CHAPTER 7

SIMULATIONS OF EFFECTUAL HYBRID TEXT CLUSTERING

7.1 INTRODUCTION

In vector space representation, defining terms as distinct single words is referred to as "bag of words" representation. Researchers stated that using phrases rather than single words to define terms produce more accurate classification results (Cohen W.W. and Singer Y., 1996, Fuernkranz J. et al., 1998, Chim H. and Deng X., 2008). But many researchers have stated that using single words as terms does not produce worse results (Dumais S. et al., 1998, Sahami M., 1998). The "bag of words" representation is the most frequently used method for defining terms and it is computationally more efficient than the phrase representation.

7.2 SIMULATION ENVIRONMENT

The performance of text clustering algorithm is evaluated in terms of the scalability of finding frequent words and the accuracy of clustering. All the algorithms have been implemented and experiments are performed in Java 1.6 on Pentium IV PC with 3.20 GHz processor and 1 GB of main memory. Java 1.6 is selected as programming language because it allows fast and flexible development.

For generating the frequent term set, a public domain implementation of the basic Apriori algorithm has been downloaded from the following link (Yibin S., 2000).
For dimension reduction, the Porter Stemming algorithm version 1.0 has been selected from the link http://tartarus.org/~martin/PorterStemmet/. This code has been modified to perform further dimension reduction and the following updates have been carried out: Elimination of words which have less than 3 characters, Inclusion of more stopwords, Inclusion of adverb and adjectives and Inclusion of non-noun verbs. All the four clustering algorithms, namely, Min-Match, Max-Match, Min-Max-Match and Cosine Similarity have been developed in Java 1.6.

7.3 DATA SET SELECTION

To test and compare clustering algorithms a pre-classified sets of documents are needed. For the performance evaluation, four groups of data sets are used. They are as follows:-

- **Books:** The first sample is selected from own pre-classified books, which consist of six documents. They are C Programming, Computer Networks, C++ Programming, Fonts, Java Programming and SQL.

- **Reuters Transcribed subset:** This data set consists of 200 documents, created from the 10 largest reuters classes in the Reuters-21578 collection. http://www.daviddlewis.com/resources/testcollections/reuters21578/.

There are 10 directories labelled by the topic name, each contains 20 documents of transcription.

http://kdd.ics.uci.edu/databases/reuters_transcribed/ReutersTranscribedSubset.zip
The ten largest Reuters classes are labelled by following names:

- ReutersTrancribedSubset/acq
- ReutersTrancribedSubset/crude
- ReutersTrancribedSubset/grain
- ReutersTrancribedSubset/money
- ReutersTrancribedSubset/trade
- ReutersTrancribedSubset/corn
- ReutersTrancribedSubset/yarn
- ReutersTrancribedSubset/interest
- ReutersTrancribedSubset/ship
- ReutersTrancribedSubset/wheat

PDF Sample: This data set is selected from 54 research documents in word format converted from pdf in the field of computer science domain.


A subset composed of 100 articles from each newsgroup is taken from the link [http://kdd.ics.uci.edu/databases/20newsgroups/mini_newsgroups.tar.gz](http://kdd.ics.uci.edu/databases/20newsgroups/mini_newsgroups.tar.gz)

One hundred Usenet articles have been taken from each of the following 20 newsgroups:

- alt.atheism
- comp.graphics
- comp.os.ms-windows.misc
- comp.sys.ibm.pc.hardware
- comp.sys.mac.hardware
- comp.windows.x
- misc.forsale
- rec.autos
- rec.motorcycles
- rec.sport.baseball
- rec.sport.hockey
sci.crypt
sci.electronics
sci.med
sci.space
soc.religion.christian
talk.politics.guns
talk.politics.mideast
talk.politics.misc
talk.religion.misc

➢ Reuters-21578: It is a collection of documents that appeared on Reuters newswire.

The collection is distributed in 22 files. Each of the first 21 files (reut2-000.sgm through reut2-020.sgm) contain 1000 documents, while the last (reut2-021.sgm) contains 578 documents. The files are in SGML format which consist of 21578 documents totally. It is obtained from the following link:-

http://kdd.ics.uci.edu/databases/reuters21578/reuters21578.tar.gz

Table 7.1 shows the description of the data sets considered for testing the algorithms.

