the detection error probabilities.

7.4 Experimental Results on Link Detection

We have conducted a tuning process to determine the language normalization parameter using a small amount of data. We conducted link detection for the testing corpus using the tuned parameters. We also compared our link detection system with and without topic type categorization.

7.4.1 Language Normalization Factor Tuning

There are three kinds of story pairs. The first kind is that both of the stories are in Chinese. The second kind is that both of the stories are in English. The third kind is that one of the story is in Chinese and the other is in English. Recall that there are three language normalization factors, namely, N_c, N_e and N_m. To determine the appropriate values of these factors, we conducted a tuning process using a small amount of data.

We used the first ten days of news from October 1st to October 10th 1998 for tuning. The data set contains 857 pairs of Chinese stories, 925 pairs of English stories, and 879 pairs of multilingual stories. We predefine two sets of component weights for each topic type categorization according to the nature of the topic type. The first set of component weight emphasizes more on the story content term while the second set puts more weights on the named entities. The first set of component weight is shown in Table 7.3. The second set of component weight is shown in Table 7.4.

We first set the language normalization factors, namely, Nc, N_e, and N_m all equal to one. Then we conducted the link detection runs on the story pairs by varying the link detection threshold θ_l. Table 7.5 and Figure 7.1 show the performance on the set of Chinese story pairs. When the link detection threshold θ_l is equal to 0.195, both of the two component sets achieve the

Topic Type	component weight set one			
	people named entity	location named entity	organization named entity	content term
Elections	0.1	0.2	0.1	0.6
Scandals/Hearings	0.15	0.1	0.15	0.6
Legal/Criminal Cases	0.2	0.1	0	0.7
Natural Disasters	0	0.2	0	0.8
Accidents	0	0.2	0	0.8
Acts Violence or War	0.05	0.15	0.05	0.75
Science and Discovery News	0.05	0	0.05	0.9
Financial News	0	0.1	0.1	0.8
New Laws	0	0.1	0	0.9
Sports News	0.05	0.1	0.15	0.7
Political and Diplomatic Meetings	0.15	0.1	0	0.75
Celebrity/Human Interest News	0.2	0.1	0	0.8

Table 7.3: First set of component weights for topic types

Topic Type	component weight set two			
	people named entity	location named entity	organization named entity	content term
Elections	0.15	0.2	0.15	0.5
Scandals/Hearings	0.15	0.15	0.15	0.55
Legal/Criminal Cases	0.15	0.15	0.05	0.65
Natural Disasters	0.05	0.3	0	0.65
Accidents	0.05	0.3	0	0.65
Acts of Violence of War	0.1	0.2	0.1	0.6
Science and Discovery News	0.15	0.1	0.05	0.7
Financial News	0.05	0.15	0.15	0.65
New Laws	0.05	0.15	0.1	0.7
Sports News	0.15	0.15	0.15	0.55
Political and Diplomatic Meetings	0.2	0.15	0.05	0.5
Celebrity/Human Interest News	0.25	0.1	0	0.65

Table 7.4: The second set of component weights for topic types

best performance.

link detection threshold	component weight set one	component weight set two
0.175	0.2434	0.3840
0.180	0.2400	0.2434
0.185	0.2333	0.2333
0.190	0.2330	0.2333
0.195	0.2097	0.2226
0.200	0.2232	0.2232

Table 7.5: Language normalization factor tuning on Chinese story pairs. The performance is measured by C_{norm}. The lower the cost, the better is the performance.

Table 7.6 and Figure 7.2 shows the performance on the set of English story pairs. When the link detection threshold θ_l is equal to 0.065, both of the two component sets achieve the best performance.

link detection threshold	component weight set one	component weight set two
0.045	0.1548	0.1548
0.050	0.1493	0.1493
0.055	0.1495	0.1495
0.060	0.1468	0.1468
0.065	0.1418	0.1440
0.070	0.1529	0.1529

Table 7.6: Language normalization factor tuning on English story pairs. The performance is measured by C_{norm}. The lower the cost, the better is the performance.

Table 7.7 and Figure 7.3 shows the performance on a set of multilingual story pairs. When the link detection threshold θ_l is equal to 0.06, both of the two component sets achieve the best performance.

