Wikipedia-based hybrid document representation for textual news classification

Marcos Antonio Mouriño-García[1*], Roberto Pérez-Rodríguez[1], Luis Anido-Rifón[1], Manuel Vilares-Ferro[2]

[1*]Department of Telematics Engineering, University of Vigo, Campus Lagoas-Marcosende, Vigo, Spain.
[2]Department of Computer Science, University of Vigo, Ourense, Spain.

*Corresponding author(s). E-mail(s): marcos@gist.uvigo.es;

Abstract

The sheer amount of news items that are published every day makes worth the task of automating their classification. The common approach consists in representing news items by the frequency of the words they contain and using supervised learning algorithms to train a classifier. This bag-of-words (BoW) approach is oblivious to three aspects of natural language: synonymy, polysemy, and multiword terms. More sophisticated representations based on concepts—or units of meaning— have been proposed, following the intuition that document representations that better capture the semantics of text will lead to higher performance in automatic classification tasks. The reality is that, when classifying news items, the BoW representation has proven to be really strong, with several studies reporting it to perform above different 'flavours' of bag of concepts (BoC). In this paper, we propose a hybrid classifier that enriches the traditional BoW representation with concepts extracted from text—leveraging Wikipedia as background knowledge for the semantic analysis of text (WikiBoC). We benchmarked the proposed classifier, comparing it with BoW and several BoC approaches: Latent Dirichlet Allocation (LDA), Explicit Semantic Analysis, and word embeddings (doc2vec). We used two corpora: the well-known Reuters-21578, composed of newswire items, and a new corpus created ex professo for this study: the Reuters-27000. Results show that (1) the performance of concept-based classifiers is very sensitive to the corpus used, being higher in the more "concept-friendly" Reuters-27000; (2) the Hybrid-WikiBoC approach proposed offers performance increases over BoW up to 4.12 and 49.35% when classifying Reuters-21578 and Reuters-27000 corpora, respectively; and (3) for average performance, the proposed Hybrid-WikiBoC outperforms all the other classifiers, achieving a performance increase of 15.56% over the best state-of-the-art approach (LDA) for the largest training sequence. Results indicate that concepts extracted with the help of Wikipedia add useful information that improves classification performance for news items.

1 Introduction

The information and communication society entails the existence of huge amounts of information distributed all across and along the Internet. Besides, the demand of information by users is growing day by day, which makes necessary and essential to automate the ordering of information (Roul et al. 2017). The automatic classification of text documents into predefined set of categories is a field that has a large number of applications and provides a solution to the problem presented above. Among these applications, we can include: the classification of books by theme, genre, or subject; sentiment analysis (Li et al. 2016); the

classification of online educational resources into their subject area or educational level (Moise et al. 2014); spam filtering (Arif et al. 2017); and the classification of textual news in its proper category (Li et al. 2016).

Automatic text classification can be modelled as a supervised machine learning problem (Sebastiani 2002). First, the classification algorithm is selected. There are many classification algorithms, being some of the most relevant in the state of the art: k-nearest neighbour, decision tree, neural networks, Bayes, random forests (RF), and support vector machines (SVM) (Khan et al. 2010). Next, the training sequence is selected—a set of examples whose category is known, which serves to train the classifier. Finally, the algorithm receives a test sequence—a set of documents whose category is unknown—so that it may predict the most appropriate category where to classify each document.

Natural Language Processing (NLP) techniques represent documents based on the features they contain, such as the structure of the document, or the frequency of words in the text (Settles 1994). Document classification makes use of NLP techniques, so that a classifier can predict the category which a document belongs to. The most commonly used representation is the bag-of-words (BoW) model, where a document is represented by a set of words and their frequency of occurrence in the text. The main drawback of the BoW model is that it does not tackle two common problems of natural language: redundancy (synonymy problem) and ambiguity (polysemy problem) (Wang et al. 2009; Huang et al. 2012; Ming and Chua 2015). Besides, a single-word-based representation is oblivious to the phenomenon of multiword terms.

In order to solve the problems presented above, several authors have proposed the bag-of-concepts (BoC) document representation, being a concept a "unit of meaning" (Wang et al. 2009; Stock 2010). By definition, concepts are not ambiguous, so that they eliminate the problems introduced by synonymy and polysemy. In accordance with this model, documents are represented by a set of concepts and their weights, which indicate their relevance in the document. Several previous works demonstrate that the BoC representation provides good results in text classification tasks (Sahlgren and Cöster 2004; Wang et al. 2009). The literature hosts several proposals for creating BoC representations of documents, and different ways to represent a concept, such as Latent Semantic Analysis (LSA) (Deerwester et al. 1990), Latent Dirichlet Allocation (LDA) (Blei et al. 2003), Explicit Semantic Analysis (ESA) (Gabrilovich and Markovitch 2009), word/document embeddings (WE/DE) (Bengio et al. 2003; Le and Mikolov 2014), and semantic annotators (Milne and Witten 2013).

The huge amount of existing information sources gen- erates immense lots of daily news, so it is necessary that these news can be organized or categorized into a finite set of categories, in such a way that it allows for an easy, quick, and efficient access to those that are of interest (Singh and Chhillar 2017). As previously stated, the most commonly used representation of text documents in classification tasks is the bag-of-words model. This model is not optimal, since it does not take into account the semantics of the words and the semantic relationships between them, causing the appearance of language problems such as redundancy an ambiguity, which negatively affects classification performance. These two above mentioned facts, and the good results offered by the combination of a concept-based representation of documents along with encyclopaedic knowledge to classify textual documents belonging to different areas of application (Sahlgren and Cöster 2004; Kim et al. 2005; Wang et al. 2009; Ni et al. 2011; Mouriño García et al. 2015; Mouriño-García et al. 2016a), lead us to the application of a bag-of-concepts representation of documents that leverages encyclopaedic knowledge, in particular the Wikipedia, to the creation of classifiers of textual news. Thus, this article describes the foundations and reports the evaluation results of a classifier of textual news that leverages the knowledge and semantic information from Wikipedia to represent text documents as bags of concepts. We call the proposed classifier WikiBoC. Furthermore, we also propose a hybrid model (Hybrid-WikiBoC) that combines the BoW and WikiBoC approaches, by enriching the bag-of- words representation of each document with concepts extracted from the document itself.

To evaluate the system, we conducted several experiments with the two approaches proposed and compared their performance in a benchmark

with four classifiers that use different state-of-the-art representations of documents: the BoW model, the ESA concept representation, the LDA model, and word/document embeddings. In order to carry out the experiments, we selected three of the most relevant algorithms in the state of the art—SVM, random forests, and naïve Bayes—and two corpora: the Reuters-21578 (Rose et al. 2002) corpus and a purpose-built corpus that comprises news of the Reuters agency, hereinafter called Reuters-27000 (Mouriño García et al. 2016b).

We consider that the main contributions of this work are the following: (1) the WikiBoC and Hybrid-WikiBoC approaches for classifying textual news; (2) the benchmarking of the state-of-the-art classification approaches; and (3) the Reuters-27000 corpus.

The remainder of this article is organized as follows. Section 2 conducts a brief review of the state of the art. Section 3 presents the *Wikipedia Miner* algorithm—which is the semantic annotator selected to create the concept-based representations of documents—the representations of documents we use, the selected classification algorithms, and the description of the corpora. Section 4 exposes the two approaches proposed: WikiBoC and Hybrid-WikiBoC. Section 5 describes the experiments conducted as well as the results obtained. Section 6 discusses and analyses the results gathered. Finally, 7 presents the main conclusions obtained.

2 Literature review

The literature hosts many studies about textual news classifi- cation, in which the bag-of-words model is the most popular approach to represent documents (Jadhav et al. 2016). Selamat et al. (2002) propose a news web page classification method that uses neural networks and principal component analysis to classify the Yahoo sports news database. Bekkerman et al. (2003) combine the distributional clustering of words and a support vector machines algorithm to classify news stories from the Reuters-21578 dataset. Van and Thanh (2017) propose the *keyword extraction with BoW* method in combination with neural network approaches to classify a corpus of Vietnamese news belonging to different topics. Singh and Chhillar (2017) use distinctive bag of words and artificial neural networks to classify a corpus of English news belonging to business, entertainment, politics, cricket, tootball, and technology categories. Kim and Kim (2016) propose a novel term-weighting scheme that can be induced from document probabilistic models such as naïve Bayes and the multinomial term model to classify textual news from Reuters-21578 dataset.

Besides, the literature holds several successful attempts to leverage concepts to represent textual news in classification tasks, which will be described in the following subsections.

2.1 Classifiers based on bag-of-concepts representations

This kind of classifiers is based on representing documents as vectors of concept weights, using, to that end, techniques such as Latent Dirichlet Allocation (LDA), Explicit Semantic Analysis (ESA), and word embeddings.

The LDA model (Blei et al. 2003) presupposes that each document within a collection comprises a small number of topics, each one of them 'generating' words. Thus, LDA finds topics in texts by 'going back' from the document and finding the set of topics that may have generated it. This approach has been leveraged by several authors to classify news stories. Colace et al. (2014) present a single-label classification method that uses LDA to represent news stories from Reuters-21578 dataset as structured vectors of features composed of weighted pairs of words, instead of using vectors of features composes of weighted words. Pavlinek and Podgorelec (2017) propose a semi-supervised method based on support vector machines and naïve Bayes algorithms, which leverages LDA topic models to classify textual news when there are little training data. Rodrigues et al. (2017) propose a supervised classification method that leverages LDA to take advantage of the inherent topical structure of documents and model their words as arising from a mixture of topics, to classify news belonging to Reuters-21578 dataset. ESA (Gabrilovich and Markovitch 2009) leverage external knowledge sources (such as Wikipedia or the Open Directory Project) to generate features from text documents. ESA analyses the text in documents and identifies topics that are explicitly present in the knowledge source used. The main drawback of this technique is its tendency to generate outliers—concepts that are related

to the text documents only marginally (Egozi et al. 2011)—which negatively affects classification performance. Gabrilovich and Markovitch (2009) evaluate the effectiveness of their method on text categorization, by classifying, among other, the news datasets Reuters-21578 and Reuters Corpus Volume 1 (RCV1). Chang et al. (2008) also use the ESA technique and Wikipedia to create a model of classification that does not need annotated training data to analyse both label and documents from a semantic point of view, thus allowing to learn classifiers.

