Is a voting approach accurate for opinion mining?

Michel Plantié[1], Mathieu Roche[2], Gérard Dray[1], Pascal Poncelet[1]

[1] Centre de Recherche LGI2P, Site EERIE Nîmes, École des Mines d'Alès - France
{michel.plantie, gerard.dray,pascal.poncelet}@ema.fr
[2] LIRMM, UMR 5506, Univ. Montpellier 2, CNRS France
mathieu.roche@lirmm.fr

Abstract. In this paper, we focus on classifying documents according to opinion and value judgment they contain. The main originality of our approach is to combine linguistic pre-processing, classification and a voting system using several classification methods. In this context, the relevant representation of the documents allows to determine the features for storing textual data in data warehouses. The conducted experiments on very large corpora from a French challenge on text mining (DEFT) show the efficiency of our approach.

1 Introduction

The Web provides a large amount of documents available for the application of data-mining techniques. Recently, due to the growing development of Web 2.0, Web documents as blogs, newsgroups, or comments of movies/books are attractive data to analyze. For example, among the tackled issues addressed by the text mining community, the automatic determination of positive or negative sentiment in these opinion documents becomes a very challenging issue. Nevertheless, the storage of this kind of data in order to apply data-mining techniques is still an important issue and some research works have shown that a data warehouse approach could be particularly well adapted for storing textual data [8]. In this context, data warehouses approaches consider two dimensional tables with the rows representing features of the documents and the columns the set of document domains. For instance, if we consider opinion documents in the movie context, the domain could be the genre of the movie (e.g. fantastic, horror, etc). In this paper we focus on text-mining approaches to find the relevant features (i.e. the first dimension of the data warehouse) to represent a document. Then we deal with the data-mining algorithms in order to classify the opinion documents using these features, i.e. classifying documents according to opinion expressed such as positive or negative mood of a review, the favorable or unfavorable aspect given by an expert, the polarity of a document (positive, neutral, negative) and/or the intensity of each opinion (low, neutral, high), etc.

The rest of the paper is organized as follows. Firstly, we present previous works on opinion mining (section 2), followed by section 3 presenting our approach based on two mains parts: the document representation techniques, and the classification process. This process is based on machine learning and "text-mining" techniques paired with a vote technique. This vote technique (section 3.5) combines several classifiers in a voting system which substantially enhance the results of other techniques, and finally section 4 presents the obtained results.

2 Related work

Classification of opinion documents as blogs or news is more and more addressed by the text mining community [21, 23, 6, 1].

Several methods exist for extracting the polarity of a document. Actually, the opinion polarities are often given by adjectives [23, 6]. The use of adverbs attached to adjectives (for instance, the adverb "very" attached to the adjective "interesting") allows to determine the intensity of phrases (group of words) [1].

For example P. Turney [21] proposes an approach based on the polarity of words in the document. The main idea is to compute correlations between both adjectives in the documents and adjectives coming from a seed set. Two seed sets are considered: positive (e.g. good, nice, ...) and negative (e.g. bad, poor, ...). The associations are calculated by statistical approaches based on the results of *(i)* search engine [20], *(ii)* LSA method [12]. Other approaches using supervised learning methods allow to define polarity degrees (positive, negative, objective) to the Wordnet lexical resource [14]. Besides many studies have shown that the grammatical knowledge are relevant for opinion mining approaches.

To calculate the polarity of words, supervised or unsupervised methods can be used to predict the polarity of a document. The supervised approaches have the advantage to automatically learn relevant features (words, phrases) to predict a domain opinion. It's important to extract domain dependent characteristics. The same word or group of words may be positive in a domain and negative for another domain: for example, the adjective "commercial" is positive for economic documents but expresses a negative sentiment to characterize a movie. Thus, these supervised methods are often used in national [7] and international [23] opinion mining challenges.

When we have well structured opinion corpora, machine learning techniques (based on training models on these corpora), outperform results. Methods based on individual word search cannot extract complete information on opinion texts and so produce less efficient classification results.

This paper proposes a new method called "COPIVOTE" (C*lassification of* OPI*nion documents by a* VOTE *system*) to classify document according to the expressed opinions. We thus define a new architecture based on coupling several techniques including a voting system adapted to each domain corpus in order to get better results. The main originality of our approach relies on associating several techniques: extracting more information bits via specific linguistic techniques, space reduction mechanisms, and moreover a voting system to aggregate the best classification results.

