Intelligent Data Analysis 23 (2019) 587–607
DOI 10.3233/IDA-184085
IOS Press

An extension to association rules using a similarity-based approach in semantic vector spaces

Brian Keith Norambuena* and Claudio Meneses Villegas
Department of Computing and Systems Engineering, Universidad Católica del Norte, Antofagasta, Chile

AUTHOR COPY

Abstract. Sentiment analysis is a field that has experienced considerable growth over the last decade. This area of research attempts to determine the opinions of people on something or someone. This article introduces a novel technique for association rule extraction in text called Extended Association Rules in Semantic Vector Spaces (AR-SVS). The objective of this analysis is to explore the feasibility of applying AR-SVS in the field of opinion mining and sentiment analysis. This new method is based on the construction of association rules, which are extended through a similarity criteria for terms represented in a semantic vector space. The method was evaluated on a sentiment analysis data set consisting of scientific paper reviews. A quantitative and qualitative analysis is done with respect to the classification performance and the generated rules. The results show that the method is competitive compared to the baseline provided by Naïve Bayes and Support Vector Machines. Furthermore, previous work on the evaluation of scientific paper reviews (the Scoring Algorithm) has been used in conjunction with association rules to obtain a method that shows a superior behaviour compared to the baseline. Finally, additional experiments are performed on various multidomain data sets in order to evaluate the results of AR-SVS in different settings.

Keywords: Sentiment analysis, data mining, association rules, semantic vector spaces

1. Introduction

The main objective of the present work is to explore a new method for generating association rules with applications in sentiment analysis. This paper is an extension to a shorter version of one [9] where the method was originally proposed. The main new contributions with respect to the original article are the addition of new experiments with the RBS representation using the Scoring algorithm from our previous work [7,8] which includes additional experiments on our review data set. Furthermore, a new section with a full AR-SVS example was added in order to give details for people who might need to implement this method from scratch. In addition, the results from new experiments on a subset of the Blitzer et al. [5] multidomain sentiment analysis data sets containing five very different domains. Finally, various additional comments on the design decisions and parametrization of the different methods.

This proposal is based on the intuitive idea that two related terms will be close to each other in the vector representation. Given this, if an association rule contains one of the terms, it is possible that other

[*] Corresponding author: Brian Keith Norambuena, Department of Computing and Systems Engineering, Universidad Católica del Norte, Antofagasta, Chile. E-mail: brian.keith@ucn.cl.

1088-467X/19/$35.00 © 2019 – IOS Press and the authors. All rights reserved

terms that are close by in the vector space can also be used in this association rule. The difficulty of this lies in finding an adequate criterion for proximity (i.e., a similarity score). This general idea can be used to build extended association rules that include similar terms. In particular, we intend to use this idea of extension by similarity of association rules to classify the polarity of documents.

Association rule learning is a method to discover patterns and interesting relationships between variables in a data set. Its objective is to identify interesting rules of the form $X \rightarrow Y$, where X and Y are sets of variables. There are various methods to extract these association rules from the data, the basic one is the Apriori algorithm [1], which is used in this work.

Association rules are not designed for classifying, instead they are a tool designed to detect co-occurrence patterns. Since these rules do not directly provide a classifier, their application to the task of polarity determination would be complementary to another method and would require various modifications. Nevertheless, there are methods that propose the usage of association rules for classification tasks [12].

Sentiment analysis (also known as opinion mining) is a growing field, particularly due to the huge growth of data available in the web such as blogs, social networks, forums, among others. One of the many applications of opinion mining is assessing products (or services) through the analysis of users' reviews. This allows discovering what people say and think about a certain product, service or organization in general [11].

In the field of opinion mining, association rules are used to find the most important aspects of a certain entity. Association rules show the relationship between words that represent entities and nouns that appear in the text. In case a strong association exists, this means there is a high chance that these nouns are related to the corresponding entities with which they co-occur [22].

On the other hand, association rules can be used to generate opinion lexicons. To do this a set of words is used as a seed, and from this set association rules are applied iteratively to expand this set [22].

The rest of this work is organized as follows: the second section shows related works. The third section formally describes the proposed method and discusses its basic components. The fourth section details the materials and methods used to evaluate the proposal, including a description of the data and the required tools. The fifth section shows the main results and the associated discussion. Finally, the last section presents the conclusions and possible research lines for future work.

AUTHOR COPY

2. Related work

There are approaches that propose the use of association rules to carry out the task of classification [12]. Associative classification differs from classic association rules in the sense that a restriction is added to the rules in such a way that in the consequent there can only be one attribute (the class).

The associative classification rules can be obtained using an algorithm similar to Apriori called CBA-RG to generate the rules and another algorithm CBA-CB to construct the classifier. The rules constructed have the class label in the consequent. From the set of generated rules, a subset is selected using a heuristic criterion [12].

There are multiple criteria to generate the rules, the most common are support and confidence. Support is the number of instances in the training set which are relevant to the rule. Confidence refers to the conditional probability that the right-hand side of the rule is satisfied if the left-hand side of the rule is satisfied [16].

Another useful metric is the Average Deviation Support which measures the discrepancy in the support distribution and allows determining the rules that discriminate the different classes [29]. This metric is defined as:

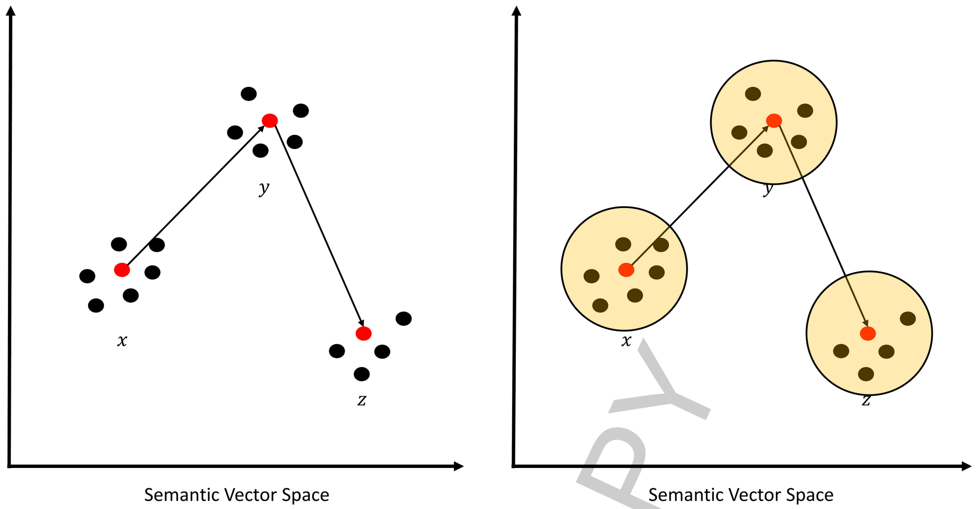

Fig. 1. Association rules in semantic vector spaces.

AUTHOR COPY

$$ADSup(I) = \frac{\sqrt{\sum_{i=1}^{n}\left[Sup(I)_i - Avg(Sup(I))\right]^2}}{Avg(Sup(I))}$$

$$Avg(Sup(I)) = \frac{1}{n}\sum_{i=1}^{n} Sup(I)_i$$

Where $Sup(I)_i$ is the support of the set of frequent items I for class i, $Avg(Sup(I))$ represents the support of the frequent element set I among all classes. The integer n is the number of classes. This metric is used because it presents good results when it is applied to the task of text classification [29].

In general, depending on how the class label is chosen, there are two kinds of approaches of associative classification: those that make predictions through a strategy of maximum likelihood and those that use multiple rules to generate scores. The methods of associative classification to classify texts have been well studied, but the use of association rules for sentiment classification has not been thoroughly explored yet [15].

3. Proposed method

3.1. Description of the AR-SVS method

This proposal seeks to exploit the capacity of the association rules for detecting interesting patterns. It is sought to generalize classic association rules in such a way that they do not represent associations between words, but between regions of the semantic vector space (as can be observed in Fig. 1). In particular, it is expected to obtain associative classification rules using the words that are located close in the semantic vector space.

In particular, associative classification rules are obtained by using the words that are close by to the original words in the rule in the semantic vector space, according to some adequate measure of similarity.