Word embeddings are dense real-valued vectors, also known as distributed representations of words (Bengio et al. 2003; Mikolov et al. 2013). They have been recently proposed, serving as rich and coherent word representations. In order to learn document-level embeddings, Le and Mikolov (2014) propose *paragraph vector*, an unsupervised framework that learns continuous distributed vector representa- tions for variable-length pieces of texts. Yao et al. (2015) proposed a method to solve data sparsity problem of short text in news classification tasks by enriching document representation with word semantic similarity information obtained using word distributed representations. Jin et al. (2016) present a text classifier of news stories by using the naïve Bayes algorithm and a bag-of-embeddings approach which exploits contextual information from text classes. Mekala et al. (2016) cluster word embeddings to capture multiple semantic contexts in which words occur. Then, they are chained to form document topic vectors that can represent multitopic documents. The approach proposed is combined with linear SVM and logistic regression algorithms to classify textual news from Reuters-21578 corpus. Finally, word embeddings are also employed to perform cross-lingual classification of news stories (Mogadala and Rettinger 2016).

2.2 Classifiers based on hybrid word-concept representations

This kind of classifiers is based on representing documents as vectors resulting from the combination of weights of words and concepts. The literature hosts several studies that state that the use of a combination of word- and concept-based approaches improves the performance of textual news classification tasks. Cai and Hofmann (2003) employ LSA to extract concepts from documents and combine them with word representations to train and test an AdaBoost classifier. Although the authors report high *F1-score* values, the relative improvement of the hybrid approach over BoW is only about 2.71%. Besides, the experiments were conducted over a subset of the Reuters-21578 corpora, which simplifies classification tasks. Sahlgren and Cöster (2004) propose a combination of words and concepts extracted by Random Indexing to train and test an SVM algorithm. Again, the experiments were performed over a subset of the Reuters-21578 corpus, and although they report high *F1-score* values, the relative improvement over BoW is not very high (about 1.37%). Elberrichi et al. (2008) and Nezreg et al. (2014) leverage semantic information provided by WordNet to enrich word-based representations of documents. Despite conducting the experiments in small subsets of Reuters-21578, the performance values reported are lower than those offered by the previous proposals. Finally, Yousif et al. (2015) leverage semantic information provided by the Arabic version of WordNet to improve the performance of a naïve Bayes (NB) classifier. The authors report relative improvements over BoW about 3.16% when classifying a corpus of news extracted from the BBC Arabic website, composed of 5258 documents belonging to one of six possible categories.

The main differences of the approach we propose compared to the approaches presented in the state of the art are the following two. (1) The use of semantic information extracted from Wikipedia to enrich BoW representations of documents. Although WordNet-based approaches also use a collection of concepts, WordNet methods are limited to individual words (Mihalcea et al. 2006; Gabrilovich and Markovitch 2007), and it does not provide word sense disambiguation (Gabrilovich and Markovitch 2007). (2) The performing of the experiments using the entire corpora instead of using a reduced number of documents or target categories, thus providing more realistic results.

3 Materials and methods

This section presents: the semantic annotator that we leverage for extracting concepts from

text, *Wikipedia Miner*; the representation of documents; the classification algorithms selected; and the corpora used in the evaluation of the different approaches.

3.1 Semantic annotator: Wikipedia Miner algorithm

A semantic annotator is a software agent that is responsible for extracting the concepts that define a document, linking these concepts to entries from external sources such as Wikipedia. Semantic annotators also perform word sense disambiguation—thus tackling synonymy and polysemy—and they assign a weight to each extracted concept in accordance with their relevance in the text. In our work, we rely on Wikipedia Miner (Milne and Witten 2013), a semantic annotator that builds on natural language processing and machine learning techniques and uses Wikipedia as knowledge base. The functioning of the algorithm is based on three steps. Figure 1 shows graphically the process of obtaining the BoC representation of a text document—being each concept a Wikipedia article—from a text document.

- First step is candidate selection. Given a text document that comprises a set of n-grams (point 1 in Fig. 1)—being an n-gram a continuous sequence of n words—the algorithm queries a vocabulary that contains all the anchor texts of Wikipedia to check if any of the n-grams are present in the vocabulary (point 2). Thus, the more relevant candidates (n-grams) are those that are used most often as anchor texts in Wikipedia.
- The next step is disambiguation (point 3). Given the same vocabulary of anchor texts, the algorithm selects the most probable target for each of the candidates. This process is based on machine learning, using as training sequence Wikipedia articles, which contain good examples of disambiguation done manually. Disambiguation is performed based on two factors: the relationship with other unambiguous terms of the context, and how common is the relationship between an anchor text and the target Wikipedia article.
- The third and final step is link detection (point 4), which consists in measuring the relevance of each concept extracted from the text. To this end, machine learning techniques are used again, using as training sequence Wikipedia articles, since each of them is an example of what constitutes a relevant link and what does not.

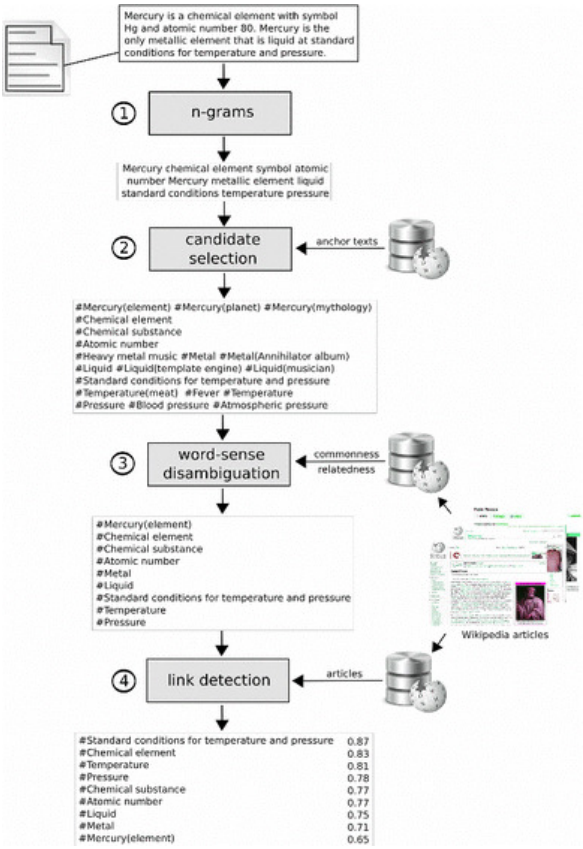

Fig. 1 Automatic extraction of concepts through *Wikipedia Miner* algorithm

3.2 Document representations

Document representations are based on the extraction of features of natural language from text. There exist different representations depending on which kind of features are extracted. In particular, we used the following three representations in our study. The first one is the bag-of-words model (Sahlgren and Cöster 2004), in which a document is represented as a vector $\bar{d} = (ww_1, ww_2, \ldots, ww_{|\mathbb{W}|})$ where ww_i are the weights—frequency of occurrence—of words in the document. The domain of features $\mathbb{W}$ is composed of the set of all words in the corpora,

ignoring 'stop words', and after applying the stemming algorithm of Porter (1980). The second one is the bag-of-concepts model (Sahlgren and Cöster 2004), in which a document is represented as a vector $\bar{d} = (cw_1, cw_2, \ldots, cw_{|\mathbb{C}|})$, where cw_i are the weights—relevance—of concepts in the document. In the case of the WikiBoC approach, we propose: (1) the domain of features $\mathbb{C}$ is composed of all the articles of the English edition of Wikipedia, and (2) in order to extract the features—concepts—we make use of the *Wikipedia Miner* algorithm previously described. Finally, the hybrid model consists in a combination of the previous two (Huang et al. 2009; Sahlgren and Cöster 2004). Thus, following the hybrid model, a document is represented as a vector $\bar{d} = (ww_1, \ldots, ww_{|\mathbb{W}|}, cw_1, cw_2, \ldots, cw_{|\mathbb{C}|})$.

3.3 Classification algorithms

3.3.1 Support vector machines

SVM is a supervised machine learning algorithm for performing—among others—regression, clustering, and classification tasks. SVM is one of most relevant state-of-the-art algorithms (Manimala et al. 2015), along with naïve Bayes, decision trees, k-nearest neighbour, and random forests. The basic idea consists in, given a set of elements each one belonging to one category, SVM algorithm builds a model that can predict whether a new element belongs to one category or another (Hearst et al. 1998). To carry out our work, we used the *scikit-learn* library, a Python module that provides a set of the most relevant machine learning algorithms in the state of the art (Pedregosa et al. 2011). In particular, in our work we use the *sklearn.svm.LinearSVC* class, with default settings, to implement the SVM classifier.

3.3.2 Random forests

Random forests (Breiman 2001) are a combination of tree predictors where each tree depends on the values of a random vector sampled independently with the same distribution for all trees in the forests. The basic principle is that a group of "weak learners" can come together to form a "strong learner". random forests are a interesting tool for making predictions considering they do not over fit because of the law of large numbers. The introduction of the right kind of randomness makes them accurate classifiers and regressors. To implement this algorithm, we used the *sklearn.ensemble.RandomForestClassifier* class of the *scikit-learn* library. We used the default settings provided by the *scikit-learn* library, except the number of trees, which was set to 10,000.

3.3.3 Naïve Bayes

Naïve Bayes (Lewis 1998) is one of the most simple, efficient, and effective algorithms for being used with data mining and machine learning purposes. Its great performance is surprising, due to its assumption of conditional independence which rarely happens in real-world applications (Zhang 2004). Given a set of documents to classify, and being those documents represented as features, the naïve Bayes classifier assumes ingenuously that those features are independent of one another. In order to evaluate our proposal, we opted for one of the classic variants of naïve Bayes used in document classification: multinomial naïve Bayes. Within *scikit-learn* library, this variant is identified with *sklearn.naive_bayes.MultinomialNB* class. We used the default settings provided by the *scikit-learn* library.