3 The Copivote approach

For efficiency reasons our method does not try to search each opinion related word. Statistic techniques are able to produce a more comprehensive document representation. This characteristic allows us to manage the large complexity and the subtleties in opinion expression contained in the language as explained in subsection 3.2. The specificity of our approach lies on pre and post treatments adapted to the corpus types. However, the overall process presented in this paper may also be adapted to other kind of corpora.

3.1 Overall process presentation

Our method uses four main steps to classify documents according to opinion:

- **Linguistic treatments for vector space model representation:** In this step we use linguistic analysis adapted to opinion texts.
- **Vector space model reduction:** in order to get better performances and limited processing time we simplify the vector space model.
- **Classification:** This stage uses classifiers to compute model and to classify new texts
- **Classifier voting system:** this phase gather the classifiers results for one document and aggregate them in one answer for each document.

3.2 Linguistic treatments for vector space model representation

Our first step is to apply several linguistic pre-treatments. The first one is based on the extraction of all linguistic units (lemmatised words or lemmas) used for document representation. For example the conjugated verb "presents" is replaced by its lemma: the infinitive verb "present".

We then eliminate words having grammar categories with a low discriminative power with regard to opinion mining: undefined articles and punctuation marks. In our approach, we keep lemmas associated with almost all grammatical categories (as adverbs) in order to specifically process opinion documents. Since we are on a machine learning approach based on corpora, we are able to use all information of documents. Each kind of word may contain opinion discriminative information even very slight . Further more we extract known expressions. extracting expressions and keeping almost all words will enhance the classification results.

For our purpose, we will call "index", the list of lemmas worked out for each corpus. Each corpus is represented by a matrix in compliance with the Salton vector space model representation [19]. In this representation, each row is associated to each document of the corpus and each column is associated with each lemma. Each matrix cell represents the number of occurrences for the considered lemma in the considered document.

In our approach, the whole set of documents of a corpus and therefore the associated vectors are used as training set.

3.3 Vector space model reduction (Index reduction)

The Vector space defined by the whole set of lemmas of the training corpus is very important in dimension. We thus perform an index reduction for each corpus. We use the method presented by Cover, which measures the mutual information between each vector space dimension and classes [5]. This method, explained in depth in [16], measures the interdependence between words and the document classifying categories by computing the entropy difference between the category entropy and the studied dimension (key word) entropy of the vector space. If the difference is high, then the discriminative information quantity of this word is high, and therefore the importance of this word is high in the categorization task.

Once the indexes are computed, we consider each computed key word in each index as the new dimensions of the new representation vectors space for each corpus of documents. The new vector spaces have a reduced number of dimensions. These new computed vectors are called: "reduced" vectors. As it is shown in [16], this reduction helps a lot to significantly improve the quality of results and drastically lower the computing times.

3.4 Use of bigrams

In this approach we take into account words to compute the document vectors and we also add bigrams of the corpora (groups of two words). Only bigrams containing special characters are rejected (mathematical characters, punctuation, etc). This richer document representation allows us to extract information more adapted to opinion corpora. As an example, in the corpora we have used for experiments, bigrams like "not convincing", "better motivate", "not enough" are groups of words much more expressive of opinions than each word separately.

This enriched document representation using bigrams improve results, as we will see in section 4. In addition to the quality of document representation that improves the classification tasks, taking into account several classifiers (see next section) remains crucial to get good quality results.

3.5 Classifier voting system

To improve the general method for classifying opinion documents, we have worked out a voting system based on several classifiers. Our vote method named CopivoteMono (c*lassification of* opi*nion documents by a* vote *system with* Mono*grams*) and CopivoteBi (c*lassification of* opi*nion documents by a* vote *system with* Bi*grams*) when bigrams are used, uses the specific data related to opinion documents presented in the previous subsections.

The voting system is based on different classification methods. We use three main classification methods presented afterwards. Several research works use voting of classifiers, Kittler and Kuncheva [10, 11] describe several ones. Rahman [18] shows that in many cases the quite simple technique of majority vote is the most efficient one to combine classifiers. Yaxin [2] compares vote techniques with summing ones. Since the probability results obtained by individual classifiers are not commensurate, vote techniques based on the final result of each classifiers is the most adequate to combine very different classifier systems.

In our approach we use four different procedures of vote:

- **Simple majority vote:** the class allocation is computed considering the majority of classifiers class allocation.
- **Maximum choice vote (respectively minimum):** the class allocation is computed as the classifier that gives the highest probability (respectively the lowest). In that situation, the probabilities expressed by each classifier must be comparable.
- **Weighted sum vote:** for each document d(i) and for each class c(j) the average of probabilities avg(i,j) is computed and the class allocated to the document i is based on the greatest average max(avg(i,j)).
- **Vote taking into account F-score, and/or recall and/or precision:** The classifier, for a given class, is elected if it produces the best result in F-score (and/or recall and/or precision) for this class. These evaluation measures (F-score, recall, precision) are defined below.

