Hybrid DIAAF/RS: Statistical Textual Feature Selection for Language-independent Text Classification

Yanbo J. Wang[1], Fan Li[1], Frans Coenen[2], Robert Sanderson[3], and Qin Xin[4]

[1] Information Management Center, China Minsheng Banking Corp., Ltd., Beijing, China
{wangyanbo, lifan}@cmbc.com.cn
[2] Department of Computer Science, University of Liverpool, Liverpool, UK
coenen@liverpool.ac.uk
[3] Los Alamos National Laboratory, Los Alamos, New Mexico, USA
rsanderson@lanl.gov
[4] Simula Research Laboratory, Oslo, Norway
xin@simula.no

Abstract. *Textual Feature Selection* (*TFS*) is an important phase in the process of *text classification*. It aims to identify the most significant textual features (i.e. *key* words and/or phrases), in a textual dataset, that serve to distinguish between text categories. In TFS, basic techniques can be divided into two groups: *linguistic* vs. *statistical*. For the purpose of building a *language-independent* text classifier, the study reported here is concerned with statistical TFS only. In this paper, we propose a novel statistical TFS approach that hybridizes the ideas of two existing techniques, DIAAF (Darmstadt Indexing Approach Association Factor) and RS (Relevancy Score). With respect to *associative* (*text*) *classification*, the experimental results demonstrate that the proposed approach can produce greater classification accuracy than other alternative approaches.

Keywords: Associative Classification, (Language-independent) Text Classification, Text Mining, Textual Feature Selection.

1 Introduction

1.1 General Background

The increasing number of electronic documents that are available to be explored on-line has led to *text mining* becoming a promising school of current research in *Knowledge Discovery in Data* (*KDD*), and is attracting increasing attention from a wide range of different groups of people. Text mining aims to extract various models of hidden, interesting, previously unknown and potentially useful knowledge (i.e. rules, patterns, regularities, customs, trends, etc.) from sets of collected textual data (i.e. web news, e-mails, research papers, meeting minutes, etc.), where a collected textual dataset can be sized in *Giga-bytes*. In a natural language context, a given textual dataset is commonly refined to produce a *documentbase* — a set of electronic documents that typically consists of thousands of documents, where each document may contain hundreds of words.

One major application of text mining is *Text Classification*/*Categorization* (*TC*) — the automated assignation of "unseen" documents into predefined text groups. TC, as a well established research filed, has been studied for almost half a century; early work on TC can be dated back to the 1960s (see for instance [21]). During the past decade, TC has been extensively investigated at the intersection of research into KDD and machine learning. Machine learning based TC focuses on *directly* assigning "unseen" documents into text categories without being concerned with presenting to end users reasons why and how the classification predictions have been made. KDD based TC typically mines and generates human readable classification rules from textual data that are further used to build a text classifier for assigning "unseen" documents into text classes; such generated textual rules can be presented to the end user. In our study, we concentrate on KDD based TC.

In general, TC can be divided into two groups: (i) *single-label* TC, which assigns each "unseen" document into exactly one (predefined) text class; and (ii) *multi-label* TC, which assigns each "unseen" document into one or more text class. With respect to single-label TC, three different approaches can be identified: (i) *one-class* TC, which learns from positive document samples only, and either assigns an "unseen" document into the predefined (text) class or ignores the assignation of this document; (ii) *two-class* (or *binary*) TC, which learns from both positive and negative document samples, and assigns each "unseen" document into the predefined class or the complement of this class; and (iii) *multi-class* TC, which simultaneously deals with all given classes comprising all document samples, and assigns each "unseen" document into the most appropriate class. This paper is concerned with the single-label multi-class TC study.

Usually text mining requires the given documentbase to be first preprocessed so that it is in an appropriate format. Hence the process of TC, in a general context, can be identified as *documentbase preprocessing* plus *data classification*. The nature of such preprocessing comprises: (i) *documentbase representation*, the process of creating a data model to precisely interpret a given documentbase in an explicit and structured manner; and (ii) *Textual Feature Selection* (*TFS*), the process of extracting the most significant textual information from the given documentbase.

In documentbase representations, the "*bag of* *" or *Vector Space Model* (*VSM*) [25] is considered to be appropriate for many text mining applications. The VSM can be described as follows: given a documentbase $Đ$, each document $D_j \in Đ$ is represented by a single numeric vector, and each vector is a subset of some vocabulary V. The vocabulary V is a representation of the set of textual features (documentbase attributes) that are used to characterize the documents. The VSM is usually presented in a *binary form*, where "*each coordinate of a document vector is zero (when the corresponding attribute is absent) or unity (when the corresponding attribute is present)*" [16]. In TC, there are two major approaches used to define the "bag of *" (vector space) model: the "*bag of words*" and the "*bag of phrases*". The experimental work, in this paper, is designed with respect to both approaches.

Theoretically speaking, the textual features of a document can include every word or phrase that might be expected to occur in a given documentbase. However, this is computationally unrealistic, so it requires some method of preprocessing documents to identify the *key* textual features that will be useful for a particular text mining application, such as TC. TFS aims to select a limited number of textual features from the entire set representing the documentbase. With respect to TFS (sometimes referred to as "*textual feature reduction*"), techniques can be generally divided into two groups: *linguistic* and *statistical*.