3.4 Corpora

This section presents the two corpora of news stories used to evaluate the approaches proposed.

3.4.1 Reuters-21578

Reuters-21578 (Rose et al. 2002) is a corpus that comprises 21,578 Reuters news classified into one or more of 60 categories available. After removing from the corpus those elements belonging to more than one category, the resulting corpus comprises 9496 documents, divided into a training sequence of 7597 documents and a test sequence that comprises 1899 documents.

3.4.2 Reuters-27000

Reuters-27000 (Mouriño García et al. 2016b) is a corpus that we expressly created for the evaluation of the proposal presented in this paper. To create the corpus, we first downloaded from Reuters website 27,000 random news articles—HTML web

pages— classified under each one of the following categories: health, art, politics, sports, science, technology, economy, and business. Next, we extracted from each article the title, the body, and the category which it belongs to. Finally, we stored in our database the title, the body, and the category of each article downloaded. As a result—after removing duplicates— we obtained a corpus that comprises 23,863 documents, which we randomly split into a training and testing sets of 14,356 and 9,507 documents, respectively.

4 Approach

This section exposes the two approaches we propose to perform classification of textual news: WikiBoC and the Hybrid-WikiBoC model.

4.1 WikiBoC classifier

The WikiBoC approach is based only on Wikipedia concepts (see Fig. 2 and Algorithm 1). During the training phase, first, it is necessary to obtain the WikiBoC representation of each document in the training set (point 1), by using the Wikipedia Miner semantic annotator (point 2) described in Sect. 3.1. After that, the WikiBoC representations of training documents (point 3) are input into the classifier in order to train it (point 4).

Algorithm 4.1 WikiBoC classifier

Input:
$\mathcal{D}$: *Training documents*
$\mathcal{C}$: *Documents to classify*
algorithm: Classification algorithm
procedure WIKIBOC($\mathcal{D}$, $\mathcal{C}$, *algorithm*)
  $i = 0$
  for each $d \in \mathcal{D}$ **do**
    $WikiBoC_d \leftarrow WikiBoC(d)$
    $i \leftarrow i + 1$
    $WikiBoC[i] \leftarrow WikiBoC_d$
  end for
  train(algorithm, WikiBoC)
  for each $c \in \mathcal{C}$ **do**
    $WikiBoC_c \leftarrow WikiBoC(c)$
    $predictedLabel \leftarrow classify(algorithm, WikiBoC_c)$
  end for
end procedure
function WIKIBOC(d)
  $WikiBoC_d \leftarrow WikipediaMinerConceptExtraction(d)$
  return $WikiBoC_d$
end function

During the classification phase, each document to classify (point 5) passes through the Wikipedia Miner semantic annotator (point 6) to obtain its WikiBoC representation (point 7). Finally, the WikiBoC representations are input into the classifier (point 8) in order to predict which category they belong to (point 9).

Algorithm 4.2 Hybrid-WikiBoC classifier

Input:
$\mathcal{D}$: *Training documents*
$\mathcal{C}$: *Documents to classify*
algorithm: Classification algorithm
1: **procedure HybridWikiBoC**($\mathcal{D}$, $\mathcal{C}$, *algorithm*)
2: $i = 0$
3: **for each** $d \in \mathcal{D}$ **do**
4: $BoW_d \leftarrow BoW(d)$
5: $WikiBoC_d \leftarrow WikiBoC(d)$
6: $Hybrid_d \leftarrow BoW_d + \text{WikiBoC}_d$
7: $i \leftarrow i + 1$
8: $Hybrid[i] \leftarrow Hybrid_d$
9: **end for**
10: *train(algorithm, Hybrid)*
11: **for each** $c \in \mathcal{C}$ **do**
12: $BoW_c \leftarrow BoW(c)$
13: $WikiBoC_c \leftarrow WikiBoC(c)$
14: $Hybrid_c \leftarrow BoW_c + \text{WikiBoC}_c$
15: $predictedLabel \leftarrow classify(algorithm, Hybrid_c)$
16: **end for**
17: **end procedure**
18: **function BoW**(d)
19: $d_{\,fileteredStopWords} \leftarrow FilterStopWords(d)$
20: $d_{stemmed} \leftarrow PorterStemmer(d_{filteredStopWords})$
21: $\text{BoW}_d \leftarrow CalculateTF(d_{stemmed})$
22: **return** BoW_d
23: **end function**
24: **function WikiBoC**(d)
25: $WikiBoC_d \leftarrow WikipediaMinerConceptExtraction(d)$
26: **return** $WikiBoC_d$
27: **end function**

4.2 Hybrid-WikiBoC classifier

In order to leverage the benefits of the traditional BoW representation along with the benefits of the WikiBoC representation, we proposed a combination of both approaches. The implementation of the Hybrid-WikiBoC classifier consists in enriching the BoW representation of each document with the concepts extracted from text by the Wikipedia Miner algorithm. Figure 3 and Algorithm 2 show the complete process.

During the training phase, first it is necessary to obtain the BoW representation of training documents (point 1). In order to create the BoW representation of documents, we first filtered stop words, then we applied the Porter stemmer (Porter 1980), and finally, we calculated the frequency of occurrence of stemmed words. At the same time, the WikiBoC representations of training documents are obtained by using the Wikipedia Miner semantic annotator (point 2). After that, BoW and WikiBoC representations are combined (point 3) and used to train the classification algorithm.

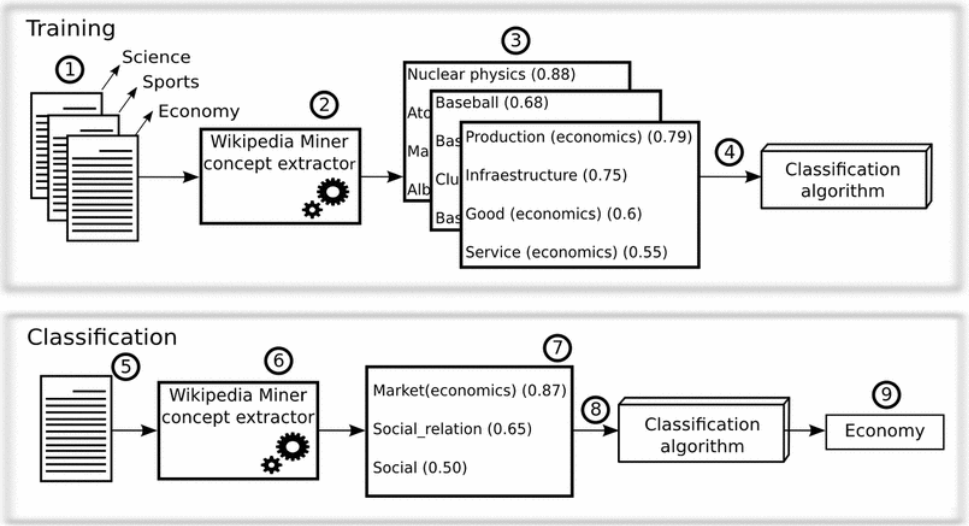

Fig. 2 Architecture of the WikiBoC classifier proposed

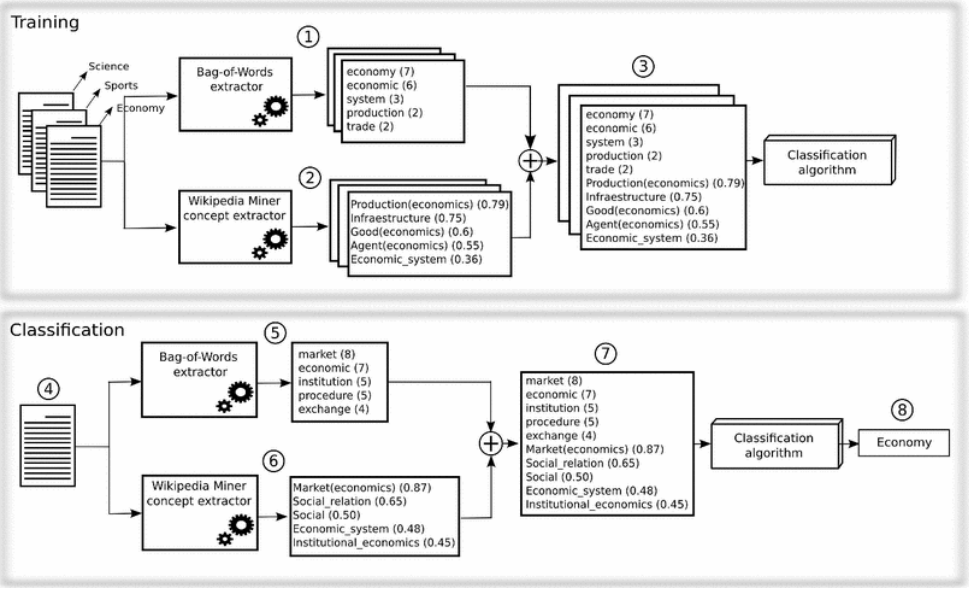

Fig. 3 Architecture of the Hybrid-WikiBoC classifier proposed

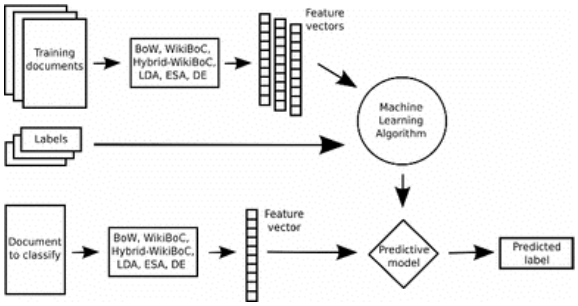

Fig. 4 Supervised machine learning classification algorithm

The process is similar during the classification phase. For each document to classify (point 4), it is necessary to obtain its BoW (point 5) and WikiBoC (point 6) representations. After that, both representations are combined (point 7) and input into the trained classifier in order to predict which category it belongs to (point 8).

5 Experiments and results

In this section, we present the experiments conducted to verify the performance of the proposed approach, as well as the results obtained.

