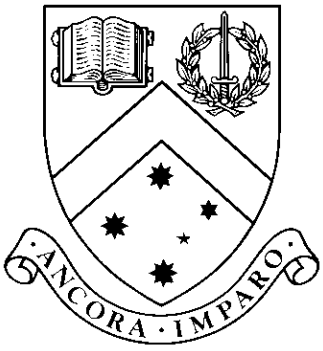
School of Computer Science and Software Engineering
Monash University



Literature Review — Semester 2, 2005

Sentence Classifier for Helpdesk Emails

[Anthony] [18734359]

Supervisors: Dr. David Albrecht,
Dr. Yuval Marom

Contents

1 Abstract

This project explores several types of features to represent sentences in classification tasks. The explored features are several variations of bag-of-words, phrases, part-of-speech, parse tree and sentence-specific features. It then employs existing text-classification feature selection methods to reduce the feature space and potentially improve classification effectiveness. A combination of feature selection methods will be investigated to achieve optimal performance. One of the methods studied in this project is Bi-Normal Separation (BNS). It is a relatively new method and has been reported to outperform existing methods, such as χ^2 and information gain. To the author's knowledge, feature selection has not yet been applied to any sentence classification studies.

2 Introduction

There has been a significant increased of email usage in the past decade. Emails have evolved from a mere communication tool to a way of organizing workflow and managing tasks (Corston-Oliver et al., 2004). Existing tasks in managing the emails involve email classification (Segal and Kephart, 1999), email summarization (Tzoukermann et al., 2001) and spam filtering (Drucker et al., 1999). The techniques to solve these problems, specifically email classification and spam filtering, tend to use low-level features, such as using only the words in the emails and discarding the information about the sentence order and word order. Using only low-level features loses much of the contextual information in the emails. It will be interesting to see if higher level features can be incorporated to existing techniques to improve their current performances.

Several studies have shown that high-level features, such as sentence types, can often be useful (McKnight and Srinivasan, 2003; Teufel and Moens, 2002). For example, an email can be assigned a priority based on the sentence types contained in that email, like TASK ("Send me the report as soon as possible") or APPOINTMENT ("We have decided to have a meeting at 3pm"). Other example include routing an email enquiry to an appropriate helpdesk operator based on the types of questions in the email enquiry. To obtain these sentence types, sentences in the emails will need to be classified.

Supervised learning can be employed to classify the sentences (Sebastiani, 2002). It generates a function that maps a set of training sentences to their corresponding sentence types. When a new sentence is given, the function will try to predict its sentence type. This function can be considered as a sentence classifier. This project aims to develop such sentence classifier.

The domain of the project is email helpdesk. Specifically, the classifier will be implemented to classify only the sentences in the email responses (user enquiries will be removed). This is because email responses tend to be more structured and have less grammatical errors compared to email enquiries as they are written by professional helpdesk operators. Once this project is completed, it can be extended to handle those more problematic email enquiries.

To the author's knowledge, there have not been many research studies on sentence classification. Most of them were not using email as the domain, and the sentence types they investigated were very abstract, such as INTRODUCTION, METHOD, AIM, BACKGROUND, BASIS, PROCEEDINGS and so on (Teufel and Moens, 2002; McKnight and Srinivasan, 2003; Hachey and Grover, 2005). Others, which used email as the domain, had little similarity

to this project in terms of the sentence types (Corston-Oliver et al., 2004; Cohen et al., 2004). These studies focused on the features to represent the sentences. None of the studies mentioned above applied any feature selection methods (see section 5) in their experiments.

Sentence classification may appear fairly similar to text classification (TC), which is the automated classification of documents to a predefined set of categories. Therefore, much of their literatures are very similar. There are several aspects in building a classifier. This project will concentrate only on three aspects, which are (1) identifying the categories (sentence types) that are suitable to the selected domain, (2) specifying the features to represent the sentences and (3) investigating the feature selection methods to select only useful features. As for the classification methods, existing ones will be used. To evaluate the effectiveness of the classifiers, standard evaluation metrics, such as precision, recall (Salton and McGill, 1983) and F_1-measure (van Rijsbergen, 1979), will be employed.

3 Tag Set

As this project aims to determine the sentence types in the domain of helpdesk email responses, a set of tags that corresponds to the sentence types will be needed. This set of tags will now be referred to as a **tag set**. This project will need to devise its own tag set since, to the author's knowledge, there has not been any research studies on the sentence classification in this domain.

Devising a tag set involves taking a sample from the corpus and examining the sentence types in that sample. These sentence types will then form the tag set. An important issue is in ensuring that the tags in the tag set are mutually exclusive, which means that each sentence should be appropriately annotated with only one tag. This allows the sentence classifier to assign a tag to a sentence with a higher certainty, thus resulting in an overall higher classification accuracy.