In order to generate these new association rules, the closest terms to each term of the *LHS* (left-hand side) and the *RHS* (right-hand side) of the association rule would be selected. Note that in general there can be several terms in the *LHS* and the *RHS*, so that there can be many neighborhoods to consider in the construction. In the case of the association rules for classification, whose *RHS* corresponds to the class label, only the closest terms to the elements of the *LHS* would be considered. This last construction for associative classification rules is the one we consider for the purposes of this article.

To determine the similarity of the terms, different methods can be utilized. It is recommended to normalize the vectors, because for the specific task of determining whether two words are similar, this has shown to provide better results [28]. The number of closest terms that will be used for the method would be a parameter defined by the final user.

This method allows capturing the semantic associations in the text, and in particular it allows making inferences on what each document really means. In particular, by extending the rules with the closest terms the method will have more information at its disposal. Furthermore, this method is considered interesting due to its possible generalizations and its intuitive nature, it is more natural to think in the existence of associated regions inside a semantic vector space than in point associations.

Another important aspect of this method is that domain knowledge could be leveraged in various ways. In particular, it is possible to use semantic vector spaces that have incorporated domain knowledge, perhaps through the use of an ontology (as in the work of Jauhar et al. [6]). On the other hand, it is also possible to use a domain specific semantic similarity measure to find the related terms, which is especially important considering that general purpose semantic similarity measures do not perform well within specific domains [25].

The method has been named AR-SVS (extended Association Rules in Semantic Vector Spaces). Note that the method is composed of several independent components, and the choice of these components is a challenge in itself. Having described the general idea of the method, each one of the steps and the design decisions involved are detailed. The method is described in a general way in Algorithm 1 under the assumptions that the model of vector representation is already trained and the association rules are in the correct format (i.e. classification rules).

AUTHOR COPY

Algorithm 1 Algorithm AR-SVS

Input: Set R of association rules, parameter $n \in \mathbb{N}$ that indicates the number of semantically similar terms to utilize

Output: Set R' of extended association rules.

```
function AR-SVS
    Let $R' = \emptyset$
    $\forall r \in R$:
        $K_r = \{\mathit{represent}(i) \mid i \in \mathit{LHS}(r)\}$
        $S_r = \bigcup_{k \in K_r} \{\mathit{closest}(k, n)\}$
        $\forall t \in S_r$:
            $r' = (\{t\} \rightarrow \mathit{RHS}(r))$
            $\mathit{sup}(r') = \mathit{sup}(r)$
            $R' = R' \cup \{r'\}$
    return $R'$
end function
```

In this definition, the functions $\mathit{LHS}(r)$ and $\mathit{RHS}(r)$ obtain the sets of the left-hand side and the right-hand side of the rule r, respectively. The function $\mathit{represent}(i)$ takes the term i and obtains its representation in the vector space. The function $\mathit{closest}(k, n)$ obtains the n terms closest to the term k according to some similarity metric. Finally, $\mathit{sup}(r)$ corresponds to the support of the rule.

Note that for simplicity, the generated rules inherit the support of the original rule (if a same rule is generated many times, it inherits the highest corresponding support). The main reason for doing this is that if one were to calculate the classic support of an extended rule it would probably be very low (since it was not part of the original rules, thus it is not considered a frequent item set). Considering this, the extended association rule should use the original support as a basis for its own support. In this case they are considered equal, but eventually it could be weighted by the semantic similarity with the original word or with some other similar schemes.

The definition of the method has been done in a general way, allowing freedom to apply the methods considered adequate in each step. The first design decision to consider is the selection of the algorithm to construct the association rules and the selection of the evaluation metrics for the rules (e.g., support and confidence). Also, it is necessary to select the representation of the terms, the similarity criterion and define the value of n. Thus, we now present the design decisions that belong to the algorithm itself:

1. Line 4 requires selecting the word/term representation (e.g. word2vec).
2. Line 5 requires selecting the similarity criterion (e.g. cosine similarity or the inverse Euclidean distance [27]) and assigning a value for n. Note that n_{k_r} terms are obtained in the set S_r.

In principle, the only a priori information required corresponds to the vector space that represents the words in the rules, the specific details of how this semantic vector space was constructed are a black box for the proposed algorithm. This vector space representation could be constructed in such a way that it uses a priori knowledge, for example, an ontology could be exploited in order to obtain word embeddings that are grounded in this ontology [6] (i.e. they exploit the knowledge stored therein).

With respect to the similarity criterion, it should be noted that semantic similarity measures designed for general purposes do not necessarily perform well in a specific domain [25]. Thus, depending on the actual domain of application, different similarity measures could be applied. In fact, a way in which domain information could be used is through the use of knowledge stored in an ontology [13,18,19,24]. Therefore, when applying AR-SVS in a specific domain, it is important to select an adequate similarity measure. For the purposes of the proof of concept contained in this article, a general purpose metric (cosine similarity) has been selected. However, if higher accuracy rates were needed then a domain specific similarity measure could prove useful.

The optimal value for the parameter n can be found empirically. From preliminary experiments it would seem that small values (1 or 2) are preferable due to the amount of combinations that would need to be taken into consideration. For the purposes of this work, this approach has been taken. However, further study is required in order to determine basic heuristics to even guide the search of the optimal value.

In this work, only support has been considered for simplicity of our implementation, but it should be noted that another metric could be selected arbitrarily, as of now, the effects of different metrics has not been thoroughly evaluated and given the large number of parameters and all the moving parts of this proposal, we believe another complete and exhaustive study for optimal parametrization and design decisions is needed, perhaps in the same way as it has been done for Semantic Vector Spaces in the work of Kiela and Clark [10]. However, such evaluation is considered to be beyond the scope of this article.

One of the limitations of the proposed algorithm is that it only considers the construction of association rules for the classification problem. Furthermore, the described proposal only generates rules of unitary length. This has been done to reduce the method's complexity, because finding all the possible combinations of valid extended association rules would require the definition of a specialized evaluation metric that could filter out uninteresting rules, in a similar fashion to what the original Apriori algorithm does.

AUTHOR COPY

However, this algorithm can be generalized, and it has value on its own, even if its application to the classification task is not considered. This method can be adapted to obtain new association rules (not only for classification) using the concept of semantic similarity, allowing it to expand the resulting rules based on the output of conventional algorithms for finding association rules.

To adapt the algorithm, it would be necessary to change the way the *RHS* is handled. In the current proposal we use the original *RHS* of the rule r, however, it is possible to replace this *RHS* with a new version RHS' made up of the terms that are semantically similar to the original terms in *RHS*. This leads to the construction of various new rules, for example, $r' = (\mathit{LHS}(r) \rightarrow \mathit{RHS}'(r))$ or $r' = (S_r \rightarrow \mathit{RHS}'(r))$. There are several possible combinations that could be considered intuitively valid to do this. Nevertheless, for the purposes of this article and our proposal, we only consider the implementation of association rules for the classification task.

However, this is not an easy task, because this would require defining a metric that is not directly based on co-occurrences (otherwise the new rules would be the same as those found by the basic Apriori algorithm); secondly, it would be necessary to design an algorithm that could generate these rules without incurring in an extremely high computational cost. This last issue could possibly be tackled through an approach similar to the Apriori algorithm or a dynamic programming variant.

Leaving these important observations behind, this work evaluates the method as described in Algorithm 1. Tackling some of these mentioned issues is thus considered out of the scope of this work and is proposed as future work.

3.2. *Classification with AR-SVS*

Although the algorithm to construct extended association rules by means of semantic vector spaces is intrinsically valuable, it is necessary to remember that the aim of this work requires using the association rules obtained to determine the semantic orientation of a document. In order to do this, it is necessary to have a classification algorithm and a scheme that allows utilizing the association rules to classify.

In particular, an approach based on scores is used to carry out the classification, this is formally described in Algorithm 2. The basic idea of this approach is to construct a scores vector that will represent each document with respect to each class. The vectors built for each document will be utilized as input for some method of traditional classification.

