Expert Systems with Applications 40 (2013) 3934–3942

Contents lists available at SciVerse ScienceDirect

Expert Systems with Applications

journal homepage: www.elsevier.com/locate/eswa

Sentiment polarity detection in Spanish reviews combining supervised and unsupervised approaches

María-Teresa Martín-Valdivia *, Eugenio Martínez-Cámara, Jose-M. Perea-Ortega, L. Alfonso Ureña-López

SINAI Research Group, Computer Science Department, University of Jaén, 23071 Jaén, Spain

ARTICLE INFO

Keywords:
Sentiment polarity detection
Multilingual opinion mining
Spanish review corpus
SentiWordNet
Metaclassifiers
Stacking algorithm
Voting system

ABSTRACT

Sentiment polarity detection is one of the most popular tasks related to Opinion Mining. Many papers have been presented describing one of the two main approaches used to solve this problem. On the one hand, a supervised methodology uses machine learning algorithms when training data exist. On the other hand, an unsupervised method based on a semantic orientation is applied when linguistic resources are available. However, few studies combine the two approaches. In this paper we propose the use of meta-classifiers that combine supervised and unsupervised learning in order to develop a polarity classification system. We have used a Spanish corpus of film reviews along with its parallel corpus translated into English. Firstly, we generate two individual models using these two corpora and applying machine learning algorithms. Secondly, we integrate SentiWordNet into the English corpus, generating a new unsupervised model. Finally, the three systems are combined using a meta-classifier that allows us to apply several combination algorithms such as voting system or stacking. The results obtained outperform those obtained using the systems individually and show that this approach could be considered a good strategy for polarity classification when we work with parallel corpora.

1. Introduction

Opinion Mining (OM), also known as Sentiment Analysis (SA) is a challenging task that combines data mining and Natural Language Processing (NLP) techniques in order to computationally treat subjectivity in textual documents. This new area of research is becoming more and more important mainly due to the growth of social media where users continually generate contents on the web in the form of comments, opinions, emotions, etc. There are several issues related to OM like subjectivity detection, opinion extraction, irony detection and so on (Pang and Lee, 2008). However, perhaps the most widely-studied task is sentiment polarity classification. This task aims to determine which is the overall sentiment-orientation (positive or negative) of the opinions contained within a given document. The document contains subjective information such as product reviews or opinionated posts in blogs.

Although different approaches have been applied to polarity classification, the mainstream basically consists of two major methodologies. On the one hand, the Machine Learning (ML) approach (also known as the supervised approach) is based on using a collection of data to train the classifiers (Pang, Lee, & Vaithyanathan, 2002). On the other hand, the approach based on Semantic Orientation (SO) does not need prior training, but takes into account the positive or negative orientation of words (Turney, 2002). This method, also known as the unsupervised approach, makes use of lexical resources like lists of opinionated words, lexicons, dictionaries, etc. Both methodologies have their advantages and drawbacks. For example, the ML approach depends on the availability of labeled data sets (training data), which in many cases are impossible or difficult to achieve, partially due to the novelty of the task. On the other hand, the SO strategy requires a large amount of linguistic resources which generally depend on the language, and often this approach obtains lower recall because it depends on the presence of the words comprising the lexicon in the document in order to determine the orientation of opinion. In order to overcome the weaknesses of both approaches, we have performed several experiments, combining ML and SO through different strategies.

Most of the studies on polarity classification only deal with English documents, perhaps due to the lack of resources in other languages. Despite the fact that Chinese, Arabic and Spanish are currently among the top ten languages most used on the Internet according to the Internet World State rank[1], there are very few resources for managing sentiments or opinions in these languages. However, people increasingly comment on their experiences, opinions, and points of views not only in English but in many other languages. Consequently, the management and study of subjectivity and sentiment analysis in languages other than English is a growing need. The work presented herein is mainly motivated by the

* Corresponding author. Tel.: +34 953212898.
E-mail addresses: maite@ujaen.es (M.-T. Martín-Valdivia), emcamara@ujaen.es (E. Martínez-Cámara), jmperea@ujaen.es (Jose-M. Perea-Ortega), laurena@ujaen.es (L.A. Ureña-López).