5.1 Experimental settings

The experiments performed consisted in the classification of the two corpora of news previously presented using the three classification algorithms selected—linear SVM, random forests, and naïve Bayes—using the two approaches proposed: the WikiBoC and the Hybrid-WikiBoC model. In order to compare in a fair way the results obtained, we conducted the same experiments with four classifiers using different state-of-the-art representations of documents: the BoW model, the ESA concept representation, the LDA model, and document embeddings. Figure 5 shows the flowchart of each approach.

The use of the three classification algorithms employed in this work is similar (see Fig. 4). During the training phase, first, documents are represented as weighted vectors of features, being these features obtained from documents using the different representation models employed in this work: Bag of Words (BoW), Wikipedia Bag of Concepts (WikiBoC), Hybrid-WikiBoC, Explicit Semantic Analysis (ESA), Latent Dirichlet Allocation (LDA), and document embeddings (DE). After that, the documents represented as vectors of features, and the label/category/class of each document is provided as inputs to the classification algorithm in order to train or to fit it.

After being fitted, the model can then be used to predict new values. To do this, it is first necessary to create the weighted vector of features of the document to be classified, using again the different representation models employed in this work: BoW, WikiBoC, Hybrid-WikiBoC, ESA, LDA, and DE. Finally, the vector of features of the document to be classified is provided to the previously fitted classification algorithm, and it will predict the most appropriate label/class/category to which it belongs to, on the basis of what has been learned during the training step.[1]

The experiments were performed on subsets of the corpora. We randomly selected training sequences of length 5, 10, 20, 50, 100, 200, 500, 1000, 2000 and 5000 elements, and as test sequences we selected 1899 and 1600 random elements for Reuters-21578 and Reuters-27000, respectively. The set of training sequences lengths has been selected according to several woks in the state of the art about text mining and text classification (Nigam et al. 2000; Wenliang et al. 2004; Kozielski et al. 2015; Jiang and Cao 2016). It should be noted that the largest training sequence is composed of 5000 documents due to computational limitations. We consider that the set of training sequences selected is sufficient since it allows to analyse the performance of the approaches proposed with different amounts of training data. The set of training sequences lengths selected allows to evaluate the performance of the approach proposed when there are little training data (short training sequences) and when there are a lot of training data (large training sequences).

In order to implement the classifier using the BoW model, it was necessary to obtain the BoW representation of documents. Then, for each document in the corpora, we first filtered the stop words, then we applied the Porter stemmer, and finally, we calculated the frequency of occurrence of stemmed words. After that, documents represented as BoW were used to train and test the three classification algorithms selected.

[1] http://scikit-learn.org/stable/ holds documentation and usage examples of SVM, RF, and NB classification algorithms.

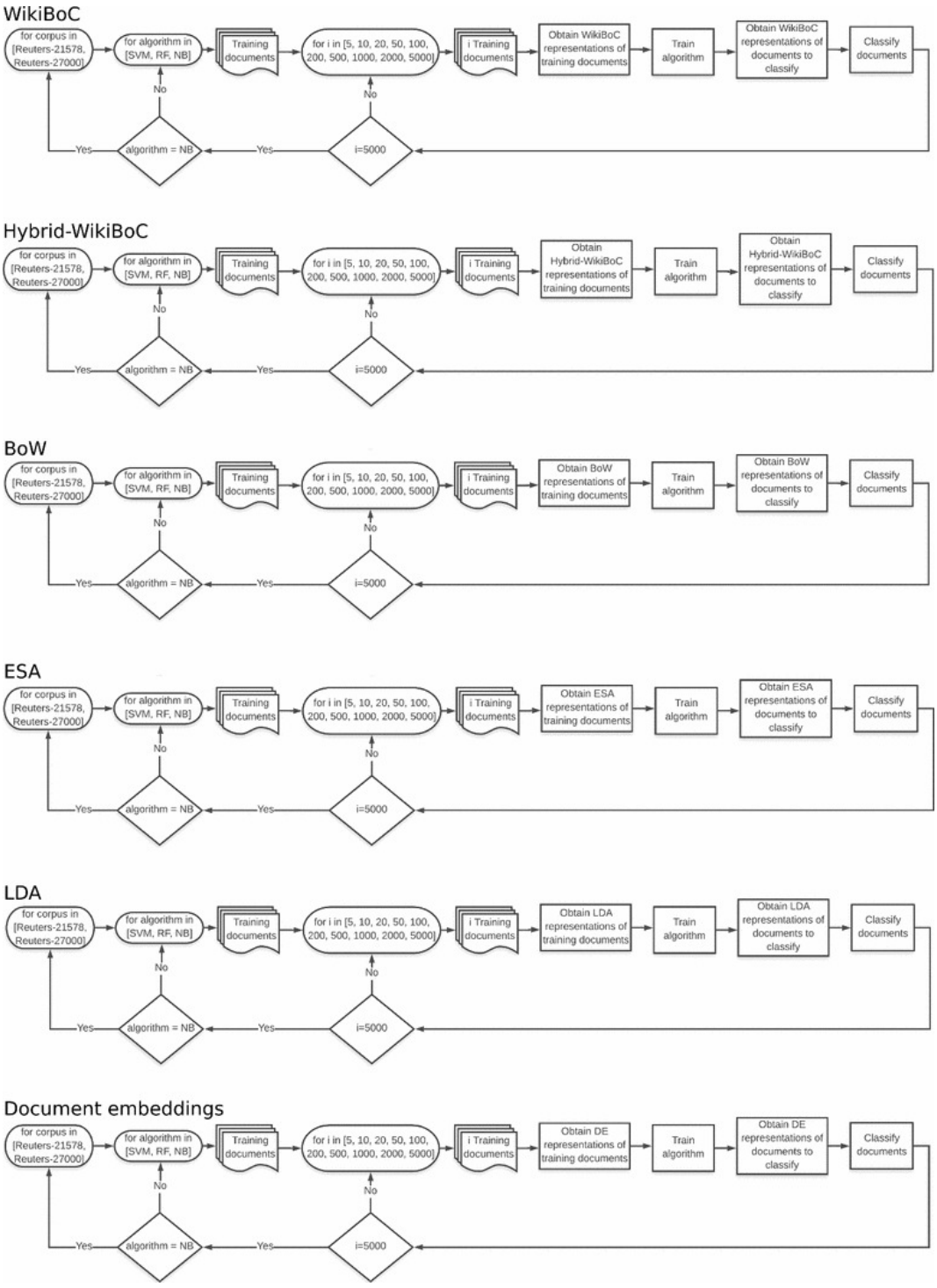

Fig. 5 Flowchart of the experimental settings

To implement the classifier based on the ESA representation of documents, it was necessary to obtain the ESA concept-based representation of each document in corpora. In order to create the ESA instance, we followed the work by Gabrilovich and Markovitch (2009) and the guidelines listed in the source code.[2]

The LDA representations of documents were obtained through the use of the *sklearn.decomposition.LatentDirichlet Allocation* class, included in the *Scikit-learn* library. In order to obtain the number of features—concepts, topics—which maximizes the classification performance, we conducted several experiments using different feature sizes. The results of the experiments show that the best performance is obtained when using a number of topics not too high—around 200 in our work—which is coherent with the studies in the state of the art. Vulić et al. (2015) state that tasks that require only coarse categorizations, such as document classification, typically use a small number of topics. Besides, although there is not a fixed threshold value for a good number of topics, since it depends on each problem in particular, previous works show the number of topics is typically in the [50–300] interval (De Smet et al. 2011; Ni et al. 2011; Vulić et al. 2015).

Finally, to implement the classifier based on document embeddings (DE), we rely on paragraph vector, an extension of word embeddings to learn document-level embeddings (Le and Mikolov 2014). Concretely, we use *doc2vec*, a widely used implementation of paragraph vectors[3] (Rehurek and Sojka 2010).

5.2 Results

The results of the experiments are presented in terms of F_1- *score*, the harmonic mean of Precision and Recall metrics (Sebastiani 2002; Sahlgren and Cöster 2004).

5.2.1 Reuters-21578

Figure 6 and Table 1 show the evolution of the F1-score for the BoW, WikiBoC, Hybrid-WikiBoC, LDA, ESA, and document embeddings approaches when varying the length of the training sequence for the Reuters-21578 corpus, using the SVM, random forests, and naïve Bayes algorithms.

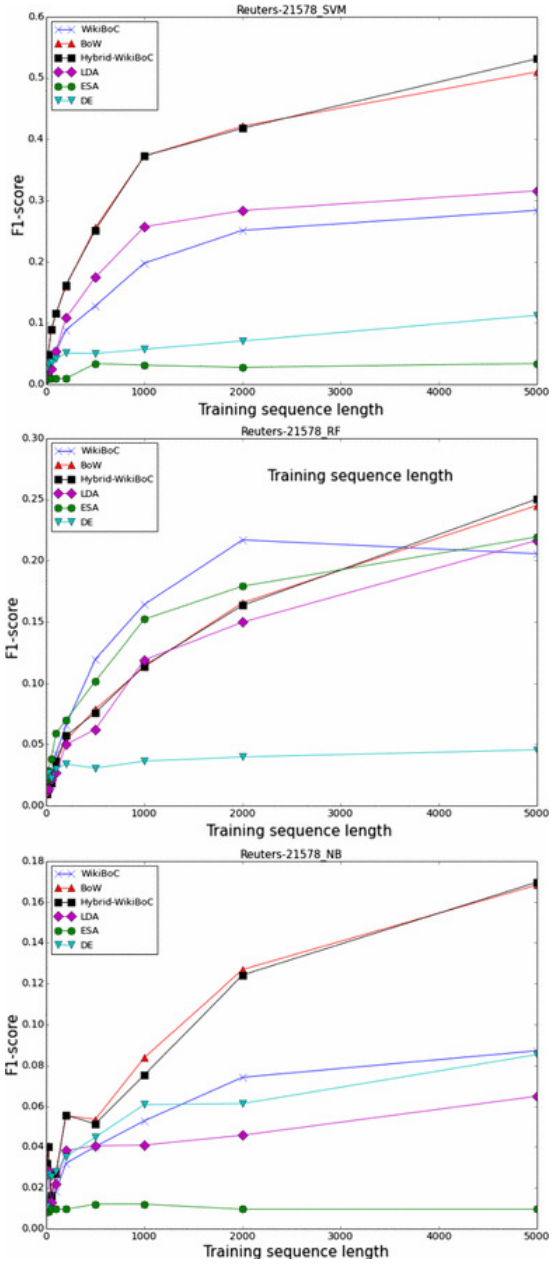

Fig. 6 Comparison of the performance of the SVM, random forests, and naïve Bayes classifiers in the Reuters-21578 corpus for the WikiBoC, BoW, Hybrid-WikiBoC, LDA, and document embeddings (DE) representations

[2]The source code is freely available at https://github.com/faraday/ wikiprep-esa.