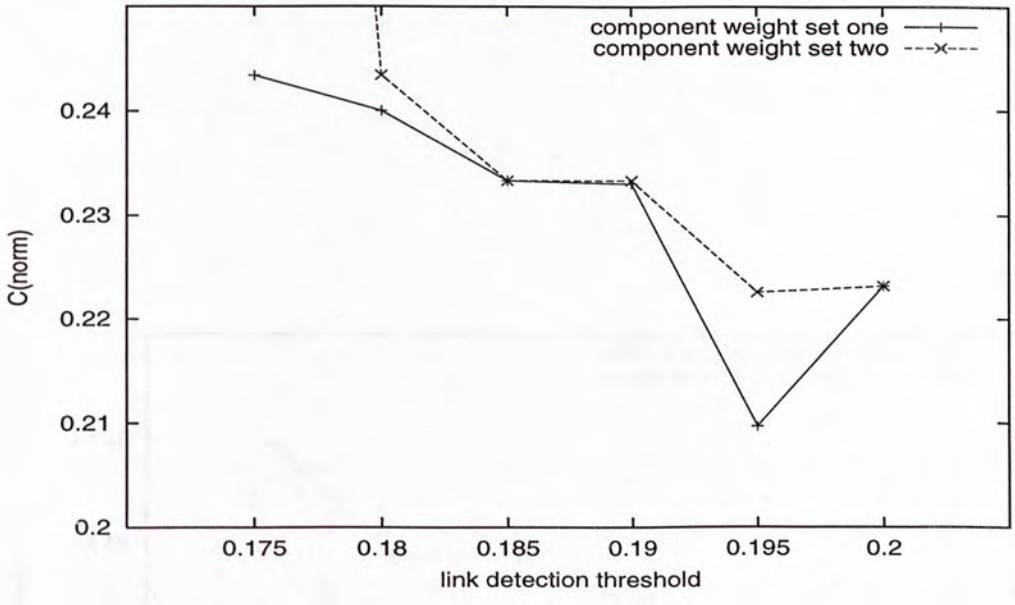

Figure 7.1: Language normalization factor tuning on Chinese story pairs. The performance is measured by C_{norm}. The lower the cost, the better is the performance.

link detection threshold	component weight set one	component weight set two
0.040	0.4541	0.4541
0.045	0.4395	0.4395
0.050	0.4346	0.4346
0.055	0.4254	0.4254
0.060	0.3431	0.3343
0.065	0.3523	0.3453

Table 7.7: Language normalization factor tuning on multilingual story pairs. The performance is measured by C_{norm}. The lower the cost, the better is the performance.

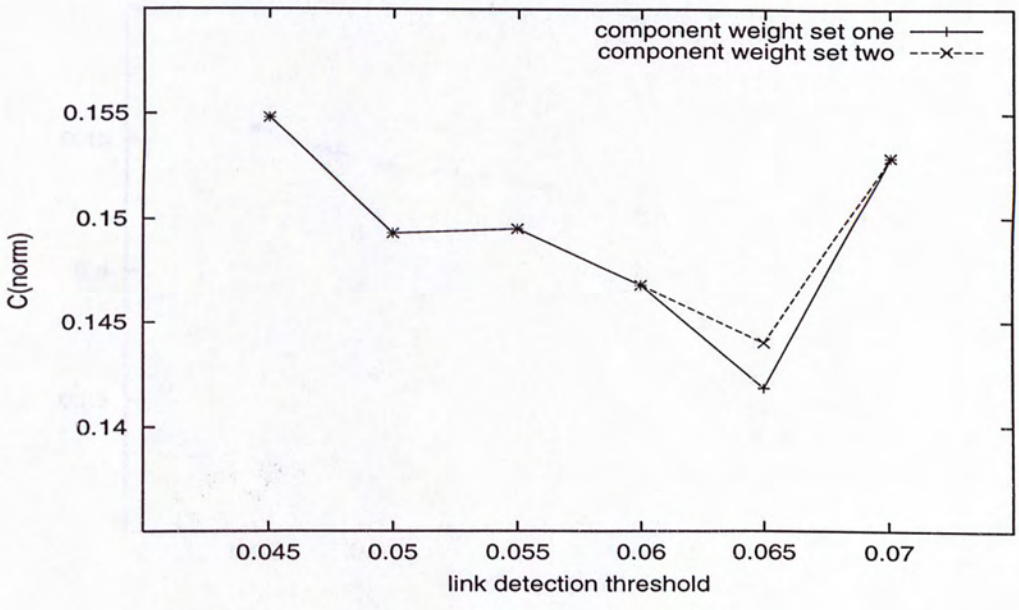

Figure 7.2: Language normalization factor tuning on English story pairs. The performance is measured by C_{norm}. The lower the cost, the better is the performance.

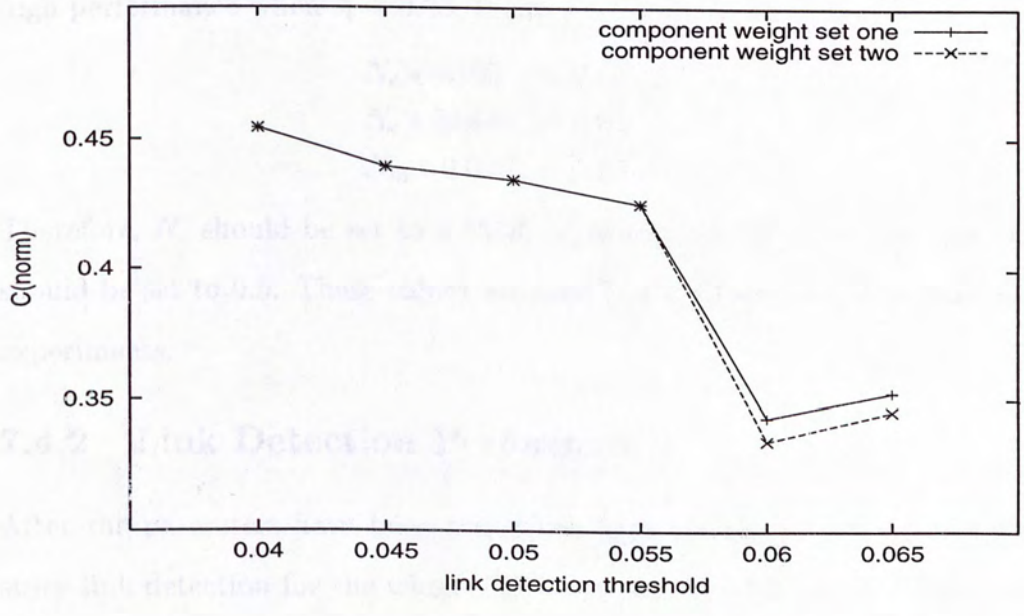

Figure 7.3: Language normalization factor tuning on multilingual story pairs. The performance is measured by C_{norm}. The lower the cost, the better is the performance.