Algorithm 2 Rule Based Scoring Algorithm (RBS)

Input: Set of association rules R and the list of documents D.
Output: c-dimensional vector representation of the documents. Where c is the number of classes.

```
function RBS
    $\forall d \in D$:
        $v_d = \mathit{zeros}(c)$
        $\forall r \in R$:
            $v_d[\mathit{RHS}(r)] += I(\mathit{LHS}(r) \subseteq d) \cdot \mathit{sup}(r)$
    return $v_d$
end function
```

Where the function $\mathit{zeros}(c)$ takes as input the number of classes c and returns a vector v_d initialized in zeros that will be used to store the score associated to each class. The function I is an indicator function that takes the value 1 if the statement in it is true and 0 otherwise. The function $\mathit{sup}(r)$ obtains the support of the rule r. The main part of the algorithm corresponds to the following: for each extended association

AUTHOR COPY

Table 1
Semantic vector space representation for the words in the full AR-SVS example

Word	x	y
Terrible	0.55	0.8
Bad	0.48	0.72
Mediocre	0.52	0.58
Regular	0.55	0.48
Good	0.61	0.32
Excellent	0.65	0.26

rules, the support of the rule is added to the vector v_d in the position corresponding to the rules' class. Thus, this algorithm is equivalent to the following formula:

$$v_d[i] = \sum_{\substack{r \in R \\ RHS(r)=i \\ LHS(r) \subseteq d}} sup(r), \forall i = 1, \ldots, c$$

Three variants of the RBS algorithm are defined:

1. *RBS-B*: the input corresponds to the set of basic rules R.
2. *RBS-X*: the input corresponds to the set of extended rules R', it does not include the basic rules R.
3. *RBS-BX*: the input corresponds to the union of both sets $R \cup R'$.

It should be noted that for longer rules it is less probable that all their elements will be a subset of a certain document. Also, their support is naturally lower, so the classification is more influenced by the rules with a small *LHS*. Intuitively this is justified by the fact that the more common rules should filter the main elements, while the longer rules (and thus less common), would be able to only make small distinctions between documents. Given that the AR-SVS method only generates rules with a unitary length for the *LHS*, this discussion is only relevant for the original rules.

It is possible to define the algorithm in such a way that it considers the presence of each element in the *LHS* and not the presence of the whole set in the document. However, empirical evaluation of this approach shows that the classification performance decreases. Given this, it has been decided to use the approach that requires the presence of the full *LHS* set in the document.

Note that the metric used to build the vectors could be different from support (e.g. confidence), thus there is some flexibility with respect to this component of our proposal. The generated vector can be used however we see fit. It is even possible to apply further transformation and preprocessing techniques on these vectors, in case it is deemed necessary.

To determine the class different approaches can be applied. The simplest way is to assign the class that has the highest score in each document, and in case of a draw, assume neutrality. Another option is training a machine learning classifier that takes as input the scores vectors. The first approach is simple and easy to implement, while the other approach requires the use of a more complex machine learning classifier. However, a machine learning classifier should be able to detect more complex patterns in the input vectors which are lost to the maximum approach. This difference might be important for the multiclass case, where there are several elements in the RBS vector.

3.3. Full AR-SVS example

In this section a full working example of AR-SVS and RBS is presented with an artificial toy data set and fictitious association rules. For simplicity and visualization concerns we work on a bi-dimensional

AUTHOR COPY

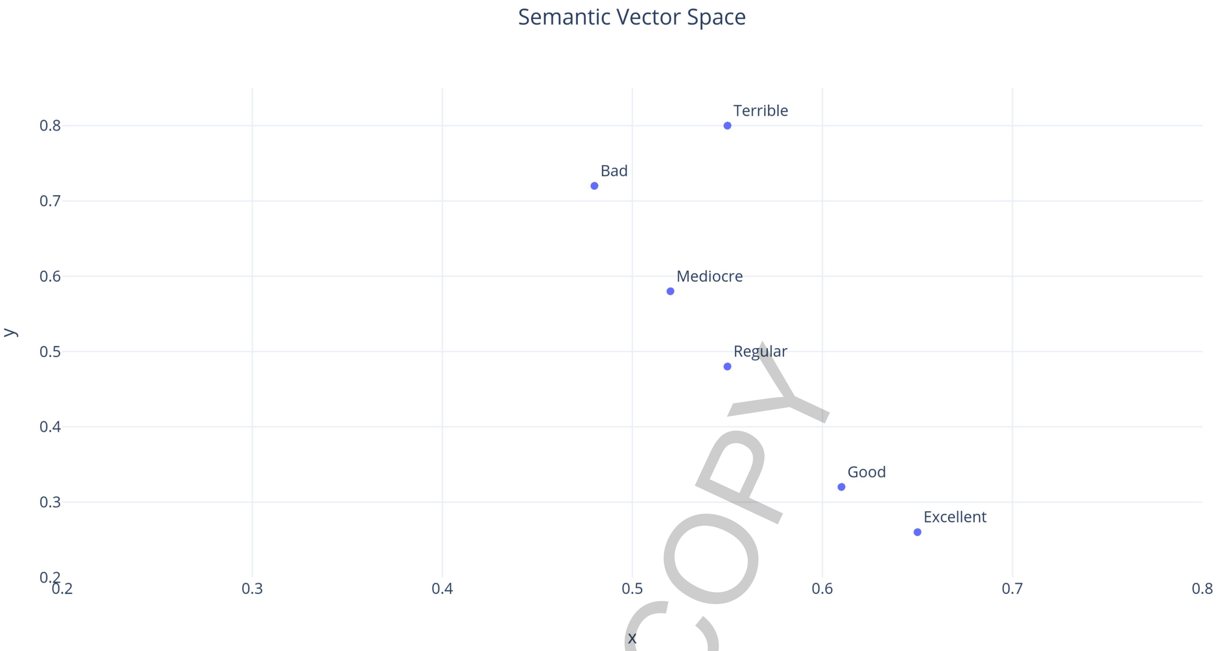

Fig. 2. Visualization of the semantic vector space for the AR-SVS example.

AUTHOR COPY

semantic vector space. Table 1 shows the data set of words that will be used throughout this example represented in the semantic vector space with coordinates x and y. The data can be visualized in Fig. 2.

The input rules are $R = \{r_1, r_2\}$ with

- $r_1 = (\{\mathit{Good}\} \to \{+\}), \mathit{sup}(r_1) = 0.4$
- $r_2 = (\{\mathit{Mediocre}, \mathit{Terrible}\} \to \{-\}), \mathit{sup}(r_2) = 0.2$

The AR-SVS algorithm will be executed using the rules R as input and choosing the top $n = 1$ similar terms. The term similarity will be measured using Euclidean distance (the bigger the distance, the less similar the terms). In this case it is easy to determine the most similar words in each case by visual inspection of Fig. 2.

Now we proceed to the execution of the AR-SVS algorithm. For the rule r_1 we get:

$$K_{r_1} = \{\mathit{represent}(\mathit{Good})\} = \{(0.61, 0.32)\}$$

$$S_{r_1} = \{\mathit{closest}((0.61, 0.32), 1)\} = \{\mathit{Excellent}\}$$

Only one semantically similar term has been found in this case, so only one rule must be added:

$$r' = (\{\mathit{Excellent}\} \to \{+\})$$

$$\mathit{sup}(r') = \mathit{sup}(r_1) = 0.4$$

$$R' = \emptyset \cup \{r'\} = \{(\{\mathit{Excellent}\} \to \{+\})\}$$

For the rule r_2 we have:

$$K_{r_2} = \{\mathit{represent}(\mathit{Mediocre}), \mathit{represent}(\mathit{Terrible})\} = \{(0.52, 0.58), (0.55, 0.8)\}$$

$$S_{r_2} = \{\mathit{closest}((0.52, 0.58), 1), \mathit{closest}((0.55, 0.8), 1)\} = \{\mathit{Regular}, \mathit{Bad}\}$$

In this case two semantically similar terms are found (one for each word in the original rule), so two new association rules must be created. The first rule would be given by:

$$r' = (\{\mathit{Regular}\} \rightarrow \{-\})$$
$$\mathit{sup}(r') = \mathit{sup}(r_2) = 0.2$$
$$R' = \{(\{\mathit{Excellent}\} \rightarrow \{+\})\} \cup \{r'\}$$
$$R' = \{(\{\mathit{Excellent}\} \rightarrow \{+\}), (\{\mathit{Regular}\} \rightarrow \{-\})\}$$

The second rule would be:

$$r' = (\{\mathit{Bad}\} \rightarrow \{-\})$$
$$\mathit{sup}(r') = \mathit{sup}(r_2) = 0.2$$

Note that this generates the set of extended association rules $R' = \{r_3, r_4, r_5\}$ given by:

- $r_3 = (\{\mathit{Excellent}\} \rightarrow \{+\}), \mathit{sup}(r_3) = 0.4$
- $r_4 = (\{\mathit{Regular}\} \rightarrow \{-\}), \mathit{sup}(r_4) = 0.2$
- $r_5 = (\{\mathit{Bad}\} \rightarrow \{-\}), \mathit{sup}(r_5) = 0.2$

This completes the first part of our example, corresponding to the generation of the extended association rules. It now remains to explain how to apply the RBS algorithm with these results.