Linguistic TFS methods identify significant textual features depending on the rules and/or regularities in semantics, syntax and/or lexicology. Typical methods in this group include: *stop*-word lists, stemming, lemmatization, *part-of-speech* tagging, etc. Such techniques are designed with particular languages and styles of language as the target, and involve deep linguistic analysis. For the purpose of building a *language-independent* text classifier (e.g. [8, 29]) that is generally applicable to *cross-lingual*, *multi-lingual* and/or *unknown-lingual* textual data collections, the statistical approach is most appropriate. This is the focus of this paper. A number of statistical TS mechanisms have been proposed, including: Darmstadt Indexing Approach Association Factor (DIAAF), Relevancy Score (RS), Mutual Information (MI), etc.

Classification (or "*data categorization*") deals with structured data, especially *tabular* data, and aims to assign "unseen" data instances into predefined data groups, based on a classifier constructed from a training set of data instances associating with (predefined) class-labels. Mechanisms on which classification algorithms have been based can be separated into two "families": (i) *classification direct learning*, classification without rule generation; and (ii) *classification rule mining* (e.g. [23]), classification with rule generation (and presentation).

Classification direct learning algorithms focus on directly categorizing "unseen" data records into predefined data groups without concern for presenting, to the end users, why and how the categorization predictions have been made. Typical mechanisms include: naïve Bayes, support vector machine and neural networks. Classification rule mining algorithms mine and generate human readable Classification Rules (CRs), again with the objective of building a classifier to classify "unseen" data instances. Typical approaches include: decision trees (C4.5) [23] and RIPPER [9] (Repeated Incremental Pruning to Produce Error Reduction).

One approach to classification rule mining other than C4.5 and RIPPER is to employ Association Rule Mining (ARM) [1] methods to identify the desired CRs, i.e. *associative classification* [2]. Associative classification mines a set of Classification Association Rules (CARs) from a *class-transactional database*. The authors of [6] and the authors of [28] together suggested that results presented in the studies of [19, 20, 32] show that in many cases associative classification offers greater classification accuracy than other classification rule mining methods, such as C4.5 and RIPPER.

During the past decade, associative classification has been applied to TC (e.g. [3, 8, 29, 33]). Note that the binary format of the VSM representation translates easily into the class-transactional format. The advantages offered by associative classification, with respect to other classification rule mining approaches, can be summarized by quoting Antonie and Zaïane [3]:

- Associative text classifier "*is fast during both training and categorization phases*", especially when handling very large databases [3].
- An associative text classifier "*can be read, understood and modified by humans*".

Given the above advantages offered by associative classification with respect to TC, this approach has been adopted in this paper to support the study of statistical TFS for language-independent TC.

1.2 Contribution

A hybrid statistical TFS approach is proposed, which integrates the ideas of two existing (statistical TFS) techniques: DIAAF (Darmstadt Indexing Approach Association Factor) and RS (Relevancy Score), namely Hybrid DIAAF/RS. The evaluation of Hybrid DIAAF/RS, under both the language-independent "bag of words" and "bag of phrases" documentbase representation settings, was conducted using the TFPC (Total From Partial Classification) associative classifier [5, 6, 7]; although any other associative classification algorithm could equally well have been employed. With respect to associative TC, the experimental results demonstrate that Hybrid DIAAF/RS can produce better classification accuracy than other statistical TFS approaches (e.g. DIAAF, RS, MI), thus improving the performance of language-independent TC.

1.3 Paper Organization

The rest of this paper is organized as follows. Section 2 describes some related work relevant to our study, where both the language-independent "bag of words" and "bag of phrases" approaches are reviewed. The DIAAF and RS as well as MI statistical TFS mechanisms are outlined in section 3. In section 4, we propose the Hybrid DIAAF/RS (statistical TFS) approach. The experimental results are presented in section 5. Finally our conclusions and open issues for further research are given in section 6.

2 Documentbase Representation

2.1 Language-independent "Bag of Words"

The "bag of words" approach has been used in TC investigation for a long time. In this approach, each document is represented by the set of words that are used in the document. Information on the ordering of words within documents as well as the structure of the documents is lost. The problem with this approach is how to effectively and efficiently select a limited, computationally manageable, subset of words from the entire set represented in the documentbase. Usually the "bag of words" approach first removes all punctuation marks (sometimes, all non-alphabetic characters, i.e. numbers, symbols, etc.) from the original documentbase. Then significant words that contribute to the TC task are selected using TFS.

In [8] the authors introduce a three-phase framework for language-independent "bag of words" construction (as follows):

1. Words are first defined in a documentbase "*as continuous sequences of alphabetic characters delimited by non-alphabetic characters, e.g. punctuation marks, white space and numbers*"; all non-alphabetic characters are then removed from the documentbase.
2. Common and rare words are collectively considered to be the *noise* words in a documentbase. They can be identified by their *support* value, i.e. the percentage of documents in the training dataset in which the word appears. Common words are words with a support value above a user-defined Upper Noise Threshold (UNT), and are referred to as upper noise words. Rare words are those with a support value below a user-defined Lower Noise Threshold (LNT), and are referred to as lower noise words. Both upper and lower noise words are then removed from the documentbase.
3. The desired set of significant words is drawn from an ordered list of potential significant words. A potential significant word also referred to as a key word is a non-noise word whose *contribution* value exceeds some user-specified threshold *G*. The contribution value of a word is a measure of the extent to which the word serves to differentiate between classes and can be calculated in a number of ways. Finally the first *K* words are selected from the ordered list of potential significant words, which are further concerned in the CRM stage of TC.