From the above analysis, it illustrates that each type of story pairs achieves the best performance at the same link detection threshold regardless of the component weight set. But three types of story pairs achieve the best performance when different link detection thresholds are used. If we need to process news that contains three types of story pairs, we have to balance these difference between the languages so that the three types of story pairs can get their best performance at the same θ_l. Suppose we want to achieve high performance when $\theta_l = 0.03$, then:

$$\begin{aligned} N_c * 0.195 &= 0.03 \\ N_e * 0.065 &= 0.03 \\ N_m * 0.060 &= 0.03 \end{aligned}$$

Therefore, N_c should be set to 0.1538, N_e should be set to 0.4615, and N_m should be set to 0.5. These values are used in the subsequent link detection experiments.

7.4.2 Link Detection Performance

After the paramters have been tuned, we have conducted the multilingual story link detection for the whole TDT3 corpus. We ran our link detection system and compared with the link detection performance without automatic topic type categorization. We used the first set of component weight because its overall performance is better than the second set. The parameter values we used were determined in the tuning process as described in Section 7.4.1. The link detection threshold was varied from 0.01 to 0.05. The component weight of the link detection system without the topic type categorization is predefined as shown in Equation 6.7. The performance of the two link detection systems is shown in Table 7.8 and Figure 7.4. The result shows that the link detection system with topic type categorization performs better

under different link detection thresholds. The best performance is obtained with link detection threshold θ_l set to 0.03 which is adjusted by the language normalization scheme.

link detection threshold θ_l	without topic type categorization	with topic type categorization
0.01	0.5196	0.5213
0.02	0.5062	0.4997
0.03	0.4782	0.4635
0.04	0.5059	0.4954
0.05	0.5768	0.5413

Table 7.8: Performance of the link detection system with and without the automatic topic type categorization method. The performance is measured by C_{norm}. Note that the lower the value, the better is the performance.

7.4.3 Link Detection Performance Breakdown

We have conducted further experiments on each type of story pairs. The parameter settings are similar to the above experiment on the link detection on the whole TDT3 corpus.

Table 7.9 shows the performance on the set of Chinese story pairs. Table 7.10 shows the performance on the set of English story pairs. The performance of multilingual story pairs are depicted in Table 7.11. All of them achieve best performance when θ_l is set to 0.03. The set of Chinese pairs gives a better performance compared to the other two sets. Multilingual story pairs have the worst performance. The reason is that the difference of translated English stories and the English stories coming from native sources. Since we apply a bilingual lexicon to do the gloss translation, the glossary for translated English is quite limited. This brings difficulty to match a

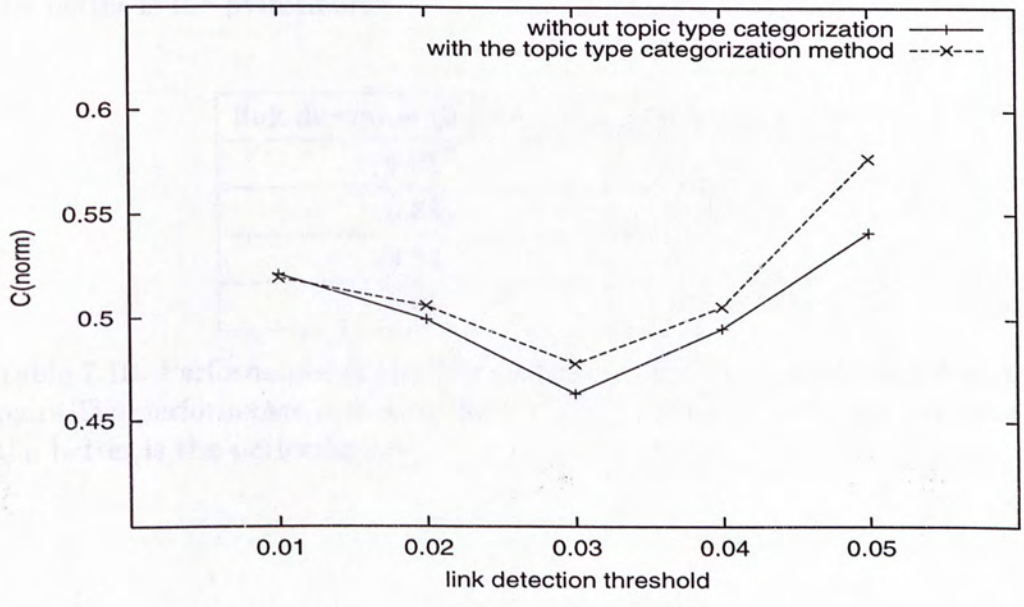

Figure 7.4: Performance of the link detection system with and without the automatic topic type categorization method for. The performance is measured by *Cnorm*. Note that the lower the value, the better is the performance.