The list of documents to classify is $D = \{d_1, d_2, d_3\}$ with:

- $d_1 =$ "The development of ideas could be considered regular at best, I recommend improving it.". This document should have a negative orientation.
- $d_2 =$ "The format of this document is very bad. However, the idea behind the proposal is excellent.". This document should have a slightly positive orientation.
- $d_3 =$ "This work is extremely bad. The development of ideas is mediocre and the proposal itself has terrible writing". This document should have a negative orientation.

To classify we use the RBS method taking as input the set $R' \cup R$ given by:

- $r_1 = (\{\mathit{Good}\} \rightarrow \{+\}), \mathit{sup}(r_1) = 0.4$
- $r_2 = (\{\mathit{Mediocre}, \mathit{Terrible}\} \rightarrow \{-\}), \mathit{sup}(r_2) = 0.2$
- $r_3 = (\{\mathit{Excellent}\} \rightarrow \{+\}), \mathit{sup}(r_3) = 0.4$
- $r_4 = (\{\mathit{Regular}\} \rightarrow \{-\}), \mathit{sup}(r_4) = 0.2$
- $r_5 = (\{\mathit{Bad}\} \rightarrow \{-\}), \mathit{sup}(r_5) = 0.2$

By executing the RBS algorithm on each document, we have that:

- For the document d_1 we initialize $v_{d_1}[+] = 0$ and $v_{d_1}[-] = 0$, note that the only *LHS* that appears in this document corresponds with *LHS*(r_4), so we get that $v_{d_1}[-] = 0 + \mathit{sup}(r_4) = 0.2$.
- For the document d_2 we initialize $v_{d_2}[+] = 0$ and $v_{d_2}[-] = 0$. Note that in this case two rules appear (r_3 and r_5). Rule r_3 is positive, so we get $v_{d_2}[+] = 0 + \mathit{sup}(r_3) = 0.4$.). On the other hand, rule r_5 is negative, so we get $v_{d_2}[-] = 0 + \mathit{sup}(r_5) = 0.2$.
- For the document d_3 we initialize $v_{d_3}[+] = 0$ and $v_{d_3}[-] = 0$. Note that in this case we have two rules again (r_2 and r_5). Rule r_2 is negative, so we get $v_{d_2}[-] = 0 + \mathit{sup}(r_2) = 0.2$.). Rule r_5 is also negative, so we get $v_{d_2}[-] = 0.2 + \mathit{sup}(r_5) = 0.4$.

Finally, to classify we will use the simple approach of choosing the class with the highest *score*. Applying this strategy we find that d_1 is negative, d_2 is positive and d_3 is negative. Note that the documents d_1 and d_2 do not have any word that appear in the original association rules, so the new rules would be able to classify new elements in some cases. With these results we have completed our full example.

AUTHOR COPY

4. Methodology

4.1. Reviews data set

The method has been evaluated on the data set of reviews of scientific articles.[1] The data set has a total of 405 reviews, from these elements the reviews written in English (17 instances) and the empty reviews (6 instances) are discarded, leaving a total of 382 reviews in Spanish. The reviews have two possible target classes ("evaluation" and "orientation") [8]:

- *Evaluation:* Review classification as defined by the reviewer, according to the 5-point scale previously described. This attribute represents the real evaluation given to the paper, as determined by the reviewers.
- *Orientation:* Review classification defined in the original papers [7,8], according to the 5-point scale previously described, obtained through the authors' systematic judgment of each review. This attribute represents the subjective perception of each review (i.e. how negative or positive the review is perceived when someone reads it).

In this work the scale "orientation" has been used, because the evaluation does not always coincide with the semantic orientation of the text [7,8].

The evaluation of the methods utilizes a holdout approach with a proportion of 70% for the training set and 30% for the tests set, carrying out 10 replicates for each method. The averages of accuracy, precision, recall and F1 with its standard deviation are reported for each case. Regarding the preprocessing, first a tokenization of the input is carried out. Then, a stopwords filter [4] is applied. Afterwards, stemming is applied by means of the Porter algorithm [21].

For the rules, a modified variant of the Apriori algorithm that considers the minimum support with respect to each class (instead of the total of the data set) and the average deviation support (*ADSup*) is used. Afterwards, AR-SVS is applied to obtain the extended rules.

The representation of the text is done using *word2vec* trained on the data set. For the construction of the set S_r the cosine similarity between the normalized vectors is considered. The threshold value n is empirically obtained by evaluating qualitatively the similarity of the obtained terms.

The vectors of documents constructed for each variant of RBS are classified using three different approaches: choosing the class with the maximum score, training a Naïve Bayes classifier, and training a support vector machine. Naïve Bayes and SVM with LSA vectors as input are used as a comparison baseline. These two latter methods have been selected due to their wide use in the literature of sentiment analysis [22]. The implementation was carried out in Python using the library *sklearn* [20].

For the baseline of NB and SVM, once the preprocessing is completed, a representation is obtained using TF-IDF. The final representation is obtained by applying LSA (utilizing the $n = 100$ most significant components). For the method AR-SVS the *word2vec* representation is used [17], implemented through the library *gensim* [23]. The representation has been trained on the data set, pre-trained vectors have not been utilized.

Given the large number of parameters, rather than looking for an optimal parametrization, in this work a satisfactory set of parameters has been used, in the sense that with these parameters the results can be considered satisfactory [14]. The obtained results of 59% for the ternary classification task (against 65.45% accuracy for the human baseline) and 46% for the five-point scale case (against 36.65% accuracy for the human baseline) are considered satisfactory [8]. Some important details about the parameter tuning process are detailed here:

[1] The data set is publicly available for download at https://archive.ics.uci.edu/ml/datasets/Paper+Reviews.

AUTHOR COPY

- Linear SVM has been used throughout the experiments for the baseline with TF-IDF-LSA, experimenting with different values of C taken from the range $[10^{-3}, 10^{-2}, 10^{-1}, \dots, 10^{5}]$ and evaluated empirically until the best result was found.
- For *word2vec* a window width of 7 has been used throughout the tests and the dimension of the semantic vector space is set to 100.
- For the case of NB with the different RBS representations different configurations of minimum support, minimum confidence and minimum deviation for the rules have been tested. For all of these parameters, values from 0.05 to 0.70 were tested in increments of 0.05.
- For the case of SVM with the different RBS representations again a Linear SVM has been used with different values of C taken from the range $[10^{-3}, 10^{-2}, 10^{-1}, \dots, 10^{5}]$ and evaluated empirically until the best result was found using the optimal values of minimum support, confidence and deviation for NB.
- The details on the parametrization of the Scoring algorithm and a detailed account of its results can be found in the original paper [8].

Finally, for binary classification, the thresholds that have been used for the Apriori algorithm are a support of 25% with respect to the class and an ADSup of 15%. On the other hand, for ternary classification a minimum support of 10% has been used and an ADSup of 40%. These values have been found empirically, evaluating the average accuracy of 10 replicates for different values from 5% to 70% with increments of 5% for both parameters, in both cases a minimum confidence of 10% has been used.

AUTHOR COPY

4.2. *Multidomain data sets*

After the initial analysis on the Paper Reviews data set, the method is evaluated in the task of determining sentiment polarity on several multidomain data sets, from the works of Blitzer et al. [5]. This multidomain data set contains different kinds of reviews. All the data sets have four classes (very negative, negative, positive and very positive, note that there is no neutral class). For purposes of this article the task will be to determine if the sentiment is positive or negative, without consideration to the intensity of this polarity. Five data sets have been chosen from the collection: *automotive* (736 instances), *magazines* (4191 instances), *camera & photos* (7408 instances), *apparel* (9246 instances) and *toys & games* (13147 instances).