In the third phase, those words whose contribution value exceeds the threshold *G* are placed into a potential significant word list, in descending ordered according to the contribution value. This list may include words that are significant for more than one class (noted as "*all words*"), or it may be decided to include only those words that are significant with respect to one class only (i.e. "*uniques*"). From the potential significant word list the final list of significant words are chosen. Two strategies can be proposed for achieving this. The first is to simply choose the first *K* words from the ordered list (the "*top K*"). This may, however, result in an unequal/unbalanced distribution of significant words between classes. The second approach chooses the top "K / $|C|$" words for each class (referred to as "*dist*"), so as to include an equal number of significant words for each class, where *C* is the set of predefined classes within a documentbase.

2.2 Language-independent "Bag of Phrases"

Instead of representing a documentbase using words, many TC studies consider the usage of phrases. In the "bag of phrases" approach, each element in a document vector represents a phrase describing an ordered combination of words appearing contiguously in sequence (sometimes with some *maximum word gap*). The motivation for this approach is that phrases carry more contextual and/or syntactic information than single words. For example Scheffer and Wrobel [26] argue that the "bag of words" representation does not distinguish between "*I have no objections, thanks*" and "*No thanks, I have objections*".

One "bag of phrases" approach is to use *n*-grams (see for instance [22]), where each sequence of *n* ordered and adjacent words in a document is identified as a phrase

($n \leq$ the size of the document). However, the main question with respect to *n*-grams is what should the value of *n* be? This remains a current research issue.

In [8] the authors propose a language-independent "bag of phrases" approach based on the language-independent "bag of words" construction (see section 2.1). In section 2.1, three categories of word were defined:

- **Upper Noise Words:** Words whose support is above a user-defined UNT (Upper Noise Threshold);
- **Lower Noise Words:** Words whose support is below a user-defined LNT (Lower Noise Threshold); and
- **Significant Words (G):** Selected key words that are expected to serve to distinguish between classes.

In this section, another two categories of word are further defined (also as introduced in [8]):

- **Ordinary Words (O):** Other non-noise words that have not been selected as significant words; and
- **Stop Marks (S):** The "key" punctuation marks: '*,*' '*.*' '*:*' '*;*' '*!*' and '*?*', referred to as *delimiters*, and used in phrase identification. All other non-alphabetic characters are ignored.

It also identifies (in [8]) two groups of categories of words:

- **Noise Words (N):** The union of upper and lower noise words; and
- **Non-noise Words:** The union of significant and ordinary words.

Significant phrases are defined as sequences of words that include at least one significant word. Four different schemes for determining phrases (and constructing a "bag of phrases") were distinguished in [8], depending on: (i) what are used as *delimiters* and (ii) what the *contents* of the phrase should be made up of:

- **DelSNcontGO:** Phrases are delimited by stop marks (S) and/or noise words (N), and made up of sequences of one or more significant words (G) and ordinary words (O). Sequences of ordinary words delimited by stop marks and/or noise words that do not include at least one significant word are ignored.
- **DelSNcontGW:** As DelSNcontGO but replacing ordinary words in phrases by *wild card* symbols (W) that can be matched to any single word. The idea here is that much more generic phrases are generated.
- **DelSOcontGN:** Phrases are delimited by stop marks (S) and/or ordinary words (O), and made up of sequences of one or more significant words (G) and noise words (N). Sequences of noise words delimited by stop marks and/or ordinary words that do not include at least one significant word are ignored.
- **DelSOcontGW:** As DelSOcontGN but replacing noise words in phrases by *wild card* characters (W). Again the idea of this scheme is to produce generic phrases.

The experimental results presented in [8] show that, with respect to the accuracy of classification, DelSNcontGO outperforms other alternative schemes. In this paper, the DelSNcontGO language-independent "bag of phrases" approach will be returned to in Section 5 (experimental results).

3 Statistical Textual Feature Selection

Statistical TFS mechanisms are desired to automatically calculate a weighting score for each textual feature in a document. A significant textual feature is one whose weighting score exceeds a user-supplied weighting threshold. These techniques do not involve linguistic analysis. With regard to TC, the common intuitions are as follows:

- The more times a textual feature appears across the documentbase in documents of all classes the worse it is at discriminating between the classes.
- The more times a textual feature uniquely appears in a class the more relevant it is to this particular class.

In the past, a number of statistical models have been proposed in statistical TFS; three major ones are introduced as follows: Darmstadt Indexing Approach Association Factor (DIAAF), Relevancy Score (RS), and Mutual Information (MI).