translated English term to a native English term.

link detection threshold θ_l	Performance
0.02	0.3790
0.03	0.3742
0.04	0.3681
0.05	0.3682

Table 7.9: Performance of the link detection system on set of Chinese story pairs. The performance is measured by C_{norm}. Note that the lower the value, the better is the performance.

link detection threshold θ_l	Performance
0.02	0.5104
0.03	0.4590
0.04	0.4754
0.05	0.4793

Table 7.10: Performance of the link detection system on set of English story pairs.The performance is measured by C_{norm}. Note that the lower the value, the better is the performance.

□ **End of chapter.**

link detection threshold θ_l	Performance
0.02	0.5328
0.03	0.5151
0.04	0.5494
0.05	0.6323

Table 7.11: Performance of the link detection system on set of multilingual story pairs.The performance is measured by C_{norm}. Note that the lower the value, the better is the performance.

Chapter 8

Conclusions and Future Work

In this chapter, we make a summary on our research and conclude our contributions. Moreover we suggest some possible future research directions

8.1 Conclusions

In this thesis, we have developed methods on handling the event and topic discovery problem as well as the story link detection problem. The goal of the event and topic discovery task is to discover new events and topics from real-time incoming news stories from diverse sources. An event includes a set of stories. A topic is composed of a set of related events. We develop a two-level hierarchical unsupervised learning approach for the discovery task. At the lower level, events are discovered from the the incoming news stories. At the higher level, topics are detected from the generated event information. A vector space representation scheme composed of different kinds of named entities and important content terms are designed for stories, events and topics. The similarity between a news story and and an event is computed by the cosine-similarity measure. Since topics and events may include a large amount of information, a query expansion technique, called relevance

model, is employed to determine the relationship of an event and a topic. Language normalization scheme is used in the event discovery step to balance the difference between the clustering properties of different languages. Since the stories related to the same event are usually happened within a short period of time, a time adjustment scheme is applied to control the relationship between a story and an event with respect to the time.

The experimental results reveal that the best event discovery performance is 0.0986, obtained with a set of parameters tuned by a small amount of data. Moreover, the event discovery system with time adjustment scheme has better performance than the one does not use the time adjustment scheme.

The story link detection system aims at determining whether two stories are related to the same topic or not. An innovative characteristic of our link detection approach is that it makes use of an automatic topic type categorization method to classify a story to some general topic types. Different emphasis can be placed on different parts of story representation during link detection process based on the categorization result. A language normalization scheme is also designed. The experiment results of story link detection reveal that the our link detection system performs better than the one that does not use topic type information.

8.2 Future Work

There are several possible directions to extend our research:

- Currently the event and topic discovery system employs unsupervised learning algorithm to discover system unknown events and topics from a set of stories. It will have additional advantage if the system can not

also deal with user defined events and topics, and track the incoming stories into those events and topics.

- The stories related to the same topic type categories may share some characteristics. In the story link detection, we use the topic type classification information to determine the component weight for each story representation. In addition to the representation component weights, there are many other useful information we can obtain from the topic type categorization. One possible direction is that we can learn more representative terms.

- Gloss term translation is conducted so that we can directly conduct unsupervised learning for Chinese and English stories. We make use of a parallel corpus to adjust the weight of each English translation. Instead of gloss term translation, we will investigate context-based translation. Term disambiguation will be conducted with the help of concurrence statistic information collected from the story context.

- In the event and topic discovery approach, we assume that each story belongs to only one event and each event belongs to only one topic. We will extend the discovery approach so that each story can belong to multiple events and each event can belong to multiple topics.

□ End of chapter.

Appendix A

List of Topic Title Annotated for TDT3 corpus by LDC

1. Cambodian Government Coalition
2. Hurricane Mitch
3. Pinochet Trial
4. Houston Chukwu Octuplets
5. Osama bin Laden Indictment
6. NBA Labor Dispute
7. Congolese Rebels vs. Pres. Kabila
8. November APEC Summit
9. Anti-Doping Proposals
10. Car Bomb in Jerusalem
11. Anwar Ibrahim Case
12. Leonid Meteor Shower
13. Dalai Lama Visits US
14. Nigerian Gas Fire
15. October Holbrooke-Milosevic Meeting

16. SwissAir 111 Crash
17. North Korean Food Shortages
18. Blair Visits China in October
19. Hong Kong Mob Boss Cheung Tze-Keung
20. 13th Asian Games
21. Thai Airbus crash
22. Chinese Labor Activists
23. Kevorkian Trial
24. Gingrich Resigns
25. Brazilian Elections
26. AOL-Netscape Merger
27. Russian Currency Crisis
28. Turkey-Syria Tension
29. Australian Yacht Race
30. Taipei Mayoral Elections
31. Space Shuttle Launch
32. China Closes ITIC Bank
33. The Euro Introduced
34. Indonesia-East Timor Violence
35. China-Taiwan Meetings
36. Nobel Prizes Awarded
37. Israeli Foreign Minister Sharon Appointed
38. Salt Lake City Olympic Bid
39. Sharif and Clinton Meet About Pakistan
40. Gaza International Airport Opened