It is useful if some tag sets that have been devised and considered as mutually exclusive in other studies can be adapted for this project. This provides an initial source of tags that can be used in the domain of this project as well as in the other domains. Thus, some of the tags can be portable to other domains easily. New tags that are specific to this domain can then be introduced while maintaining the mutual exclusive properties. Dialog Act (DA) tagging is one of the research areas that has devised a number of tag sets that can be useful for this project (Warnke et al., 1997; Wright, 1998; Taylor et al., 1998; Chu-Carroll, 1998). The following discusses how DA tagging and sentence classification can be related and describes why the tag sets from DA tagging can be incorporated into the project.

DA tagging involves assigning a predefined tag to a particular utterance (Stolcke et al., 2000). This is similar to sentence classification by treating utterances as sentences. Although utterances can be much shorter than sentences in the email responses, some of the sentence types in email responses are very similar to those found in DA tagging, such as REQUEST, SUGGESTION, INSTRUCTION and so on. In addition, the email responses are also written dialogues. This suggests that existing tag sets for DA tagging can be useful for this project. The following discusses several selected tag sets in more details. The first tag set is very general and forms the basis from which other tag sets are derived (Jurafsky et al., 1997b; Ivanovic, 2005).

The natural language community has designed a Dialog Act Markup in Several Layers (DAMSL) tag set, which was used for coding task-oriented dialogs. It aims to provide a set of domain-independent, high-level communicative actions that are applicable to different types of dialogues. By using DAMSL annotation scheme, multiple tags can be applied to an utterance at the same time. For example, the sentence *"Yeah, that'll will a good time"* can be simultaneously be annotated as RESPONDING-TO-A-QUESTION, PROMISING, and INFORMING (Core and Allen, 1997). Although a domain-independent tag set is useful, this annotation scheme is clearly not appropriate for the project. According to Clark and Popescu-Belis (2004), there are about 4 million possible combinations of DAMSL tags. Therefore, tags selection will be required in order to make the tag set highly relevant to a domain of interest.

The DAMSL tag set was then adapted to Switchboard data (human-to-human telephone conversations). Some classes in DAMSL were omitted, while some other classes were further expanded to provide more variety of tags. The final tag set is called SWBD-DAMSL tag set and consists of 42 mutually exclusive tags for automatic annotation of discourse structure (Jurafsky et al., 1997b). These mutually exclusive tags can be a very good source of initial tags for the project. This tag set and its annotation manual is available in (Jurafsky et al., 1997a).

Other related tag sets include (Jekat et al., 1995) and (Taylor et al., 1998). These studies have focused on task-oriented dialogues. The former focuses on conversation between two speakers, in which the first speaker is trying to guide the second speaker to route a map. The latter focuses on dialogues to schedule a meeting. Some of the tags from these tag sets can be related to this project as a substantial number of sentences in the email responses are task-oriented, consisting of instruction or queries from helpdesk operators to the users.

Once the tag set for the project has been devised, it is necessary to measure how reliable the tag set can be used to annotate the sentences. If different annotators have significant disagreement in annotating the sentences using the same tag set, the tag set will need to be revised. Kappa statistics K will be used to measure the reliability (Siegel and Castellan, 1988). This is a standard approach used in most research studies that need to measure the agreement between different human annotators in tagging certain objects.

4 Sentence Representation and Feature Set

In general, texts can not be directly processed by most classification algorithm, even if they are already stored in machine readable format, such as HTML and PDF (Joachims, 2002). Therefore, they must be transformed into an appropriate representation for classification tasks. A sentence is usually represented as a set of features. This section discusses the features that can be used for the selected domain. It reviews the use of the features in both sentence and text classification since they are closely related. The project will focus on using a combination of features to optimize the overall sentence classification effectiveness. This will also be one of the major contributions of the project.

4.1 Bag-of-words

A common representation used in many classification tasks is *bag-of-words* approach. With this approach, each distinct word in the text corresponds to a feature, and the text is transformed to a vector of N elements ($< w_1, w_2, \ldots, w_n >$), where N is the total number of distinct words in the entire corpus, and w_k is the weight of the $word_k$. Information

about the structure of the text, sentence order and word order are discarded (Scott and Matwin, 1999). There are several methods of implementing bag-of-words as discussed below.

- Binary bag-of-words
  This approach uses only 0 or 1 as the value of w_k, where $w_k = 0$ indicates that $word_k$ does not occur in the text, while $w_k = 1$ indicates that it occurs (Soucy and Mineau, 2001). This is not usually used in text classification as words that occur more frequently in the text are generally regarded to have different significance compared to words that occur just once, except stop-words, like "the", "of", "a" and so on.
  However, this approach may be useful for sentence classification, especially when the sentences are short. It is quite rare to have some words occurring more than once in a short sentence, excluding those stop-words. This is supported after examining the sentences in a small sample taken from the corpus. In addition, storing only binary values is simpler and requires less computation time compared to other approaches discussed later.
  Binary bag-of-words was used in one of the studies to categorize sentence types in medical abstract (McKnight and Srinivasan, 2003). The result showed a macro-averaged F_1-measure of below 0.75. When location of the sentence was incorporated as a feature, it had a substantial improvement in overall classification effectiveness, recording a macro-averaged F_1-measure of above 0.8. This may be an indicator of how a combination of features can improve overall classification accuracy.