0957-4174/$ - see front matter © 2012 Elsevier Ltd. All rights reserved.
http://dx.doi.org/10.1016/j.eswa.2012.12.084

need to develop polarity detection systems in languages other than English. Specifically, we focus on Spanish sentiment polarity detection and we use a corpus of movie reviews written in Spanish called MuchoCine corpus (MC corpus) (Cruz, Troyano, Enriquez, & Ortega, 2008).

According to Mihalcea, Banea, and Wiebe (2007), there are two main approaches in the context of multilingual sentiment analysis. The first one is a Lexicon-based approach, where a target-language subjectivity classifier is generated by translating an existing lexicon into another idiom. The second one is a Corpus-based approach, where a subjectivity-annotated corpus for the target language is built through projection, training a statistical classifier on the resulting corpus. In this paper we follow this second approach and we generate an English parallel corpus called MCE (MuchoCine English version). The MCE corpus is built by applying automatic machine translation techniques to the Spanish MC corpus.

In this paper we apply the two approaches, ML and SO, over the parallel corpus in Spanish and English (MC and MCE). We have generated three initial models that finally we combined using different strategies in order to improve the performance of the final system. The first one, MC-ML, applies ML over the original MC corpus in Spanish. The second one, MCE-ML, follows the same strategy as MC-ML but using MCE parallel corpus in English. And the third one, MCE-SO, uses the SentiWordNet (Baccianella, Esuli, & Sebastiani, 2010) resource over the MCE corpus in order to integrate lexical knowledge and generate an unsupervised polarity classifier. Finally, the output of these three individual models has been combined using different approaches. On the one hand, we have applied a voting system based on the majority rule, and the other we have assessed four algorithms as meta-classifiers following a strategy of stacking. The results obtained are very promising and show that our proposal is a good strategy when we need to deal with languages that have few lexical resources for tackling the polarity classification problem.

The rest of the paper is organized as follows: the next section presents work related to polarity detection dealing with languages other than English and multilingual opinion mining. Section 3 presents the approach proposed in this work. Section 4 describes the different resources used in our experiments including the MC and MCE corpora and SentiWordNet. The different experiments carried out and the results obtained are expounded in Section 5. Finally, the main conclusions and ideas for further work are expounded in Section 6.

2. Related work

As we have already commented, many of the research papers on sentiment analysis have been applied to English, but work on other languages is still growing. There are some interesting papers that have studied the problem using non-English collections including German, Chinese or Arabic.

Kim and Hovy (2006) compared opinion expression between an aligned corpus of emails in German and English. They developed two models: The first one translates German emails into English and then applies opinion-bearing words. The second one translates English opinion-bearing words into German and then analyzes the German emails using the German opinion-bearing words. The results showed that Model 1 works slightly better than Model 2. Following this work, Denecke (2008) worked on German comments collected from Amazon. These reviews were translated into English using standard machine translation software. Then the translated reviews were classified as positive or negative, using three different classifiers: LingPipe, SentiWordNet with classification rule, and SentiWordNet with machine learning.

Tan and Zhang (2008) were among the first researchers to study opinion mining in Chinese. They carried out a widely experimental revision using lots of different models. Zhang, Zeng, Li, Wang, and Zuo (2009) applied Chinese sentiment analysis on two datasets. In the first one euthanasia reviews were collected from different web sites, while the second dataset was about six product categories collected from Amazon (Chinese reviews). They proposed a rule-based approach including two phases: firstly, determining each sentence's sentiment based on word dependency, and secondly, aggregating sentences in order to predict the document sentiment. Wan (2009) studied the sentiment polarity identification of Chinese product reviews using a semantic orientation. He makes use of bilingual knowledge including both Chinese resources and English resources. The corpus is composed of 886 Chinese documents that were translated into English by using Google Translate and Yahoo Babel Fish. In addition, the approach used ensemble methods to combine the individual results over Chinese and English datasets. The results for the combination methods improved the performance of individual results.