41. Jiang's Historic Visit to Japan
42. PanAm Bombing Trial
43. Lankan Gov't. vs Tamil Rebels
44. Kurd Separatist Abdullah Ocalan Arrested
45. Mobil-Exxon Merger
46. House Speaker-Elect Livingston Resigns
47. Space Station Module Zaria Launched
48. IMF Bailout of Brazil
49. North Korean Nuclear Facility
50. Mid-term Elections
51. Bosnian War Crimes Tribunal
52. Typhoon Zeb
53. Clinton's Gaza Trip
54. China Human Rights Treaty
55. D'Alema's New Italian Government
56. Chechnya Rebel Violence
57. India Train Derailment
58. Energy Sec'y. Richardson Visits Taiwan
59. Russian Politico Starovoitova Assasinated
60. Hyundai Corp. Aids North Korean Economy
61. S. Africa Truth & Reconciliation Committee Report
62. Pope Visits Balkans
63. Capitol Shooter Indictment
64. Columbian Air Force Drug Scandal
65. Zapatistas & Mexican Gov't talks

66. Immigrant Smuggling Ring
67. Dominique Moceanu vs. Parents
68. Matthew Shepard Murder
69. New Turkish Gov't: Bulent Ecevit
70. European Cold Wave
71. Typhoon Babs
72. Princess Di Crash Investigation
73. Abortion Doctor Slepian Killed
74. Chinese Missile Scientist Arrested
75. Japan Apology to Korea
76. ATP Shanghai Open
77. Chretien Visits China
78. Zorig Killed (Mongolian Politics)
79. AIDS Vaccine Testing
80. Ukraine Mining Disasters
81. Fossilized Dinosaur Embryos Found
82. Swedish Dance Hall Fire
83. Kyoto Energy Protocol
84. South Africa Weapons Purchase
85. Kenyan Teachers on Strike
86. Yankees vs. Padres in World Series
87. Iranian National Elections
88. Hurricane George in Carribbean
89. Azerbaijani Presidential Elections
90. Jesse 'the Body' Ventura

91. US Federal Budget
92. Yeltsin's Illness
93. Microsoft Anti-Trust Case
94. New Orleans Sues Handgun Manufacturers
95. G-7 World Finance Meeting
96. Joe DiMaggio Illness
97. Japanese and Russian Leaders Meet
98. American Embassy Bombing Trial
99. China Denies Bugs
100. slamic Extremists Sentenced
101. Philippine Airlines Closes
102. New Paris Subway Line
103. Lebanon Elects New President
104. Merceded/Chrysler Merger
105. Chinese Army Ordered to shut down Industry
106. Buddhist Seeks Asylum
107. Environmentalist Hill in a Tree
108. South Korean Pres. Visits China
109. Australian PM 'apologizes' to Aboriginies
110. China will Not Allow Opposition Parties
111. Beijing Applies to Host 2008 Olympics
112. Florida Law to Reduce Divorce Rates
113. South Korean Vets Escape NK
114. ASEAN Meeting
115. US and Pakistan Settle Dispute

116. South Korean Police Forced into Temple
117. Chinese Dissident Wei Visits Taiwan
118. Tonga Switches Diplomatic Relations to China
119. Hundreds Protest Financial Scandal in China
120. Japan Political Coalition

□ **End of chapter.**

Appendix B

List of Manually Annotated Events for TDT3 Corpus

1. 一名国际反兴奋剂专家戴·科文在11月 10号警告说,假如在悉尼奥运会之前不采用血液 检查的话,反兴奋剂的战争将面临失败的危险。
2. French lawmakers adopted a bill that gave France one of the world's toughest anti-doping laws on November 19, 1998.
3. The fight against doping in sports got a million-dollar boost from the White House on November 25th, 1998.
4. All Olympic sports, except for soccer, tennis and cycling, agreed to a package of measures aimed at unifying the fight against banned drugs on November 27th, 1998.
5. Analyzing the possible reason for the air accident, there was an increasing chance a commercial airline flight could be interrupted by something as simple as an odor.
6. An in-house publication of the airline said the temperatures rose to 300 degrees (570 degrees F) without leaving traces of fire in the front part of Swissair Flight 111 before it crashed on November 5th 1998.
7. The airline's chief executive said that Swissair "did everything correctly" in installing a state-of-the-art enter tainment system switched off last month in the wake of the crash of Flight 111, November 22nd, 1998.

8. "The Washington Post" reported that an agency subcommittee received a memo from a home insulation expert in 1988 calling the FAA's insulation flammability test "meaningless". The subcommittee reportedly ignored that warning.

9. The Federal Aviation Administration abruptly ordered airlines that fly MD-11s to inspect two cockpit switches because one model in wide use can give off smoke at certain settings on November 12th, 1998.

10. After the crash of Swissair Flight 111 , the airline gave the grieving families of the victims all the help customarily offered in such disasters.

11. Investigators looking into the crash of Swissair flight 111 said that they had discovered one of the three engines was not working when the plane crashed on November 20th, 1998.

12. More than two months have passed since Swissair flight 111 crashed off novia Scotia, killing all 229 people aboard. For the families and for investigators, finding the cause has been painfully slow.