- Word-Frequency bag-of-words
  With this approach, w_k stores the number of occurrences of $word_k$ in the text. This was used in text classification in (Apte et al., 1994; Johnson et al., 2002). For long sentences, some non-stop-words may repeat. This information can be captured by this approach. However, there are usually just a few non-stop-words repeating in a long sentence. When the sentences are short, this approach may have very similar performance to the simpler binary bag-of-words.

- Term Frequency Inverse Document Frequency (TFIDF)
  Initially introduced by Salton and Buckley (1988), TFIDF assigns higher weights to words that occur frequently in a document but rarely in all other documents. This is based on the intuition that words that occur often in a document are considered important to that document, but they become less discriminating if they occur often in many other documents. It has been considered as a standard approach (Liao et al., 2003); thus, it is widely-used in a number of studies (Joachims, 1998; Guo et al., 2004). As its name sounds, it is intended to be used in text classification. TFIDF can be easily applied to sentence classification by treating sentences as documents.
  Some issues arise when TFIDF is applied to sentence classification. There are some words that are cues to identify certain sentence types. For example, "thank" usually signals a THANKING sentence, "apologize" usually signals a APOLOGY sentence and "what" may often signal a QUESTION. They generally occur only once in a sentence but quite frequent in the entire email corpus. Hence, TFIDF will assign them low weights, but in fact, they provide much information to discriminate certain sentence types. This suggests that TFIDF may not be appropriate for sentence classification. In a study to classify emails into speech act (Cohen et al., 2004), TFIDF was reported to have an inferior performance compared to binary bag-of-words.

Obviously, bag-of-words approach loses much of the contextual information in the text. However, it is widely used due to its simplicity and computational efficiency (Cardoso-Cachopo and Oliveira, 2003). Good classification effectiveness have also been achieved in

text classification when bag-of-words is used as the only type of features (Dumais et al., 1998; Joachims, 1998; Soucy and Mineau, 2001; Guo et al., 2004). It is usually used as the baseline features. Other features are incorporated to work with bag-of-words.

4.2 Phrase Features

This approach is essentially the same as bag-of-words but uses a sequence of words instead of just a single word to represent a feature. For example, *bigram* uses a sequence of two words, while *trigram* uses a sequence of three words. The general term is called word n-gram, which uses a sequence of n words. Phrase features are better than bag-of-words in capturing semantic relations of successive words (Bekkerman and Allan, 2004). For example, the phrase "supervised learning" is useful to identify that the text may be related to artificial intelligence. If simpler bag-of-words is used, "supervised" may be related to many other fields, such as research or education, while "learning" may be related to education.

The main problem in using phrase features is the significant increase in the total number of features as each feature corresponds to a sequence of two or more words in the text (Scott and Matwin, 1999). To avoid a significant computational overhead, bigram is more often considered for experiments than other higher word n-gram.

Even though bigram is able to preserve some information between successive words, it has not been able to show significant improvement over bag-of-words in a number of text classification experiments, and sometimes it even causes performance degradation (Lewis, 1992a,b; Apte et al., 1994; Caropreso et al., 2001; Koster and Seutter, 2002; Liao et al., 2003). Performance degradation is often observed when bigram is used as the only features without bag-of-words. When both are used together, a slight improvement may potentially be obtained (Bekkerman and Allan, 2004). Lewis (cited in (Scott and Matwin, 1999)) argued that phrase feature performed poorly in text classification due to its high dimensionality, skewed distribution of feature values, high redundancy among features and lots of noise in feature values.

Phrase features were used in a study to classify sentences in emails (Corston-Oliver et al., 2004). But, the study did not report the impact that phrase features had on the classification performance. Other studies on sentence classification that the author knows of did not use bigram.

4.3 Part-of-Speech

Some words have the same spelling but different meaning. Bag-of-words ignores this by treating all words that have the same spelling as the same. Part-of-speech (PoS) can capture this information as it describes the roles that the words have in the sentence (Scott and Matwin, 1999). The roles can be a *verb*, *adjective*, *noun* and so on. For example, the word "book" can be considered as a noun or a verb, depending on how it is used in the sentence.

Several sentence classification experiments have incorporated PoS, but its impact was not discussed (Corston-Oliver et al., 2004; Teufel and Moens, 2002; Hachey and Grover, 2005). On the other hand, Cohen et al. (2004) reported in an email classification study that there was a marginal increase in classification effectiveness when PoS was applied. PoS is also needed when parse tree (discussed next) is incorporated as features. The downside of PoS is that it will cause an increase in feature space as a word may correspond to several features, for example, "book" can be treated as "book_noun" and "book_verb".