<table>
<thead>
<tr>
<th>Data set</th>
<th>Number of documents</th>
<th>Size (KB)</th>
<th>Words</th>
</tr>
</thead>
<tbody>
<tr>
<td>Books</td>
<td>6</td>
<td>237</td>
<td>38999</td>
</tr>
<tr>
<td>PDF Sample</td>
<td>54</td>
<td>2999</td>
<td>448489</td>
</tr>
<tr>
<td>Reuters Transcribed Subset</td>
<td>200</td>
<td>249</td>
<td>40489</td>
</tr>
<tr>
<td>Newsgroups</td>
<td>2000</td>
<td>4567</td>
<td>620546</td>
</tr>
<tr>
<td>Reuters-21578</td>
<td>22</td>
<td>20275</td>
<td>3133193</td>
</tr>
</tbody>
</table>

### 7.4 DOCUMENT PREPROCESSING

One challenge emerging, when terms are defined as single words, is that the feature space becomes very highly dimensional. In addition, words which are in the same
context, such as biology and biologist, are defined as different terms. So, in order to define words that are in the same context with the same term, and consequently to reduce dimensionality, the terms are defined as stemmed words. To stem the words, Porter’s Stemming Algorithm (Porter M.F., 1980), which is the most commonly used algorithm for word stemming in English have been chosen.

Preprocessing and document representation phase, which is implemented in Java 1.6, consists of the following steps: Parsing the documents and case-folding, Removing stopwords, Stemming and Dimensionality reduction. These steps are described briefly in the following sections.

**Parsing the Documents and Case-folding**

<table>
<thead>
<tr>
<th>a</th>
<th>at</th>
<th>by</th>
<th>down</th>
<th>every</th>
</tr>
</thead>
<tbody>
<tr>
<td>across</td>
<td>be</td>
<td>can</td>
<td>during</td>
<td>everybody</td>
</tr>
<tr>
<td>again</td>
<td>became</td>
<td>cannot</td>
<td>each</td>
<td>everyone</td>
</tr>
<tr>
<td>all</td>
<td>because</td>
<td>cant</td>
<td>eg</td>
<td>ex</td>
</tr>
<tr>
<td>alone</td>
<td>been</td>
<td>certain</td>
<td>eight</td>
<td>except</td>
</tr>
<tr>
<td>also</td>
<td>behind</td>
<td>consequently</td>
<td>either</td>
<td>far</td>
</tr>
<tr>
<td>am</td>
<td>being</td>
<td>could</td>
<td>else</td>
<td>few</td>
</tr>
<tr>
<td>an</td>
<td>below</td>
<td>did</td>
<td>enough</td>
<td>first</td>
</tr>
<tr>
<td>and</td>
<td>better</td>
<td>do</td>
<td>et</td>
<td>five</td>
</tr>
<tr>
<td>any</td>
<td>both</td>
<td>does</td>
<td>etc</td>
<td>for</td>
</tr>
<tr>
<td>arc</td>
<td>brief</td>
<td>doing</td>
<td>even</td>
<td>four</td>
</tr>
<tr>
<td>as</td>
<td>but</td>
<td>done</td>
<td>ever</td>
<td>from</td>
</tr>
</tbody>
</table>

*Figure 7.1 Portion of Stopwords List used*

All the HTML or SGML mark-up tags and non-alpha characters are removed from the documents in the document corpora. Case-folding (stands for converting all the
characters in a document into the same case) is performed by converting all the characters into lower-case. Terms consisting of alpha characters are extracted.

**Removing Stopwords**

Removing stopwords from the documents is very common in information retrieval. Elimination of stopwords from the documents, will lead to a drastic reduction in the dimensionality of the feature space. A list of 429 words is used (Salton G. et al., 1975). Figure 7.1 shows a portion of the stopword list. This stopword list is obtained from [http://www.lextek.com/manuals/onix/stopwords.html](http://www.lextek.com/manuals/onix/stopwords.html).

**Stemming**

The similar terms “computer”, “computers” and “computing” are reduced to the word stem “compuf’. After stemming, terms that are shorter than three characters are also removed as they do not carry much information about the content of a document. Mostly, document clustering is based on context matching, which is denoted by nouns. Based on the above assumption, the adverbs and adjectives are eliminated for further dimension reduction. The portion of list of adverbs and adjectives list are shown in Figures 4.2 and 4.3 in Chapter 4 under the section 4.2. All the above stemming as well as the elimination process are implemented by using the improved stemming algorithm, which is shown in section 4.2.