For this part only the accuracy metric has been considered for simplicity. Furthermore, Naïve Bayes will be used with TF-IDF and LSA (with the top $n = 100$ most significant components) for document representation. Finally, for AR-SVS and RBS, the simplest approach to classify the RBS representation (*MAX*) has been considered, considering that the *MAX* approach does not have any additional parameters that needs to be defined and since other methods and metrics have been thoroughly analyzed for the Paper Reviews data set. As in the previous case, the evaluation of the methods utilizes a holdout approach with a proportion of 70% for the training set and 30% for the tests set, carrying out 10 replicates for each method. The average accuracy is reported for each case.

The threshold values for support, confidence and deviation were taken from 0.05 to 0.20 in increments of 0.05. For the first four data sets (*automotive*, *magazines*, *camera & photos* and *apparel*) the values for the minimum support, confidence and deviation were 0.10, 0.10 and 0.15, respectively. For the *toys & games* data set the values were all equal to 0.10. Furthermore, for the number of similar terms $n = 2$ has been used in the AR-SVS algorithm. Finally, the same *word2vec* parameters that were used for the Paper Reviews data set have been used to train the word embeddings for these data sets.

Table 2
Summary of results obtained for binary classification

Binary classification					
Classifier	Representation	*Accuracy*	*Precision*	*Recall*	F_1
NB	TF-IDF-LSA	0.68 ± 0.05	0.67 ± 0.06	0.68 ± 0.05	0.64 ± 0.06
	RBS-B	0.63 ± 0.03	0.62 ± 0.04	0.63 ± 0.03	0.61 ± 0.04
	RBS-X	0.64 ± 0.04	0.64 ± 0.03	0.64 ± 0.04	0.63 ± 0.04
	RBS-BX	0.63 ± 0.03	0.63 ± 0.03	0.63 ± 0.02	0.63 ± 0.03
SVM	TF-IDF-LSA	0.7 ± 0.05	0.7 ± 0.05	0.7 ± 0.05	**0.69 ± 0.06**
	RBS-B	**0.72 ± 0.06**	**0.72 ± 0.07**	**0.72 ± 0.06**	**0.69 ± 0.06**
	RBS-X	0.62 ± 0.05	0.46 ± 0.15	0.62 ± 0.05	0.52 ± 0.10
	RBS-BX	0.67 ± 0.06	0.65 ± 0.14	0.66 ± 0.06	0.65 ± 0.11
MAX	RBS-B	0.65 ± 0.05	0.59 ± 0.16	0.65 ± 0.05	0.52 ± 0.07
	RBS-X	0.65 ± 0.06	0.64 ± 0.11	0.65 ± 0.06	0.54 ± 0.08
	RBS-BX	0.64 ± 0.05	0.53 ± 0.17	0.64 ± 0.05	0.51 ± 0.07

AUTHOR COPY

5. Results and discussion

5.1. Classification with AR-SVS and RBS

The results for binary classification are shown in Table 2. The best results are obtained with the RBS-B method, followed by the SVM base. This is the only instance of the method that exceeds the baseline. Although the other variants fail to overcome the performance of NB or SVM in all of the metrics, these present a competitive behavior in accuracy and recall. A larger difference is observed in the results of precision and F_1.

It can also be observed that the use of the extended rules (the X and BX variants) does not produce improvements in the classification results. However, it must be highlighted that even using only the new rules it is possible to classify the documents in a competitive way. The use of both rules (basic and extended) does not produce a consistent effect on the different evaluation metrics and the differences are not significant anyway.

The results in binary classification show that the representation generated by the RBS algorithm can be used to classify the documents in an adequate way. However, it is necessary to observe that its good performance depends on the set of rules used as an input, because the three variants B, X and BX have shown different behaviors on this data set.

The results for ternary classification are shown in Table 3. The behavior of the RBS method is in general similar to the binary case. Again, the good performance of the RBS-B variant in all the metrics can be noted.

The results of the RBS variants show a more competitive behavior. It must be noted that the RBS-BX method obtains similar results to the SVM base. Although RBS-B outperforms this method, in this case, adding the extended rules led to a slightly decreased performance.

The results in ternary classification corroborate those observed in the binary case with regards to the usefulness of the RBS representation. As before, it can be observed that the use of basic rules allows obtaining a better classification performance. Although unlike the binary case, the variants RBS-X and RBS-BX present a more competitive behavior.

Considering the previous results, we conclude that the RBS representation method is adequate, but that it depends on the set of the initial rules. There still exist several ways to improve the current proposal. The most evident one being finding a way to generate rules that allow more than one element in the *LHS*. On this particular data set, using the basic association rules provides slightly better results. However, it

Table 3
Summary of results obtained for ternary classification

Ternary classification					
Classifier	Representation	*Accuracy*	*Precision*	*Recall*	F_1
NB	TF-IDF-LSA	0.46 ± 0.03	0.42 ± 0.05	0.46 ± 0.03	0.41 ± 0.05
	RBS-B	0.41 ± 0.05	0.42 ± 0.07	0.41 ± 0.05	0.37 ± 0.04
	RBS-X	0.47 ± 0.04	0.38 ± 0.05	0.47 ± 0.04	0.41 ± 0.05
	RBS-BX	0.42 ± 0.05	0.36 ± 0.05	0.42 ± 0.05	0.37 ± 0.04
SVM	TF-IDF-LSA	0.48 ± 0.05	0.46 ± 0.06	0.48 ± 0.06	0.46 ± 0.06
	RBS-B	**0.49 ± 0.05**	0.48 ± 0.05	0.49 ± 0.05	**0.47 ± 0.06**
	RBS-X	0.45 ± 0.06	0.29 ± 0.08	0.45 ± 0.06	0.35 ± 0.07
	RBS-BX	0.48 ± 0.05	0.47 ± 0.04	0.48 ± 0.05	0.46 ± 0.05
MAX	RBS-B	**0.49 ± 0.05**	**0.52 ± 0.1**	**0.5 ± 0.05**	0.41 ± 0.07
	RBS-X	0.45 ± 0.04	0.44 ± 0.12	0.45 ± 0.04	0.36 ± 0.06
	RBS-BX	0.47 ± 0.05	0.45 ± 0.13	0.48 ± 0.05	0.35 ± 0.07

is possible that in other circumstances the approaches that use the extended rules could generate better results. As future work we consider the evaluation of this on a different, and possibly bigger, data set.

5.2. Extending the RBS representation

Finally, we perform some additional experiments to explore a different approach to classification using RBS in conjunction with other methods. This experiment takes the previous results and methods from [8]. In particular, we propose a different scheme based on the following observations:

- The usage of association rules for classification has produced slightly better results than SVM with TF-IDF+LSA.
- Combining TF-IDF+LSA with the Scoring Algorithm generates better results than the Scoring Algorithm on its own in the case of multiclass classification and competitive results in the binary case [8].
- Among all the RBS approaches used in our experiments, the one that provided the best results was the one using the original rules generated by Apriori (RBS-B).

Specifically, we have taken the output from RBS-B and we have added a new entry to this vector containing the output of Scoring Algorithm from [8]. This new vector is fed to a SVM and used as a new representation for the document. Thus, the method works as follows:

1. Take the output score vector generated by RBS-B (using the basic rules generated by Apriori).
2. Concatenate to each document vector the Score output generated by the Scoring Algorithm for the corresponding document.
3. Train a SVM using this new document representation.

We shall denote this new method by EHS-SVM (Extended Hybrid Scoring – SVM). We present the results for binary classification in Tables 4 and 5 we can find the results for ternary classification. We also include the results for classification with all five classes in Table 6.

In the case of binary classification, the best result is still given by the Scoring Algorithm, whereas for the case of ternary classification, the best result is given by the new EHS-SVM scheme. This new approach obtains a competitive performance in the binary case and a superior performance in the ternary case. Also, we can observe that in the five classes case the method also provides a good result in comparison with the other methods. Taking into account the obtained results, we affirm that, for the scientific paper reviews data set, the EHS-SVM method presents a better behaviour with respect to the effect of increasing the number of classes on the performance on various metrics.