13. American families who lost loved ones when Swissair flight 111 crashed off Nova Scotia returned to the crash scene on November 27, 1998 to remember those who died.

14. A North Korean man arrived in Seoul and sought asylum after escaping his hunger-stricken homeland on November 4th, 1998.

15. North Korea is entering its fourth winter of chronic food shortages with its people malnourished and at risk of dying from normally curable illnesses, November, 1998.

16. 北韩领袖金正日呼吁大规模发展经济，提高陷入饥荒的并韩人的生活水平。

17. Despite catastrophic hunger at home , North Korea plans to send 317 athletes and officials to next month 's Asian Games in Thailand , South Korean officials said on November 19th, 1998.

18. North Korea may be cheating on an agreement to freeze it 's huge nuclear weapon program at a time when the people are starving , November , 1998.

19. When a South Korean cruise ship sailed into the North Korean port of Changjon in late November 1998, passengers could hear the shouts of drilling soldiers carry clearly on the cold winter air.

20. 由于遭受严重的自然灾害，朝鲜今年粮食欠收，呼吁国际社会继续提供援助。

21. 选民们不仅要投票在候选人中进行选择，而且还要对一些颇为引起争论的问题做出抉择，密歇根州选民在投票中要决定的提案，是应不应该将医生协助病人自杀予以合法化。

22. "60 minutes" broadcasted a film clip of Dr. Jack Kevorkian as assisting a terminally ill patient to commit suicide, November 22nd, 1998.

23. Kevorkian attempted to challenge the statute prohibiting assisted suicide in the state of Michigan.

24. The tape broadcasted on "60 minutes," prompted strong reaction.

25. George Annas, the chairman of the health and law department of Boston University school of public health, and Derek Humphrey, author of six books on the subject of euthanasia, discussed the assisted suicide assisting on November 23rd, 1998.

26. In the pile-on were Roman Catholic prelates, medical ethicists, editorial writers , television reviewers and teachers of journalism , all expressing indignation , shock , outrage that the pre-eminent news magazine should have run a video of Dr. Jack Kevorkian giving lethal injections to Thomas Youk , who suffered from Lou Gehrig 's disease.

27. Kevorkian would be charged with first-degree murder as a result of his latest death on November 25th, 1998.

28. A Boston couple knew the joy of a life reclaimed because of new equipment on some airplanes November 25th, 1998.

29. Jack Kevorkian was free on $ 750,000 bail after he was charged with first degree premeditated murder in the televised death of a patient with Lou Gehrig 's disease November 25th, 1998.

30. Some Washington-based foreign journalists were asked in "The World", how they were covering America 's top stories November 27th, 1998.

31. Almost half of Michigan residents apparently believe Jack Kevorkian should be criminally charged in the death of a man who had Lou Gehrig 's disease .

32. An opposition candidate in December 's legislative election apologized for flinging live piglets at aides to Taiwan Governor James Soong on November 1st, 1998.

33. “三合一”选战进入密锣紧鼓的阶段。 各党派的候选人，近日为了争取选民的支持，各种竞选花招纷纷出炉，1998。

34. 台湾年底立法委员选举参选人林瑞图指陈水扁秘密到澳门的事件，在1998年11月4日进一步扩大。 民进党支持者还到联合报抗议撕报纸，要求该报作出道歉。

35. 1998年11月7日，台湾的选战已经进入最后阶段，为了吸引年青人的支持，各党派莫不挖空心思举办活动。

36. 民进党的“竞选花车”-由一辆双层巴士改装成的“金达尼号”，1998年11月6日在烟火及欢呼声中,举行“开航”仪式，

37. 1998年11月10日,马英九和陈水扁进行了一场一对一的辩论,就两岸关系发表了看法。

38. 1998年11月台北市长竞选活动进入白热化之际，台湾监察院因一致通过一项议案，纠正台北市政府，结果卷入一场竞选风波之中。

39. 台湾监察院纠正台北市政府事件1998年11月13日进一步扩大。 民进党籍和独派候选人到监察院抗议，并与声援监委翟宗泉的新党候选人和支持者发生冲突。

40. 台北市年底三合一选举第二届市长、 第四届立委与第八届市议员候选人于1998年11月14日进行号次抽签，以及决定公办政见会发言顺序，现场造势活动热闹滚滚，有如一场嘉年华会。

41. 1998年11月17日,台北市长选情紧绷，两大竞争对手，民进党的陈水扁和国民党的马英九仍处平分秋色局面，抛离对手，双方都打出新牌。

42. 1998年11月19日，目前距离月日举行的台湾立法委员选举只有两周左右，根据受访政治观察者的预测，虽然民进党积极鼓吹三党不过半，但国民党还是有可能取得过半的席位。

43. 台湾年底三项选举的竞选活动在1998年11月20日正式开始，警调单位昨日凌晨即展开选前的扫黑、 肃枪和扫毒大行动，以确保竞选活动期间的社会治安。

44. 1998年11月23日，台北市长选情激烈，两名热门人选陈水扁、 马英九龙虎相争。

45. Taiwan's Foreign Ministry blamed “administrative negligence” for an incident in which Nobel Peace Prize winner Josi Ramos-Horta was left stranded at the airport for hours after being refused entry on November 26th, 1998.

