Generating Features from Textual Documents Through Association Rules

Rafael Geraldeli Rossi, Solange Oliveira Rezende

[1]Instituto de Ciências Matemáticas e de Computação - Universidade de São Paulo
São Carlos – SP – Brazil
{ragero,solange}@icmc.usp.br

Abstract. *The Text Mining techniques are used to organize, manage and extract knowledge from the huge amount of textual data available in digital format. In order to use these techniques, the textual documents need to be represented in an appropriate format. The common way to represent text collections is by using the bag-of-words approach, in which each document is represented by a vector. Each word in the document collection represents a dimension of the vector. This approach has well known problems as the high dimensionality, and sparsity of data. Besides, most of the concepts are described by a set of words, such as "text mining", "association rules", and "machine learning". The approaches, which generate features compounded by a set of words to solve this problem, suffer from other problems, such as the generation of features without meaning, and the need to analyze the high dimensionality of the bag-of-words in order to generate the features. An approach named bag-of-related-words is proposed to generate features compounded by a set of related words that avoids the problems as mentioned above. The features are generated from each textual document of a collection through association rules. Experiments were carried out using classification algorithms with different paradigms in order to evaluate the generated features. The obtained results demonstrated that the proposed approach is similar to the bag-of-words with much lower dimensionality and features which are easy to understand.*

1. Introduction

The information available in digital format at the world wide web has been increasing incessantly. Part of the data in the digital universe is in text format, such as e-mails, reports, papers, and web-pages contents. Due to the huge amount of information in textual format, Text Mining techniques are fundamental to manage and extract knowledge. The textual documents need to be represented in an appropriate way in order to use these techniques.

The Vector Space Model (VSM) [Salton 1989] is generally used to represent textual documents. In this model, each document is represented by a vector, and each position of this vector corresponds to features (dimensions) of a textual document collection. The common approach based on the VSM is the bag-of-words representation. In this representation, each word in a collection of documents becomes a dimension (feature) in the vector space. However, this approach presents some problems as the high dimensionality, a high sparsity data, and the relative positions of the words is lost.

One of the major problem presented by the bag-of-words is that concepts are usually represented by a set of words, as "data mining", "neural networks", "association rules", and so on. These concepts can not be found in the bag-of-words, and they can help in the learning from textual document collection, as presented in [Liu and Hu 2007]. For example, single words as "learning", or "mining", can represent documents about machine learning or represent documents about teaching, or mineral extraction respectively.

To obtain set of words as features, representations were developed based on phrases [Carvalho et al. 2010, Carvalho and Cohen 2006, Fürnkranz 1998, Mladenic and Grobelnik 1998, Fagan 1989], and based on set of n words as features [Zhang and Zhu 2007, Tesar et al. 2006, Bekkerman and Allan 2004, Yang et al. 2003, Ahonen-Myka et al. 1999]. These types of representations usually add the features to the bag-of-words approach, which increases the already high dimensionality. Most of the researches analyze the entire collection to generate the features, which can have a high computational cost, due the dimensionality and features without meaning. Normally it is necessary labeled textual document collections to perform a process of supervised feature selection.

This paper proposes an approach that generates related words and uses them as features. The proposed approach, named *bag-of-related-words*, generates features from each document of a collection through the association rules to discover relations among items in a dataset. Another intention of using association rules is to reduce the dimensionality of the bag-of-words. This is possible because not all the words of a document are used as features, but only the words that occur with a frequency above a certain threshold and the words that cooccur in the document.

A mapping of the document into transactions is necessary to enable the generation of association rules from each document. This can be done by considering the sentences, paragraphs, or a sliding window as transactions. This way of mapping the document into transactions can produce more understandable features for the user, since the words must be related in specific contexts of the documents. The related words generated by our approach enable the pattern extraction with better results, and can be useful in situations as text categorization, text clustering, cluster labeling, and document identification.

After the feature generation, a representation is created based on the VSM. In this paper we analyze if the features generated by the proposed approach are feasible to be used as document descriptors or cluster labels according to their predictive power. The representations will be submitted to text categorization algorithms for evaluation and comparison with the bag-of-words.

This paper is organized as follows. Section 2 presents the related works about representation using features compounded by more than one word. Section 3 presents the bag-of-related-words. Section 4 shows the results obtained with the proposed approach and the bag-of-words for the categorization of textual documents. Finally, Section 5 presents the conclusions and future works.

2. Related Works

There are several works that extract set of words to use them as features. Usually this can be done using *n*-grams (or statistical phrases) or using set of words. The *n*-grams are sequences of words in the text collection.