Ghorbel and Jacot (2010) used a corpus with movie reviews in French. They applied a supervised classification combined with SentiWordNet in order to determinate the polarity of the reviews. French is also managed in Balahur and Turchi (2012), along with Spanish and German. Different machine translation systems and meta-classifiers were tested in order to demonstrate that multilingual SA using these techniques is comparable to the English performance.

In Rushdi-Saleh, Martín-Valdivia, Ureña-López, and Perea-Ortega (2011a) a corpus of movies reviews in Arabic annotated with polarity is presented and several experiments using machine learning techniques are performed. Subsequently, they generated the parallel EVOCA corpus (English version of OCA) by translating the OCA corpus automatically into English. The results showed that, although the experiments with EVOCA are worse than OCA, the results are comparable to other English experiments, since the loss of precision due to the translation is very slight (Rushdi-Saleh, Martín-Valdivia, Ureña-López, & Perea-Ortega, 2011b).

Regarding opinion mining focused on Spanish, there are also some remarkable studies. For example, Banea, Mihalcea, Wiebe, and Hassan (2008) proposed several approaches to cross lingual subjectivity analysis by directly applying the translations of opinion corpus in English to training an opinion classifier in Romanian and Spanish. This work showed that automatic translation is a viable alternative for the construction of resources and tools for subjectivity analysis in a new target language. Boldrini, Balahur, Martínez-Barco, and Montoyo (2009) aimed to build up a corpus with a fine-gained annotation scheme for the detection of subjective elements. The data were collected manually from 300 blogs in three different languages: Spanish, Italian and English. (Brooke et al., 2009) presented several experiments dealing with Spanish and English resources. They conclude that although the ML techniques can provide a good baseline performance, it is necessary to integrate language-specific knowledge and resources in order to achieve an improvement. They proposed three approaches: the first one uses Spanish resources generated manually and automatically. The second one applies ML to a Spanish corpus. The last one translates the Spanish corpus into English and then applies the SO-CAL (Semantic Orientation CALculator), a tool developed by themselves (Taboada, Voll, & Brooke, 2008).

Finally, the MuchoCine (MC) corpus used in this work was manually recollected by Cruz et al. (2008). This corpus contains annotated Spanish movie reviews from the MuchoCine website[1]. The MC corpus was generated in order to develop a sentiment polarity

[1] http://www.muchocine.net.

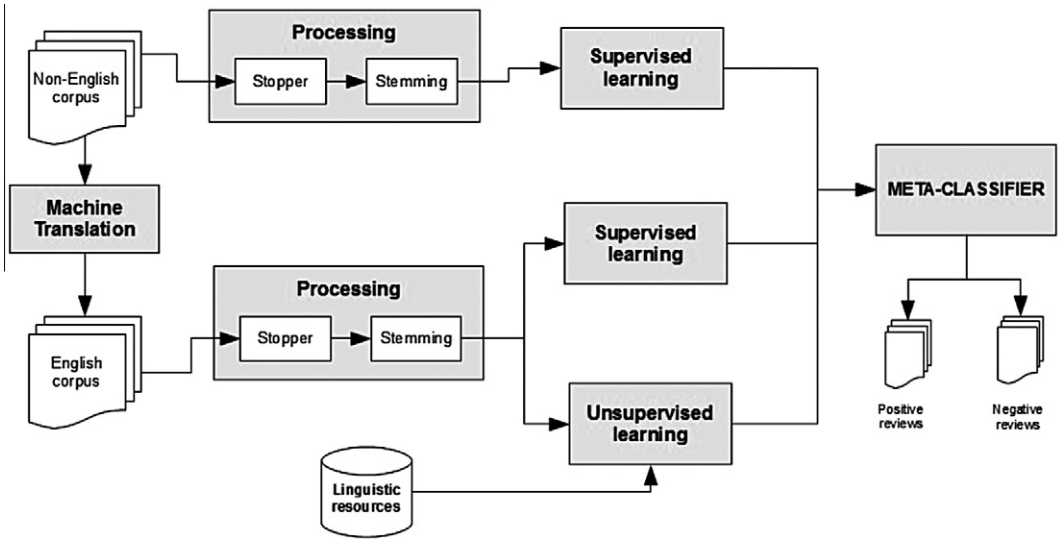

Fig. 1. Overview of the approach proposed.

classifier based on semantic orientation. On the contrary, Martínez-Cámara, Martín-Valdivia, and Ureña-López (2011) applied the supervised approach to the MC corpus using different ML algorithms (SVM, NB, BBR, KNN, C4.5). The results are much better than those obtained with the unsupervised approach proposed by Cruz et al. (2008).