[3]https://radimrehurek.com/gensim/.

5.2.2 Reuters-27000

Similarly, Fig. 7 and Table 2 show the evolution of the F1-score for the BoW, WikiBoC, Hybrid-WikiBoC, LDA, ESA, and document embeddings approaches when varying the length of the training sequence for the Reuters-27000 corpus, using the SVM, random forests, and naïve Bayes algorithms.

6 Discussion

The discussion section is divided into three subsections. The first two are devoted to discuss the performance of the evaluated approaches for each corpus. The third subsection is dedicated to analyse the computational complexity of the system and the evaluated approaches.

6.1 Reuters-21578 corpus

Figure 6 and Table 1 clearly show that the SVM algorithm offers the best performance for all document representations except for ESA, where random forests offer the best performance. Then, we consider the SVM algorithm as the most suitable for classifying Reuters-21587 corpus. Thus, the remainder of the discussion of Reuters-21578 results will be focused on the results offered by the SVM algorithm.

6.1.1 The BoW and the Hybrid-WikiBoC approaches

The performance of the BoW approach is higher than any of pure concept-based approaches (WikiBoC, LDA, ESA, and Documents embeddings) in the whole range of training sequences lengths, offering performance increases for the largest training sequence of 79.58, 61.39, 1.400, and 351.33%, respectively. The performance of the Hybrid-WikiBoC approach is equal or greater than the performance offered by the classifier based on the BoW representation for almost every training sequence length, being the one that offers the highest performance (F_1-scor e = 0.531), showing relative increases over BoW, LDA, ESA, and DE of 4.12, 68.04, 1461.76, and 369.91%, respectively. It should be noted that the relative increase over BoW is 4.12the offered by the state-of-the-art approaches proposed by Cai and Hofmann (2003) (2.71%), Sahlgren and Cöster (2004) (1.37%), and Yousif et al. (2015) (3.16%). It means that the features—Wikipedia concepts—used to enrich the BoW representation add useful information for the classifier, thus improving its performance (Cai and Hofmann 2003; Sahlgren and Cöster 2004; Huang et al. 2009; Yousif et al. 2015).

6.1.2 Pure concept-based approaches (WikiBoC, LDA, document embeddings and ESA)

The pure concept-based approaches seem to be not suitable for classifying Reuters-21578, since the performance they offer falls well below the performance offered by the BoW and Hybrid-WikiBoC approaches.

WikiBoC

The BoW and the Hybrid-WikiBoC approaches outperform the WikiBoC model in the complete range of training sequences, obtaining performance increases for the largest training sequence of 79.58 and 86.97%, respectively. The performance of the WikiBoC approach depends heavily on the ability of the semantic annotator to extract con- cepts from documents. Reuters-21578 documents contain lots of abbreviations, measures, and other words that the semantic annotator fails to translate into concepts. Table 3 shows an example of two documents randomly selected from Reuters-21578 and Reuters-27000 corpora, and the concepts extracted by the Wikipedia Miner semantic annotator (Wik-iBoC concepts) from each of them. It can be seen that the quality of concepts extracted from Reuters-21578 document is clearly inferior than the quality of concepts extracted from Reuters-27000 document: lower number of concepts, and poorly related to the document. This means that the concepts extracted do not represent the document in an optimal way, which has a negative impact on the performance of classification algorithms.

LDA

The performance of the classifier based on LDA is also quite low, and the BoW and Hybrid-WikiBoC approaches outperform it in the whole range of training sequences, achieving relative performance improvements for the largest training sequence of 61.39 and 68.04%, respectively. As well as the WikiBoC model, LDA topics— concepts—are

Table 1 Comparison of the performance of the SVM, random forests, and naïve Bayes classifiers in the Reuters-21578 corpus for the WikiBoC, BoW, Hybrid-WikiBoC, LDA, and document embeddings (DE) representations when varying the length of the training sequence (upper row)

	5	10	20	50	100	200	500	1000	2000	5000
SVM										
WikiBoC	0.013	0.015	0.022	0.031	0.054	0.089	0.128	0.197	0.251	0.284
BoW	**0.028**	**0.028**	**0.049**	**0.089**	**0.116**	0.160	**0.255**	**0.372**	**0.421**	0.510
Hybrid-WikiBoC	**0.028**	**0.028**	**0.049**	**0.089**	0.115	**0.161**	0.251	**0.372**	0.418	**0.531**
LDA	0.001	0.016	0.026	0.025	0.053	0.108	0.174	0.257	0.283	0.316
ESA	0.010	0.008	0.010	0.010	0.010	0.010	0.034	0.031	0.028	0.034
DE	0.025	0.027	0.033	0.036	0.042	0.051	0.050	0.057	0.071	0.113
Random forests										
WikiBoC	**0.013**	0.015	0.016	**0.019**	**0.043**	**0.065**	**0.120**	**0.164**	**0.217**	0.206
BoW	0.009	**0.017**	**0.019**	**0.019**	0.035	0.053	0.079	0.113	0.166	0.245
Hybrid-WikiBoC	0.009	0.016	0.018	**0.019**	0.036	0.057	0.076	0.114	0.163	**0.250**
LDA	0.012	0.011	0.013	0.022	0.027	0.050	0.062	0.120	0.150	0.216
ESA	0.021	0.022	0.028	0.038	0.059	0.070	0.102	0.152	0.179	0.219
DE	0.024	0.026	0.027	0.022	0.030	0.034	0.031	0.036	0.040	0.046
Naïve Bayes										
WikiBoC	0.012	0.009	0.010	0.011	0.018	0.032	0.040	0.053	0.074	0.087
BoW	**0.028**	**0.032**	**0.040**	**0.016**	**0.027**	0.055	**0.054**	**0.084**	**0.127**	0.168
Hybrid-WikiBoC	**0.028**	**0.032**	**0.040**	**0.016**	**0.027**	**0.056**	0.051	0.075	0.124	**0.170**
LDA	0.013	0.029	0.028	0.013	0.022	0.038	0.041	0.041	0.046	0.065
ESA	0.010	0.008	0.008	0.010	0.010	0.010	0.012	0.012	0.010	0.010
DE	0.011	0.026	0.027	0.025	0.028	0.035	0.044	0.061	0.061	0.085

Values in bold indicate the highest value for each training sequence length for each algorithm

obtained from documents' text, so it is clear that the quality of concepts is strongly dependent on the content of documents. Row 'LDA concepts' of Table 3 shows the first ten LDA topics for each corpus. We can see that the topics extracted from Reuters-21578 corpus contain lots of abbreviations and numbers, and many of them are not very informative.

Document embeddings

The performance of the classifier based on document embeddings is also very low. The WikiBoC, BoW, and Hybrid-WikiBoC approaches beat it in the complete range of training sequences, showing performance increases for the largest training sequence of 151.33, 351.33, and 369.91%, respectively. As this approach obtains features from external corpora, it seems clear that these corpora have a crucial role in the features extracted, which is line with previous works (Mouriño-García et al. 2017; Lau and Baldwin 2016). In order to verify this, we performed four different experiments using different corpora to obtain the features. The first two experiments were conducted using two pre-trained models, the first one obtained from the entire Wikipedia and the second one from a set of Associated Press news[4] (Lau and Baldwin 2016). The last experiments were conducted using two models created by us, using news from Reuters-21578 and Reuters-27000 corpora, respectively.[5] Figure 8 shows the performance of the document embeddings approach using the four models, where we can see that there are significant variations on performance depending on the external corpora used, being the best results those obtained when using Reuters-27000 and Reuters-21578 corpora.

ESA

Finally, the worst results are those obtained using the ESA concept-based representation, being

[4] https://github.com/jhlau/doc2vec.

[5] Models are freely available at http://www.itec-sde.net/doc2vec_ reuters27000_reuters21578_models.zip.

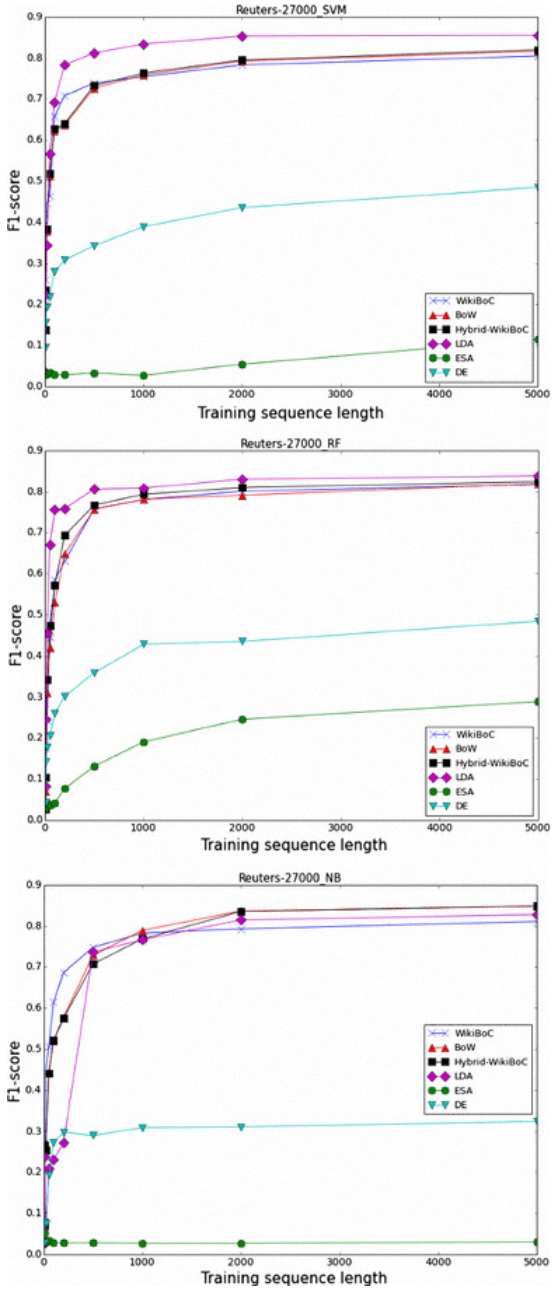

Fig. 7 Comparison of the performance of the SVM, random forests, and naïve Bayes classifiers in the Reuters-27000 corpus for the WikiBoC, BoW, Hybrid-WikiBoC, LDA, and document embeddings (DE) representations