In [Mladenic and Grobelnik 1998] the bag-of-words is increased with *n*-gram features. The *n*-grams are obtained by *n* passes over the textual document collection, where the *i*-grams are generated based on the (*i* - 1)-grams from the previous pass. After this step, a process of feature selection to detect the best features is performed. The results demonstrated that features with n close to 3 are better than only single words.

In [Tan et al. 2002a], first the words with higher frequency are sorted in each document and only the most frequent words (U) are considered. Then, the words that appear in sequence with the words in U are used to generate the bigrams. After this a process of feature selection is carried out using the measures *tf-idf* and *Mutual Information*. The results were improved with the addition of the generated bigrams.

In [Carvalho et al. 2010], the automatic detection of *n*-grams is done and the impact of the addition of the n-grams in a system of information retrieval is investigated. The n in this work was set to 2. First, all the bigrams of the textual document collection are generated. Then a classification model using the Support Vector Machine method is built using a labeled collection of valid and invalid bigrams. For some domains, a domain specialist is necessary to create valid and invalid bigrams. The results using the *n*-grams improve the precision up to 50%.

Another way to use more than one word as features is by using a set of n words. The n words that compound the features can not appear together or close to each other in a textual document.

In [Yang et al. 2003] the "association features" is defined. This approach does not consider where the words occur in the documents. In order to generate the features, first the documents of a collection are represented by a bag-of-words, then a process of frequent itemset extraction is used to obtain the words that cooccur in the text document collection. The association features are added to the bag-of-words. The results demonstrated improvements using the Naïve Bayes classification algorithm.

In [Tesar et al. 2006], the use of bigrams and 2-itemsets is analyzed. 2-itemsets are sets compounded by two words. For the extraction of the 2-itemsets, a list of unique words of each textual document is created. Then, all the unique words are combined to form the 2-itemsets. As the number of 2-itemsets is huge, only the words with a frequency higher than a threshold are considered. Both the bigrams and the 2-itemsets are added to the bag-of-words representation. The results demonstrated better results with statistical significance using the Multinomial Naïve Bayes classification algorithm with the use of bigrams and 2-itemsets.

In the work of [Zhang and Zhu 2007], the "loose n-grams" were proposed. The loose n-grams are set of words that cooccur in a limited space, as sentences, or a sequence of words (window). The relative position of the words are not considered. Only the words that appear in a minimum number of documents are considered. The words that have a good discriminative power are separated. Then, sets of words compounded by words that cooccur in the collection are obtained, and these sets of words are filtered using the χ^2 method. The results demonstrated a little improvement in comparison to the bag-of-words.

Most of the related researches about the generation of features compounded by more than one word generated a number of features that is much higher than the bag-of-words approach and in most of the times a process of feature selection is necessary. However, due the high dimensionality of the collection, this process can be computationally extremely expensive, and sometimes impossible to be carried out. Besides, in unlabeled collections, which is very common, this process can be more difficult and might produce worse results.

Moreover, some approaches that analyze the entire collection may obtain features without meaning, as "*introduction_conclusion*" if we consider a document collection of scientific articles. Sometimes collections with few documents of some kind of vocabulary may not have their features compounded by more than one word. For example, suppose 5000 documents about chemistry and 20 about computer science. Suppose that some words of the documents about computer science cooccur only in 5 documents, the minimum cooccurrence must be 0.001. On the other hand a huge number of features about chemistry could be generated. Besides the cooccurrence analysis is executed in a bag-of-words representation, which can be computationally expensive.

The proposed approach treats all the problems as mentioned before i) it avoids the generation of a huge number of features, ii) it does not need labeled collections or collections with balanced topics, and iii) the features are more understandable.

3. Bag-of-related-words

The proposed approach was named *bag-of-related-words*. This approach generates features compounded by a set of related words from textual documents using association rules. An association rule is a rule of the type $A \Rightarrow B$, in which A and B are group of items, called itemsets, and $A \cap B = \emptyset$ [Agrawal and Srikant 1994]. The association rules extract the relations of the items in a dataset, in which $A \Rightarrow B$ means that when A (body) occurs, B (head) also tends to occur. Two classical measures to generate association rules are *support* and *confidence*. Support measures the joint probability of an itemset in a database, that is $sup(A \Rightarrow B) = n(A \cup B)/N$, in which $n(A \cup B)$ is the number of transactions that A and B occur together, and N is the total number of transactions. Confidence indicates the probability of A and B occur together given the occurrence of A, that is, $conf(A \Rightarrow B) = n(A \cup B)/n(A)$. Minimum values of support and confidence are defined to generate the rules.