The amount of increase in feature space depends on whether the experiment uses all possible PoS of a word or just a subset of them, such as considering only *noun*, *verb*, *adjective* and *adverb* PoS (Gliozzo and Strapparava, 2005).

4.4 Parse Tree

Parse tree examines the syntactical structure of the sentences (Benko and Katona, 2005). Two sentences that have the same set of words but different structures can have different meaning. For example, "Are there error messages" and "There are error messages" have different meaning. Since bag-of-words loses this information, parse tree can keep this information. To be able to parse the sentences, part-of-speech is needed.

Parse tree has been used in a sentence classification experiment (Corston-Oliver et al., 2004). However, no significant improvement was observed. The overhead of parsing was not reported. Other studies on sentence classification that the author knows of did not incorporate parse tree as features.

Parsing sentences is a research in its own. This project will not focus on building a good parser. It will just use any working parsers, such as the one available in Monash University User Modeling and Natural Language group [1].

4.5 Sentence-specific Features

Teufel and Moens (2002) devised a set of features that are more related to sentence classification than to general text classification. Their feature set was intended for the summarization of scientific articles. Only a subset of their features that are considered as useful for the project are reviewed below:

Location : Location of a sentence may help determine certain sentence types, such as SALUTATION, THANKING, APOLOGY and so on, which usually appear at the beginning or end of the email responses. The location can be computed relative to the size of the containing email. If such information is not available, location feature is not useful.

Action : This feature clusters the verbs into certain number of classes. Semantic concepts, such as similarity and textual structure, are uses as the basis of the clustering. For example, verbs like *"conjecture"* and *"speculate"* are clustered into one class. This may be considered as a feature selection method as well. The study did not report the effect of including such feature.

5 Feature Selection

Since there can be thousands or even tens of thousands distinct words in the entire email corpus, the feature space tends to be large and can be inefficient for computation. This leads to feature selection study to remove some uninformative features and reduce the processing time of the classification tasks. In addition, careful selection of features may be able to improve classification effectiveness substantially (Forman, 2003). Another major contribution of this project will be the investigation of the feature selection methods or combination of them to achieve a better performance in terms of the computation time or classification effectiveness.

[1] http://www.csse.monash.edu.au/research/umnl/

Several successful feature selection methods are discussed in this section. Other methods, such as **Mutual Information**, **Term Strength**, have been reported to have far inferior performance (Yang and Pedersen, 1997); therefore they will not be experimented in this project. The methods discussed have generally been applied to text classification. These methods, except **Stop-words removal**, have not yet been applied to any of the sentence classification studies within the author's knowledge. Therefore, they are generally reviewed in the context of text classification. This project will apply them to sentence classification by treating sentences as documents.

5.1 Stop-words removal

Generally, the first step to reduce the feature space is to remove the stop-words (connective words, such as "of", "the", "in"). These words are very common words and are conjectured to provide no information to the classifier, thus removing them are not likely to affect the classification accuracy. To identify these words, a threshold value t can be set to eliminate words that occur in more than t sentences, or a stopword list can be supplied (Forman, 2003). Stop-words removal is used in almost all text classification experiments (Scott and Matwin, 1999).

For sentence classification, stop-words removal raises some issues. First, removing stop-words in a short sentence may result in even fewer words left. This may pose a problem since the sentence classifier will need to determine the sentence type based on only a very few available words. Second, certain stop-words that are frequently used in text classification, such as "what", "how", "please" and so on, are actually useful to determine some sentence types in this domain. Therefore, if stop-words removal is applied to sentence classification, it is necessary to be more selective in choosing what words to be included in stop-words list. This has to take the domain of interest into consideration.

5.2 Document Frequency (DF)

Document Frequency is the number of documents in which a word appears. Only words that have document frequency above a predetermined threshold are retained (Sebastiani, 2002). The optimal threshold can be obtained empirically by executing the classifier using different threshold value each time and selecting the highest threshold value where the classification accuracy can still maintain unaffected.

There is a well-known assumption in information retrieval, which states that rare-words are important for certain category prediction. In contrast, in a series of text classification experiments, Yang and Pedersen (1997) showed that DF could remove 90% of distinct words without affecting the classification effectiveness (in terms of average precision). The removed words are rare-words. They claimed that rare-words were less important in text classification.

DF has been used in several text classification experiments, together with other feature selection methods (Furnkranz, 1998; Lam and Lee, 1999; Liao et al., 2003). It has a linear computational complexity, which is an advantage over other feature selection methods discussed next.