- **DIAAF:** Originally, the Darmstadt Indexing Approach (DIA) [13] was "*developed for automatic indexing with a prescribed indexing vocabulary*" [14]. In machine learning, the author of [27] indicates that DIA "*considers properties (of terms, documents, categories, or pairwise relationships among these) as basic dimensions of the learning space*". Examples of such properties include document length, occurrence frequency between textual features and predefined classes, training data generality of each predefined class, etc. One pair-wise relationship in consideration herein is the term-category relationship, noted as the DIA Association Factor (DIAAF) [27], which can be employed to select significant textual features for TC problems. The computation of DIAAF score, also reported in [12], is achieved by using a probabilistic (**Pr**) form:

  $$diaaf\_score(u_h, C_i) = \mathbf{Pr}(C_i \mid u_h) = count(u_h \in C_i) \;/\; count(u_h \in Đ)\ ,$$

  where $Đ$ represents a given documentbase, u_h represents a textual feature in $Đ$, C_i represents a set of documents (in $Đ$) labeling with a particular text class, $count(u_h \in C_i)$ is the number of documents containing u_h in C_i, and $count(u_h \in Đ)$ is the number of documents containing u_h in $Đ$. The DIAAF score expresses the proportion of the feature's occurrence in the given class divided by the feature's documentbase occurrence.
- **RS:** The initial concept of RS was given by Salton and Buckley [24], as relevancy weight. It aims to measure how "unbalanced" a textual feature (term) u_h is across documents in a documentbase $Đ$ with and without a particular text class C_i. They define a term's relevancy weight as: "*the proportion of relevant documents in which a term occurs divided by the proportion of nonrelevant*

items in which the term occurs" [24]. In [31] the idea of RS was based on relevancy weight with the objective of selecting significant textual features in Đ for the TC application. A term's relevancy score can be defined (in logarithm) as: the number of relevant (the target text class associated) documents in which a term occurs divided by the number of non-relevant documents in which a term occurs. Sebastiani [27] and Fragoudis *et al.* [12] calculate the RS score in probabilistic (**Pr**) form using:

$$relevancy\_score(u_h, C_i) = log((\mathbf{Pr}(u_h \mid C_i) + d) / (\mathbf{Pr}(u_h \mid \neg C_i) + d)) \ ,$$

where $\neg C_i$ (equals to $Đ - C_i$) represents the set of documents labeling with the complement of the predefined class C_i, and d is a constant damping factor. In [31] the value of d was initialized as 1/6. This formula can also be written in the following form:

$$relevancy\_score(u_h, C_i) = log((count(u_h \in C_i) / |C_i| + d) \\ / (count(u_h \in (Đ - C_i)) / |Đ - C_i| + d)) \ ,$$

where $|C_i|$ is the size function of set C_i, $|Đ - C_i|$ is the size function of set $Đ - C_i$, and $count(u_h \in (Đ - C_i))$ is the number of documents containing u_h in $Đ - C_i$.

- **MI:** Another important existing statistical TFS mechanism other than DIAAF and RS is Mutual Information (MI). Early study of MI can be seen in [4] and [11]. This statistical model is applied to determine whether a genuine association exists between two textual features or not. In TC, MI has been broadly utilized in a variety of approaches to select the most significant textual features that serve to classify documents. The computation of the MI score between a textual feature u_h and a predefined text class C_i, also reported in [12], is achieved using:

  $$mi\_score(u_h, C_i) = log(\mathbf{Pr}(u_h \mid C_i) / \mathbf{Pr}(u_h)) \ .$$

  This score expresses the proportion (in a logarithmic term) of the frequency with which the feature occurs in documents of the given class divided by the feature's documentbase frequency.

4 Proposed Textual Feature Selection

With respect to language-independent TC, we propose a novel statistical TFS technique in this section. In the previous section, two statistical TFS mechanisms DIAAF and RS were described. The proposed technique is a variant of the original RS approach that makes use of the DIAAF approach, namely Hybrid DIAAF/RS.

Recall that the formula for calculating the RS score is given by:

$$relevancy\_score(u_h, C_i) = log((\mathbf{Pr}(u_h \mid C_i) + d) / (\mathbf{Pr}(u_h \mid \neg C_i) + d)) \ .$$

The core computations here can be recognized as $\mathbf{Pr}(u_h \mid C_i)$ and $\mathbf{Pr}(u_h \mid \neg C_i)$. The DIAAF score is calculated using:

$$diaaf\_score(u_h, C_i) = \mathbf{Pr}(C_i \mid u_h) \ .$$

Substituting for the core computations into the RS score formula using the DIAAF (related) formula, a new RS fashion formula (Hybrid DIAAF/RS) is defined:

$$diaaf\text{-}relevancy\_score(u_h, C_i) = log((\mathbf{Pr}(C_i \mid u_h) + d) \,/\, (\mathbf{Pr}(C_i \mid \neg u_h) + d))\ ,$$

where $\neg u_h$ represents a document that does not involve the feature u_h, and d is a constant damping factor (as mentioned in the original RS). The formula can be further expanded as:

$$diaaf\text{-}relevancy\_score(u_h, C_i) = log((count(u_h \in C_i) \,/\, count(u_h \in \text{Đ}) + d) \,/ \\ ((count(\neg u_h \in C_i) \,/\, count(\neg u_h \in \text{Đ}) + d))\ ,$$

where $count(\neg u_h \in C_i)$ is the number of documents containing no u_h in C_i, and $count(\neg u_h \in \text{Đ})$ is the number of documents containing no u_h in *Đ*.