The goal of our approach is to use related words that are repeated over the document in limited spaces as features. The four main steps to generate the features of the documents are:

1. Mapping the textual document into transactions;
2. Extracting association rules from the transactions;
3. Using the itemsets of the rules to compound the features;
4. Using the features to construct the *document-term* matrix;

The step of mapping of the textual document into transactions allows the extraction of association rules. In this step, Text Mining preprocessing techniques can be applied, as stopwords removal and stemming. The transactions can be obtained considering the sentences, paragraphs, or a sliding window as transactions. To illustrate this step, the text in Table 1 about Data Mining taken from Wikipedia[1] is considered.

[1] http://en.wikipedia.org/wiki/Data_mining (May 13, 2010).

Table 1. Example text.

Data mining is the process of extracting patterns from data. Data mining is becoming an increasingly important tool to transform this data into information. It is commonly used in a wide range of profiling practices, such as marketing, surveillance, fraud detection and scientific discovery.

Data mining can be used to uncover patterns in data but is often carried out only on samples of data. The mining process will be ineffective if the samples are not a good representation of the larger body of data. Data mining cannot discover patterns that may be present in the larger body of data if those patterns are not present in the sample being ”mined”. Inability to find patterns may become a cause for some disputes between customers and service providers. Therefore data mining is not foolproof but may be useful if sufficiently representative data samples are collected. The discovery of a particular pattern in a particular set of data does not necessarily mean that a pattern is found elsewhere in the larger data from which that sample was drawn. An important part of the process is the verification and validation of patterns on other samples of data.

After the preprocessing steps, the text from Table 1 is mapped into transactions. Table 2 presents the transactions considering the sentences.

Table 2. Transactions considering the sentences of the preprocessed text from Table 1.

1. data mine process extract pattern data
2. data mine increasingli import tool transform data inform
3. commonli wide rang profil practic market surveil fraud detect scientif discoveri
4. data mine uncov pattern data carri sampl data
5. mine process ineffect sampl good represent larger bodi data
6. data mine discov pattern present larger bodi data pattern present sampl mine
7. inabl find pattern disput custom servic provid
8. data mine foolproof suffici repres data sampl collect
9. discoveri pattern set data necessarili pattern found larger data sampl drawn
10. import part process verif valid pattern sampl data

The next step consists of extracting association rules from transactions. The rules extracted from the text of Table 2 considering a minimum support of 30% and a minimum confidence of 75% are presented on the Table 3. The values on the right of the rules are the support and the confidence respectively.

Table 3. Rules obtained considering the transactions from Table 2.

$\emptyset \Rightarrow data$ (80.0, 80.0)	data ⇒ mine (60.0, 75.0)	mine pattern ⇒ data (30.0, 100.0)
process ⇒ data (30.0, 100.0)	data ⇒ sampl (60.0, 75.0)	larger data ⇒ sampl (30.0, 100.0)
larger ⇒ sampl (30.0, 100.0)	sampl ⇒ data (60.0, 100.0)	pattern data ⇒ sampl (40.0, 80.0)
larger ⇒ data (30.0, 100.0)	mine ⇒ data (60.0, 100.0)	mine sampl ⇒ data (40.0, 100.0)
pattern ⇒ data (50.0, 83.3)	larger sampl ⇒ data (30.0, 100.0)	pattern sampl ⇒ data (40.0, 100.0)

The proposed approach also considers features compounded by single words. In this case, the items of the rules with empty set[2] like $data \Rightarrow \emptyset$ will be used as features compounded by single words. In this case the feature “$data$” would be generated.

The third step consists of using the items of the obtained association rules to compound the features of a document. For example, the rule “$data \Rightarrow mine$” will generate the feature “*data_mine*”. In order to avoid that rules with the same items generate different features, the items of the rule are sorted lexicographically or according to the order that they occur on the document. For example, “$association \Rightarrow rule$” and “$rule \Rightarrow association$” generate the feature “*association_rule*”.

In this step, interest measures besides support and confidence can be used to generate different relations among the items, to rank and prune the association rules. The intention to use interest measures is to obtain more understandable features

[2] The empty set are only present on the body of the rule

and reduce even more the dimensionality of the obtained representation. In this approach, the used measures were *Confidence*, *Lift*, *Yules'Q*, *Linear Correlation Coefficient*, *Mutual Information*, *Gini Index*, *Kappa*, and *J-Measure* [Guillet and Hamilton 2007, Geng and Hamilton 2006, Blanchard et al. 2005, Tan et al. 2002b]. The interest measures are compared to verify which measure produces better results for the textual document classification task.

The fourth step consists of using the generated features of each document to build a representation in the vector space model.