AUTHOR COPY

Table 4
Binary classification results with EHS-SVM

Binary classification					
Method		*Accuracy*	*Precision*	*Recall*	F_1
Baseline	SVM	0.70 ± 0.05	0.70 ± 0.05	0.70 ± 0.05	0.69 ± 0.06
Methods from [8]	Score	**0.81**	**0.81**	**0.81**	**0.81**
	HS-SVM	0.79 ± 0.05	0.80 ± 0.05	0.79 ± 0.05	0.79 ± 0.05
Association rules	RBS-B SVM	0.72 ± 0.06	0.72 + 0.07	0.72 ± 0.06	0.69 ± 0.06
	EHS-SVM	0.75 ± 0.03	0.76 ± 0.03	0.75 ± 0.03	0.75 ± 0.03

Table 5
Ternary classification results obtained with EHS-SVM

Ternary classification					
Method		*Accuracy*	*Precision*	*Recall*	F_1
Baseline	SVM	0.48 ± 0.05	0.46 ± 0.06	0.48 ± 0.06	0.41 ± 0.05
Methods from [8]	Score	0.51	0.58	0.51	0.52
	HS-SVM	0.56 ± 0.04	0.54 ± 0.04	0.56 ± 0.04	0.54 ± 0.04
Association rules	RBS-B SVM	0.49 ± 0.05	0.48 ± 0.05	0.49 ± 0.05	0.47 ± 0.06
	EHS-SVM	**0.59 ± 0.04**	**0.58 ± 0.04**	**0.59 ± 0.04**	**0.58 ± 0.04**

Table 6
Classification results with five classes obtained with EHS-SVM

Classification with 5 classes					
Method		*Accuracy*	*Precision*	*Recall*	F_1
Baseline	SVM	0.4 ± 0.05	0.38 ± 0.04	0.41 ± 0.03	0.37 ± 0.03
Methods from [8]	Score	0.41	**0.5**	0.41	0.4
	HS-SVM	**0.46 ± 0.05**	0.45 ± 0.06	0.46 ± 0.05	0.43 ± 0.05
Association rules	RBS-B SVM	–	–	–	–
	EHS-SVM	**0.46 ± 0.04**	0.44 ± 0.05	**0.47 ± 0.05**	**0.44 ± 0.04**

5.3. Generated rules

Having analyzed the main results in terms of classification, the rules generated by the AR-SVS method are now discussed. Table 7 shows potential English translations of each word used by the example rules and Table 8 shows some of the original rules and the extended rules generated by the method for the binary classification for exemplification purposes.

Table 8 shows four association rules obtained through Apriori and the extended rules constructed using AR-SVS. It must be noted that in many cases the terms are repeated (e.g., the word "uso" (use) appears both in the positive and negative cases), furthermore, it is possible that the extended rules contain the same terms as the original rules (e.g. in the last rule, the term "deber" (must) was determined similar to "uso" (use) and "trabajo" (work). That is, there are terms that are related both by semantic similarity and co-occurrences evaluated by the algorithm Apriori, even allowing for the generation of cyclical relationships.

The graph of Fig. 3 shows the relationships among the different terms of Table 8. Bidirectional edges represent a co-ocurrence relationship and are labeled with "CO". Unidirectional edges represent a relationship of semantic similarity according to *word2vec* and are labeled with "w2v". It must be noted that the nodes can represent many words with different grammatical functions. This semantic multiplicity is due to the fact that during analysis words have been reduced to their root. Given this, for illustrative purposes just one representative has been chosen.

We present an analysis for the obtained rules and their relationships shown in Table 8:

AUTHOR COPY

Table 7
Potential translations for each word in the example rules

Spanish word	English translations
Interés	'Interest' is the direct translation, in this context possibly referring to an interesting paper, idea or application.
Aplicación	'Application' is the direct translation.
Artículo	(Scientific) 'Article' is the direct translation.
Faltar	'To Lack' is the direct translation.
Uso	'Usage' or 'Use' (noun) are the direct translations, it could also be related to the verb 'To Use'.
Aspecto	'Aspect' is the direct translation.
Ser	'To Be' is the direct translation.
Mejor	'Better' or 'Best' are both potential translations, depending on the context.
Hacer	'To Do' or 'To Make' are both potential translations, depending on the context.
Embargo	This is part of the compound expression 'Sin Embargo' which is an adversative clause and could be translated to 'Nevertheless' or similar words.
Deber	'Should' or 'Must' are both potential translations, depending on the context.
Trabajo	'Work' is the direct translation, in the context of these rules this probably refers to a Scientific Article.
Paper	This is an anglicism referring to Scientific Articles in this context.

Table 8
Examples of the obtained rules

Parametrization		
A minimum support of 20% is used and a minimum ADSup of 5% for this example. The two most similar terms are obtained ($n = 2$).		
Examples	Original rule	Extended rules
Regla 1	('Interés') $\Longrightarrow$ ('1')	('Aplicación') $\Longrightarrow$ ('1') ('Artículo') $\Longrightarrow$ ('1')
Regla 2	('Faltar') $\Longrightarrow$ ('-1')	('Uso',) $\Longrightarrow$ ('-1') ('Aspecto',) $\Longrightarrow$ ('-1')
Regla 3	('Ser', 'Mejor') $\Longrightarrow$ ('1')	('Uso') $\Longrightarrow$ ('1') ('Hacer') $\Longrightarrow$ ('1') ('Aspecto') $\Longrightarrow$ ('1') ('Embargo') $\Longrightarrow$ ('1')
Regla 4	('Deber', 'Trabajo', 'Ser') $\Longrightarrow$ ('-1')	('Uso') $\Longrightarrow$ ('-1') ('Trabajo') $\Longrightarrow$ ('-1') ('Uso') $\Longrightarrow$ ('-1') ('Paper') $\Longrightarrow$ ('-1') ('Uso') $\Longrightarrow$ ('-1') ('Trabajo') $\Longrightarrow$ ('-1')

AUTHOR COPY

- Rule 1 is isolated in the graph. In this case, the word "interés" ("interest") has no co-occurrence relations with the other terms that appear in this subset of rules. However, we can distinguish two semantic relationships obtained from similarity in *word2vec* (we could have found more if we increased the value of the n parameter).
  Rule 1 is associated with the positive class, so it is of interest to analyze its semantic relationships given by *word2vec* in this light ("application" and "article"). In particular, it is possible that this semantic relationship comes from reviewers indicating that the applications or the article itself are interesting. This would be a positive commentary with respect to the article, and thus it would be natural to associate it with the positive class.
- Rule 2 is associated with the negative class, so we will analyze its relationships taking this into account. In this case there are no co-occurrences with other words and there are only two semantic relationships with *word2vec*.
  In particular, the verb "faltar" ("to lack") is related with "uso" ("usage") and "aspecto" ("aspect"),

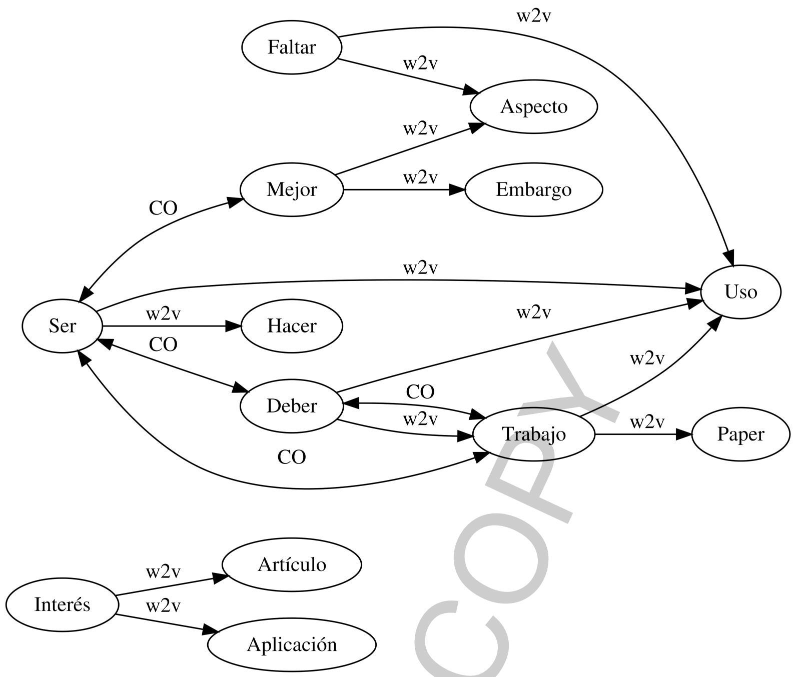

Fig. 3. Graph of relationships for the terms of Table 8.

