© International Journal of Combinatorial Optimization Problems and Informatics, Vol. 6, No. 1, Jan-April 2015, pp. 7-19. ISSN: 2007-1558.

Latent Dirichlet Allocation complement in the vector space model for Multi-Label Text Classification

Víctor Carrera-Trejo[1], Grigori Sidorov[1],
Sabino Miranda-Jiménez[2], Marco Moreno Ibarra[1]
and Rodrigo Cadena Martínez[3]
Centro de Investigación en Computación[1],
Instituto Politécnico Nacional, México DF, México
Centro de Investigación e Innovación en Tecnologías
de la Información y Comunicación (INFOTEC)[2], Ags., México
Universidad Tecnológica de México[3] – UNITEC MÉXICO
jvcarrera@ipn.mx, {sidorov,marcomoreno}@cic.ipn.mx,
sabino.miranda@infotec.com.mx
rocadmar@mail.unitec.mx

Abstract. In text classification task one of the main problems is to choose which features give the best results. Various features can be used like words, n-grams, syntactic n-grams of various types (POS tags, dependency relations, mixed, etc.), or a combinations of these features can be considered. Also, algorithms for dimensionality reduction of these sets of features can be applied, like Latent Dirichlet Allocation (LDA). In this paper, we consider multi-label text classification task and apply various feature sets. We consider a subset of multi-labeled files from the Reuters-21578 corpus. We use traditional tf-IDF values of the features and tried both considering and ignoring stop words. We also tried several combinations of features, like bigrams and unigrams. We also experimented with adding LDA results into Vector Space Models as new features. These last experiments obtained the best results.

Keywords: Multi-label text classification, Reuters-21578, Latent Dirichlet Allocation, tf-idf, Vector Space Model.

1 Introduction

Textual information can be classified into different themes, so it is important to identify features allowing organize this information on different topics. In this paper, we present a set of features focused on the task of automatic classification of multi-labelled documents.

In the automatic multi-labelled text classification, i.e. classification of documents given a set of themes (topics), can be assigned different labels or topics in a document according to its content. For example, a text with information about an oil well can be included in topics related to geographic, ecological, financial, political, etc. concepts.

The automatic classifiers are based on models of representation of a set or a corpus of documents, using a Vector Space Model (VSM). Using VSM means that we choose features of the documents and the values of these features and build a vector representation.

The most traditional features for text classification are words or n-grams that are present in the corpus with their values, tf-IDF weights (see below for more detailed description), are the most widely used model, however this representation has several problems [12]:

- High dimensionality of the vector space. The documents contain a lot of words or n-grams used as features, whose number determines a high dimensionality of the VSM.
- A small number of features can be considered as irrelevant.

Received April 23, 2015 / Accepted Jun 12, 2015

There are various ideas for dealing with these problems. For example, there are methods for reducing the dimensionality of Vector Space Models, for example, Latent Semantic Analysis or Latent Dirichlet Allocation (used in this paper). On the other hand, irrelevant features (for example, stop words) can be filtered using the idf measure or predefined lists of these words.

Recently, we proposed [30] to construct a vector space representation complementing it with other features, i.e., adding other features into the traditional VSMs. In a similar way, in [20] (the work presented at the same conference MICAI 2014, as our work [30]), it is also proposed to complement the traditional VSM (they considered only lexical features: words and stems) with LDA for various number of topics (100, 200, 300, 400, 500) and with semantic spaces (HAL, COALS). By semantic spaces they mean distributional Vector Space Models, i.e., words from the context are used as features for each word. The stems were important for the authors because they experimented with highly inflective Czech language. Probably, using very traditional lemmatization would be sufficient in that situation. In our case, we have also exploited various n-gram models and not only lexical features as in [0]. We also complemented our models with LDA. We conducted our experiments for the traditional corpus for text classification: a subset of the Reuters' collection (in English). Interestingly, in this case n-gram models obtain better results. It is the unexpected conclusion, because it is well-known that for text classification task using words as features is usually the preferred alternative.

So, this paper proposes a method that allows the construction of a Vector Space Model for multi-label text classification using LDA complement. We used the Reuters-21578 corpus for the multi-label classification using the ideas mentioned above: (1) The Vector Space Model is generated using traditional tf-idf values for words and/or lexical n-grams as features; (2) It is complemented with its LDA representation. LDA allows recognizing the set of most probable topics that correspond to each document.

This method was tested with the multi-labeling algorithm called Rakel1 [9] using the software called Meka [11]. Rakel1 uses a Naive Bayes classifier as the base classifier. We always use lemmas of words and consider both possibilities for treatment of stop words (prepositions, articles, conjunctions, etc.): including or excluding stop words [12, 13, 14]. As the baseline, we use the traditional tf-idf measure with words as features [14, 15, 16]. Our best result is obtained using a combination of words and lexical bigrams with the LDA complement.

The structure of this paper is as follows. Section 2 describes the main concepts used in the paper. Section 3 presents related works. In Section 4, we describe our method. In Section 5, we discuss the obtained results and compare them with the baseline. Finally, we give conclusions and ideas for the future work in Section 6.

2 Text Classification Task

This section presents general concepts that are necessary for understanding of the further discussion.