The bag-of-related-words avoids several problems as discussed in Section 2. For instance, the approach does not analyze the whole dimensionality of the textual document collection, which can be extremely expensive. The proposed approach analyzes each document individually, in which the number of words can be orders of magnitude smaller than all the words in a text document collection.

Another characteristic of the bag-of-related-words is that it does not need to set the value of n, since the rules will be generated until the items have values of support and some interest measure higher than the thresholds informed by the user. To aid the user with the minimum support setting, we proposed the following equation.

$$AutSup = \frac{GeneralMeanFreq}{NumberTrans} \tag{1}$$

where $GeneralMeanFreq$ means the sum of the frequency of all the words, and $NumberTrans$ means the number of transactions. This formula is used for each document of the collection to the minimum support setting.

The definition of the thresholds of other interest measures can be also set automatically. In this paper we used the mean ($\overline{x}$) of all the rules extracted of a document as a threshold. According to the thresholds chosen by the user during the process, the dimensionality can be much lower than the bag-of-words, and the application of feature selection methods is not necessary. However, as the features are represented in the vector space model, these methods can be applied as an additional task.

Therefore, the proposed approach is independent on the i) natural language processing, which is normally computationally expensive ii) the interference of domain specialists, and iii) knowledge base.

4. Experiments and Results

To evaluate the predictive power of the generated features by the proposed approach with the different interest measures and compare them with the bag-of-words, 4 classification algorithms with different paradigms were used: (i) *Naïve Bayes* (probabilistic learning); (ii) J48[3] (symbolic learning); (iii) *SMO* (statistical learning); and (iv) *KNN* (instance based learning). The Weka tool [Witten and Frank 2005] with standard options was used to execute the classification algorithms.

Experiments were carried out using 8 textual document collections of proceedings

[3] J48 is an open source Java implementation of the C4.5

from the ACM digital library[4] and the Reuters-21578[5] collection. Each ACM collection has 5 classes and approximately 90 documents per class. For this collection, the 10-fold-cross validation was used to evaluate the error classification rate.

For the Reuters-21578 collection, only the content of the tags *TITLE* e *BODY* was used. Moreover, the documents with single topic of the 15 more frequent classes were considered. The *LEWISSPLIT* division (*"TRAIN"* or *"TEST"*) was considered to build and to evaluate the classification model respectively. 2537 documents for training and 952 documents for test were used.

The bag-of-related-words was obtained following the steps presented in section 2. In the step referring to the conversion of a document into transactions, the stopwords were removed and the words were stemmed using the Porter's algorithm [Porter 1980]. A sliding window of size 5 to map the document into transactions was considered, since it obtained the best results [Rossi and Rezende 2010]. The value of minimum support was calculated from each document of the collection based on the proposed Equation 1.

The measures mentioned in Section 3 were used to prune the association rules. For each measure, a set of thresholds was used. The values of the thresholds were obtained empirically based on the number of features. Also the threshold based on mean ($\overline{x}$) was used. Table 4 presents the used thresholds. As the Lift ranges from 0 to ∞, the standardization proposed in [McNicholas et al. 2008] was used.

Table 4. Thresholds of the interest measures used in the experiments.

Objective Measure	**ACM**	**Reuters**
Confidence	0.25; 0.50; $\overline{x}$	0.25; 0.50; $\overline{x}$
Lift	0.10; 0.20; $\overline{x}$	0.10; 0.20; $\overline{x}$
Yule's Q	0.50; 0.75; $\overline{x}$	0.50; 0.75; $\overline{x}$
Correlation	0.25; 0.50; $\overline{x}$	0.25; 0.50; $\overline{x}$
Mutual Information	0.005; 0.01; $\overline{x}$	0.01; 0.05; $\overline{x}$
Gini Index	0.005; 0.01; $\overline{x}$	0.01; 0.04; $\overline{x}$
Kappa	0.15; 0.20; $\overline{x}$	0.30; 0.50; $\overline{x}$
J-Measure	0.01; 0.02; $\overline{x}$	0.02; 0.05; $\overline{x}$

In order to compare the results, the Pretext tool [Soares et al. 2008] was used to generate the bag-of-words representation. The metric used to compare the results obtained with the representations was the error classification rate.

We verified if there were significant statistically difference among the obtained results with the thresholds used for each interest measure. The Friedman test was applied considering the results of the 8 ACM collections for each interest measure. The results obtained by this test demonstrated that although the number of features vary significantly for each threshold, the differences in general were not statistically significant. For Naïve Bayes, there were differences with the thresholds for the Mutual Information and Gini Index measures, in which the threshold 0.01 was better than 0.005; and Correlation, in which the threshold based on mean was statistically better than the other thresholds. For the KNN, there were statistical difference among the thresholds of the Kappa measure, in which the threshold based on mean was better than the other thresholds.