3. Meta-classifier: an approach based on the combination of supervised and non-supervised models

The aim of the approach proposed in this study is to improve the polarity classification of the opinions provided by a corpus whose documents are in a language other than English. The main idea is to translate the original corpus into English and work with parallel corpora, generating several learning models by using both corpora. Furthermore, as we have a corpus translated into English we can make use of a semantic resource for opinion mining tasks such as SentiWordNet in order to apply a non-supervised approach to that corpus. In this way, the models (supervised and unsupervised) generated using the parallel corpora can be combined in a meta-classifier that could apply different algorithms to establish the final polarity classification. Fig. 1 illustrates this approach.

One of the advantages of our architecture is its modularity, allowing the use of different supervised algorithms for both corpora (original and translated) and even in the meta-classifier for combining previous generated models. As can be seen in Fig. 1, we apply a processing to the corpora, which usually consists of a stemming process for extracting the root of each word after removing the words without semantic meaning (*stopwords*).

Once the corpora are processed, we generate the learning models that will be used later in the meta-classifier. This supervised method has been applied to both corpora and it allows the use of different learning algorithms such as SVM or NB. However, the unsupervised approach is applied solely to the translated corpora because the linguistic resources, such as SentiWordNet or Word Affect, are available in English only.

Finally, the meta-classifier process combines several features from the supervised and unsupervised models previously generated, and also allows us to apply different combination algorithms, as explained in Section 5.2.

In order to improve the results of supervised methods and to gain some independence from the domain it is necessary to use linguistic and semantic resources. There are very few linguistic and semantic resources in Spanish, so multilingual techniques should be utilized to improve the results in sentiment analysis of Spanish documents. One of the simplest multilingual techniques is to translate the text in the source language to a target language (usually English), and then apply the linguistic resources to the translated version. In our case the source language is Spanish and the target one is English, so we translated the MC corpus to English. As can be seen in Fig. 1, the availability of the English version of MC (MCE) allows us to use any English linguistic resource as SentiWordNet to carry out the unsupervised learning experiments.

4. Resources used for the experiments

Although sentiment analysis is not considered a recent research task, most of the NLP resources available on the Internet are based on English. The lack of resources based on Spanish makes research related to sentiment analysis of Spanish documents particularly difficult.

For the experiments carried out in this study we have used a corpus called MuchoCine (MC) composed of opinions in Spanish. Next the main features of this corpus are described along with the procedure conducted for its machine translation. Finally we present a brief description of SentiWordNet, the resource employed in the unsupervised experiments.

4.1. The MC corpus

In order to demonstrate the effectiveness of our approach, we have selected the MuchoCine corpus (MC), available for the SA research community in Spanish[2]. It has been described in Cruz et al. (2008) and widely used in different studies such as del-Hoyo, Hupont, Lacueva, and Abadía (2009), Barreiro and Gonçalo (2011), Malvar-Fernández and Pichel-Campos (2011), Martínez-Cámara, Martín-Valdivia and Ureña-López (2011) and Martínez Cámara, Martín-Valdivia, Perea-Ortega, and Ureña-López (2011).

The corpus consists of 3878 movie reviews collected from the MuchoCine website. The reviews are written by web users instead of professional film critics. This increases the difficulty of the task because the sentences found in the documents may not always be grammatically correct, or they may include spelling mistakes or

[2] http://www.lsi.us.es/~fermin/corpusCine.zip.

Table 1
Rating distribution.

Rating	# Reviews
1	351
2	923
3	1253
4	890
5	461
Total	3875

Table 2
Binary classification of the MC corpus.