5.3 Chi-Squared (χ^2)

χ^2 measures the lack of independence between a feature and a category. If the independency is high, then the feature is considered not predictive for the category. It is computed

using the following equation, taken from (Seki and Mostafa, 2005):

$$\chi^2(w,c) = \frac{N \times (AD - CB)^2}{(A+C) \times (B+D) \times (A+B) \times (C+D)},$$

where w is a word, c is a category, A is the number of documents containing word w in category c, B is the number of documents containing w in categories other than c, C is the number of documents not containing w in c, D is the number of documents not containing w in categories other than c, and N is the total number of documents. For each word w, χ^2 is computed for each category, and the maximum score is taken as χ^2 statistics for that word w: $\chi^2(w) = max_i\ \chi^2(w, c_i)$. Only the top n words with highest χ^2 are retained. χ^2 has a quadratic complexity.

χ^2 has been reported to have a good performance, removing 98% of the distinct words in one experiment, while having a slight improvement in the classification effectiveness (in terms of average precision) (Yang and Pedersen, 1997). When more features are removed above a critical threshold value, performance starts to degrade. That experiment was a comparative study on feature selection. It showed that χ^2 was one of the best performers. This is further supported in (Rogati and Yang, 2002; Forman, 2003; Gabrilovich and Markovitch, 2004).

5.4 Information Gain (IG)

Information gain measures the entropy when the feature is present versus the entropy when the feature is absent (Forman, 2003). The entropy is indicator for predicting the sentence type. It is quite similar to χ^2 in a sense that it considers the usefulness of a feature not only from its presence, but also from its absence in each category. It is computed by using the following equation (Yang and Pedersen, 1997):

$$\begin{aligned} G(w) = &- \sum_{i=1}^{m} Pr(c_i) log Pr(c_i) \\ &+ Pr(w) \sum_{i=1}^{m} Pr(c_i|w) log Pr(c_i|w) \\ &+ Pr(\bar{w}) \sum_{i=1}^{m} Pr(c_i|\bar{w}) log Pr(c_i|\bar{w}) \end{aligned}$$

where w is a word, c is the category, m is the number of categories and G is the information gain of w. IG is computed for each distinct word w, and only words with IG above a predetermined threshold will be retained. The conditional probability of a category given a word has a time complexity of $O(N)$ and a space complexity of $O(VN)$, while the time complexity of computing the entropy is $O(Vm)$. N is the number of training instances, V is the number of words and m is the number of categories. Thus, its overall computation complexity is quadratic.

IG has been reported by Gabrilovich and Markovitch (2004) to have similar performance to χ^2. The difference is statistically insignificant. This result supports the same claim made by Yang and Pedersen (1997) and Forman (2003).

5.5 Bi-Normal Separation (BNS)

Bi-Normal Separation is a relatively new feature selection method introduced by Forman (2003). It is defined as $|F^{-1}(P(w_k|c_i)) - F^{-1}(P(w_k|\bar{c_i}))|$, where $P(w_k|c_i)$ is the conditional probability that a document D contains the word w_k, given that D belongs to

category c_i, and F is the cumulative probability function of the standard Normal distribution (Gabrilovich and Markovitch, 2004). In his study, Forman (2003) has shown that BNS is as competitive as χ^2 and IG. When macro-averaged F_1-measure was used as the evaluation metric, BNS performed best with a substantial margin when using 500 to 1000 features. This was because BNS obtained much higher recall than other methods. If precision was used as the evaluation metric, χ^2 and IG were preferred. The study was conducted on a dataset that had substantial high class skew.

In another study, Gabrilovich and Markovitch (2004) demonstrated that χ^2, IG and BNS were the top performers among other feature selection methods, and the differences between three of them were not statistically significant. On the other hand, Keerthi (2005) reported that IG outperformed BNS in his experiment, but stated that his dataset did not have a large class skew.

6 Classification Methods

As mentioned in the Introduction section, the project will use existing classification methods without extending them since they are not the focus of the project. As such, these methods are only briefly described in this review, together with their advantages and disadvantages. For the detail implementation, refer to the references.

6.1 Naïve Bayes

Naïve Bayes (NB) classification is based on probabilistic model to estimate $P(C|S)$, the probability of a category C given a sentence S (McCallum and Nigam, 1998). It incorporates a strong assumption that the words (w_1, ..., w_2) in a sentence are independent of each other given the category of that sentence. This assumption is clearly invalid. For example, when the word "thank" is detected in a sentence, the probability of the following word being detected as "you" is higher than other words.

The advantage of the independence assumption is that it makes the overall computation of $P(C|S)$ much simpler; thus, NB has a very good performance in terms of training time. In an environment where computation resource is limited, NB is suitable. If the independence assumption is not considered, the computation of $P(C|S)$ increases significantly.

In several text classification experiments, NB has shown inferior performance compared to DT and SVMs (Joachims, 1998; Dumais et al., 1998; Yang and Liu, 1999). In one sentence classification study, NB was also outperformed by other methods (Hachey and Grover, 2005). Other sentence classification studies did not compare the performance of classification methods.