46. 台湾举行的台北及高雄市长和市议员选举，还有立法委员选举已经全面展开竞选活动，一向与台湾各级选举脱离不了关系的买票活动，随着选举日的日益接近，也纷纷出笼。

47. 1998年11月28日，香港民主党党魁李柱铭率领人代表团到台湾，观察在12月举行的立委和市长选举。

48. NASA and the Russian Space Agency agreed to set aside a last-minute Russian request to launch an international space station into an orbit closer to Mir , officials announced on November 13th, 1998.

49. The first piece of the long-delayed international space station was scheduled to be launched from Russia on November 19th night, 1998.

50. The first piece of the international space station was orbiting Earth on November 20th, 1998, sprouting antennae and unfolding solar power panels as it awaited other segments, which will eventually grow into the largest orbital laboratory in history.

51. Russian space officials gave the first module of the international space station a routine tweak on November 21st, 1998 to push it into higher orbit , and convened a meeting on Earth to map out its future.

52. 日升空的国际空间站第一舱－“曙光” 号功能货物舱在经过多次飞行轨道的调整后于日成功进入工作轨道，目前舱上所有系统工作正常。

53. The Space Shuttle Endeavour was scheduled to blast off. The six-member crew was carrying with it the second part of the international space station.

54. 俄罗斯的太空进入了一个新时代，俄罗斯太空当局不久前发射了一个叫做日出的空间站日出空间站是计划发射的第一个永久性的进入地球轨道的国际空间站，空间站的太空舱是在哈萨克斯坦的克拜努尔发射场，由俄罗斯的一枚直子火箭发射升空的。

55. The Space Shuttle Endeavour was due to lift off early morning of November 30th,1998 from Cape Canaveral , Florida with five U.S. astronauts and one Russian Cosmonaut.

56. U.S. shuttles and Russian rockets will carry out 45 missions to assemble the station , an outpost for research and platform for space exploration

57. In November 1998, Hong Kong bankers were concerned about a brewing nightmare in China , the possibility of more defaults on foreign loans by financial institutions on the mainland.

58. 1998年11月广东省内主要直辖企业及国投公司和重组方案已正式敲定。

59. 1998年11月，被迫关闭的中国第二大信贷机构广东国际信托投资公司由于没有能够按期支付证券利息，成为中国第一家不履行国际借债条约的金融机构。 广信的倒闭也直接影响了中国国际信托公司的信誉。

60. 中国人民银行行长戴相龙表示，在全国多家信托投资公司中，只有“ 极少数” 的公司会被关闭，央行也会考虑为部分有付款困难的公司提供担保。

61. 由于产权不清及未能确定可以收回多少应收帐，帐面资产值亿港元的广信实业暂时仅能确认亿港元的资产可以变现。

62. 1998年11月15号,有消息表明，中国将在短期内推出一套新法规，现有的二百四十家国际信托投资公司中最多可达百分之七十的公司将被关闭。

63. 在广信集团倒闭事件之后，中国总理朱(金容)基要求各省管好自己的驻港窗口公司，要求绝对不能出问题。

64. China's central bank chief warned that risks to the country's financial system "can no longer be ignored" and pledged tougher regulation of the shaky banking sector in November 30th, 1998.

□ **End of chapter.**

Bibliography

[1] J. Allan, V. Lavrenko, D. Frey, and V. Khandelwal. UMass at TDT 2000. In *The Topic Detection and Tracking Workshop 2000*, pages 109–115, February 2001.

[2] J. Allan, V. Lavrenko, D. Malin, and R. Swan. Detection, bounds and timelines: UMass and TDT-3. In *The DARPA Topic Detection and Tracking Workshop - TDT3*, pages 167–174, February 2000.

[3] J. Allan, R. Papka, and V. Lavrenko. On-line new event detection and tracking. In *Proceedings of the Annual International ACM SIGIR Conferenc Research and Development in Information Retrieval*, pages 37–45, 1998.

[4] E. Brill. Some advances in transformation-based part of speech tagging. In *National Conference on Artificial Intelligence*, pages 722–727, 1994.

[5] E. Brill. Transformation-based error-driven learning and natural language processing: A case study in part of speech tagging. In *Association of Computational Linguistics*, volume 21, pages 543–565, 1995.

[6] R. D. Brown, T. Pierce, Y. Yang, and J. G. Carbonell. Link detection - results and analysis. In *The DARPA Topic Detection and Tracking Workshop - TDT3*, February 2000.

[7] J. Carbonell, Y. Yang, and J. Lafferty. CMU report on TDT-2: Segmentation, detection and tracking. In *Proceedings of the DARPA Broadcast News Workshop - TDT2*, pages 117–120, February 1999.

[8] H. H. Chen and L. W. Ku. Description of a topic detection algorithm on tdt3 mandarin text. In *The DARPA Topic Detection and Tracking Workshop - TDT3*, pages 165–166, February 2000.

[9] H. H. Chen and L. W. Ku. An NLP & IR approach to topic detection. In J. Allan, editor, *Proceedings of the Topic Detection And Tracking Event-based Information Organization*, pages 243–264. Kluwer Academic Publishers, 2002.