Classes	# Reviews
Positive	1274
Negative	1351
Total	2625

informal expressions. The corpus contains about 2 million words and an average of 546 words per review.

The opinions are rated on a scale from 1 to 5. One point means that the movie is very bad and 5 means very good. Films with a rating of 3 can be considered as "neutral", which means that the user considers the film is neither bad nor good. Table 1 shows the number of reviews per rating.

In our experiments we have neglected the neutral examples. In this way, opinions rated with 3 were not considered, and the opinions with ratings of 1 or 2 were considered as positive and those with ratings of 4 or 5 were considered as negative. Table 2 shows the class distribution of the binary classification of MC.

4.2. The MCE corpus

The MuchoCine English corpus (MCE) is the version of MC translated into English[3]. It was generated by applying a machine translation process. Several tools were tested in carrying out this process. Next we explain the main issues that we took into account in choosing the selected tool.

First of all we were interested in free tools. Since we had to translate a lot of documents, the possibility of using free web services was unfeasible. Therefore, powerful web services such as Worldlingo[4], Babelfish[5] or Google Translate[6] were discarded. Then we decided to look for free APIs that would allow us to translate the MC corpus easily.

The Google Translate API[7] v1 was no longer available from December 1, 2011 and was replaced by Google Translate API v2. Google Translate API v1 was officially deprecated on May 26, 2011. The decision to deprecate the API and replace it with the paid service was made due to the substantial economic burden caused by extensive abuse. We therefore discarded the Google Translate API because it is only available as a paid service.

Among the most interesting alternatives we found two main tools. One of them was iTranslate4,[8] a European project supported by the EU Competitiveness and Innovation Framework Programme (ICT PSP). The project has developed the first European web portal providing free online translation across all European languages. Due to the competitive nature of the approach used, this service guarantees that users receive the best quality translator available. Moreover, iTranslate4 has unified leading European machine translation companies such as SYSTRAN[9] or PROMT[10] into one consortium. They provide a common API for translation services with the drawback that only 30,000 characters per API-key can be translated. After that one must to buy a license in order to use this API.

The other alternative was to use the Microsoft Translator API,[11] formerly known as the Bing Translator. Specifically we have used the Microsoft Translator Java API[12] in order to carry out the automatic translation process of MC. First you must subscribe to the Microsoft Translator API on the Windows Azure Marketplace[13] and then register your application. Basic subscriptions, up to 2 million characters a month, are free. Once you reach this maximum number of characters translated in a session, you have to renew the subscription.

The last issue we had to take into account was related to another restriction that the Microsoft Translator API has. This is the limitation of textual blocks with a maximum size of 10,240 bytes per query. Therefore during the translation process it was necessary to divide the opinions from MC into blocks of this size. This split was carried out considering complete words and sentences, always with a size in bytes lower than the limit established by the API.

4.3. SentiWordNet

Regarding the unsupervised approach applied to the MCE corpus, we incorporated the knowledge extracted from SentiWordNet version 3.0 (Baccianella, Esuli, & Sebastiani, 2010). SentiWordNet (SWN) is a publicly available lexical resource for opinion mining which assigns three sentiment scores to each synset of WordNet[14]: positivity (how positive the word is), negativity (how negative the word is) and objectivity (how objective the word is). Each of the scores ranges from 0 to 1, and their sum equals 1. SentiWordNet word values have been semi-automatically computed based on the use of weakly supervised classification algorithms. Examples of "subjectivity" scores associated to WordNet entries are shown in Fig. 2. As can be seen in that Figure, the entries contain the parts of speech category of the displayed entry, its positivity, its negativity, and the list of synonyms. We show various synsets related to the words "good" and "bad". There are four senses of the noun "good", 21 senses of the adjective "good", and two senses of the adverb "good" in WordNet. There is one sense of the noun "bad", 14 senses of the adjective "bad", and 2 senses of the adverb "bad" in WordNet.