6.2 Decision Tree

Decision Tree (DT) is a tree structure, in which internal nodes represent features, edges from the nodes represent the tests on the value of the features in the sentence, and the leafs represents categories (Sebastiani, 2002). A sentence S is categorized by recursively testing the weights of the features in S until a leaf node C is found. Thus, S is categorized into C.

The constructed decision tree can be considered as an "if-then-else" rules (Hasan and Rahman, 2003). It will be interesting to see what combination of words determines a sentence type, and DT is able visualize that. Cue words for certain sentence types, that are not previously observed by humans, may also be automatically identified as well.

DT has been applied to many text classification tasks and achieved a good performance in terms of classification effectiveness (Dumais et al., 1998; Johnson et al., 2002). In some studies, it was one of the best performer.

6.3 Support Vector Machines

Support Vector Machines (SVMs) for classification are a learning based on Structural Risk Minimization principle (Vapnik, 1995). The idea of the principle is to find a hyperplane that separates data points into two classes with maximum-margin. Thus, it is a binary classification. To apply it to sentence classification, m-binary classifiers will be built, where m equals the number of categories. Each binary classifier classifies a category. The downside of SVMs is that when there are many categories, the training time may increase substantially as more binary classifiers will need to be built.

A number of text classification experiments have reported that SVMs have the best classification effectiveness among existing methods (Dumais et al., 1998; Yang and Liu, 1999; Forman, 2003). SVMs are considered to be able to handle high-dimensional feature space, and it does not require feature selection (Joachims, 1998).

SVMs have also been used in several sentence classification studies (McKnight and Srinivasan, 2003; Corston-Oliver et al., 2004; Hachey and Grover, 2005). Only one of the studies involved comparison with other classification methods, and SVMs was one of the top performers.

7 Conclusion

Research on sentence classification has not been extensive compared to the research on text classification. Previous studies on sentence classification have explored several types of features that are very similar to those used in text classification. Bag-of-words is usually used as the baseline representation of a sentence/text in classification tasks. Other more complex features are incorporated together with bag-of-words. In text classification, using more complex features have not had significant improvement over classification accuracy. In sentence classification, it is not really clear how each type of features perform since some of the studies did not report the feature performances. This project will attempt to find the right combination of features that are suitable to represent the sentences in the selected domain.

Previous sentence classification experiments have not been observed to employ any feature selection methods. On the other hand, feature selection has been successfully employed in text classification, removing a significant number of irrelevant features and making a slight improvement over the classification effectiveness. This prompts an interest for this project to investigate whether existing text classification feature selection methods can be applicable to sentence classification as well. A combination of the methods may be used to achieve a better performance.

For the classification methods, SVMs has been generally considered as the best performer. DT performs slightly below SVMs but above NB. However, NB has a faster training time.

Since this project's domain is different from those of previous studies, it is necessary to devise its own tag set that suits the properties of the selected domain. It will be better

if some of the tags can be easily portable to the other domains. Thus, devising a tag set, exploring the features and investigating the (combination of) feature selection methods will become the major contributions of this project.

References

Apte, C., Damerau, F. and Weiss, S. M. (1994). Automated learning of decision rules for text categorization, *ACM Transactions on Information Systems* **12**(3): 233–251.

Bekkerman, R. and Allan, J. (2004). Using bigrams in text categorization, *Technical Report IR-408*, Center for Intelligent Information Retrieval, University of Massachusetts.

Benko, B. K. and Katona, T. (2005). On the efficient indexing of grammatical parse trees for information retrieval, *Proceedings of Innovations in Intelligent Systems and Applications*, Istanbul, Turkey, pp. 366–369.

Cardoso-Cachopo, A. and Oliveira, A. L. (2003). An empirical comparison of text categorization methods, *String Processing and Information Retrieval, 10th International Symposium*, Brazil, pp. 183–196.

Caropreso, M. F., Matwin, S. and Sebastiani, F. (2001). A learner-independent evaluation of the usefulness of statistical phrases for automated text categorization, *in* A. G. Chin (ed.), *Text Databases and Document Management: Theory and Practice*, Idea Group Publishing, Hershey, US, pp. 78–102.

Chu-Carroll, J. (1998). A statistical model for discourse act recognition in dialogue interactions, *Applying Machine Learning to Discourse Processing*, AAAI Press, Menlo Park, CA, pp. 12–17.

Clark, A. and Popescu-Belis, A. (2004). Multi-level dialogue act tags, *in* M. Strube and C. Sidner (eds), *Proceedings of the 5th SIGDIAL Workshop on Discourse and Dialogue*, Association for Computational Linguistics, Cambridge, Massachusetts, USA, pp. 163–170.

Cohen, W. W., Carvalho, V. R. and Mitchell, T. M. (2004). Learning to classify email into "Speech Acts", *Proceedings of EMNLP*, Association for Computational Linguistics, Barcelona, Spain, pp. 309–316.