AUTHOR COPY

this relationship could come from the fact that reviewers are pointing out to the authors that they are lacking in certain aspect or they are not using certain method. This kind of commentary is negative, since it indicates that the article lacks a certain element according to the reviewer, thus it is natural to associate it with the negative class.

- Rule 3 is associated with the positive class, so its relationships must be analyzed with this perspective in mind. In this case we observe a co-occurrence with the verb "ser" (which means "to be" and includes its conjugations) and the word "mejor" (which means "better" and could be both an adverb or an adjective depending on the context, also it should be noted that only the roots of the words are considered, so it could also come from the verb "mejorar" which means "to improve").

  The verb "ser" is related with "uso" ("to use") and "hacer" ("to do"). However, it is hard to find a significant relationship between these words and a verb as common as "to be" and all its conjugations. Thus, we cannot find a meaningful relationship between these three terms. The most important word in this rule is "mejor", which is probably the one that carries the positive implication of this rule, note that the verb "to be" on itself should not express any opinion.

  The word "mejor" is related to "aspecto" ("aspect") and "embargo" (this term comes from "sin embargo", which can be translated as "nevertheless" or something similar). The relationship between "mejor" and "aspecto" could be due to two different motives. The first possibility is that this is indicating that a certain aspect is the best part of an article (since "mejor" could also mean "best" depending on the context) and the other one is that some aspect must be improved (taking the other sense of the word). Given that this rule is positive we shall take the first interpretation as the valid one. As said before, the term "embargo" comes from the adversative clause "sin embargo", so in practical terms this semantic relationship is not useful for our analysis.

  The generated rules show that semantic relationships are not always directly interpretable, meaningful or useful. It is possible that with a larger data set the relationships found with *word2vec* would

Table 9
Results for the five multidomain data sets (*automotive*, *magazines*, *camera & photos*, *apparel* and *toys & games*)

Data sets	NB		MAX	
	TF-IDF-LSA	*RBS-B*	*RBS-X*	*RBS-BX*
Automotive	0.790 ± 0.030	0.739 ± 0.039	0.558 ± 0.111	0.700 ± 0.083
Magazines	0.793 ± 0.026	0.799 ± 0.010	0.747 ± 0.011	0.794 ± 0.010
Camera & photos	0.797 ± 0.012	0.848 ± 0.006	0.716 ± 0.023	0.851 ± 0.007
Apparel	0.709 ± 0.009	0.804 ± 0.003	0.716 ± 0.020	0.830 ± 0.004
Toys & games	0.721 ± 0.008	0.804 ± 0.008	0.621 ± 0.016	0.805 ± 0.007

be more significant.

- Rule 4 is associated with the negative class and consists of three terms related by co-occurrences in the *LHS*. Again, the verb "ser" is too common and general to be considered as important in this analysis, so we shall consider "deber" (it has various meanings, but in this context it probably translates as "must") and "trabajo" ("work" as in the article that is being reviewed or maybe as the verb "to work").

  The word "deber" is related with "uso" and "trabajo", this relationship could be caused by the usage of expressions that indicate that something must be used or that the authors must work on certain aspect of their article. In general, the usage of this word is negative, since it indicates that something must be done to improve the article. For this reason, it is natural to consider this as a negative rule.

  On the other hand, the word "trabajo" is related with "uso" and "paper". The relationship between "trabajo" and "paper" in this case is direct, since in this domain they are basically synonyms. The relationship with "uso" (or any conjugation of the verb "to use") is related with the level co-occurrences that these two words have rather than a direct semantic relationship.

  It should be noted that the term "uso" is related by *word2vec* in various rules, but it does not appear by itself in any of the shown rules. One cause could be that this word is too common to generate an interesting association rule (it would be discarded by Apriori). However, it is related to terms that do generate rules, possibly by appearing in similar contexts.

The quality of the semantic similarity will depend on the training set of the *word2vec* model. In this particular case it is possible that the data set is too small to obtain deep and relevant semantic relationships. It is possible that in a larger data set the results of the RBS-X and RBS-BX methods would be better. Hence, it would be possible to discover more useful relationships between the different terms present in the text.

5.4. *Evaluation on other data sets*

The results obtained for the different multidomain data sets can be seen in Table 9. In general, it can be seen that RBS-X has the lowest performance out of the three RBS approaches. However, as in the Paper Reviews data set case, considering that RBS-X only contains information obtained through association rules derived by semantic similarity rather than co-occurrences it shows good results. The worst performance is found in the *automotive* data set, which could be caused by the small size of this data set, which could prevent the construction of good word embeddings, although this could not be said for the second lowest case given by the *toys & games* data set, which has the most instances of the selected data sets.

The effects of combining the basic rules and the extended rules compared to using the basic rules alone are mixed, in some cases it produces a marginal improvement (see *apparel* that actually shows a slight but statistically significant increase over RBS-B), while in others it actually decreases (*automotive*) or

AUTHOR COPY

stays the same (*magazines*, *camera & photos* and *toys & games*). With respect to the NB baseline there are clear improvements (with RBS-B or RBS-BX) in the case of the *camera & photos* (RBS-B and RBS-BX performed the best in this case), *apparel* (RBS-BX) and *toys & games* (RBS-B and RBS-BX).

During the parameter tuning experiments, it was found that allowing too many rules (by lowering the minimum support and other threshold values) leads to a greater number of extended rules, which in turn hurt the performance of the RBS-X case (and partially in the RBS-BX). However, the basic rules used for RBS-B proved more robust to this effect. Thus, the tuning of the threshold requires taking this performance issue into account (for example, for the Automotive data set, lowering all the thresholds to 0.05 caused RBS-X to have a performance around 25%). It would also seem that the size of the data set has an effect in the parameter choice, however the relationships are still unclear and further study is required.

5.5. *Discussion*

AUTHOR COPY

Extended association rules arise in the context of semantic vector spaces as an idea to exploit the concept of semantic similarity. This allows us to obtain new association rules, which are similar to the original, but are not generated on the basis of direct co-occurrences. Even though this idea is quite general and flexible, the version implemented in this work has some limitations. In particular, the extended rules are only allowed to have one term in their antecedent (*LHS*).

In terms of performance, the extended association rules do not generate the best results, but their results are still competitive, especially considering the fact that these rules were generated through the concept of semantic similarity rather than direct co-occurrences. Nevertheless, the original rules were the best when used with the RBS approach. In fact, by combining the output from RBS and the Scoring Algorithm output from the previous work in [8] in the domain of scientific paper reviews we have obtained a better classification performance in the multiclass case.

As mentioned previously, association rules have not been exploited exhaustively in the field of sentiment analysis. Thus, this work presents a contribution in terms of a new way to apply association rules for sentiment analysis, and in particular, this method allows us to tackle classification tasks using association rules (either by using the original rules or the extended ones).

With regards to the extended rules evaluated in this work, the similar terms have been found applying the *word2vec* model. However, it is possible to apply other approaches, like Latent Semantic Analysis (LSA), to obtain similar terms. In this context, it is important to mention that both empirical and theoretical results [3] suggest that the representation generated by *word2vec* is related to an older model to determine similar terms, known as PMI (*Pointwise Mutual Information*) which is based on the estimation of occurrence probabilities of each term with respect to the presence of other terms [26]. On the other hand, models such as LSA and more advanced variants do not show an evident relationship with PMI. In particular, there is no direct relationship between LSA and *word2vec*, thus we would expect to see that the relationships they find to be different [3]. The fundamental difference between LSA and *word2vec* is that the first is based on word frequency and occurrence counting, while *word2vec* is based on a predictive model [2].

In this case, the similarity criterion would look for latent semantic relationships, which could have a different behaviour than those presented in this work. In spite of this, it should be noted that the underlying hypothesis of the semantic vector space models are general [27], and this implies that both models should, at the very least, share these hypothesis as their basic assumptions.

While the extended rules themselves do not improve the classification performance, it is interesting to note that the classifiers that take the new rules by themselves or in conjunction with the original rules have a similar performance to the classifier that uses the original rules, although a bit worse. We consider this to make sense, considering the fact that the construction of the frequent item sets made by the Apriori algorithm finds terms that have a statistically strong relationship, and on the other hand, the search for similar terms in a semantic vector spaces could be interpreted in the same way, since according to the representation hypothesis [27] the similarity of these terms implies a strong statistical relationship.