Precision for a given class i corresponds to the ratio between the documents rightly assigned to their class i and all the documents assigned to the class i. Recall for a given class i corresponds to the ratio between the documents rightly assigned to their class i and all the documents appertaining to the class i. Precision and recall may be computed for each of the classes. A trade-off between recall and precision is then computed: the F-score (F-score is the harmonic average of recall and precision).

3.6 Classification

We adapt the classification method for each training set. We have kept the most competitive classification method for a given corpus. The results are evaluated using the cross validation method on each corpus, based on the precision, recall and F-score measures.

Having described the vote system, we will now briefly present the different classification methods used by Copivotemono and Copivotebi hereafter. A more detailed and precise description of these methods is given in [4].

- **Bayes Multinomial.** The Bayes Multinomial method [22] is a classical approach in text categorization; it combines the use of the probability well known Bayes law and the multinomial law.

- **Support Vector Machine (S.V.M.).** The SVM method [9, 17] draws the widest possible frontier between the different classes of samples (the documents) in the vector space representing the corpus (training set). The support vectors are those that mark off this frontier: the wider the frontier, the lower the classification error cases.

- **RBF networks (Radial Basis Function).** RBF networks are based on the use of neural networks with a radial basis function. This method uses a "k-means" type clustering algorithm [13] and the application of a linear regression method. This technique is presented in [15].

Our contribution relies on the association of all the techniques used in our method. First the small selection in grammatical categories and the use of bigrams enhance the information contained in the vector representation, then the space reduction allows to get more efficient and accurate computations, and then the voting system enhance the results of each classifiers. The overall process comes to be very competitive.

4 Results

4.1 Corpora description

The third edition of the French DEFT'07 challenge (http://deft07.limsi.fr/) focused on specifying opinion categories from four corpora written in French and dealing with different domains.

- **Corpus 1:** Movie, books, theater and comic books reviews. Three categories: good, average, bad,
- **Corpus 2:** Video games critics. Three categories: good, average, bad,
- **Corpus 3:** Review remarks from scientific conference articles. Three categories: accepted, accepted with conditions, rejected,
- **Corpus 4:** Parliament and government members speeches during law project debates at the French Parliament. Two categories: favorable, not favorable,

These corpora are very different in size, syntax, grammar, vocabulary richness, opinion categories representation, etc. For example, table 1 presents the allocation of classes for each corpus. This table shows that corpus 4 is the largest, and corpus 3 is the smallest. On the other hand, we may find similarities between the corpora (for example, the first class is smaller for the 3 first corpora), Table 1 shows important differences with respect to the number of documents in each class.

Classes	Corpus 1	Corpus 2	Corpus 3	Corpus 4
Class 1	309	497	227	10400
Class 2	615	1166	278	6899
Class 3	1150	874	376	∅
Total	2074	2537	881	17299

Table 1. Allocation of the corpus classes for the DEFT'07 challenge.

Table 2 shows the vector space dimensions reduction associated to each corpus. This operation drastically decreases the vector spaces for all the DEFT07 challenge corpora with a reduction percentage of more than 90%.

Corpus	**Initial Number of linguistic units**	**Number of linguistic units after reduction**	**Reduction percentage**
Corpus 1	36214	704	98.1%
Corpus 2	39364	2363	94.0%
Corpus 3	10157	156	98.5%
Corpus 4	35841	3193	91.1%

Table 2. Number of lemmas for each corpus before and after reduction.

4.2 detailed Results

Table 4 shows that the vote procedures globally improve the results. Firstly, all the vote methods (see section 3) give rise to the same order of improvement even if some results of the ”weighted sum vote” are slightly better (also called ”average vote”). Secondly, the bigram representations associated to vote methods (CopivoteBi) globally improved the results compared to those obtained without using bigrams (CopivoteMono).

Table 3 shows the classification methods used in our vote system. We notice that the Bayes Multinomial classifier is very competitive with a very low computing time. Almost every time the SVM classifier gives the best results. The RBF Network classifier gives disappointing results.