largely surpassed by the other representations. This behaviour is due to its poor performance when extracting concepts from texts (Mouriño-García et al. 2017). This is because of the following reasons. First, Gabrilovich and Markovitch (2009) state that considering the document as a whole can be wrong, because its text might be too diverse to be mapped to the right set of concepts, while notions mentioned only briefly may be omitted. Besides, ESA trends to generate outliers (Egozi et al. 2011)—concepts that are not related to the document, or concepts that are related to the document only marginally—which hinders its usefulness in classification tasks. 'ESA concepts' row in Table 3 shows an example of the concepts extracted by ESA from a text document randomly selected from Reuters-21578 corpus. It is observed that although the ESA approach was able to extract a large number of concepts, they are not related to the document. This implies that the ESA concepts extracted do not represent the document correctly, which has a negative impact on the performance of the classifier.

6.2 Reuters-27000 corpus

Regarding the Reuters-27000 corpus (Fig. 7 and Table 2), it is not easy to determine at first glance which algorithm performs better. The highest performance is obtained by using the SVM algorithm along with the LDA representation of documents (achieving a F1-score of 85.5%), but the performance offered by other combinations of document representations and classification algorithms offer values very close to it. This behaviour is line with King et al. (1995), which state that the performance of classification algorithms depends critically on the dataset and on the features of the dataset.

6.2.1 The BoW and the Hybrid-WikiBoC approaches

The Hybrid-WikiBoC approach outperforms BoW in the whole range of training sequence lengths for the SVM and random forests algorithm, and with the largest training sequence for the naïve Bayes, achieving performance increases up to 49.3%. Again, enriching BoW representations with Wikipedia concepts extracted from text add valuable information for the classifier, thus improving its performance. Although the LDA-based approach is the one that offers the highest performance, the Hybrid-WikiBoC approach performs very well, being the one that offers the second best performance. Besides, and unlike LDA, the Hybrid-WikiBoC approach we propose does a

Table 2 Comparison of the performance of the SVM, random forests, and naïve Bayes classifiers in the Reuters-2700 corpus for the WikiBoC, BoW, Hybrid-WikiBoC, LDA, and document embeddings (DE) representations when varying the length of the training sequence (upper row)

	5	10	20	50	100	200	500	1000	2000	5000
SVM										
WikiBoC	**0.152**	**0.259**	**0.446**	0.465	0.658	0.708	0.738	0.754	0.783	0.805
BoW	0.138	0.233	0.378	0.512	0.622	0.636	0.726	0.758	0.792	0.816
Hybrid-WikiBoC	0.139	0.236	0.384	0.518	0.627	0.639	0.732	0.763	0.795	0.820
LDA	0.034	0.224	0.344	**0.567**	**0.692**	**0.783**	**0.812**	**0.834**	**0.853**	**0.855**
ESA	0.028	0.030	0.030	0.034	0.028	0.028	0.033	0.027	0.054	0.115
DE	0.094	0.157	0.194	0.220	0.280	0.310	0.343	0.389	0.440	0.485
Random forests										
WikiBoC	0.030	0.177	0.242	0.445	0.584	0.630	0.758	0.780	0.801	0.818
BoW	0.028	0.069	0.310	0.420	0.530	0.648	0.757	0.781	0.791	0.819
Hybrid-WikiBoC	0.028	0.104	0.343	0.474	0.571	0.693	0.767	0.793	0.810	0.824
LDA	**0.083**	**0.246**	**0.455**	**0.671**	**0.756**	**0.759**	**0.806**	**0.809**	**0.830**	**0.838**
ESA	0.028	0.029	0.042	0.036	0.042	0.077	0.131	0.190	0.246	0.289
DE	0.044	0.141	0.178	0.206	0.260	0.301	0.359	0.428	0.435	0.484
Naïve Bayes										
WikiBoC	0.067	**0.272**	**0.445**	**0.505**	**0.614**	**0.686**	**0.748**	0.783	0.793	0.811
BoW	0.067	0.264	0.254	0.444	0.519	0.577	0.729	**0.789**	**0.836**	0.848
Hybrid-WikiBoC	**0.071**	0.267	0.254	0.441	0.522	0.575	0.707	0.768	0.835	**0.849**
LDA	0.028	0.239	0.031	0.210	0.231	0.272	0.736	0.766	0.815	0.828
ESA	0.028	0.052	0.030	0.034	0.028	0.028	0.028	0.027	0.027	0.030
DE	0.028	0.078	0.075	0.194	0.273	0.298	0.290	0.309	0.311	0.324

Values in bold indicate the highest value for each training sequence length for each algorithm

very decent role in both corpora, either 'concept-friendly' or not. Table 4 and Fig. 9 show the average of F_1-*score* values of LDA and Hybrid-WikiBoC approaches over both corpora, where it can be seen that the Hybrid-WikiBoC approach outperforms LDA in almost the whole range of training sequences length, achieving a performance increase of 15.56% for the largest training sequence.

6.2.2 Pure concept-based approaches (WikiBoC, LDA, document embeddings and ESA)

WikiBoC

Unlike Reuters-21578, the Reuters-27000 corpus seems suitable for using the WikiBoC concept-based representation, since the performance it offers is very close or even superior to that offered by word- and hybrid-based approaches. This confirm what we stated previously: the quality of concepts is strongly dependent on the content of corpora. On the one hand, the performance of the

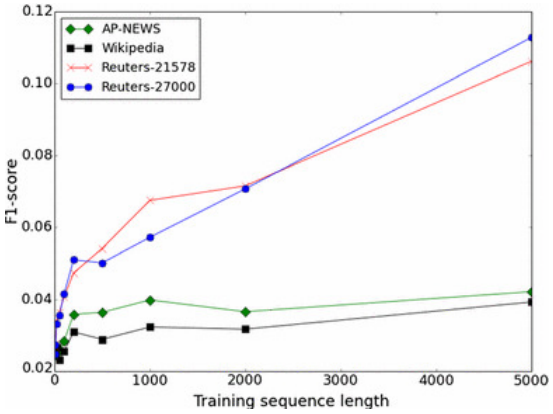

Fig. 8 Corpora influence on document embeddings approach for the Reuters-21578 corpus

WikiBoC approach depends heavily on the ability of the semantic annotator to extract concepts from documents. Row 'WikiBoC concepts' in Table 3 shows that the quality of concepts extracted from Reuters-27000 documents is clearly superior to that of those extracted from Reuters-21578 documents. The set of Wikipedia concepts extracted by

Table 3 Reuters-21578 and Reuters-27000 documents and concepts extracted from them

Corpus	Reuters-21578			Reuters-27000		
Text	Shr 39 cts vs 50 cts Net 1,545,160 vs 2,188,933 revs 25.2 mln vs 19.5 mln Year Shr 1.53 dlrs vs 1.21 dlrs Net 6,635,318 vs 5,050,044 Revs 92.2 mln vs 77.4 mln NOTE: Results include adjustment of 848,600 dlrs or 20 cts shr for 1986 year and both 1985 periods from improvement in results of its universal life business than first estimated. Reuter			The drug, when given in addition to standard treatment, extended median overall survival in 50% of newly diagnosed glioblastoma multiforme (GBM) patients to 2 years in a mid-stage study. Usually GBM patients succumb to the disease in 1 year. (Reporting by Natalie Grover in Bangalore; Editing by Joyjeet Das)		
WikiBoC concepts	Nordisk Mobiltelefon (Sweden) 1986, 1985,	635, 848, 933, Universal life insurance,	Pandan Bikol language, Business Universality (philosophy)	Therapy, Disease, Median, Drug, Glioblastoma multiforme,	Bangalore Theatre, Editing, Standarization,	Natalie Cole, Standard treatment, Survival rate Report, Grover
ESA concepts	Photomask, German Aerospace Center, London Borough of Newham, Cheng Man-ch'ing, Transport for London, Cadence Design Systems, Canary Whark tube station,	Docklands Light Railway, Isle of Dogs, GM Sigma platform, Greenwich station, Stratford station, Jubilee Line, Job Control Language, Oyster card, UBS AG, STS-55,	GM High Feature engine language, Guatemalan Party of Labour, Bank and Monument stations, Deptford, Canary Wharf, Poplar,London, Digital video Carpal tunnel, syndrome	Game Boy Micro,	Gliobastoma multiforme,	Hong Kong honours system
LDA cancepts	#1: Versus 000 ct shr net #2: Said product compani research develop #3: Said program s agricultur u #4: Pct year februari 0 price #5: 1 reuter 3 28 2 #6: Said s futur chang requir #7: Price said market demand thi #8: Earn dlr share compani said #9: March reuter 31 13 16 #10: Industri product said output manufactur			#1: Immigr said s border reform #2: Execut s chief wa year #3: Univers particl matter cern scientist #4: Job report labour unemploy month #5: Veteran va said wait day #6: Debt s govern default argentina #7: Play s music perform broadway #8: Ecb euro zone rate said #9: Said wa comment did ani #10: Said thi think s time		

Outliers are underlined. We are only showing the first 5 words of the ten first LDA topics (concepts) extracted for each corpus

Table 4 Average of F_1 -score values of LDA and Hybrid- WikiBoC approaches over both Reuters-21578 and Reuters-27000 corpora

	5	10	20	50	100	200	500	1000	2000	5000
Av. F1 LDA	0.022	0.120	0.185	0.296	**0.373**	**0.446**	**0.493**	0.545	0.568	0.585
Av. F1 Hybrid-WikiBoC	**0.083**	**0.132**	**0.216**	**0.304**	0.371	0.400	**0.493**	**0.568**	**0.606**	**0.676**

Upper row indicates the length of the training sequence
Values in bold indicate the highest value for each training sequence length

the Wikipedia Miner semantic annotator from the Reuters-27000 document is composed of a greater number of concepts, and most of them are very related to the document.