The algorithm for identifying significant textual features (i.e. key words in our situation, with regard to sections 2.1 and 2.2) in *Đ*, based on Hybrid DIAAF/RS, is given in Algorithm 1 (as follows):

```
Algorithm 1: Key Word Identification - Hybrid DIAAF/RS
Input:(a) A documentbase Đ (the training part, where the
          noise words have been removed);
      (b) A user-defined significance threshold $G$;
      (c) A constant damping factor $d$;
Output: A set of identified key words $S_{KW}$;
Begin Algorithm:
    $S_{KW}$ ← an empty set for holding the identified key
               words in Đ;
    $C$ ← catch the set of predefined text classes
            within Đ;
    $W_{GLO}$ ← read Đ to create a global word set, where the
               word documentbase support $supp_{GLO}$ is
               associated with each word $u_h$ in $W_{GLO}$;
    for each $C_i \in C$ do
        $W_{LOC}$ ← read documents that reference $C_i$ to
                   create a local word set, where the local
                   support $supp_{LOC}$ is associated with each
                   word $u_h$ in $W_{LOC}$;
        for each word $u_h \in W_{LOC}$ do
            contribution ← $log(((u_h.supp_{LOC} / u_h.supp_{GLO})$
                           $+ d) / (((|C_i| - u_h.supp_{LOC})$
                           $/ (|Đ| - u_h.supp_{GLO}) + d))$;
            if (contribution $\geq G$) then
                add $u_h$ into $S_{KW}$;
        end for
    end for
    return ($S_{KW}$);
End Algorithm
```

An example of Hybrid DIAAF/RS score calculation is provided in Table 1. Given a documentbase *Đ* containing 100 documents equally divided into 4 classes (i.e. 25

per class), and assuming that word u_h appears in 30 of the documents and that the value of d (constant damping factor) is 0, then the Hybrid DIAAF/RS score per class can be calculated as shown in the table.

Table 1. An example of the Hybrid DIAAF/RS score calculation.

Class	**# docs per class**	**# docs with u_h per class**	**# docs without u_h per class**	**# docs with u_h in Ð**	**# docs without u_h in Ð**	**$\Pr(C_i \mid u_h) + d$**	**$\Pr(C_i \mid \neg u_h) + d$**	**Hybrid DIAAF/ RS Score**
1	25	15	10	30	70	0.500	0.143	0.544
2	25	10	15	30	70	0.333	0.214	0.192
3	25	5	20	30	70	0.167	0.286	-0.234
4	25	0	25	30	70	0	0.357	$-\infty$

The rationale of this approach is that a significant textual feature (term) with respect to a particular text class should have:

1. A high ratio of the class based term support (document frequency) to the documentbase term support; and/or
2. A low ratio of the class based term support of non-appearance to the documentbase term support of non-appearance.

5 Experimental Results

In this section, we present an evaluation of our proposed statistical TFS approach, using three popular text collections: Usenet Articles, Reuters-21578 and MedLine-OHSUMED. The aim of this evaluation is to assess the approach with respect to the accuracy of classification in both language-independent "bag of words" (section 2.1) and "bag of phrases" (section 2.2) settings. All evaluations given in this section were conducted using the TFPC[1] associative classification algorithm; although any other associative classifier could equally well have been employed. All algorithms involved in the evaluation were implemented using the standard Java programming language. The experiments were run on a 1.87 GHz Intel(R) Core(TM)2 CPU with 2.00 GB of RAM running under Windows Command Processor.

5.1 Experimental Data Description

For the experiments outlined in the following subsections, five individual documentbases were used. Each was extracted (with regard to the documentbase extraction idea in [30]) from one of the three above mentioned text collections.

[1] TFPC software may be obtained from http://www.csc.liv.ac.uk/~frans/KDD/Software/Apriori-TFPC/aprioriTFPC.html

The Usenet Articles collection is a popular text collection compiled by Lang [17] from 20 different newsgroups, and is sometimes referred to as the "20 Newsgroups" collection. Each newsgroup represents a predefined class. There are exactly 1,000 documents per class with one exception, the class "soc.religion.christian" that contains 997 documents only. In comparison with other common text collections, the structure of "20 Newsgroups" is relatively "*neat*", every document is labeled with one class only, and almost all documents have a "proper" text-content. In the context of this paper, a proper text-content document is one that contains at least *q recognized* words. The value of *q* is usually small (*q* is set to be 20 in our study). Previous TC studies have used this text collection in various ways. For example, in [10] the entire "20 Newsgroups" was randomly divided into two non-overlapping and (almost) equally sized documentbases covering 10 classes each. In this paper we adopted the approach of [10]. The entire collection was randomly split into two documentbases covering 10 classes each: 20NG.D10000.C10 and 20NG.D9997.C10.