**Dimensionality Reductions**

Term Frequency Weighting methods are applicable only when the selected terms of the documents are known in advance. In new document environment, frequent term set
generation is the most applicable method to find frequent terms as well as for dimension reduction. Frequent terms are determined by scanning the document and collecting those terms that satisfy the minimum support threshold value. These processes are implemented as a single level selection procedure as shown in Figure 4.9 under the section 4.5.

7.5 APPLICATIONS

The applications of text categorization (TC) are manifold and the common qualities (Sebastiani F., 2002) are 1) Need to handle and organise documents in which the textual component is either the unique or dominant or simplest to interpret, component, 2) Need to handle and organise large quantities of such documents, i.e., large enough that their manual organisation into classes is either too expensive or not feasible within the time constraints imposed by the application, and 3) The set of categories is known in advance and its variation over time is small.

The nature of the documents, that is, structured texts (such as scientific articles), newswire stories, classified ads, image captions, e-mail messages, transcripts of spoken texts, hypertexts or others. If the documents are hyper textual, rather than textual, very different techniques may be used, since links provide a rich source of information on which classifier learning activity can gain leverage. The structure may be flat or hierarchical in the classification scheme. Hierarchical classification schemes may in turn be tree-shaped or allow for multiple inheritance. The nature of the task may be single-label or multi-label.
7.5.1 Document Organisation

Indexing with a controlled vocabulary is an instance of the general problem of document base organisation. In general, many issues pertaining to document organisation and filing, be it for purposes of personal organisation or structuring of a corporate document (Sebastiani F., 2002). For instance, at the offices of a newspaper incoming "classified" advertisements must be, prior to publication, categorized under categories such as Personals, Cars for Sale, Real Estate, etc. Newspapers dealing with a high volume of classified advertisements would benefit from an automatic system that chooses the most suitable category for a given advertisement. Other applications are the organisation of patents into categories for making their search easier (Larkey L.S., 1999), the automatic filing of newspaper articles under the appropriate sections (e.g., Politics, Home News, Lifestyles, etc.) or the automatic grouping of conference papers into sessions.

7.5.2 Text Filtering

Text filtering is the activity of classifying a stream of incoming documents dispatched in an asynchronous way by an information producer to an information consumer. A typical case is a newsfeed, where the producer is a news agency and the consumer is a newspaper (Hayes P.J. et al., 1990). In this case, the filtering system should block the delivery of the documents in which the consumer is not interested in (e.g., all news not concerning sports, in the case of a sports newspaper). Filtering can be seen as a case of single-label TC, that is, the classification of incoming documents into two disjoint categories, the relevant and irrelevant. In the example above, all articles about sports should be further classified, so as to allow journalists specialised in individual sports to
access only documents of prospective interest for them. Similarly, an e-mail filter might be trained to discard "junk" mail (Drucker H. et al., 1999) and further classify non-junk mail into topical categories of interest to the user.

A filtering system can be installed at the producer end, in which case it must route the documents to the interested consumers only at the consumer end, in which case it must block the delivery of documents deemed uninteresting to the consumer. In the former case, the system builds and updates a "profile" for each consumer, while in the later case a single profile is needed.

7.5.3 Hierarchical Categorization of Web Pages

TC has aroused a lot interest also for its possible application to automatically classifying web pages or sites, under the hierarchical catalogues hosted by popular Internet portals. When web documents are catalogued rather than issuing a query to a general-purpose web search engine, a searcher may find it easier to first navigate in the hierarchy of categories and then restrict a search to a particular category of interest. Classifying Web pages automatically has obvious advantages, as the manual categorization of a large enough subset of the web is unfeasible (Chakrabarti S. et al., 1998). Automatic web page categorization has two essential peculiarities, namely, the hyper textual nature of the documents and the typically hierarchical structure of the category set.
7.5.4 Word Sense Disambiguation

Word Sense Disambiguation (WSD) is the activity of finding, given the occurrence in a text of an ambiguous (i.e., polysemous or homonymous) word, the sense of this particular word occurrence. For instance, bank may have at least two different senses in English, as in the Bank of England (a financial institution) or the bank of river Thames (a hydraulic engineering artifact). It is thus, a WSD task to decide the sense of the occurrence of ‘bank’ in “Last week I borrowed some money from the bank”. WSD can be a single-label TC task (Escudero G. et al., 2000). Once given a word $\omega$, the contexts of occurrence of $\omega$ is viewed as documents and the senses of $\omega$ as categories.