[4] `http://sites.labic.icmc.usp.br:8088/ragero/Acm-Collection`

[5] `http://www.daviddlewis.com/resources/testcollections/reuters21578`

To compare the representations obtained by different measures and the bag-of-words, the representations with threshold based on mean were chosen, since in most of the situations, they did not present statistical differences in relation to the other thresholds. Again, the Friedman test was applied with the results of the 8 ACM collections. For the Naïve Bayes, Correlation measure presented better results than the measures Confidence and Lift with significant statistical difference. For SMO, only the representation using the Gini Index measure presented worse results than the bag-of-words with significant statistical difference.

Tables 5, 6, 7, and 8 present the results of the representations that obtained the smallest error classification rates of the ACM collection using the Naïve Bayes, J48, SMO, and KNN classification algorithm respectively[6]. The lowest classification errors for each collection are highlighted in these tables. It can be noticed that our approach obtained the smallest error rate in 5 of the 8 collection using the Naïve Bayes, 2 of the 8 using the J48, 1 of the 8 using the SMO, and 5 of the 8 using the KNN. It must be highlighted that some representations, as [J-Measure 0.02], [Mutual Information 0.2], [Confidence 0.50], [Correlation 0.5], obtained the best results for some ACM collection using the classification algorithms Naïve Bayes, J48, SMO, and KNN respectively, with a reduction of the feature number in approximately 55%, 38%, 80%, and 76% respectively.

Table 5. Naïve Bayes - results of the representations with lowest error classification rates for the ACM collection.

Representation	**ACM-1**	**ACM-2**	**ACM-3**	**ACM-4**	**ACM-5**	**ACM-6**	**ACM-7**	**ACM-8**
Bag-of-words	***5.51** ± 0.14	7.80 ± 0.10	7.55 ± 0.14	11.42 ± 0.08	***2.76** ± 0.09	7.29 ± 0.10	7.89 ± 0.10	***5.86** ± 0.14
Correlation $\overline{x}$	9.27 ± 0.19	***3.66** ± 0.12	***3.13** ± 0.11	1.78 ± 0.08	3.40 ± 0.12	3.66 ± 0.12	4.48 ± 0.13	6.46 ± 0.16
Mutual Inf. 0.005	9.27 ± 0.19	3.90 ± 0.12	3.37 ± 0.12	***1.52** ± 0.08	3.61 ± 0.12	3.66 ± 0.12	5.33 ± 0.14	6.87 ± 0.17
Kappa 0.15	9.77 ± 0.20	4.39 ± 0.13	3.85 ± 0.12	***1.52** ± 0.08	3.61 ± 0.12	3.89 ± 0.12	4.69 ± 0.13	7.27 ± 0.17
Kappa $\overline{x}$	9.52 ± 0.20	4.15 ± 0.13	3.85 ± 0.12	1.78 ± 0.08	3.61 ± 0.12	3.66 ± 0.12	***4.05** ± 0.13	7.47 ± 0.17
J-Measure 0.02	9.77 ± 0.20	4.15 ± 0.12	3.85 ± 0.12	2.28 ± 0.10	3.61 ± 0.12	***3.43** ± 0.12	5.76 ± 0.15	7.47 ± 0.17

Table 6. J48 - results of the representations with lowest error classification rates for the ACM collection.

Representation	**ACM-1**	**ACM-2**	**ACM-3**	**ACM-4**	**ACM-5**	**ACM-6**	**ACM-7**	**ACM-8**
Bag-of-words	***11.53** ± 0.22	***7.07** ± 0.19	12.97 ± 0.22	***6.09** ± 0.13	***14.01** ± 0.22	11.62 ± 0.20	***11.30** ± 0.20	***10.30** ± 0.20
Lift 0.1	14.79 ± 0.24	13.66 ± 0.23	11.54 ± 0.21	9.39 ± 0.19	18.47 ± 0.26	***8.01** ± 0.18	18.55 ± 0.26	16.97 ± 0.25
Lift 0.2	15.04 ± 0.24	13.17 ± 0.22	***9.86** ± 0.19	8.12 ± 0.18	17.41 ± 0.26	10.53 ± 0.2	18.55 ± 0.26	19.39 ± 0.27
Yule's Q 0.5	14.04 ± 0.23	13.41 ± 0.23	11.06 ± 0.2	10.15 ± 0.2	17.41 ± 0.26	***8.01** ± 0.17	18.98 ± 0.26	18.59 ± 0.26
Yule's Q $\overline{x}$	14.54 ± 0.24	13.90 ± 0.23	10.10 ± 0.19	9.39 ± 0.18	17.41 ± 0.26	***8.01** ± 0.17	18.55 ± 0.26	18.38 ± 0.26