2.1 Multi-Label Text Classification

Classification of documents is important within the area of information management, since it allows knowing different contexts or topics that correspond to documents. It is considered that words and word combinations (n-grams) from documents represent different semantic relations with other words and phrases within a particular context [1] and in this manner define the text's meaning. Detection of semantic relations (in a very broad sense) is included in the field of text classification [2], which refers to sorting a set of documents into a set of categories (topics, classes or labels). There are situations when some documents are assigned to more than one topic, which is known as multi-labeling [2, 6, 9, 10, 18]. Text classification is based on techniques both from information retrieval and machine learning [2, 3, 4, 5].

2.2 TF, IDF and TF-IDF measures

The most common representation used in tasks of text classification [14, 15] is the Vector Space Model (VSM), in which each document is represented as a vector, whose components are the values of the VSM features. Each feature corresponds to a dimension in the vector space and the total size of the vector is given by the total number of features [8]. The features in VSM are usually words (or their morphological invariants: lemmas or stems). Another widely exploited possibility is to use instead of words n-grams or syntactic n-grams [19].

VSM is also the representation used in multi-labeling algorithms. In case of texts, each document is transformed into a set of words/n-grams/syntactic n-grams which correspond to its vector representation. In the VSM, each element w_i (word, etc.) is a feature for a document, whose value is determined by its frequency of appearance in the document. This frequency is called the term frequency (*tf*). Note that n-grams or syntactic n-grams also have *tf*.

Another value, which is measured with respect to the corpus, is called Inverse Document Frequency (*idf*). *Idf* characterizes how distinctive each feature is, i.e., if the feature is present in all documents, it cannot distinguish between them, so its weight should be low. The ideal situation for *idf* is when a feature is present exactly in one document, so it will allow finding this document immediately. It is common to use the combination of these two values, *tf* and *idf*, which is known as *tf-idf* measure, whose value is determined by equation 1:

$$tf\text{-}idf_{w,d} = tf_{w,d} \times idf_w, \quad (1)$$

where w refers to the word w_i in the document d, the term $tf_{w,d}$ is calculated using the equation 2:

$$tf_{w,d} = frequency(w, d), \quad (2)$$

where w refers to a word in the document d. Finally, the term idf_w is the value that determines the importance of a word with respect to the corpus. Its value is given by equation 3:

$$idf_w = log \frac{N}{df_w}, \quad (3)$$

where N represents the total of documents in the corpus and df_w is the number of documents that contain the term w_i at least once [17].

It is common practice (for example, Joachims [12, 13]) to avoid using stop words (prepositions, articles, etc.), because they carry no lexical meaning and are present equally in all documents. But both possibilities: using them or excluding them should be analyzed experimentally.

2.3 Latent Dirichlet Allocation (LDA)

In natural language processing, Latent Dirichlet Allocation (LDA) refers to a probabilistic generative model for a corpus of documents, when the documents are represented as a combination of the probabilities of belonging to each topic in a Vector Space Model. Each topic is characterized by a probabilistic distribution based on a set of words (or n-grams, or syntactic n-grams) from the corpus. The number of topics is a parameter needed to choose the number of partitions to divide the corpus, when the LDA model is created.

For its computation, the LDA model takes as input the *tf-idf* values from documents and the number of topics given by a user. After this, the generative model determines the probability of the membership of a document for each topic, generating new Vector Space Model of LDA topics, as shown in Figure 1.

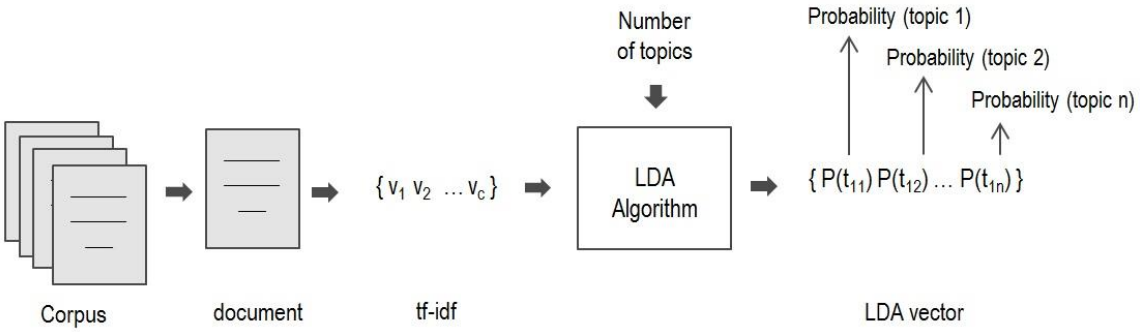

Fig 1. LDA model

The LDA algorithm contains the following steps.

- For each document:
  - Determine the number of possible features —words or n-grams—in the document using the Poisson distribution, i.e., only the features that correspond to this distribution are considered.
  - Determine the probability of the membership of the document in each topic using the Dirichlet distribution.
- In an iterative process, for each word w_i in the document, the algorithm chooses the topic t_n using multinomial distribution and conditional multinomial probability $P\left({}^{w_i}/_{t_n}\right)$ [7, 10].
- Finally, the algorithm returns a new set of features for each topic and the probability of each document to belong to each topic.

3 Related Work

In the last years multi-label document classification has received special interest focusing primarily on the search for automatic document classification methods. The purpose is to identify groups of labels that correspond to the documents and replace the currently used human-assisted methods. These new methods are based on various feature sets and different classification algorithms.