The automatic extraction of opinions, emotions and sentiments in text (subjectivity analysis) to support applications such as product review mining, summarization, question answering and information extraction is an active area of research in NLP (Cem Akkaya et al., 2009). A Subjectivity Word Sense Disambiguation (SWSD), which is to automatically determine which word instances in a corpus are being used with subjective senses, and which are being used with objective senses.

7.5.5 Automated Survey Coding

Survey coding is the task of assigning a symbolic code from a predefined set of such codes to the answer that a person has given in response to an open-ended question in a questionnaire. This task is usually carried out in order to group respondents according to a predefined scheme based on their answers. Survey coding has several applications, especially in the social sciences, where the classification of respondents is functional to
the extraction of statistics on political opinions, health and lifestyle habits, customer satisfaction, brand fidelity and patient satisfaction.

Survey coding is a difficult task, since the code that should be attributed to a respondent based on the answer given is a matter of subjective judgment and thus requires expertise. The problem can be formulated as a single-label TC problem (Giorgetti D. and Sebastiani F., 2003), where the answers play the role of the documents and the codes that are applicable to the answers returned to a given question play the role of the categories. Different questions thus correspond to different TC problems.

7.5.6 Automated Authorship Attributes and Genre Classification

Authorship attribution is the task of determining the author of a text of disputed or unknown paternity, choosing from a predefined set of candidate authors (Vel O.Y.D. et al., 2001, Diederich J. et al., 2003). Authorship attribution has several applications, ranging from the literary (e.g., discovering the author of a recently discovered sonnet) to the forensic (e.g., identifying the sender of an anonymous letter or checking the authenticity of a letter allegedly authored by a given person). Authorship attribution can also be seen as a single-label TC task, with possible authors playing the role of the categories. This is an application in which a TC system typically cannot be taken at face value. Usually, its result contributes an "opinion" on possible author might be, but the final decision has to be taken by a human professional. As a result, a TC system that ranks the candidate authors in terms of their probability of being the true author.
The intuitions that must be brought to bear in these applications are orthogonal to those that are at play in topic-based classification, since an author normally writes about multiple topics. Because of this, it is unlikely that topic-based features can be good at discriminating among authors. Rather, stylistic features are the most appropriate choice. For instance, vocabulary richness (i.e. ratio between number of distinct words and total number of words), average word length and average sentence length are important, in the sense that it is these features that tend "to give an author away".

Genre classification is also an applicative context which bears remarkable similarities to authorship attribution. There are applicative contexts in which it is desirable to classify documents by genre, rather than by topic (Finn A. et al., 2002, Lee Y.B. and Myaeng S.H., 2002). For instance, it might be desirable to classify articles about scientific subjects into one of the two categories namely, Popular Science and Hard Science, in order to decide whether they are suitable for publication in popular science magazines or not. Likewise, distinguishing between ProductReviews and Advertisements might be useful for several applications. In genre classification too, topic-dependent words are not good separating features, and specialised features need to be devised, which are often similar to the ones used for authorship attribution applications.

Genre based classifications focus on functional purposes and classifies web pages into categories such as online shopping, technical paper or discussion forum. Until now, genre classifications are not well developed due to the subjectivities and difficulties to define the genre, the features and even the categories (Chen G. and Choi B., 2008).
7.6 DISCUSSION

This chapter discusses the various simulation environments, especially hardware, software and data set selection. Throughout this work, all the algorithms have been developed using Java 1.6 on Pentium IV PC with 3.20 GHz processor. Experiments have been conducted to test the validity of the algorithm by using 5 different data sets. In document preprocessing, words which are used frequently and carry no useful information about the content are removed as stopwords. The stemming process removes the morphological word suffixes, leaving only the stems or roots. The adverb and adjectives terms are also removed for further dimension reduction, because, the context matching in document clustering is based on nouns. The various applications of TC are also discussed.