Table 9 presents the results of the representation with thresholds based on mean and the bag-of-words for the Reuters collection. For Naïve Bayes all the representations obtained an error classification rate lower than the bag-of-words. The lowest error rate was obtained by the representation [Kappa $\overline{x}$], with an error of 24.05%, which was 3.89% lower than the obtained by the bag-of-words. For the J48, again all the representations obtained an error classification rate lower than the bag-of-words. The lowest error rate

[6] The complete results are presented in http://sites.labic.icmc.usp.br:8088/ragero/ENIA2011/docint.pdf

Table 7. SMO - results of the representations with lowest error classification rates for the ACM collection.

Representation	**ACM-1**	**ACM-2**	**ACM-3**	**ACM-4**	**ACM-5**	**ACM-6**	**ACM-7**	**ACM-8**
Bag-of-words	***3.26** ± 0.32	***2.20** ± 0.32	***3.54** ± 0.32	***0.25** ± 0.32	***2.12** ± 0.32	***1.14** ± 0.32	3.84 ± 0.32	***4.04** ± 0.32
Confidence 0.25	5.26 ± 0.32	3.41 ± 0.32	4.09 ± 0.32	1.52 ± 0.32	3.40 ± 0.32	2.29 ± 0.32	***2.77** ± 0.32	8.08 ± 0.32
Confidence 0.5	5.01 ± 0.32	4.39 ± 0.32	4.33 ± 0.32	1.52 ± 0.32	3.18 ± 0.32	2.06 ± 0.32	***2.77** ± 0.32	7.68 ± 0.32
Yule's Q $\overline{x}$	5.26 ± 0.32	3.41 ± 0.32	4.57 ± 0.32	1.78 ± 0.32	2.76 ± 0.32	2.29 ± 0.32	***2.77** ± 0.32	7.07 ± 0.32
Correlation 0.50	5.26 ± 0.32	3.90 ± 0.32	4.33 ± 0.32	1.52 ± 0.32	3.18 ± 0.32	2.29 ± 0.32	***2.77** ± 0.32	7.68 ± 0.32
Mutual Inf. 0.005	5.26 ± 0.32	3.41 ± 0.32	4.33 ± 0.32	1.78 ± 0.32	2.76 ± 0.32	2.97 ± 0.32	***2.77** ± 0.32	6.67 ± 0.32

Table 8. KNN - results of the representations with lowest error classification rates for the ACM collection.

Representation	**ACM-1**	**ACM-2**	**ACM-3**	**ACM-4**	**ACM-5**	**ACM-6**	**ACM-7**	**ACM-8**
Bag-of-words	10.02 ± 0.06	7.32 ± 0.04	4.96 ± 0.03	2.04 ± 0.02	5.74 ± 0.03	***1.82** ± 0.03	***6.18** ± 0.03	***4.44** ± 0.03
Yule's Q 0.5	11.01 ± 0.06	8.78 ± 0.07	6.48 ± 0.04	0.76 ± 0.01	***3.18** ± 0.02	3.66 ± 0.03	6.61 ± 0.02	8.47 ± 0.04
Correlation 0.5	***8.78** ± 0.04	10.49 ± 0.04	5.49 ± 0.04	1.27 ± 0.01	4.65 ± 0.04	4.13 ± 0.03	7.25 ± 0.02	8.69 ± 0.04
Correlation $\overline{x}$	9.26 ± 0.05	***6.83** ± 0.04	5.55 ± 0.04	1.01 ± 0.02	4.04 ± 0.03	2.98 ± 0.02	6.40 ± 0.03	8.07 ± 0.05
J-Measure $\overline{x}$	11.52 ± 0.04	8.78 ± 0.04	***4.81** ± 0.03	***0.50** ± 0.02	4.02 ± 0.03	4.13 ± 0.03	7.46 ± 0.03	9.08 ± 0.02

was obtained by the representations [Mutual Inf. 0.05], [Gini Index 0.01], e [J-Measure 0.02], being 28.89%, which is 2.83% lower than the bag-of-words. For the SMO, 11 of the 24 representations obtained an error classification rate lower than bag-of-words. The lowest error classification rate was obtained by the representation [Kappa 0.50] with an error of 22.90%, which is 0.42% smaller than the bag-of-words. For KNN, all the error rates obtained by the proposed representation were higher than the bag-of-words. The best result obtained by our approach was obtained by the representation [Confidence 0.50%] with an error rate of 17.58%, which is 2.56% higher than the bag-of-words.