Usually, these methods use the Vector Space Model and Term Frequency-Inverse Document Frequency (*tf-IDF*) weighting to calculate the importance of the features. Various classification algorithms can be used, for example, Naive Bayes, decision trees, neural networks, support vector machines (SVM), etc. [13, 27].

Among many works developed in the multi-label context we would like to mention some, which are closer related to our paper.

As we already mentioned, in [20] it is presented a new kind of characterization based on unsupervised stemmer, latent Dirichlet allocation (LDA) and semantic spaces (HAL and COALS) using the corpus of the Czech News Agency (CTK), where a document can belong to several topics such as politics, sports, culture, business, and others. This work uses as baseline (1) *tf-idf* values calculated for words only, (2) *tf-idf* values for stems, which are important for Czech language, and (3) the pure LDA calculation [22]. The new proposal is based on the use of semantic spaces: HAL [23] and COALS [24]. Semantic spaces correspond to the Distributional Semantic Model, when words or stems are grouped into clusters based on their semantic distance. Further these clusters replace words or stems. The work compares different results generated using various feature sets: words, stems, combination of words and stems, words and each semantic space, words and LDA. The best results are obtained using the combination of words, stems, LDA and the semantic space COALS. MALLET [21] is used as the software for experiments.

Another work [10] proposes a new feature selection method called Labeled LDA, L-LDA. This paper refers to a probabilistic graphical model describing a process of generation of a set of labelled documents. This process is similar to the LDA process by regarding each document as a mixture of words belonging to different topics, except that the L-LDA process is supervised during the model creation. The L - LDA can be considered mainly as a model related to models of multinomial Naive Bayes classification.

In [26], it is shown that L-LDA is competitive. They use as the test corpus a Yahoo corpus with 4,000 documents and 20 labels. A set of binary classifiers based on support vector machines (SVM) is applied.

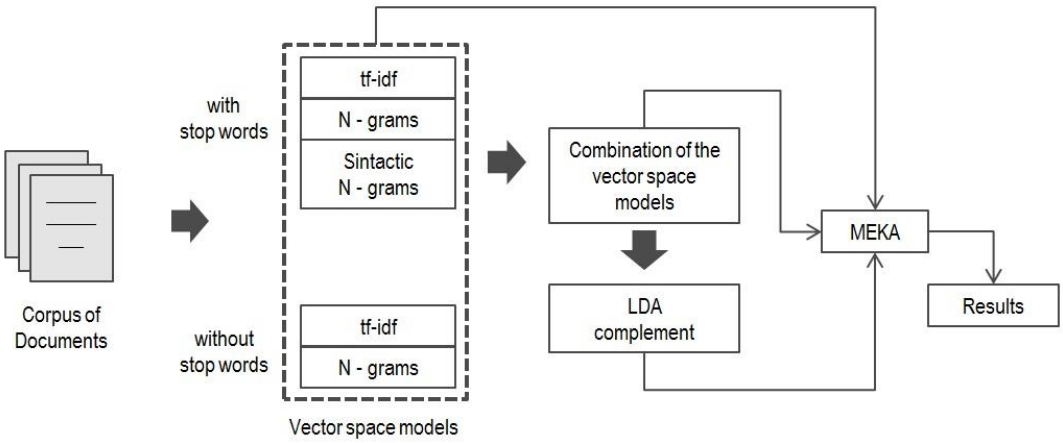

Fig 2. Proposed method

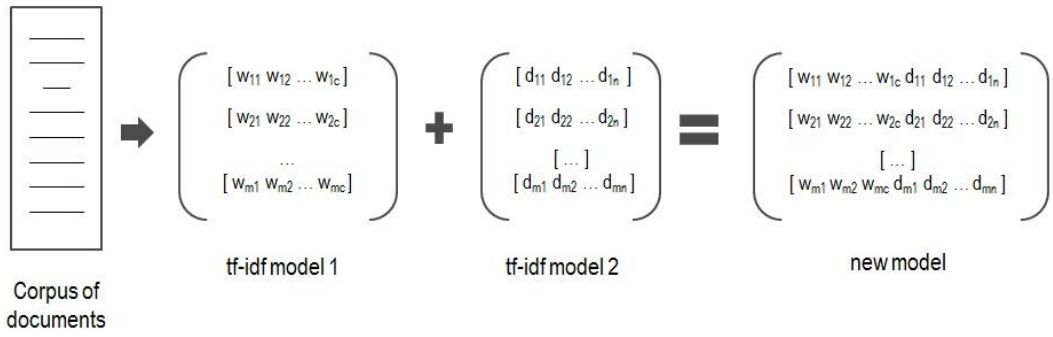

Fig 3. Combination of VSMs.

In [28], an algorithm called Rakel for multi-label classification is described (RAndom k-labELsets). It uses a method based on ensemble of classifiers LP (Label Powerset), in which each subset of tags is treated as a single label and from this assumption the classification is performed.

Note that the algorithms for multi-label classification can be divided into two different groups [9]: (i) problem transformation, and (ii) adaptation algorithms. The first type refers to independent algorithms, where the problem of multi-label classification is divided into a particular bi-class classification and they are resolved as particular problems. Some algorithms in this group are label ranking [26], regression, etc. The second group uses adaptation of different classification algorithms to handle multi-label data. We can mention between these algorithms decision trees, support vector machines, neural networks, etc.