LDA

We have seen that LDA does not perform well in the Reuters-21578 corpus, but its performance is very high in the Reuters-27000 corpus, being the one which offers the highest performance for the SVM and random forests algorithms. It shows performance increases over the WikiBoC, BoW, Hybrid-WikiBoC, ESA, and DE approaches of 6.21, 4.78, 4.27, 643.48, and 76.29%, respectively, when using the SVM algorithm, and of 2.44, 2.32, 1.70, 189.96, and 73.14% when using random forests. For the naïve Bayes algorithm, the LDA model only outperforms the WikiBoC, ESA, and DE approaches, being surpassed by the BoW and Hybrid-WikiBoC approaches, the latter being the one that offers the high performance.

This behaviour is because LDA needs 'concept-friendly' corpora for extracting topics. Row 'LDA concepts' of Table 3 clearly suggests that LDA topics obtained from Reuters-27000 documents are more informative and of more quality than those extracted from Reuters-21578 news items.

Document embeddings

The performance of the classifier based on document embeddings is quite far away from the offered by BoW, WikiBoC, Hybrid-WikiBoC, and LDA representations regardless of the classification algorithm employed. For the support vector machines, the BoW, WikiBoC, Hybrid-WikiBoC, and LDA models outperform document embeddings approach in the whole range of training sequences length, obtaining performance increases for the largest training sequence of 68.25, 65.98, 69.07, and 76.29%, respectively. For the random

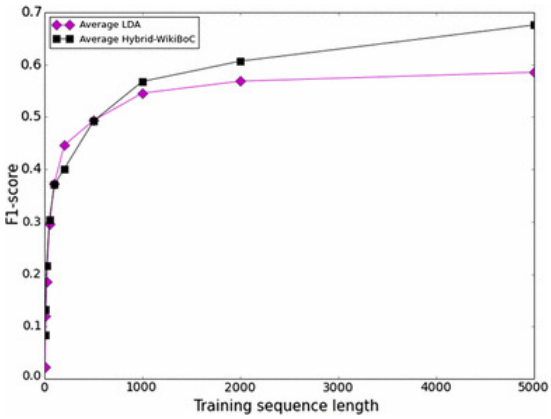

Fig. 9 Average of F_1-*score* values of LDA and Hybrid- WikiBoC approaches over both Reuters-21578 and Reuters-27000 corpora

forests classifier, the BoW, WikiBoC, Hybrid-WikiBoC, and LDA approaches show performance increases for the largest training sequence of 69.21, 69, 70.25, and 73.14%, respectively. Finally, for the naïve Bayes algorithm, the BoW, WikiBoC, Hybrid-WikiBoC, and LDA representations obtain performance increases for the largest training sequence of 161.73, 150.31, 162.04, and 155.56%, respectively. As we previously stated, external corpora used to obtain features have a crucial role in the features extracted. Figure 10 shows the variation in performance on the document embeddings approach depending on the corpora used to obtain the features. Again, there are significant variations on performance depending on external corpora used, being the best results those obtained using Reuters-27000 news as external corpora.

ESA

Finally, the ESA approach offers the worst results, being amply surpassed by the other representations. Again, ESA fails when extracting concepts from entire documents. 'ESA concepts' row in Table 3 shows the concepts extracted from a text

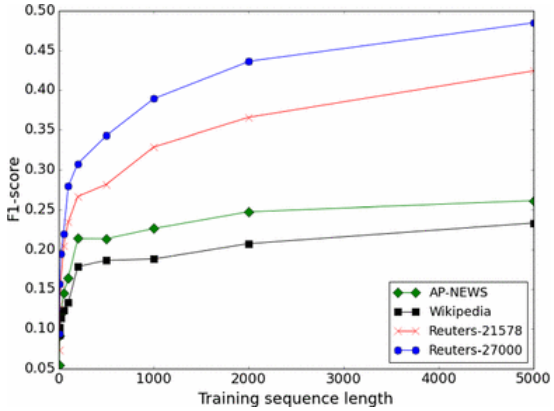

Fig. 10 Corpora influence on document embeddings approach for the Reuters-27000 corpus

document randomly selected from Reuters-27000 corpus. It can be seen that ESA was only able to extract three concepts from the document, two of which are not related to it. Again, the concepts extracted do not represent the document correctly, which has a negative impact on the performance of the classification algorithm.

6.3 Computational complexity

Insofar, complexity of the infrastructure is concerned, and there are no special requirements to implement the approach proposed. Both the implementation of the Wikipedia Miner semantic annotator and the realization of the classification experiments were conducted using office-grade personal computers (i.e. Intel® Core™ 7-4770 CPU @ 3.40 GHz × 8 with 16GB RAM).

Insofar, complexity time is concerned, and the CPU times needed to train and test the three classification algorithms, using the six representations of documents and the two corpora employed in this work, are depicted in Tables 5 and 6.

These times enabled us to draw the following observations:

- The random forests algorithm is the one that needs more time to be trained and for classifying documents, regardless both of the corpus and the representation of documents selected.
- The support vector machines and naïve Bayes algorithms show similar training and classification times for each pair corpus-document's representation.
- The training and classification times when using the LDA approach are similar for both corpora. This is because the complexity is dependent on the number of features (dimensionality), and for LDA, the number of features used to represent each document is preselected. In particular, the LDA model uses 200 features to represent each document, regardless of the corpora. The behaviour is similar for document embeddings model, which also uses 200 features to represent each document.
- The training and classification times when using the WikiBoC, BoW, Hybrid-WikiBoC, and ESA models are greater when classifying the Reuters-27000 corpus than when classifying Reuters-21578 corpus, independently of the classification algorithm. Again, this is because the complexity is dependent on the number of features. Reuters-27000 documents are, in general, more extensive than Reuters-21578 documents, which causes that the number of features which represents a document from Reuters-27000 is greater than the number of features which represent a Reuters-21578 document.
- In the same way, the training and classification times when using the Hybrid-WikiBoC representation of documents are greater than the training and classification times when using the BoW and the WikiBoC approaches, regardless of the classification algorithm employed. Again, the number of features used to represent a document following the Hybrid-WikiBoC model is greater than the number of features used to represent a document following the BoW and WikiBoC approaches. Specifically, the number of features used to represent a document according to the Hybrid-WikiBoC representation is the sum of the number of BoW and WikiBoC features.

7 Conclusions

The study presented in this paper attempts to provide solutions aimed at increasing the performance of automatic news classification systems. To that end, we presented an automatic news classification system using three of the most relevant algorithms in the state of the art—SVM, random forests, and naïve Bayes—and two document representations: WikiBoC, only based on Wikipedia concepts; and Hybrid-WikiBoC, that leverage the advantages of the traditional BoW paradigm and the WikiBoC approach.

Table 5 CPU time needed to train and test the three classification algorithms using the six representations of documents for the Reuters-21578 corpus

	SVM		RF		NB	
	Train	Test (ms/doc.)	Train	Test (ms/doc.)	Train	Test (ms/doc.)
WikiBoC	8″	0.89	34′51″	530.35	7″	1.02
BoW	5′12″	32.65	49′42″	837.28	5′4″	31.07
Hybrid-WikiBoC	7′25″	70.04	54′55″	973.04	5′44	36.86
LDA	9″	0.63	10′38″	505.01	6″	0.62
ESA	9′48″	54.24	225′06″	977.36	8′34″	52.66
DE	20″	0.579	15′14″	487.10	5″	1.77

Training times are shown in minutes and seconds. Testing times are shown in milliseconds per document

Table 6 CPU time needed to train and test the three classification algorithms using the six representations of documents for then Reuters-27000 corpus

	SVM		RF		NB	
	Train	Test (ms/doc.)	Train	Test (ms/doc.)	Train	Test (ms/doc.)
WikiBoC	12′27	77.5	94′22″	945.63	13′38″	71.25
BoW	32′40″	205.63	96′40″	1180.63	31′51″	203.75
Hybrid-WikiBoC	77′21″	488.75	186′3″	1806.25	78′1″	439.38
LDA	6″	0.63	9′53″	546.88	6″	0.60
ESA	35″	3.63	215″	687.15	37″	3.875
DE	25″	0.625	8′36″	500.63	5″	0.731

Training times are shown in minutes and seconds. Testing times are shown in milliseconds per document

The results of the experiments conducted allow us to obtain the following conclusions:

- The enrichment of the BoW representation with concepts extracted from documents through the Wikipedia Miner semantic annotator adds valuable information to the classifier, thus improving its performance. Experiments conducted show performance increases over BoW up to 4.12 and 49.35% when classifying Reuters-21578 and Reuters-27000 corpora, respectively.
- The performance of the classification algorithm depends on the dataset used, and on their features, but SVM is the one that offers the best performance for classifying both corpora.
- Regarding the representation of documents, the performance of the different approaches—particularly those purely based on concepts—also strongly depends on the particular corpora, being the Hybrid-WikiBoC and LDA models the best options for classifying Reuters-21578 and Reuters-27000 corpora, respectively. Finally, we consider the Hybrid-WikiBoC model as the best overall choice, since it offers the highest average performance over both corpora.

Acknowledgments. Work supported by the European Regional Development Fund (ERDF) and the Galician Regional Government under agreement for funding the Atlantic Research Centre for Information and Communication Technologies (AtlantTIC), and the projects R2014/034 (RedPlir), and R2014/029 (TELGalicia).

Compliance with ethical standards

Conflict of interest All authors declared that they have no conflict of interest.