We have used nouns, adjectives, verbs and adverbs as linguistic features in order to generate the MCE_SWN corpus. As a first step, the English documents from MCE were processed by applying a POS tagger like TreeTagger[15] (Schmid, 1994). The aim of this process was to obtain all the nouns, adjectives, verbs and adverbs of each review. Then in a second step we generated a total of 15 sub-corpora from MCE by making a combination of the four possibilities (nouns, adjectives, verbs and adverbs) in order to analyze the impact of each type of word. Finally, we calculated the SWN score for each document as the polarity score of the document. This score was obtained following the method proposed by Denecke (2008) based on the calculation of a triple combination of positivity, negativity and objec-

[3] MCE is freely available in http://sinai.ujaen.es/wiki/index.php/MCE_Corpus_(English_version).
[4] http://www.worldlingo.com.
[5] http://www.babelfish.com.
[6] http://translate.google.com.
[7] https://developers.google.com/translate.
[8] http://www.itranslate4.eu.
[9] http://www.systransoft.com.
[10] http://www.online-translator.com.
[11] http://www.bing.com/translator.
[12] http://code.google.com/p/microsoft-translator-java-api.
[13] https://datamarket.azure.com.
[14] WordNet is a large lexical database of English. Nouns, verbs, adjectives and adverbs are grouped into sets of cognitive synonyms (synsets), each expressing a distinct concept. It is available in http://wordnet.princeton.edu.
[15] http://www.ims.uni-stuttgart.de/projekte/corplex/TreeTagger.

Category	WNT Number	pos	neg	synonyms
a	1006645	0.25	0.375	good#a#15 well#a#2
a	1023448	0.375	0.5	good#a#23 unspoilt#a#1 unspoiled#a#1
a	1073446	0.625	0.0	good#a#22
a	1024262	0.0	1.0	spoilt#a#2 spoiled#a#3 bad#a#4
a	1047353	0.0	0.875	defective#a#3 bad#a#14
a	1074681	0.0	0.875	bad#a#13 forged#a#1

Fig. 2. Fragment of SentiWordNet.

tivity scores.

5. Experiments and results

In this section we describe the experimental framework used to evaluate the experiments carried out in this study. Then we show the results obtained for each corpus individually by applying both supervised and non-supervised approaches along with the results obtained using the combination method (meta-classifier) proposed.

Some of the tools used in order to extract the linguistic features from the MCE corpus and apply the unsupervised approach have already been discussed above, such as SentiWordNet or TreeTagger.. The Rapid Miner[16] tool was used to run the different machine learning experiments. It provides a text mining plug-in, along with different tools designed to assist in the preparation of text documents for mining tasks (namely tokenization, stop word removal and stemming, among others). Rapid Miner is an environment for machine learning and data mining processes that includes learning operators such as SVM, NB, BBR and KNN.

5.1. Individual experiments

The advantage of supervised learning methods versus unsupervised ones is that the first usually achieve better results than the second. However, the main drawbacks of the supervised methods are their need of a large amount of labeled data and their high dependence on the domain.

The main purpose of the first experiments was to prove that the automatic translation process from Spanish to English does not entail a loss in the quality of the classification. In Martínez Cámara et al. (2011) we presented a study of the influence of the weighting scheme, the reduction method of features (stopper and stemmer) and the machine learning algorithm in polarity classification over Spanish documents using the MC corpus. In that study each review was represented as a vector of unigrams which were weighted by the following four weighting methods:

- Term Frequency-Inverse Document (TF-IDF): This is well-known by the Information Retrieval research community and was presented in (Salton & McGill, 1986).
- Term Frequency (TF): The relative frequency of a term in a document. The resulting vector for each document is normalized to the Euclidean unit length.
- Term Occurrences (TO): The absolute number of occurrences of a term. The resulting vector is not normalized.
- Binary Term Occurrences (BTO): A term receives 1 if it is present in the document or 0 otherwise.

In the study two of the learning algorithms most used in SA were assessed, i.e., Support Vector Machines (SVM) and Naïve Bayes (NB). The best results were achieved by SVM, and they are presented in Table 3.