4 Proposed Method

The method developed in this work combines the Vector Space Model with its corresponding LDA model, which is used as a complement to create a new Vector Space Model. Our hypothesis is that it can improve the classification results. For classification, we use the multi-labeling algorithm Rakel1 implemented in Meka [11].

The proposed method comprises four stages, which are listed below and shown in Figure 2.

- Construction of the traditional Vector Space Models (VSMs) using various feature sets,
- Considering various combinations of these Vector Space Models,
- Construction of the LDA complement and its inclusion into VSMs,
- Multi-label classification (with MEKA).

As the first step, we create various traditional Vector Space Models using different feature sets: words (including stop words or ignoring them), n-grams and syntactic n-grams. We use lemmas of words and *tf-idf* values for construction of these Vector Space Models.

Further we consider combinations of each VSM with other VSMs, for example, we combine VSMs based on bi-grams with syntactic bi-grams. The combination of VSMs is presented in Figure 3.

For each Vector Space Model we apply the LDA algorithm using various numbers of topics to obtain it corresponding LDA-VSM. After this, a new Vector Space Model is created using a traditional or combined Vector Space Model and its respective LDA model. The combination consists in joining each vector of the first model with the respective vector of the LDA model.

We consider that the LDA models are a good complement because if two documents are similar, i.e., they have the same thematic labels, then their corresponding LDA vectors should be similar too. Our experiments show that using LDA complement allows obtaining better results. Note that the LDA complements have relatively small size, i.e., few dimensions are added. Still, they improve the classification.

For multi-label classification, we apply the Rakel1 algorithm [9] implemented in Meka for each VSM generated at the previous steps.

5 Experimental Setup

We developed the corresponding software in Python 2.7 using *nltk* and *gensim* libraries. We also use the Meka 1.3 and the software Core NLP of the Stanford University as lemmatizer.

We constructed the corpus that is part of the corpus Reuters-21578. Namely, we took the documents that are labelled with more than one class for 4 topics.

Table 1. Test corpus distribution

Combinations of topics	Number of files
interest money-fx	112
wheat grain	102
corn grain	67
dlr money-fx	65

We use files in the corpus that were labeled with the following topics: *interest*, *money-fx*, *wheat*, *grain*, *corn* and *dlr*, considering 346 test files distributed in certain combinations as shown in Table 1. This is a subset of all topics in the corpus.

5.1 Construction of the vector space models

For each file in the corpus, we identify words, applying typical tokenization process, creating a set of words, lemmatizing them and calculating their frequencies. Further, we consider them as features and also construct from them n-grams and syntactic n-grams. For obtaining syntactic n-grams, we apply the Stanford parser first [31]. We also calculate *tf-idf* values. Let us remind that n-grams are linear combinations of elements in order as they appear in texts, while syntactic n-grams use the order of elements according to the paths in corresponding syntactic trees.
First of all, we construct two baseline VMSs: one of them considers all words in the corpus, denoted as unigrams-with_sw (where with_sw means “with stop words”), while the other one does not consider stop words (it is denoted as “unigrams”).

We also constructed two VSMs using bigrams as features. One of them considers all words, while the other one does not consider stop word: called bigrams-with_sw and bigrams. Also, the VSM that considers syntactic bigrams (s-bigrams) was constructed. This VSM contains stop words.

Table 2 shows different number of features in each new Vector Space Model.

Table 2. Number of features in the new vector models.

Vector Space Model	Features
unigrams	3,776
unigrams-with_sw	4,067
bigrams	18,534
bigrams-with_sw	23,667
s-bigrams	26,164

Table 3 contains an extract of the VSM with bigram, which corresponds to a *tf-idf* vector for a document, where the first number represents the feature number and the second number is its *tf-idf* numeric value. It does not include the features whose values are 0.

Table 3. Example of a vector in VSM with bigrams.

{ (70 2.2380), (288 2.2380), … , (16821 0.8488), (18274 2.2380) }

We also considered combinations of unigrams and unigrams-with_sw within other models as shown in Table 2. The numbers of features for the new models are shown in Table 4.

Table 4. Number of features in the combined Vector Space Models.

Vector Space Model	Features
unigrams + bigrams	22,310
unigrams-with_sw + bigrams	22,601
unigrams + bigrams-with_sw	27,443
unigrams-with_sw + bigrams-with_sw	27,734
unigrams + s-bigrams	29,940
unigrams-with_sw + s-bigrams	30,231

5.2 Construction of the LDA complement

Finally, for each Vector Space Model we generated new Vector Space Model complementing it with the results of the LDA algorithm. We used the values of 6, 50, 100, 500 and 1000 as the LDA parameter for the number of topics. In all cases the number of features of each LDA model directly corresponds to the number of topics, i.e., if the number of topics is 6 then the VSM is going to have only 6 additional features. In Table 5, we show an example of the LDA vector for the data from Table 3 using 6 topics. It is easy to see that the vector has non-zero value for the first topic only.

Table 5. Example of an LDA vector for data in Table 3.

{ Topic0 0.9934, Topic1 0.0, Topic2 0.0, Topic3 0.0, Topic4 0.0, Topic5 0.0 }

In Table 6 we present only the best cases after the classification using Meka. We used the parameters and the values shown in Table 8. The LDA complement is represented as *lda_n* where *n* corresponds to the number of topics. Table 6 contains the proposed VSMs and their number of features. Classification results are presented in the next section.