[10] Y. Chen and H. H. Chen. NLP and IR approaches to monolingual and multilingual link detection. In *International Conference on Computational Linguistics*, pages 176–182, 2002.

[11] I. S. Dhillon, Y. Guang, and J. Kogan. Iterative clustering of high dimensional text data augmented by local search. In *Proceedings of IEEE Conference on Data Mining (ICDM)*, December 2002.

[12] C. Ding, X. He, H. Zha, and H. D. Simon. Adaptive dimension reduction for clustering high dimensional data. In *Proceedings of IEEE Conference on Data Mining (ICDM)*, December 2002.

[13] M. Franz, J. McCarley, T. Ward, and W.-J. Zhu. Segmentation and detection at IBM : Hybrid statistical models and two-tiered clustering. In *The DARPA Topic Detection and Tracking Workshop - TDT3*, pages 149–153, February 2000.

[14] W. Gale and K. Church. A program for aligning sentences in bilingual corpora. In *Proceedings of the Annual Conference of the Association for Computational Linguistics*, pages 177–184, 1991.

[15] R. Huang and W. Lam. Support story link detection using automatic topic type categorization. In *The 2003 International Conference on Information and Knowledge Engineering (IKE'03)*, June 2003.

[16] R. Huang, W. Lam, and Y.-Y. Law. Discovering multilingual news events and term associations from the web. In *The 7th World Multiconference on Systemics, Cybernetics and Informatics (SCI2003)*, July 2003.

[17] T. Joachims. Text categorization with support vector machines: Learning with many relevant features. In *Proceedings of the European Conference on Machine Learning (ECML)*, pages 137–142, 1998.

[18] W. Lam, H. Meng, and K. Hui. Multilingual topic detection using a parallel corpus. In *The Topic Detection and Tracking Workshop 2000*, pages 87–91, February 2001.

[19] W. Lam, C. K. Tsang, T. L. Wong, and H. Meng. CUHK's link detection system for the TDT2001 evaluation. In *DARPA TDT Workshop*, 2001.

[20] V. Lavrenko, J. Allan, E. DeGuzman, D. LaFlamme, V. Pollard, and S. Thomas. Relevance models for topic detection and tracking. In *Proceedings of Human Language Technology (HLT) Conference*, pages 104–110, March 2002.

[21] T. Leek, H. Jin, S. Sista, and R. Schwartz. The BBN crosslingual topic

detection and tracking system. In *The DARPA Topic Detection and Tracking Workshop - TDT3*, pages 123–127, February 2000.

[22] X. Liu, Y. Gong, W. Xu, and S. Zhu. Document clustering with cluster refinement and model selection capabilities. In *Proceedings of ACM SIGIR Conference on Research and Development in Information Retrieval 2002*, pages 191–198, 2002.

[23] National Institute of Standards. The 2002 Topic Detection and Tracking Task Definition and Evalutaion Plan. In *ftp://jaguar.ncsl.nist.gov/tdt/tdt2002/evalplans/TDT02.Eval.Plan.v1.1.ps*, 2002.

[24] National Institute of Standards. Topic detection and tracking annotation guide TDT 2002. In *http://www.ldc.upenn.edu/Projects/TDT4/Annotation/label_instructions.html*, 2002.

[25] P. Pantel and D. Lin. Document clustering with committees. In *Proceedings of ACM SIGIR Conference on Research and Development in Information Retrieval 2002*, pages 199–206, 2002.

[26] R. Papka, J. Allan, and V. Lavrenko. UMASS approach to detection and tracking at TDT2. In *Proceedings of the DARPA Broadcast News Workshop - TDT2*, pages 111–116, February 1999.

[27] S. Sista, R. Schwartz, T. R. Leek, and J. Makhoul. An algorithm for unsupervised topic discovery from broadcast news stories. In *Proceedings of Human Language Technology (HLT) Conference*, March 2002.

[28] F. Walls, H. Jin, S. Sista, and R. Schwartz. Topic detection in broadcast news. In *Proceedings of the DARPA Broadcast News Workshop - TDT2*, pages 193–198, February 1999.

[29] Y. Yang, J. G. Carbonell, R. D. Brown, T. Pierce, B. T. Archibald, and X. Liu. Learning approaches for detecting and tracking news events. In *IEEE Intelligent Systems, Special Issue on Application of Intelligent Information Retrieval*, volume 14, pages 32–43, 1999.

[30] Y. Yang and J. O. Pedersen. A comparative study on feature selection in text categorization. In *Proceedings of the 14th International Conference on Machine Lea rning (ICML)*, pages 412–420, 1997.

[31] Y. Yang, T. Pierce, T. Ault, and J. Carbonell. Combining multiple learning strategies to improve tracking and detection performance. In *The DARPA Topic Detection and Tracking Workshop - TDT3*, pages 129–134, February 2000.

[32] Y. Yang, T. Pierce, and J. Carbonell. A study on retrospective and on-line event detection. In *Proceedings of the Annual International ACM SIGIR Conference Research and Development in Information Retrieval*, pages 28–36, 2000.

[33] Y. Yang, J. Zhang, J. Carbonell, and C. Jin. Topic-conditioned novelty detection. In *ACM SIGKDD Internaltional Conference on Knowledge Discovery and Data Mining*, pages 688–693, 2002.