Table 9. Results for the Reuters collection using the Naïve Bayes (N.B.), J48, SMO, and KNN classification algorithms.

Representation	**N.B.**	**J48**	**SMO**	**KNN**
Bag-of-words	27.94	31.72	23.32	***15.02**
Lift $\overline{x}$	24.90	30.04	23.21	18.41
Confidence $\overline{x}$	25.00	30.36	24.05	18.41
Yule's Q $\overline{x}$	24.79	***29.52**	23.53	18.49
Correlation $\overline{x}$	24.26	29.62	23.63	18.09
Mutual Inf. $\overline{x}$	24.90	30.04	23.21	18.72
Gini Index$\overline{x}$	24.79	30.57	23.21	18.09
Kappa $\overline{x}$	***24.05**	29.73	***23.00**	18.84
J-Measure $\overline{x}$	25.00	29.83	***23.00**	18.41

For the Reuters collection, several representations based on the proposed approach for the classification algorithms Naïve Bayes, J48, SMO, obtained better results than the bag-of-words with a reduction of the feature number up to 20%.

The results using the Naïve Bayes classification algorithm demonstrated that the proposed approach is appropriate, since the best results were obtained for most of the collections, and with a feature number smaller than the bag-of-words

We observed that our approach obtained error rates higher than the bag-of-words for the most of the ACM collection using the algorithm J48. We noticed that this is related to the features of the collection as conference places, author names, and so on. For instance, some nodes of the decision tree obtained using the bag-of-words and our approach for the collection ACM-8 are presented on the Table 10.

Table 10. Example of rules generated by the J48 algorithm using the bag-of-words representation and the proposed approach.

Bag-of-words	**Proposed Approach**
"hawaii > 0: Hypertext_Hypermedia"	"number_vertice > 0: Hypertext_Hypermedia"
"jaschk > 0: Hypertext_Hypermedia"	"system_time > 0: Embedded_Systems"
"piezzo > 0: Embedded_Systems"	"packet_process > 0: Embedded_Systems"
"rajwar > 0: Microarchtecture"	"architecture_ comput > 0: Microarchitecture"

It can be noticed that the rules generated by the proposed approach are more understandable and bring more knowledge than the obtained rules using the bag-of-words representation. In situations in which there are no authors cited in several documents of a class, or there are no places in some documents of a class, our approach can obtain better results than the bag-of-words. Besides the fact that the bag-of-words obtained the lowest error classification rates for most of ACM collection, there were not significant statistical differences between the bag-of-words and the bag-of-related-words.

The results using the classification method SMO demonstrated that the proposed representation is appropriate, since there were no significant statistical differences between the bag-of-related-words and the bag-of-words. Again, features as author's names, places and conferences names might have influenced the results obtained by the bag-of-words, since the support vector might use vectors with these features as support vectors. For the Reuters-21578 collection, most of the representations of the proposed approach obtained better results with a number of features smaller than the bag-of-words.

For KNN, the smallest error classification rates were obtained for most of the ACM collections. However unlike the other classification algorithms, the proposed approach did not obtain the smallest error classification rate for the Reuters-21578 collection. Other experiments involving similarity measures demonstrated that the proposed approach led to good results when used in collections with long texts, and the results with short texts are not as good as the obtained results by the bag-of-words.

5. Conclusions and Future Work

The proposed approach demonstrated that it is capable of generating features that can predict the classes as good as the features of the bag-of-words for different classification algorithms. The initial dimensionality of our approach, even containing features composed by n words, in most cases was much smaller than the bag-of-words with better results. Analyzing the rules obtained by a symbolic algorithm, we can notice that the features are really easy to understand.

Compared to the previous paper that used only frequent itemsets as features of the textual documents [Rossi and Rezende 2010], the use of interest measures reduced the number of features and in most cases led to better results. The different interest measures used to generate the features demonstrated to be equivalent in most cases. Only for the

Naïve Bayes classification algorithm the Correlation measure presented better results than the Confidence and Lift measures, and for the classification algorithm SMO, the Lift measure presented worse results than the bag-of-words.

Another aspect that we are currently investigating is related to the determination of the interest measure threshold. This setting is not trivial and can be time consuming for the user. An automatically threshold setting was proposed in this paper. The results obtained with the automatically set thresholds were equivalent or better than the manually threshold setting.

The analysis of each document individually, besides avoids the entire dimensionality of the collection, allow the proposed approach to be used in dynamic contexts, in which the analysis of the entire document collection to generate the features is unfeasible for each new document.