Table 6. Size of the Vector Space Models with the LDA complement.

Vector Space Model	Features
unigrams	3,776
unigrams + lda_6	3,780
unigrams + lda_50	3,826
unigrams + lda_100	3,876
unigrams + lda_500	4,276
unigrams + lda_1000	4,776
bigrams	18,534
bigrams + lda_6	18,540
bigrams + lda_50	18,584
bigrams + lda_100	18,634
bigrams + lda_500	19,034

Vector Space Model	Features
bigrams + lda_1000	19,534
unigrams + bigrams	22,310
unigrams + bigrams + lda_6	22,316
unigrams + bigrams + lda_50	22,360
unigrams + bigrams + lda_100	22,410
unigrams + bigrams + lda_500	22,810
unigrams + bigrams + lda_1000	23,310

For example, for the vector shown in Tables 3 and 5, the result with the LDA complement is presented in Table 7. The LDA complement is marked in boldface, the features with zero values are not shown.

Table 7. An example of the vector with the LDA complement

{ (70 2.2380), (288 2.2380), …, (16821 0.8488), (18274 2.2380), (18533, 0.0), (18534 0.9934) }

Each Vector Space Model was used in topic classification applying Meka with the parameters shown in Table 8.

Table 8. Parameters of Meka

Parameter	Value
Multi-labeling algorithm	Rakel1
Base classifier	Naïve Bayes
Validation method	k-Fold cross with k = 10

6 Experiments

In this section, we present experimental results obtained using the proposed method and several baseline methods.

First of all, we build several VSMs based on traditional features and use them as the most obvious baseline method for comparison. Table 9 shows the results of the multi-labeling classification (Rakel1) using these VSMs ordered from the best case to the worst.

Table 9. F1 measure for traditional VSMs without combinations.

Model	F1 measure
bigrams	0.8970
unigrams	0.8955
unigrams-with_sw	0.8935
bigrams-with_sw	0.8800
s-bigrams (syntactic)	0.8405

Traditionally, it is considered that the best results in thematic classification are obtained using unigrams (i.e., words). As it can be seen, the baseline F1 measure for unigrams is 0.8955. Interestingly, the best case for our corpus is the use of bigrams without stop words, because its F1 measure value is 0.8970, though the difference between them is only 0.0015.
In Figure 4, we can see a comparative graphic with different F1 measures for various cases from Section 5.1.

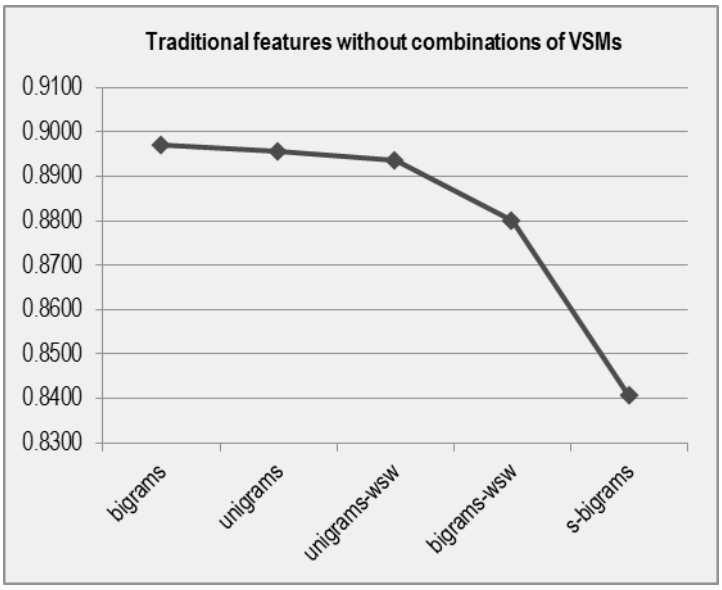

Fig 4. Graphic representation of the results of the VSMs with traditional features.

On the other hand, Table 10 shows the results for classification using only the LDA model for the two best cases from Table 9. It is obvious that if we want to classify using only the LDA model, then our classification will be unsuccessful. The best case using LDA models is for unigrams with 100 topics with F1 measure of 0.6065, which is much worse than the results in Table 9 (0.8970 using bigrams).

Table 10. F1 measures for LDA only models from Table 6 using the best features from Table 9.

LDA model	topics	F1 measure
unigrams	100	0.6065
unigrams	50	0.5940
unigrams	100	0.5260
unigrams	6	0.5125
bigrams	100	0.5025
unigrams	500	0.4435
bigrams	50	0.4405
bigrams	6	0.4265
bigrams	500	0.4055
bigrams	1000	0.3920

Previously, we proposed considering combinations of different features to construct a new set of VSMs and classify using them. In Table 4, we presented the new features for different VSMs, Table 11 shows the different F1 measures for each VSM after the classification process.

Table 11. F1 measure for VSMs with single features.

Model	F1 measure
unigrams + bigrams	0.9180
unigrams + bigrams-with_sw	0.9120
unigrams-with_sw + bigrams	0.9120
unigrams-with_sw + bigrams-with_sw	0.9035
unigrams + s-bigrams	0.9025
unigrams-with_sw + s-bigrams	0.8975

We can see a slight improvement of the results. The difference between our baseline case and the best combination is 0.0225. In Figure 5 we present the comparative graphic for the used combinations of features.