Finally, it should be noted that without considering the aspect of classification, the proposal allows finding new association rules. Thus, the AR-SVS method developed can be used independently from the RBS algorithm. Taking this into account, it is possible to apply this proposal on other data sets in text form for exploratory purposes. This work has planted the seeds for future research on extended association rules.

AUTHOR COPY

6. Conclusions

In this work, the concept of extended association rules has been developed, specifically focusing on the case of association rules for classification in sentiment analysis. The results show that the proposed method is competitive, but there are still opportunities of improvement. Association rules have not been exhaustively exploited in the field of sentiment analysis, so this work presents a contribution in terms of a new way of applying them. The classification algorithm based on association rules is a simple approach, and even though it has been shown to be effective, it could still be improved using a different approach.

It is important to note that the proposed method has several adjustable parameters and design decisions that must be made by the final user. The search of an optimal design is of course a pending challenge that must be addressed, possibly through the use of heuristics.

From the obtained results, we consider exploring the relationship that exists between the similarity of terms in the context of a semantic vector space and those generated by frequent item sets generated by the association rules. We also consider interesting studying the relationship that could exist between those two concepts and the latent semantic relationships between terms as described by LSA. In general, we believe that further evaluation using different representation methods for words is required.

Regarding the limitations of the method, having just one term in the LHS of the rule is one of the serious limitations of the method in this proposal. However, as a proof of concept for the extension of association rules, this proposal meets its objective. The construction of extended association rules with more terms in the *LHS* would require the development of new ways of evaluating the rules. The classic metrics of support and confidence would not be sufficient, because these are based on the concept of co-occurrences, while the terms found by the AR-SVS method will not necessarily be able to be evaluated by a co-occurrence criterion, since their relationship can be deeper or more indirect than this.

On the other hand, we must consider the combinatorial explosion that could be caused by attempting to generate rules with the new terms. It is possible that a similar approach to that of the one used in the Apriori algorithm could be useful to solve this problem. We propose as future work the development of a method that allows for the generation of association rules using these semantically related terms.

An initial idea for this is to include a mixed similarity measure that uses classic association rules metrics (such as Support and Confidence) in conjunction with a semantic similarity measure (such as Cosine Similarity). The new evaluation metric could be a multiplication of these values and then an Apriori-like (i.e. Dynamic Programming) approach could be applied.

Finally, the use of vector representations of the terms to extend the rules can be seen as a method to find association rules in itself for data in text form. Future work will have to consider the development of metrics and algorithms that enable the construction of extended rules that allow more than one term both in the consequent and in the antecedent.

References

[1] R. Agrawal, T. Imieliński and A. Swami, Mining association rules between sets of items in large databases, in: *Acm Sigmod Record*, ACM, Vol. 22, 1993, pp. 207–216.

[2] E. Altszyler, M. Sigman and D.F. Slezak, Comparative study of lsa vs word2vec embeddings in small corpora: a case study in dreams database, arXiv preprint arXiv:1610.01520, 2016.

[3] S. Arora, Y. Li, Y. Liang, T. Ma and A. Risteski, A latent variable model approach to pmi-based word embeddings, *Transactions of the Association for Computational Linguistics* **4** (2016), 385–399.

[4] R. Baeza-Yates, A. Moffat and G. Navarro, Searching large text collections, in: *Handbook of Massive Data Sets*, Springer, 2002, pp. 195–243.

[5] J. Blitzer, M. Dredze, F. Pereira et al., Biographies, bollywood, boom-boxes and blenders: Domain adaptation for sentiment classification, in: *ACL*, Vol. 7, 2007, pp. 440–447.

[6] S.K. Jauhar, C. Dyer and E. Hovy, Ontologically grounded multi-sense representation learning for semantic vector space models, in: *Proceedings of the 2015 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, 2015, pp. 683–693.

[7] B. Keith, E. Fuentes and C. Meneses, A hybrid approach for sentiment analysis applied to paper reviews, 2017.

[8] B. Keith, E. Fuentes and C. Meneses, Sentiment analysis and opinion mining applied to scientific paper reviews, Intelligent data analysis, 2019 (in press).

[9] B. Keith Norambuena and C. Meneses Villegas, Extended association rules in semantic vector spaces for sentiment classification, in: Á. Rocha, H. Adeli, L.P. Reis and S. Costanzo, eds, *Trends and Advances in Information Systems and Technologies*, Cham, 2018, pp. 649–658. Springer International Publishing.

[10] D. Kiela and S. Clark, A systematic study of semantic vector space model parameters, in: *Proceedings of the 2nd Workshop on Continuous Vector Space Models and their Compositionality (CVSC)*, 2014, pp. 21–30.

[11] B. Liu, Web data mining: exploring hyperlinks, contents, and usage data, Springer Science & Business Media, 2011.

[12] B. Liu, W. Hsu and Y. Ma, Integrating classification and association rule mining, in: *Proceedings of the Fourth International Conference on Knowledge Discovery and Data Mining*, AAAI Press, 1998, pp. 80–86.

[13] H. Liu and P. Wang, Assessing text semantic similarity using ontology, *JSW* **9**(2) (2014), 490–497.

[14] C. Lorena, I.G. Costa, N. Spolaôr and M.C. de Souto, Automatic parameters selection in machine learning, *Neurocomputing* **75** (2012), 1–2.

[15] Y. Man, O. Yuanxin and S. Hao, Investigating association rules for sentiment classification of web reviews, *Journal of Intelligent & Fuzzy Systems* **27**(4) (2014), 2055–2065.

[16] W. Medhat, A. Hassan and H. Korashy, Sentiment analysis algorithms and applications: A survey, *Ain Shams Engineering Journal* **5**(4) (2014), 1093–1113.

[17] T. Mikolov, I. Sutskever, K. Chen, G.S. Corrado and J. Dean, Distributed representations of words and phrases and their compositionality, in: *Advances in Neural Information Processing Systems*, 2013, pp. 3111–3119.

[18] A. Pawar and V. Mago, Calculating the similarity between words and sentences using a lexical database and corpus statistics, arXiv preprint arXiv:1802.05667, 2018.

[19] T. Pedersen, S.V. Pakhomov, S. Patwardhan and C.G. Chute, Measures of semantic similarity and relatedness in the biomedical domain, *Journal of Biomedical Informatics* **40**(3) (2007), 288–299.

[20] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg et al., Scikit-learn: Machine learning in python, *Journal of Machine Learning Research* **12**(Oct) (2011), 2825–2830.

[21] M.F. Porter, An algorithm for suffix stripping, *Program* **14**(3) (1980), 130–137.

[22] K. Ravi and V. Ravi, A survey on opinion mining and sentiment analysis: Tasks, approaches and applications, *Knowledge-Based Systems* **89** (2015), 14–46.

[23] R. Rehurek and P. Sojka, Software framework for topic modelling with large corpora, in: *In Proceedings of the LREC 2010 Workshop on New Challenges for NLP Frameworks*, Citeseer, 2010.

[24] G. Soğancıoğlu, H. Öztürk and A. Özgür, Biosses: a semantic sentence similarity estimation system for the biomedical domain, *Bioinformatics* **33**(14) (2017), i49–i58.

[25] K. Sugathadasa, B. Ayesha, N. de Silva, A.S. Perera, V. Jayawardana, D. Lakmal and M. Perera, Synergistic union of word2vec and lexicon for domain specific semantic similarity, arXiv preprint arXiv:1706.01967, 2017.

[26] P.D. Turney, Thumbs up or thumbs down? Semantic orientation applied to unsupervised classification of reviews, in: *Proceedings of the 40th Annual Meeting on Association for Computational Linguistics*, ACL '02, Association for Com-

AUTHOR COPY

putational Linguistics, 2002, pp. 417–424.

[27] P.D. Turney, P. Pantel et al., From frequency to meaning: Vector space models of semantics, *Journal of Artificial Intelligence Research* **37**(1) (2010), 141–188.

[28] B.J. Wilson and A.M. Schakel, Controlled experiments for word embeddings, arXiv preprint arXiv:1510.02675, 2015.

[29] M. Yuan, Y.X. Ouyang and Z. Xiong, A text categorization method using extended vector space model by frequent term sets, *Journal of Information Science and Engineering* **29**(1) (2013), 99–114.

AUTHOR COPY