Core, M. G. and Allen, J. F. (1997). Coding dialogues with the DAMSL annotation scheme, *in* D. Traum (ed.), *Working Notes: AAAI Fall Symposium on Communicative Action in Humans and Machines*, American Association for Artificial Intelligence, Menlo Park, California, pp. 28–35.

Corston-Oliver, S., Ringger, E., Gamon, M. and Campbell, R. (2004). Task-focused summarization of email, *Text Summarization Branches Out: Proceedings of the ACL-04 Workshop*, Association for Computational Linguistics, Barcelona, Spain, pp. 43–50.

Drucker, H., Wu, D. and Vapnik, V. N. (1999). Support Vector Machines for spam categorization, *IEEE Transactions on Neural Network* **10**(5): 1048–1054.

Dumais, S., Platt, J., Heckerman, D. and Sahami, M. (1998). Inductive learning algorithms and representation for text categorization, *Proceedings of 7th International Conference on Information and Knowledge Management*, pp. 148–155.

Forman, G. (2003). An extensive empirical study of feature selection metrics for text classification, *Journal of Machine Learning Research*, Vol. 3, MIT Press, USA, pp. 1289–1305.

Furnkranz, J. (1998). A study using n-gram features for text categorization, *Technical Report OEFAI-TR-98-30*, Austrian Institute for Artificial Intelligence.

Gabrilovich, E. and Markovitch, S. (2004). Text categorization with many redundant features: Using aggressive feature selection to make SVMs competitive with C4.5, *Proceedings of the Twenty-First International Conference on Machine Learning*, Alberta, Canada, pp. 321–328.

Gliozzo, A. M. and Strapparava, C. (2005). Domain kernels for word sense disambiguation, *Proceedings of the 43th Annual Meeting of the Association for Computational Linguistics*, Ann Arbor, Michigan, pp. 403–410.

Guo, G., Wang, H., Bell, D. A., Bi, Y. and Greer, K. (2004). An kNN model-based approach and its application in text categorization, *Computational Linguistics and Intelligence Text Processing, 5th International Conference, CICLing 2004*, Springer, Seoul, Korea, pp. 559–570.

Hachey, B. and Grover, C. (2005). Sequence modelling for sentence classification in a legal summarisation system, *Proceedings of the 2005 ACM Symposium on Applied Computing*, ACM Press, pp. 292–296.

Hasan, M. M. and Rahman, C. M. (2003). Text categorization using association rule based decision tree, *Proceedings of the 6th International Conference on Computer and Information Technology*, Bangladesh, pp. 453–456.

Ivanovic, E. (2005). Dialogue act tagging for instant messaging chat sessions, *Proceedings ACL Student Workshop 2005*, Association for Computational Linguistics, Ann Arbor, Michigan, pp. 79–84.

Jekat, S., Klein, A., Maier, E., Maleck, I., Mast, M. and Quantz, J. (1995). Dialogue Acts in VERBMOBIL, *Technical Report Verbmobil-Report 65*, Universitat Hamburg, DFKI GmbH, Universitat Erlangen and TU Berlin.

Joachims, T. (1998). Text categorization with Support Vector Machines: Learning with many relevant features, *European Conference on Machine Learning (ECML)*, pp. 137–142.

Joachims, T. (2002). *Learning to Classify Text Using Support Vector Machines: Methods, Theory and Algorithms*, Kluwer Academic Publishers, Massachusetts, USA.

Johnson, D. E., Oles, F. J., Zhang, T. and Goetz, T. (2002). A Decision-Tree-based symbolic rule induction system for text categorization, *IBM Systems Journal*, Vol. 41, pp. 428–437.

Jurafsky, D., Bates, R., Coccaro, N., Martin, R., Meteer, M., Ries, K., Shriberg, E., Stolcke, A., Taylor, P. and Ess-Dykema, C. V. (1997b). Automatic detection of discourse structure for speech recognition and understanding, *Proceedings of the IEEE Workshop on Speech Recognition and Understanding*, Santa Barbara, CA, pp. 88–95.

Jurafsky, D., Shriberg, E. and Biasca, D. (1997a). Switchboard SWBD-DAMSL shallow-discourse-function annotation coders manual, *Technical Report 97–02*, University of Colorado, Institute of Cognitive Science.
URL: *http://www.colorado.edu/ling/jurafsky/manual.august1.html*

Keerthi, S. S. (2005). Generalized LARS as an effective feature selection tool for text classification with SVMs, *in* L. D. Raedt and S. Wrobel (eds), *Proceedings of the 22nd International Machine Learning Conference*, ACM Press, Germany.

Koster, C. H. A. and Seutter, M. (2002). Taming wild phrases, *Proceedings 25th European Conference on IR Research*, pp. 161–176.