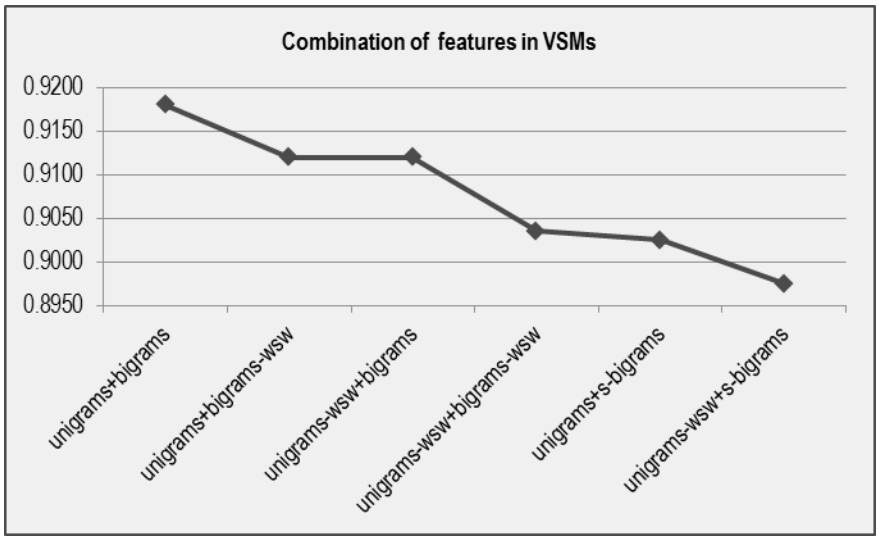

Fig 5. Graphic representation of the results of combinations of features in VSMs.

Now, we can also consider combining different LDA models for the best case from Table 11. In Table 12 we present F1 measures being the best case 0.5870 using 50 topics. This value is very low as compared with 0.9180, which is obtained using a combination of unigrams and bigrams (both without stop words).

Table 12. F1 measure for the LDA models.

LDA model	Topics	F1 measure
unigrams+bigrams	50	0.5870
unigrams+bigrams	100	0.4605
unigrams+bigrams	500	0.4525
unigrams+bigrams	6	0.4355
unigrams+bigrams	1,000	0.3750

Finally, we propose using different LDA models combining them with the other VSMs. In this case, we choose the best cases in the previous experiments and our baseline case. In Table 13, we present the best results for the new VSMs.

It can be seen that the results are improved. The best case has F1 measure of 0.9240 and the value for our baseline case is 0.8955 being the difference 0.0285. The best value of 0.8970 which was obtained using traditional features is improved too being the difference of 0.0270.

The best case using the LDA complement improves the best case using combinations of features, being the F1 measure for the combinations 0.9180: the difference of 0.0060.

Table 13. F1 measure for VSMs based on LDA complement.

Model	F1 measure
unigrams + bigrams + lda_100	0.9240
unigrams + bigrams + lda_6	0.9235
unigrams + lda_1000	0.9030
unigrams + lda_500	0.9010
bigrams + lda_6	0.8990

We can see that in some cases there is an improvement using the LDA complement, while in other cases there is no real improvement. All F1 measures using the LDA complement improve the F1 measures when we use only traditional features, but it is not so for the combinations of features. In Figure 6, we present the comparative graphic for the VSMs based on LDA complement.

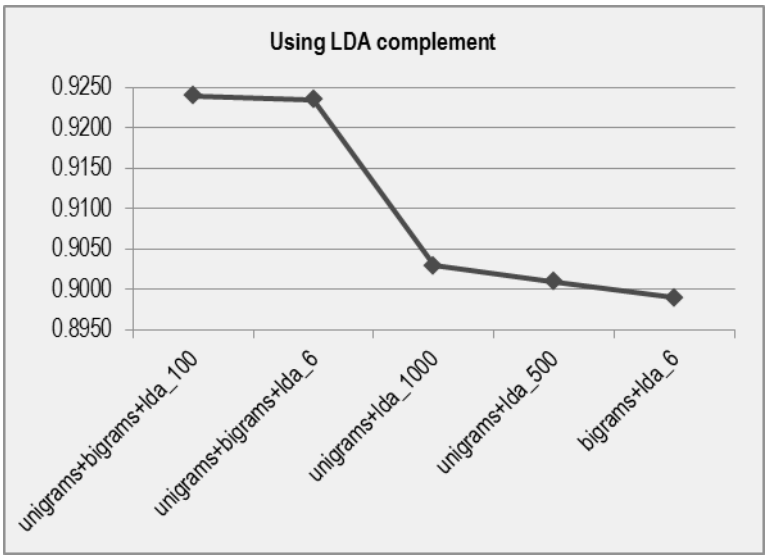

Fig 6. Graphic representation of the results of VSMs based on LDA complement.

Finally, in Table 14 we summarize the results for all experiments presented in this paper.

Table 14. Summary of all experimental values.

Model	F1 measure
unigrams + bigrams + lda_100	0.9240
unigrams + bigrams + lda_6	0.9235
unigrams + bigrams	0.9180
unigrams + bigrams-with_sw	0.9120
unigrams–with_sw + bigrams	0.9120
unigrams–with_sw + bigrams-with_sw	0.9035
unigrams + lda_1000	0.9030
unigrams + s-bigrams	0.9025
unigrams + lda_500	0.9010
bigrams + lda_6	0.8990
unigrams-with_sw + s-bigrams	0.8975
bigrams	0.8970
unigrams (baseline)	0.8955
unigrams-with_sw	0.8935
bigrams-with_sw	0.8800
s-bigrams	0.8405

Figure 7 shows a comparative graphic for all experiments described in the previous sections. For the sake of clarity of the graphic we do not show the values on X axis (they are present on other graphics).