As can be seen in Table 3, TF-IDF was the weighting scheme that obtained the best results. The difference regarding the other schemes is remarkable. Comparing the best results among the weighting schemes and taking into account the F1 measure, TF-IDF improves by 10.16% the TF score, 13.22% the TO score and 4.16% the BTO score. Regarding the behavior of using stopper and stemmer, the best result was obtained without applying any of the two heuristics. Nevertheless, the difference when the stopwords are removed is very low (+0.045% and +0.034% regarding F1 and accuracy, respectively). For this reason we have considered the configuration that removes the stopwords as the best one because, by removing the stopwords the reduction of the number of features is significant, so the classification process is more efficient. For this reason, we have remarked in bold, not only the best result but also the result obtained with TF-IDF applying stopwords because this last will be use as a base in the following experiments. Therefore, the configuration that consists of removing stopwords, using TF-IDF and applying SVM was considered as the best and most efficient (MC_SVM).

The next step was to apply the same experiments over the MCE corpus. The results, which are shown in Table 4, presented a similar behavior to those obtained with the MC corpus. As in the original corpus, TF-IDF was the weighting scheme that obtained the best results. Regarding the results achieved with other weighting schemes, it is important to note that in some cases they were better than those obtained using MC as corpus. This means that the translation process performed well, and MCE could be used to improve the results of polarity classification over the original corpus MC.

Regarding the use of stopper and stemmer, the best results were also obtained without applying either of them. Comparing these results with those obtained using the original corpus, the best results obtained using MCE were slightly worse than expected,

Table 3
Results obtained using SVM over the MC corpus (MC_SVM).

	Stop	Stem	Precision	Recall	F1	Accuracy
TF-IDF	✓	✓	0.8684	0.8667	0.8675	0.8674
	✓		**0.8771**	**0.8763**	**0.8767**	**0.8766**
		✓	0.8680	0.8664	0.8672	0.8670
			0.8773	**0.8769**	**0.8771**	**0.8769**
TF	✓	✓	0.7981	0.7944	0.7962	0.7958
	✓		0.7748	0.7708	0.7728	0.7722
		✓	0.7974	0.7942	0.7958	0.7954
			0.7706	0.7665	0.7685	0.7680
TO	✓	✓	0.77660	0.7689	0.7727	0.7665
	✓		0.7405	0.7300	0.7352	0.7269
		✓	0.7786	0.7709	0.7747	0.7684
			0.7425	0.7312	0.7368	0.7280
BTO	✓	✓	0.8423	0.8420	0.8421	0.8419
	✓		0.8394	0.8391	0.8392	0.8385
		✓	0.8413	0.8412	0.8412	0.8411
			0.8411	0.8416	0.8413	0.8404

[16] Rapid Miner is freely available in http://rapid-i.com.

Table 4
Results obtained using SVM over the MCE corpus (MCE_SVM).

	Stop	Stem	Precision	Recall	F1	Accuracy
TF-IDF	✓	✓	0.8502	0.8497	0.8499	0.8499
	✓		**0.8704**	**0.8693**	**0.8698**	**0.8697**
		✓	0.8586	0.8582	0.8584	0.8583
			0.8776	**0.8769**	**0.8772**	**0.8769**
TF	✓	✓	0.8423	0.8415	0.8419	0.8419
	✓		0.8554	0.8540	0.8547	0.8544
		✓	0.7853	0.7845	0.7849	0.7847
			0.7809	0.7803	0.7806	0.7806
TO	✓	✓	0.8463	0.8460	0.8461	0.8461
	✓		0.8497	0.8489	0.8493	0.8491
		✓	0.7651	0.7647	0.7649	0.7642
			0.7483	0.7474	0.7478	0.7467
BTO	✓	✓	0.8276	0.8267	0.8271	0.8270
	✓		0.8462	0.8451	0.8456	0.8453
		✓	0.8281	0.8273	0.8277	0.8274
			0.8442	0.8430	0.8436	0.8430

because almost all automatic translation tools introduce some noise during the process.

As for the MC corpus, the configuration that achieved the best performance consisted of not applying either stopper or stemmer. However, we have also considered the configuration that uses TF-IDF as the weighting scheme, removing the stopwords and applying the SVM algorithm,as the most efficient for MCE (MCE_SVM) due to the reduction of the number of features.