Reuters-21578 is another well known text collection widely applied in text mining. It comprises 21,578 documents collected from the Reuters newswire service with 135 predefined classes. However, many TC studies (see for example [18, 34]) have used only the 10 most populous classes. There are 68 classes that consist of fewer than 10 documents, and many others consist of fewer than 100 documents. The extracted documentbase, suggested in [18] and [34], is referred to as Reuters.D10247.C10 and comprises 10,247 documents with 10 classes. However this documentbase includes multi-labeled documents that are inappropriate for a single-label TC investigation (the approach adopted in our study). In this paper, the processing of the Reuters-21578 based documentbase comprised two stages: (1) identification of the top-10 populous classes, as in [18] and [34]; and (2) removal of multi-labeled and/or non-text documents from each class. As a consequence the class "wheat" had only one "*qualified*" document, and no document was found for class "corn". Hence, the final documentbase, namely Reuters.D6643.C8, omitted the "wheat" and "corn", classes leaving a total of 6,643 documents in 8 classes.

The MedLine-OHSUMED text collection, collected by Hersh *et al.* [15], consists of 348,566 records relating to 14,631 predefined MeSH (Medical Subject Headings) categories. The OHSUMED collection accounts for a subset of the MedLine text collection for 1987 to 1991. The process of extracting a documentbase from MedLine-OHSUMED in our study can be detailed as follows. First, the top-100 most populous classes were identified in the collection. These included many super-and-sub class-relationships. Due to the difficulty of obtaining a precise description of all the possible taxonomy-like class-relationships, we simply selected two sets (groups) of 10 target-classes from these classes by hand, so as to exclude obvious super and sub class-relationships in each group. Documents that are either multi-labeled or without a proper text-content (containing < *q* recognized words) were then removed from each class. Finally two documentbases, namely OHSUMED.D6855.C10 and OHSUMED.D7427.C10, were created.

5.2 Results using the "Bag of Words" Representation

This section, reports on a set of experiments to evaluate the proposed Hybrid DIAAF/RS TFS approach, in comparison of alternative mechanisms (i.e. DIAAF, RS, and MI), with respect to the "bag of words" representation. Accuracy figures, describing the proportion of correctly classified "unseen" documents, were obtained using Ten-fold Cross Validation (TCV). A *support* threshold value of 0.1%, a *confidence* threshold value of 35% and a Lower Noise Threshold (LNT) value of 0.2% were used as suggested in [8] and [29]. The Upper Noise Threshold (UNT) value was set to be 20%. Following the main findings of [8] the evaluations were conducted using: (i) the "all words" rather than "uniques" strategy in the construction of a potential significant word list, and (ii) the "dist" rather than "top *K*" strategy for choosing the final significant words. The parameter *K* (maximum number of selected final significant words) was set to 1,000. To ensure that sufficient potential significant words were generated for each category, the *G* parameter was given a zero minimal value so that the parameter could be ignored. In both RS and Hybrid DIAAF/RS, 0 was used as the constant damping factor value.

Table 2. Classification accuracy — comparison of the four statistical TFS approaches in the language-independent "bag of words" setting.

	DIAAF	**RS**	**MI**	**DIAAF/RS**
20NG.D10000.C10	76.72	76.72	76.72	**77.01**
20NG.D9997.C10	80.61	80.61	80.61	**80.75**
Reuters.D6643.C8	85.40	86.34	86.56	**86.81**
OHSUMED.D6855.C10	77.54	**79.28**	79.27	79.17
OHSUMED.D7427.C10	**78.97**	77.21	77.45	78.12
Average Accuracy	79.85	80.03	80.12	**80.37**
# of Best Accuracies	1	1	0	**3**

The results presented in Table 2 compare 20 classification accuracy values (using the "bag of words" representation) using the test documentbases. From Table 2 it can be seen that the proposed Hybrid DIAAF/RS technique worked better than the other alternative approaches:

1. The overall average classification accuracy throughout can be ranked in order as: Hybrid DIAAF/RS (80.37%), MI (80.12%), RS (80.03%) and DIAAF (79.85%).
2. The number of cases of best classification accuracies obtained throughout the five documentbases can be ranked in order as: Hybrid DIAAF/RS (3 out of 5 cases), DIAAF (1 case), RS (1 case), and MI (none of any case).

5.3 Results using the "Bag of Phrases" Representation

In this section, we present the experimental results comparing the proposed Hybrid DIAAF/RS TFS approach with previously developed TFS methods (i.e. DIAAF, RS, and MI) using the language-independent "bag of phrases" representation. According to the results presented in [8], the DelSNcontGO phrase generation scheme outperforms other alternative schemes, thus DelSNcontGO was selected to be used in our experiments. All parameters in this section were kept consistent to the parameter setting described in section 5.2 except that K was set to 900 for the OHSUMED documentbases. The reason to decrease the value of K was that using K = 1,000 generated more than 2^{15} while the TFPC associative classifier limited the total number of identified attributes[2] (significant words/phrases) to 2^{15}.

Table 3. Classification accuracy — comparison of the four statistical TFS approaches in the language-independent "bag of phrases" setting.