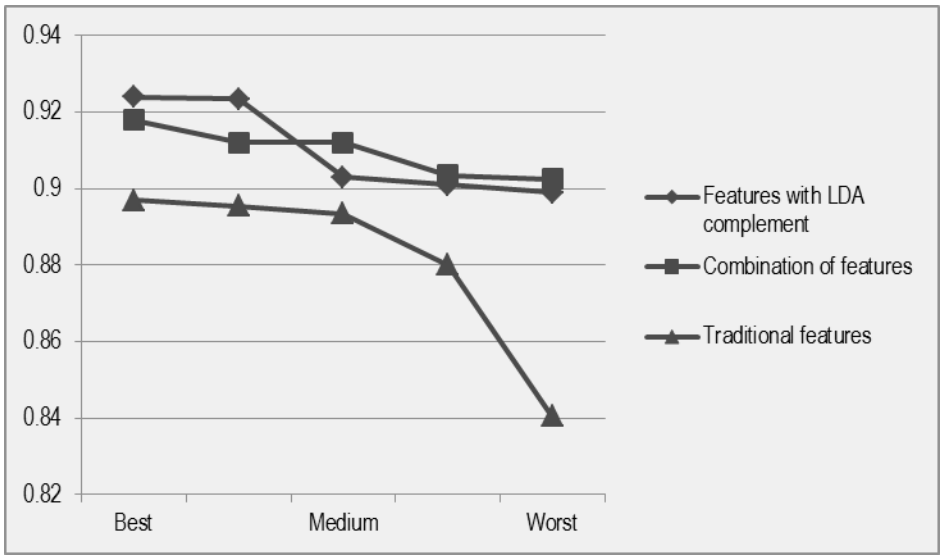

Fig 7. Schematic comparative graphic for all experiments.

7 Conclusions and Future Work

In this paper, we considered the task of multi-labelled text classification. Experiments were conducted when we applied combination of various features (like bigrams and unigrams). The obtained results show certain improvements over baseline methods. Besides, we proposed to use a VSM complement based on the LDA, adding the LDA results as features to Vector Space Models. Our experiments showed that it allows improving the classification results for multi-label thematic text classification, giving the best F1 measure.

It is worth mentioning that the LDA model was used very successfully in classification for exactly one topic per document [13], but in case of multi-label classification the results of the LDA alone are very poor.
In our case, we assumed that if two documents are labelled with the same topics by LDA, then these documents really belong to the same topics. This is the reason why the LDA complement works for improving the results.

As it is mentioned in [12, 13, 19, 20], it is important to consider particular features for each topic during classification. In this sense, Latent Dirichlet Allocation is the method that allows finding these features for each topic. It is important to mention that we used various empiric values for a number of topics for the LDA algorithm. As future work, we plan to define the optimal number of topics that allows obtaining better results. We also plan to explore other methods based on LDA, for example L-LDA [10], and to verify other methods for dimensionality reduction such as latent semantic analysis (LSA).

Finally, the plan to try the ideas based on the soft similarity [8] applied to multi label classification. The soft similarity proposes to weight similarity of pairs of features in VSMs. For example, we can consider the similarity of topics using WordNet similarity of words that belong to each pair of topics.

8 Acknowledgment

Work done under partial support of the Mexican government (CONACYT, SNI) and Instituto Politécnico Nacional, Mexico (projects SIP 20144274 and 20144534, COFAA, PIFI), FP7-PEOPLE-2010-IRSES: Web Information Quality-Evaluation Initiative (WIQ-EI) European Commission project 269180, and Cátedras CONACYT program.

9 References

1 Cilibrasi, R.L., Vitányi, M.B.P.: The Google Similarity Distance. IEEE Transactions on knowledge and data engineering, Vol. 19, No. 3, 2007. (doi:10.1109/TKDE.2007.48). http://doi.ieeecomputersociety.org/10.1109/TKDE.2007.48

2 Sebastiani, F.: Text Categorization. In: Alessandro Zanasi (ed.), Text Mining and its Applications, WIT Press, Southampton, UK, 2005.