Table 4 shows the results expressed with the F-score measure (globally and for each class) obtained by the cross validation process on each training set. These results point out the classes that may or may not be difficult to process for each corpus. For example, we notice in table 4 that the corpus 2 gives well balanced results according to the different classes. On the contrary, the neutral class (class 2) of corpus 1 leads to poor results meaning that this class is not very discriminative. This may be explained by the nearness of the vocabulary used to describe a film or a book in a neutral way comparatively to a more clear-cut opinion.

Table 5 shows the results associated with the test corpora given by the DEFT’07 challenge committee. Tables 4 and 5 give very close results, showing that the test corpus is a perfectly representative sample of the training data. Table 5 shows that only one corpus gives disappointing results: corpus 3 (reviews of conference articles). This may be explained by the low number of documents in the training set and by the noise contained in the data (for example, this corpus contains a lot of spelling errors). The vector representation of the documents is then poor and noise has a bad effect on the classification process. The bigram representation does not provide any improvement for this corpus. More effort should be made on linguistic pre-treatment on this corpus in order to improve results.

The outstanding results for corpus 4 (parliamentary debates) may be explained by its important size that significantly support the statistic methods used. With this corpus, the vote system improves a lot the results obtained by each of the classifiers (see table 3). We may notice that the F-score value exceeds for more than 4% the best score of the DEFT'07 challenge.

In table 5, we compare our results with the DEFT'07 challenge best results. It shows that our results was of the same order or even slightly better with CopivoteBi .

Corpus	**SVM**	**RBF-Network**	**Naive Bayes Mult.**	Copivote	CopivoteBi
Corpus 1	61.02%	47.15%	59.02%	60.79%	61.28%
Corpus 2	76.47%	54.75%	74.16%	77.73%	79.00%
Corpus 3	50.47%	X	50.07%	52.52%	52.38%
Corpus 4	69.07%	61.79%	68.60%	74.15%	75.33%

Table 3. F-score average for the different methods used in Copivote on the test corpus .

Corpus	Copivote				CopivoteBi			
	class 1	**class 2**	**class 3**	**global**	**class 1**	**class 2**	**class 3**	**global**
Corpus 1	64.6%	42.7%	75.2%	60.8%	64.8%	43.8%	75.3%	61.3%
Corpus 2	74.9%	76.9%	82.6%	78.1%	75.8%	79.1%	82.4%	79.1%
Corpus 3	52.3%	43.0%	62.7%	52.7%	47.9%	45.0%	64.48%	52.4%
Corpus 4	80.0%	68.5%	∅	74.2%	81.2%	69.6%	∅	74.2%

Table 4. F-score per class and global, based on Learning corpus (cross validation).

Corpus	**Vote type**	Copivote	CopivoteBi	**Best submission of DEFT07**
Corpus 1	Minimum	60.79%	61.28%	60.20%
Corpus 2	Average	77.73%	79.00%	78.24%
Corpus 3	Minimum	52.52%	52.38%	56.40%
Corpus 4	Average	74.15%	75.33%	70.96%
Total		**66.30%**	**67.00%**	**66.45%**

Table 5. F-score of Test corpus of DEFT07.

4.3 Discussion: The use of linguistic knowledge

Before text classification, we also tried a method to improve linguistic treatments. Specific syntactic patterns may be used to extract nominal terms from tagged corpus [3] (*e.g.* Noun Noun, Adjective Noun, Noun Preposition Noun, etc). In addition to nominal terms, we extracted adjective and adverb terms well adapted to opinion data [23, 6, 1]. For instance the "Adverb Adjective" terms are particularly relevant in opinion corpora [1]. For example, *still insufficient*, *very significant*, *hardly understandable* extracted from the scientific reviews corpus (corpus 3) of the DEFT'07 challenge may be discriminative in order to classify opinion documents. We used the list of these extracted terms to compute a new index for vector representation. We obtained poor results.

Actually our approach CopivoteBi takes into account words and all the bigrams of the corpus to have a large index (before its reduction presented in section 3.3). Besides the number of bigrams is more important without the application of linguistic patterns. Then our CopivoteBi approach combining a voting system and an expanded index (words and all the bigrams of words) can explain the good experimental results presented in this paper.

5 Conclusion and future work

This paper lay down a new approach based on combining text representations using key-words associated with bigrams while combining a vote system of several classifiers. The results are very encouraging with a higher F-score measure than the best one of the DEFT'07 challenge. Besides, our results show that the relevant representation of documents for datawarehouses is based on words and bigrams after the application of linguistic and index reduction process.

In our future work, we will use enhanced text representations combining key-words, bigrams and trigrams which may still improve the obtained results. We also want to use vote systems based on more classifiers. Finally, a more general survey must be undertaken by using other kinds of corpora and moreover textual data in different languages.