	DIAAF	**RS**	**MI**	**DIAAF/RS**
20NG.D10000.C10	76.96	76.96	76.96	**77.32**
20NG.D9997.C10	81.72	81.72	81.72	**82.09**
Reuters.D6643.C8	87.63	87.94	87.99	**88.53**
OHSUMED.D6855.C10	79.20	**80.16**	80.04	80.03
OHSUMED.D7427.C10	**78.24**	75.80	75.75	77.07
Average Accuracy	80.75	80.52	80.49	**81.01**
# of Best Accuracies	1	1	0	**3**

Table 3 gives the 20 classification accuracy values obtained using the given documentbases. From Table 3 it can be seen that the proposed Hybrid DIAAF/RS approach outperforms the other alternative approaches:

1. The overall average classification accuracy can be ranked ordered as follows: Hybrid DIAAF/RS (81.01%), DIAAF (80.75%), RS (80.52%) and MI (80.49%).
2. The number of cases of best classification accuracies obtained throughout the five documentbases can be ranked in order as: Hybrid DIAAF/RS (3 out of 5 cases), DIAAF (1 case), RS (1 case), and MI (none of any case).

6 Conclusions

This paper is concerned with an investigation of the statistical textual feature selection for (single-label multi-class) language-independent text classification. An overview of the language-independent documentbase preprocessing, in terms of the "bag of words" and the "bag of phrases" documentbase representations, was provided in section 2. Both the DIAAF and RS statistical TFS techniques were reviewed in section 3. A Hybrid DIAAF/RS (statistical) TFS approach was consequently introduced in section 4, which integrates the ideas of DIAAF and RS. From the

[2] The TFPC algorithm stores attributes as a signed short integer.

experimental results, it can be seen that the proposed Hybrid DIAAF/RS approach outperforms other alternative (statistical TFS) mechanisms in both the language-independent "bag of words" and "bag of phrases" settings regarding the approach of associative classification, Hybrid DIAAF/RS produced the greatest average classification accuracy and the highest number of cases of best classification accuracies throughout the five chosen textual datasets (documentbases). This in turn improves the performance of language-independent text classification.

The results presented in this paper corroborate that the traditional text classification problem can be solved, with good classification accuracy, in a language-independent manner. Further research is suggested to identify the improved statistical textual feature selection mechanism and further improve the performance of language-independent text classification.

Acknowledgments. The authors would like to thank Dr. Jiongyu Li from the China Minsheng Banking Corp., Ltd., Professor Paul Leng and Chuntao Jiang from the University of Liverpool, Songpo Wang from IBM China, Fan Wu from the Beijing eMay Softcom Technology Ltd., and Zhijie Jia from the Beijing Friendship Hotel for their support with respect to the work described here.

References

1. Agrawal. R., Imielinski, T., Swami, A.: Mining Association Rules between Sets of Items in Large Database. In: Proceedings of the 1993 ACM SIGMOD International Conference on Management of Data, Washington, DC, USA, May 1993, pp. 207-216. ACM Press (1993)
2. Ali, K., Manganaris, S., Srikant, R.: Partial Classification using Association Rules. In: Proceedings of the Third International Conference on Knowledge Discovery and Data Mining, Newport Beach, CA, USA, August 1997, pp. 115-118. AAAI Press (1997)
3. Antonie, M.-L., Zaïane, O.R.: Text Document Categorization by Term Association. In: Proceedings of the 2002 IEEE International Conference on Data Mining, Maebashi City, Japan, December 2002, pp. 19-26. IEEE Computer Society (2002)
4. Church, K.W., Hanks, P.: Word Association Norms, Mutual Information, and Lexicography. In: Proceedings of the 27th Annual Meeting on Association for Computational Linguistics, Vancouver, BC, Canada, pp. 76-83. Association for Computational Linguistics (1989)
5. Coenen, F., Leng, P.: An Evaluation of Approaches to Classification Rule Selection. In: Proceedings of the 4th IEEE International Conference on Data Mining, Brighton, UK, November 2004, pp. 359-362. IEEE Computer Society (2004)
6. Coenen, F., Leng, P., Zhang, L.: Threshold Tuning for Improved Classification Association Rule Mining. In: Proceedings of the 9th Pacific-Asia Conference on Knowledge Discovery and Data Mining, Hanoi, Vietnam, May 2005, pp. 216-225. Springer-Verlag (2005)
7. Coenen, F., Leng, P.: The Effect of Threshold Values on Association Rule based Classification Accuracy. Journal of Data and Knowledge Engineering 60, 2, 345-360 (2007)
8. Coenen, F., Leng, P., Sanderson, R., Wang, Y.J.: Statistical Identification of Key Phrases for Text Classification. In: Proceedings of the 5th International Conference on Machine Learning and Data Mining, Leipzig, Germany, July 2007, pp. 838-853. Springer-Verlag (2007)
9. Cohen, W.W.: Fast Effective Rule Induction. In: Proceedings of the 12th International Conference on Machine Learning, Tahoe City, CA, USA, July 1995, pp. 115-123. Morgan Kaufmann Publishers (1995)
10. Deng, Z.-H., Tang, S.-W., Yang, D.-Q., Zhang, M., Wu, X.-B., Yang, M.: Two odds-radio-based Text Classification Algorithms. In: Proceedings of the Third International Conference on Web Information Systems Engineering workshop, Singapore, December 2002, pp. 223-231. IEEE Computer Society (2002)
11. Fano, R.M.: Transmission of Information — A Statistical Theory of Communication. The MIT Press (1961)
12. Fragoudis, D., Meretaskis, D., Likothanassis, S.: Best Terms: An Efficient Feature-selection Algorithm for Text Categorization. Knowledge and Information Systems 8, 1, 16-33 (2005)