References

Arif MH, Li J, Iqbal M, Liu K (2017) Sentiment analysis and spam detection in short informal text using learning classifier systems. Soft

Comput 1–11. https://doi.org/10.1007/s00500-017-2729-x

Bekkerman R, El-Yaniv R, Tishby N, Winter Y (2003) Distributional word clusters vs. words for text categorization. J Mach Learn Res 3:1183–1208

Bengio Y, Ducharme R, Vincent P, Jauvin C (2003) A neural proba- bilistic language model. J Mach Learn Res 3:1137–1155

Blei DM, Ng AY, Jordan MI (2003) Latent dirichlet allocation. J Mach Learn Res 3:993–1022

Breiman L (2001) Random forests. Mach Learn 45(1):5–32

Cai L, Hofmann T (2003) Text categorization by boosting automatically extracted concepts. In: Proceedings of the 26th annual international ACM SIGIR conference on Research and development in informaion retrieval. ACM, pp 182–189

Chang MW, Ratinov LA, Roth D, Srikumar V (2008) Importance of semantic representation: dataless classification. AAAI 2:830–835

Colace F, De Santo M, Greco L, Napoletano P (2014) Text classification using a few labeled examples. Comput Hum Behav 30:689–697

De Smet W, Tang J, Moens MF (2011) Knowledge transfer across mul- tilingual corpora via latent topics. Adv Knowl Discov Data Min 549–560. https://doi.org/10.1007/978-3-642-20841-6_45

Deerwester SC, Dumais ST, Landauer TK, Furnas GW, Harshman RA (1990) Indexing by latent semantic analysis. JAsIs 41(6):391–407

Egozi O, Markovitch S, Gabrilovich E (2011) Concept-based informa- tion retrieval using explicit semantic analysis. ACM Trans Inf Syst (TOIS) 29(2):8

Elberrichi Z, Rahmoun A, Bentaallah MA (2008) Using wordnet for text categorization. Int Arab J Inf Technol 5(1):16–24

Gabrilovich E, Markovitch S (2007) Computing semantic relatedness using wikipedia-based explicit semantic analysis. IJCAI 7:1606–1611

Gabrilovich E, Markovitch S (2009) Wikipedia-based semantic interpretation for natural language processing. J Artif Intell Res 34:443–498

Hearst MA, Dumais ST, Osman E, Platt J, Scholkopf B (1998) Support vector machines. IEEE Intell Syst Appl 13(4):18–28

Huang A, Milne D, Frank E, Witten IH (2009) Clustering documents using a wikipedia-based concept representation. In: Advances in knowledge discovery and data mining. Springer, pp 628–636

Huang L, Milne D, Frank E, Witten IH (2012) Learning a concept- based document similarity measure. J Am Soc Inform Sci Technol 63(8):1593–1608

Jadhav BR, Mahajan M, GHR CEM W, (2016) Dual sentiment analysis using adaboost algorithm sentiment analysis. Int J Eng Sci 6(6):7641–7645

Jiang M, Cao J-Z (2016) Positive-unlabeled learning for pupylation sites prediction. BioMed Res Int 2016:4525786 https://doi.org/10.1155/2016/4525786

Jin P, Zhang Y, Chen X, Xia Y (2016) Bag-of-embeddings for text classification. Int Jt Conf Artif Intell 25:2824–2830

Khan A, Baharudin B, Lee LH, Khan K (2010) A review of machine learning algorithms for text-documents classification. J Adv Inf Technol 1(1):4–20

Kim HK, Kim M (2016) Model-induced term-weighting schemes for text classification. Appl Intell 45(1):30–43

Kim H, Howland P, Park H (2005) Dimension reduction in text classification with support vector machines. J Mach Learn Res 6:37–53

King RD, Feng C, Sutherland A (1995) Statlog: comparison of classification algorithms on large real-world problems. Appl Artif Intell Int J 9(3):289–333

Kozielski S, Mrozek D, Kasprowski P, Kostrzewa D et al (2015) Beyond databases, architectures and structures. Springer, Berlin

Lau JH, Baldwin T (2016) An empirical evaluation of doc2vec with practical insights into document embedding generation. ACL 2016:78

Le Q, Mikolov T (2014) Distributed representations of sentences and documents. In: Proceedings of the 31st international conference on machine learning (ICML-14), pp 1188–1196

Lewis DD (1998) Naive (bayes) at forty: the independence assumption in information retrieval. In: Machine learning: ECML-98. Springer, pp 4–15

Li J, Fong S, Zhuang Y, Khoury R (2016) Hierarchical classification in text mining for sentiment analysis of online news. Soft Comput 20(9):3411–3420

Manimala K, David IG, Selvi K (2015) A novel data selection technique using fuzzy c-means clustering to enhance svm-based power quality classification. Soft Comput 19(11):3123–3144

Mekala D, Gupta V, Karnick H (2016) Text classification with sparse composite document vectors. arXiv preprint arXiv:1612.06778

Mihalcea R, Corley C, Strapparava C et al (2006) Corpus-based and knowledge-based measures of text semantic similarity. AAAI 6:775–780

Mikolov T, Chen K, Corrado G, Dean J (2013) Efficient estimation of word representations in vector space. arXiv preprint arXiv:1301.3781

Milne D, Witten IH (2013) An open-source toolkit for mining wikipedia. Artif Intell 194:222–239

Ming ZY, Chua TS (2015) Resolving polysemy and pseudonymity in entity linking with comprehensive name and context modeling. Inf Sci 307:18–38

Mogadala A, Rettinger A (2016) Bilingual word embeddings from parallel and non-parallel corpora for cross-language text classification. In: Proceedings of NAACL-HLT, pp 692–702

Moise G, Vladoiu M, Constantinescu Z (2014) Maseco: a multi-agent system for evaluation and classification of oers and ocw based on quality criteria. In: E-Learning paradigms and applications. Springer, pp 185–227

Mouriño García MA, Pérez Rodríguez R, Anido Rifón LE (2015) Biomedical literature classification using encyclopedic knowledge: a wikipedia-based bag-of-concepts approach. Peer J 3:e1279

Mouriño-García M, Pérez-Rodríguez R, Anido-Rifón L, Gómez-Carballa M (2016a) Bag-of-concepts document representation for bayesian text classification. In: 2016 IEEE international conference on computer and information technology (CIT). IEEE, pp 281–288

Mouriño García MA, Pérez Rodríguez R, Anido Rifón L (2016) Reuters 27000 corpus. URL http://dx.doi.org/10.17632/3cw44dk29f.2

Mouriño-García MA, Pérez-Rodríguez R, Anido-Rifón L (2017) Wikipedia-based cross-language text classification. Inf Sci 406–407:12–28. https://doi.org/10.1016/j.ins.2017.04.024

Nezreg H, Lehbab H, Belbachir H (2014) Conceptual representation using wordnet for text categorization. Int J Comput Commun Eng 3(1):27

Ni X, Sun JT, Hu J, Chen Z (2011) Cross lingual text classification by mining multilingual topics from wikipedia. In: Proceedings of the

fourth ACM international conference on Web search and data mining. ACM, pp 375–384

Nigam K, McCallum AK, Thrun S, Mitchell T (2000) Text classification from labeled and unlabeled documents using em. Mach Learn 39(2–3):103–134

Pavlinek M, Podgorelec V (2017) Text classification method based on self-training and lda topic models. Expert Syst Appl 80:83–93

Pedregosa F, Varoquaux G, Gramfort A, Michel V, Thirion B, Grisel O, Blondel M, Prettenhofer P, Weiss R, Dubourg V et al (2011) Scikit-learn: machine learning in python. J Mach Learn Res 12:2825–2830

Porter MF (1980) An algorithm for suffix stripping. Program 14(3):130– 137

Rehurek R, Sojka P (2010) Software framework for topic modelling with large corpora. In: Proceedings of the LREC 2010 workshop on new challenges for NLP frameworks, Citeseer

Rodrigues F, Lourenco M, Ribeiro B, Pereira FC (2017) Learning supervised topic models for classification and regression from crowds. IEEE Trans Pattern Anal Mach Intell 39(12):2409–2422

Rose T, Stevenson M, Whitehead M (2002) The reuters corpus volume 1-from yesterday's news to tomorrow's language resources. LREC 2:827–832

Roul RK, Asthana SR, Kumar G (2017) Study on suitability and importance of multilayer extreme learning machine for classification of text data. Soft Comput 21(15):4239–4256

Sahlgren M, Cöster R (2004) Using bag-of-concepts to improve the performance of support vector machines in text categorization. In: Proceedings of the 20th international conference on Computational Linguistics. Association for Computational Linguistics, p 487

Sebastiani F (2002) Machine learning in automated text categorization. ACM Comput Surv (CSUR) 34(1):1–47

Selamat A, Yanagimoto H, Omatu S (2002) Web news classification using neural networks based on pca. In: SICE 2002. Proceedings of the 41st SICE annual conference, vol 4. IEEE, pp 2389–2394

Settles B (1994) Active learning literature survey. Mach Learn 15(2):201–221

Singh A, Chhillar SK (2017) News category classification using distinctive bag of words and ann classifier. Int J Emerg Res Manag Technol 6(6):311–317

Stock WG (2010) Concepts and semantic relations in information science. J Am Soc Inform Sci Technol 61(10):1951–1969

Van TP, Thanh TM (2017) Vietnamese news classification based on bow with keywords extraction and neural network. In: 2017 21st Asia Pacific symposium on intelligent and evolutionary systems (IES). IEEE, pp 43–48

Vulić I, De Smet W, Tang J, Moens MF (2015) Probabilistic topic modeling in multilingual settings: an overview of its methodology and applications. Inf Process Manag 51(1):111–147

Wang P, Hu J, Zeng HJ, Chen Z (2009) Using wikipedia knowledge to improve text classification. Knowl Inf Syst 19(3):265–281

Wenliang C, Xingzhi C, Huizhen W, Jingbo Z, Tianshun Y (2004) Automatic word clustering for text categorization using global information. In: Asia information retrieval symposium. Springer, pp 1–11

Yao D, Bi J, Huang J, Zhu J (2015) A word distributed representation based framework for large-scale short text classification. In: 2015 international joint conference on neural networks (IJCNN)

Yousif SA, Samawi VW, Elkabani I, Zantout R (2015) The effect of combining different semantic relations on arabic text classification. World

Comput Sci Inf Technol J 5(1):12–118

Zhang H (2004) The optimality of naive bayes.
AA 1(2):3