3 Yang, H., Callan, J.: Near-Duplicate Detection by Instance-level Constrained Clustering. In Proc. of the 29th Conference on research and development in IR, 2006. (doi:10.1145/1148170.1148243).
4 Henzinger, M.: Finding Near-Duplicate Web Pages: a Large-Scale Evaluation of Algorithms. In: Proc. of the 29th Conference on research and development in IR, 2006. (doi:10.1145/1148170.1148222).
5 Stein, B., Meyer zu Eiben, S.: Near Similarity Search and Plagiarism Analysis. In: From Data and Information Analysis to Knowledge Engineering. Springer, 2006.
6 Zhang, M-L., Zhou, Z.-H.: A Review on Multi-label Learning Algorithms. IEEE Transactions on Knowledge and Data Engineering, 26(8), 2014. (doi:10.1109/TKDE.2013.39)
7 Blei, D.M., Ng, A.Y., Jordan, M.I.: Latent Dirichlet Allocation. Journal of Machine Learning Research 3, 2003.
8 Sidorov, G., Gelbukh, A., Gómez-Adorno, H., Pinto, D.: Soft Similarity and Soft Cosine Measure: Similarity of Features in Vector Space Model. Computación y Sistemas 18(3), 2014. (doi:10.13053/CyS-18-3-2043).
9 Tsoumakas, G., Katakis, I., Vlahavas, I.: Mining Multi-label Data. In: Data Mining and Knowledge Discovery Handbook, Springer, 2010.
10 Ramage, D., Hall, D., Nallapati, R., Manning, D.C.: Labeled LDA: A supervised topic model for credit attribution in Multi-labeled Corpora. In: Proceedings of the 2009 Conference on Empirical Methods in Natural Language Processing, Singapore, 2009.
11 Meka: A Multi-label Extension to Weka. http://meka.sourceforge.net/
12 Joachims, T.: Text Categorization with Support Vector Machines: Learning with Many Relevant Features. In: Proceedings 10th European Conference on Machine Learning, Springer Verlag, 1998.
13 Dumais, S., Platt, J., Heckerman, D., Sahami, M.: Inductive Learning Algorithms and Representations for Text Categorization. In: Proceedings of the 7th International Conference on Information and knowledge management, ACM Press, 1998.
14 Salton, G., McGill, M.: Introduction to Modern Information Retrieval. McGraw Hill, 1983.
15 Salton, G.: Automatic Text Processing: the Transform, Analysis, and Retrieval of Information by Computer. Addison-Wesley Longman Publishing Co. Inc., Boston, MA., USA, 1989.
16 Salton, G., Buckley, C.: Term weighting approaches in automatic text retrieval. Information Processing and Management, 24(5), 1988. (doi:10.1016/0306-4573(88)90021-0)
17 Schütze, H., Manning, D.C., Raghavan, P.: Introduction to Information Retrieval, Cambridge University Press, 2008. (doi:10.1017/CBO9780511809071). http://dx.doi.org/10.1017/CBO9780511809071
18 Tahir, M.A., Kittler, J., Mikolajczyk, K., Yan, F.: Improving Multilabel Classification Performance by Using Ensemble of Multi-label Classifiers, MCS 2010, Springer-Verlag, 2010. (doi:10.1007/978-3-642-12127-2_2)
19 Sidorov, G., Velasquez, F., Stamatatos, E., Gelbukh, A., Chanona-Hernández, L.: Syntactic N-grams as Machine Learning Features for Natural Language Processing, Expert Systems with Applications, 41(3), 2014. (doi:10.1016/j.eswa.2013.08.015).
20 Brychcín, T., Král, P.: Novel Unsupervised Features for Czech Multi-label Document Classification, 13th Mexican International Conference on Artificial Intelligence, MICAI, 2014. (doi:10.1007/978-3-319-13647-9_8).
21 Konkol, M.: Brainy: A machine learning library. In: Rutkowski, L., Korytkowski, M.. Scherer, R., Tadeusiewicz, R., Zadeh, L.A.., Zurada, J.M. (eds.) Artificial Intelligence and Soft Computing, Lecture Notes in Computer Science, vol. 8468. Springer Berlin Heidelberg, 2014.
22 McCallum, A.K.: Mallet: A machine learning for language toolkit (2002), http://mallet.cs.umass.edu
23 Lund, K., Burgess, C.,: Producing high-dimensional semantic spaces from lexical co-occurrence, Behavior Research Methods Instruments and Computers 28(2), 203-208, 1996. (doi:10.3758/BF03204766).
24 Rohde, D.L.T., Gonnerman, L.M., Plaut, D.C.: An improved method for deriving word meaning from lexical co-ocurrence, Cognitive Psychology 7, 573 – 605, 2004.
25 Tsoumakas, G., Katakis, I., Vlahavas, I.: Mining multi-label data, In Data Mining and Knowledge Discovery Handbook, 667 – 685, 2010.
26 Lewis, D.D., Yang, Y., Rose, T.G., Dietterich G., Li, F.: RCV1: A new benchmark collection for text categorization research, JMLR 5, 361- 397, 2004.
27 Della Pietra, S., Della Pietra, V., Lafferty, J.: Inducing features of random fields. IEEE Transactions on Pattern Analysis and Machine Intelligence 19(4), 380 – 397, 1997. (doi:10.1109/34.588021).
28 Tsoumakas, G., Vlahavas, I.: Random k-Labelsets: An Ensamble Method for Multilabel Classification, ECML '07 Proceedings of the 18th European conference on Machine Learning, Springer-Verlag Berlin, Heidelberg 2007. (doi:10.1007/978-3-540-74958-5_38).
29 Yang, Y., Pedersen, J: A Comparative Study on Feature Selection in Text Categorization, ICML '97 Proceedings of the Fourteenth International Conference on Machine Learning, Morgan Kaufmann Publishers Inc. San Francisco, CA, USA 1997.
30 Carrera-Trejo, V., Sidorov, G., Miranda-Jiménez, S., Moreno Ibarra, M., Cadena Martínez, R.: Using Soft Similarity in Multi-label Classification for Reuters-21578 Corpus, 13th Mexican International Conference on Artificial Intelligence, MICAI 2014.
31 The Stanford Parser: A statistical parser. http://nlp.stanford.edu/software/lex-parser.shtml