For future research, we are going to evaluate the quality of the clustering used to build a topic taxonomy and a subjective evaluation with users browsing in big text document collections represented by the bag-of-related-words. Also we are going to evaluate the ensemble of the results of different interestingness measures, and the clustering of the rules to reduce the number of features.

Acknowledgments

The authors are grateful to the financial support of CNPq.

References

Agrawal, R. and Srikant, R. (1994). Fast algorithms for mining association rules in large databases. In *VLDB'94: International Conference on Very Large Data Bases*, pages 487–499. Morgan Kaufmann Publishers Inc.

Ahonen-Myka, H., Heinonen, O., Klemettinen, M., and Verkamo, A. I. (1999). Finding co-occurring text phrases by combining sequence and frequent set discovery. In *IJCAI-99: Workshop on Text Mining: Foundations, Techniques and Applications*, pages 1–9.

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

Blanchard, J., Guillet, F., Gras, R., and Briand, H. (2005). Using information-theoretic measures to assess association rule interestingness. In *ICDM'05: Internation Conference on Data Mining*, pages 66–73.

Carvalho, A. L. C., Moura, E. S., and Calado, P. (2010). Using statistical features to find phrasal terms in text collections. *Journal of Information and Data Management*, 1(3):583–597.

Carvalho, V. R. and Cohen, W. W. (2006). Improving "email speech acts" analysis via n-gram selection. In *ACTS '09: Workshop on Analyzing Conversations in Text and Speech*, pages 35–41. Association for Computational Linguistics.

Fagan, J. (1989). The effectiveness of a nonsyntactic approach to automatic phrase indexing for document retrieval. *Journal of the American Society for Information Science*, 40(2):115–132.

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

Geng, L. and Hamilton, H. J. (2006). Interestingness measures for data mining: A survey. *ACM Computing Surveys*, 38(3):9.

Guillet, F. and Hamilton, H. J., editors (2007). *Quality Measures in Data Mining*, volume 43 of *Studies in Computational Intelligence*. Springer.

Liu, S. and Hu, H. (2007). Text classification using sentential frequent itemsets. *Journal of Computer Science and Technology*, 22(2):334–337.

McNicholas, P. D., Murphy, T. B., and O'Regan, M. (2008). Standardising the lift of an association rule. *Computational Statistics & Data Analysis*, 52(10):4712–4721.

Mladenic, D. and Grobelnik, M. (1998). Word sequences as features in text-learning. In *ERK'98: Electrotechnical and Computer Science Conference*, pages 145–148.

Porter, M. F. (1980). An algorithm for suffix stripping. *Readings in Information Retrieval*, 14(3):130–137.

Rossi, R. G. and Rezende, S. O. (2010). The use of frequent itemsets extracted from textual documents for the classification task. In *WTI 2010: International workshop on Web and Text Intelligence located on International Joint Conference (SBIA, SBRN, JRI)*, pages 1–10.

Salton, G. (1989). *Automatic text processing: the transformation, analysis, and retrieval of information by computer*. Addison-Wesley Longman Publishing Co., Inc.

Soares, M. V. B., Prati, R. C., and Monard, M. C. (2008). PRETEXT II: Descrição da reestruturação da ferramenta de pré-processamento de textos. Technical Report 333, ICMC-USP.

Tan, C.-M., Wang, Y.-F., and Lee, C.-D. (2002a). The use of bigrams to enhance text categorization. *Information Processing and Management*, 38(4):529–546.

Tan, P.-N., Kumar, V., and Srivastava, J. (2002b). Selecting the right interestingness measure for association patterns. In *ACM SIGKDD'2002: International Conferenceon Knowledge Discovery and Data Mining*, pages 32–41. ACM.

Tesar, R., Strnad, V., Jezek, K., and Poesio, M. (2006). Extending the single words-based document model: a comparison of bigrams and 2-itemsets. In *DocEng'06: ACM Symposium on Document Engineering*, pages 138–146.

Witten, I. H. and Frank, E. (2005). *Data Mining: Practical machine learning tools and techniques*. Morgan Kaufmann, 2 edition.

Yang, Z., Zhang, L., Yan, J., and Li, Z. (2003). Using association features to enhance the performance of naïve bayes text classifier. In *ICCIMA '03: International Conference on Computational Intelligence and Multimedia Applications*, page 336. IEEE Computer Society.

Zhang, X. and Zhu, X. (2007). A new type of feature - loose n-gram feature in text categorization. In *IbPRIA'07: Iberian Conference on Pattern Recognition and Image Analysis*, pages 378–385. Springer.