Lam, S. L. and Lee, D. L. (1999). Feature reduction for Neural Network based text categorization, *in* A. L. Chen and F. H. Lochovsky (eds), *Proceedings of DASFAA-99, 6th IEEE International Conference on Database Advanced Systems for Advanced Application*, IEEE Computer Society Press, Los Alamitos, US, Hsinchu, TW, pp. 195–202.

Lewis, D. D. (1992a). An evaluation of phrasal and clustered representations on a text categorization task, *Proceedings of SIGIR-92, 15th ACM International Conference on Research and Development in Information Retrieval*, Copenhagen, Denmark, pp. 37–50.

Lewis, D. D. (1992b). Feature selection and feature extraction for text categorization, *Proceedings of Speech and Natural Language Workshop*, Morgan Kaufmann, California, USA, pp. 212–217.

Liao, C., Alpha, S. and Dixon, P. (2003). Feature preparation in text categorization, *Proceedings of Autralasian Data Mining Worksop*, Canberra, Australia.

McCallum, A. and Nigam, K. (1998). A comparison of event models for Naive Bayes text classification, *AAAI-98 Workshop on Learning for Text Categorization*, AAAI Press, pp. 41–48.

McKnight, L. and Srinivasan, P. (2003). Categorization of sentence types in medical abstracts, *Proceedings of the American Medical Informatics Association Annual Symposium*, pp. 440–444.

Rogati, M. and Yang, Y. (2002). High-performing feature selection for text classification, *Proceedings of the Eleventh International Conference on Information and Knowledge Management*, ACM Press, pp. 659–661.

Salton, G. and Buckley, C. (1988). Term weighting approaches in automatic text retrieval, *Information Processing and Management* **24**(5): 513–523.

Salton, G. and McGill, M. J. (1983). *An Introduction to Modern Information Retrieval*, McGraw-Hill.

Scott, S. and Matwin, S. (1999). Feature engineering for text classification, *Proceedings of ICML-99, 16th International Conference on Machine Learning*, Slovenia, pp. 379–388.

Sebastiani, F. (2002). Machine learning in automated text categorization, *ACM Computing Surveys* **34**(1): 1–47.

Segal, R. and Kephart, J. O. (1999). Mailcat: An intelligent assistant for organizing e-mail, *International Conference on Autonomous Agents*, pp. 276–282.

Seki, K. and Mostafa, J. (2005). An application of text categorization methods to gene ontology annotation, *Proceedings of the 28th annual international ACM SIGIR conference on Research and development in information retrieval*, ACM Press, pp. 138–145.

Siegel, S. and Castellan, N. J. (1988). *Nonparametric Statistics for the Behavioural Sciences*, 2nd edn, McGraw-Hill, New York.

Soucy, P. and Mineau, G. W. (2001). A simple kNN algorithm for text categorization, *Proceedings IEEE International Conference on Data Mining*, San Jose, USA, pp. 647–648.

Stolcke, A., Ries, K., Coccaro, N., Shriberg, E., Bates, R., Jurafsky, D., Taylor, P., Martin, R., Ess-Dykema, C. V. and Meteer, M. (2000). Dialogue Act modeling for automatic tagging and recognition of conversational speech, *Computational Linguistics* **26**(3): 339–373.

Taylor, P., King, S., Isard, S. and Wright, H. (1998). Intonation and dialog context as constraints for speech recognition, *Language and Speech* **41**(3–4): 489–508.

Teufel, S. and Moens, M. (2002). Summarising scientific articles - experiments with relevance and rhetorical status, *Computational Linguistics* **28**(4): 409–445.

Tzoukermann, E., Muresan, S. and Klavans, J. L. (2001). GIST-IT: Summarizing email using linguistic knowledge and machine learning, *Proceedings of the HLT and KM Workshop, EACL/ACL 2001*, France.

van Rijsbergen, C. J. (1979). *Information Retrieval*, 2nd edn, Butterworths, London.

Vapnik, V. N. (1995). *The Nature of Statistical Learning Theory*, Springer, New York.

Warnke, V., Kompe, R., Niemann, H. and Noth, E. (1997). Integrated Dialog Act segmentation and classification using prosodic features and language models, *Proceedings of the 5th European Conference on Speech Communication and Technology*, Vol. 1, Rhodes, Greece, pp. 207–210.

Wright, H. (1998). Automatic utterance type detection using suprasegmental features, *in* R. H. Mannell and J. Robert-Ribes (eds), *Proceedings of the International Conference on Spoken Language Processing*, Vol. 4, Sydney, pp. 1403–1406.

Yang, Y. and Liu, X. (1999). A re-examination of text categorization methods, *22nd Annual International SIGIR*, Berkley, pp. 42–49.

Yang, Y. and Pedersen, J. O. (1997). A comparative study on feature selection in text categorization, *in* D. H. Fisher (ed.), *Proceedings of ICML-97, 14th International Conference on Machine Learning*, Morgan Kaufmann Publishers, San Francisco, US, Nashville, US, pp. 412–420.