References

1. F. Benamara, C. Cesarano, A. Picariello, D. Reforgiato, and V.S. Subrahmanian. Sentiment analysis: Adjectives and adverbs are better than adjectives alone. In *Proceedings of ICWSM conference*, 2007.
2. Y. Bi, S. McClean, and T. Anderson. Combining rough decisions for intelligent text mining using dempster's rule. *Artificial Intelligence Review*, 26(3):191–209, 2006.
3. E. Brill. Some advances in transformation-based part of speech tagging. In *AAAI, Vol. 1*, pages 722–727, 1994.
4. A. Cornuéjols and L. Miclet. *Apprentissage artificiel, Concepts et algorithmes.* Eyrolles, 2002.

5. T.M. Cover and J.A. Thomas. *Elements of Information Theory.* John Wiley, 1991.
6. A. Esuli and F. Sebastiani. PageRanking wordnet synsets: An application to opinion mining. In *Proceedings of the 45th Annual Meeting of the Association for Computational Linguistics (ACL'07), Prague, CZ*, pages 424–431, 2007.
7. C. Grouin, J-B. Berthelin, S. El Ayari, T. Heitz, M. Hurault-Plantet, M. Jardino, Z. Khalis, and M. Lastes. Présentation de deft'07 (défi fouille de textes). In *Proceedings of the DEFT'07 workshop, Plate-forme AFIA, Grenoble, France*, 2007.
8. Himanshu Gupta and Divesh Srivastava. The data warehouse of newsgroups. In *In Proceedings of the Seventh International Conference on Database Theory, LNCS*, pages 471–488, 1999.
9. T. Joachims. Text categorisation with support vector machines: Learning with many relevant features. In *Proceedings of the European Conference on Machine Learning (ECML)*, pages 137–142, 1998.
10. J. Kittler, M. Hatef, R.P.W. Duin, and J. Matas. On combining classifiers. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 20(3):226–239, 1998.
11. L.I. Kuncheva. *Combining Pattern Classifiers: Methods and Algorithms.* John Wiley and Sons, Inc., 2004.
12. T. Landauer and S. Dumais. A solution to plato's problem: The latent semantic analysis theory of acquisition, induction and representation of knowledge. *Psychological Review*, 104(2):211–240, 1997.
13. J.B. MacQueen. Some methods for classification and analysis of multivariate observations. In *Proceedings of the 5th Berkeley Symposium on Mathematical Statistics and Probability.*, 1967.
14. G.A. Miller, R. Beckwith, C. Fellbaum, D. Gross, and K.J. Miller. Introduction to WordNet: an on-line lexical database. *International Journal of Lexicography*, 3(4):235–244, 1990.
15. J. Parks and I.W. Sandberg. Universal approximation using radial-basis function networks. *Neural Computation*, 3:246–257, 1991.
16. M. Plantié. *Extraction automatique de connaissances pour la dcision multicritre.* PhD thesis, École Nationale Supérieure des Mines de Saint Etienne et de l'Université Jean Monnet de Saint Etienne, Nîmes, 2006.
17. J. Platt. Machines using sequential minimal optimization. In *In Advances in Kernel Methods - Support Vector Learning: B. Schoelkopf and C. Burges and A. Smola, editors.*, 1998.
18. A.F.R. Rahman, H. Alam, and M.C. Fairhurst. *Multiple Classifier Combination for Character Recognition: Revisiting the Majority Voting System and Its Variation*, pages 167–178. 2002.
19. G. Salton, C.S. Yang, and C.T. Yu. A theory of term importance in automatic text analysis. *Journal of the American Society for Information Science*, 26:33–44, 1975.
20. P.D. Turney. Mining the Web for synonyms: PMI–IR versus LSA on TOEFL. In *Proceedings of ECML conference, LNCS, Spinger-Verlag*, pages 491–502, 2001.
21. P.D. Turney and M. Littman. Measuring praise and criticism: Inference of semantic orientation from association. *ACM Transactions on Information Systems*, 21(4):315–346, 2003.
22. Y. Wang, J. Hodges, and B. Tang. Classification of web documents using a naive bayes method. In *Proceedings of the 15th IEEE International Conference on Tools with Artificial Intelligence*, pages 560–564, 2003.
23. H. Yang, L. Si, and J. Callan. Knowledge transfer and opinion detection in the trec2006 blog track. In *Notebook of Text REtrieval Conference*, 2006.