13. Fuhr, N.: Models for Retrieval with Probabilistic Indexing. Information Processing and Management 25, 1, 55-72 (1989)
14. Fuhr, N., Buckley, C.: A Probabilistic Learning Approach for Document Indexing. ACM Transactions on Information System 9, 3, 223-248 (1991)
15. Hersh, W.R., Buckley, C., Leone, T.J., Hickman, D.H.: OHSUMED: An Interactive Retrieval Evaluation and New Large Test Collection for Research. In: Proceedings of the 17th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval, Dublin, Ireland, July 1994, pp. 192-201. ACM/Springer (1994)
16. Kobayashi, M., Aono, M.: Vector Space Models for Search and Cluster Mining. In: Berry, M.W. (ed.) Survey of Text Mining — Clustering, Classification, and Retrieval, pp. 103-122. Springer-Verlag New York, Inc. (2004)
17. Lang, K.: News Weeder: Learning to Filter Netnews. In: Proceedings of the Twelfth International Conference on Machine Learning, Tahoe City, CA, USA, July 1995, pp. 331-339. Morgan Kaufmann Publishers (1995)
18. Li, X., Liu, B.: Learning to Classify Texts using Positive and Unlabeled Data. In: Proceedings of the Eighteenth International Joint Conference on Artificial Intelligence, Acapulco, Mexico, August 2003, pp. 587-594. Morgan Kaufmann Publishers (2003)
19. Li, W., Han, J., Pei, J.: CMAR: Accurate and Efficient Classification based on Multiple Class-association Rules. In: Proceedings of the 2001 IEEE International Conference on Data Mining, San Jose, CA, USA, November-December 2001, pp. 369-376. IEEE Computer Society (2001)
20. Liu, B., Hsu, W., Ma, Y.: Integrating Classification and Association Rule Mining. In: Proceedings of the Fourth International Conference on Knowledge Discovery and Data Mining, New York, NY, USA, August 1998, pp. 80-86. AAAI Press (1998)
21. Maron, M.E.: Automatic Indexing: An Experimental Inquiry. Journal of the ACM 8, 3, 404-417 (1961)
22. Moschitti, A., Basili, R.: Complex Linguistic Features for Text Classification: A Comprehensive Study. In: Proceedings of the 26th European Conference on IR Research, Sunderland, UK, April 2004, pp. 181-196. Springer-Verlag (2004)
23. Quinlan, J.R.: C4.5: Programs for Machine Learning. Morgan Kaufmann Publishers, San Francisco, CA, USA (1993)
24. Salton, G., Buckley, C.: Term-weighting Approaches in Automatic Text Retrieval. Information Processing & Management 24, 5, 513-523 (1988)
25. Salton, G., Wong, A., Yang, C.S.: A Vector Space Model for Automatic Indexing. Information Retrieval and Language Processing 18, 11, 613-620 (1975)
26. Scheffer, T., Wrobel, S.: Text Classification Beyond the Bag-of-words Representation. In: Proceedings of the Workshop on Text Learning, held at the Nineteenth International Conference on Machine Learning, Sydney, Australia (2002)
27. Sebastiani, F.: Machine Learning in Automated Text Categorization. ACM Computing Surveys 34, 1, 1-47 (2002)
28. Shidara, Y., Nakamura, A., Kudo, M.: CCIC: Consistent Common Itemsets Classifier. In: Proceedings of the 5th International Conference on Machine Learning and Data Mining, Leipzig, Germany, July 2007, pp. 490-498. Springer-Verlag (2007)
29. Wang, Y.J., Coenen, F., Leng, P., Sanderson, R.: Text Classification using Language-independent Pre-processing. In: Proceedings of the Twenty-sixth SGAI International Conference on Innovative Techniques and Applications of Artificial Intelligence, Peterhouse College, Cambridge, UK, December 2006, pp. 413-417. Springer-Verlag (2006)
30. Wang, Y.J., Sanderson, R., Coenen, F., Leng, P.: Document-base Extraction for Single-label Text Classification. In: Proceedings of the 10th International Conference on Data Warehousing and Knowledge Discovery, Turin, Italy, September 2008, pp. 357-367. Springer-Verlag (2008)
31. Wiener, E., Pedersen, J.O., Weigend, A.S.: A Neural Network Approach to Topic Spotting. In: Proceedings of the 4th Annual Symposium on Document Analysis and Information Retrieval, Las Vegas, NV, USA, April 1995, pp. 317-332 (1995)
32. Yin, X., Han, J.: CPAR: Classification based on Predictive Association Rules. In: Proceedings of the Third SIAM International Conference on Data Mining, San Francisco, CA, USA, May 2003, pp. 331-335. SIAM (2003)
33. Yoon, Y., Lee, G.G.: Practical Application of Associative Classifier for Document Classification. In: Proceedings of the Second Asia Information Retrieval Symposium, Jeju Island, Korea, October 2005, pp. 467-478. Springer-Verlag (2005)
34. Zaïane, O.R., Antonie, M.-L.: Classifying Text Documents by Associating Terms with Text Categories. In: Proceedings of the 13th Australasian Database Conference, Melbourne, Victoria, Australia, January-February 2002, pp. 215-222. CRPIT 5 Australian Computer Society (2002)