Finally, regarding the unsupervised experiments carried out individually over the MCE corpus, as has already been introduced in Section 4.3, we generated different sub-corpora from MCE. Then we calculated the SWN score for each document according to the method proposed by Denecke (2008) based on the calculation of a triple combination of positivity, negativity and objectivity scores for each review. Table 5 shows these results. Analyzing this table, we highlight the following facts:

1. When the lexical categories are considered individually the results obtained are similar, with the exception of the adverbs whose recall value is very low in general.
2. The poor performance of adverbs in general means that all the sub-corpora that contain this feature obtain worse results.
3. The lexical categories with the most semantic information are verbs and adjectives. These achieve good results individually and, as expected, the best results are obtained when they are combined in the same sub-corpus.

As can be seen in Table 6 the MC_SVM and MCE_SVM results are very similar, so we highlight the quality of the translation process. These results encourage us to apply a strategy in which the different models generated individually can be combined in a meta-classifier with the aim of achieving the best possible combination, and thus attempt to improve the results obtained by the models individually. The difference between the supervised and unsupervised results is not strange and highlights the fact that in general the supervised methods have a better performance than the unsupervised ones, because supervised methods take advantage of prior knowledge.

5.2. Combined experiments

In Data Mining when several results are achieved by several classifiers over the same data set, it is common to ensemble all the models in order to obtain better precision. After carrying out the individual experiments we propose the following method: if we use several classifiers for the same data then we will obtain several models that have learned different patterns from that data. In this manner it is very likely that the correct combination of the models achieves better results than those obtained by each classifier individually. In the literature we can find several methods for combining classifiers that achieve a good performance. Some of the most widely used are boosting (AdaBoost), bagging, melting methods (voting schemes) or hybrid methods (stacking, cascading). An extensive classification of the ensemble classification methods can be found in (Dietterich, 2000).

With the aim of carrying out the approach proposed above we adapted the idea of the ensemble classifiers, but working with parallel corpora instead of the same corpus. For the combined experiments we used MC and MCE as corpora, and three models (MC_SVM, MCE_SVM and MCE_SWN) generated after applying the supervised and unsupervised approaches to them. Specifically we tried two of the most widely used combination algorithms, Voting and Stacking, which will be described in the following sections.

5.2.1. Voting system

One of the combination algorithms employed in the meta-classifier was the well-known voting system called majority rule (Johnson, 2005). In this manner, we combined the classification results obtained by applying the supervised and unsupervised approaches over the MC and MCE corpora.

Voting models are based on voting schemes usually used in economics and politics for group choice decision-making. Due to the fact that we work with two possible candidates for each review

Table 5
Results obtained applying the unsupervised approach over the MCE as corpus (MCE_SWN).

MCE sub-corpus	Rev POS		Rev NEG		P	R	F1	Acc
	Pred POS	Pred NEG	Pred POS	Pred NEG				
Only-noun	1055	219	962	389	0.5231	0.8281	0.6411	0.5501
Only-adj	1006	268	751	600	0.5726	0.7896	0.6638	0.6118
Only-verb	1139	135	1025	326	0.5263	0.8940	0.6626	0.5581
Only-adv	323	951	252	1099	0.5617	0.2535	0.3494	0.5417
Adj + noun	1106	168	853	498	0.5646	0.8681	0.6842	0.6110
Adj + verb	1114	160	851	500	0.5669	0.8744	**0.6879**	0.6149
Adj + adv	727	547	436	915	0.6251	0.5706	0.5966	0.6255
Noun + verb	1144	130	1060	291	0.5191	0.8980	0.6578	0.5467
Noun + adv	644	630	470	881	0.5781	0.5055	0.5394	0.5810
Verb + adv	531	743	396	955	0.5728	0.4168	0.4825	0.5661
Adj + noun + verb	1151	123	934	417	0.5520	0.9035	0.6853	0.5973
Adj + noun + adv	894	380	557	794	0.6161	0.7017	0.6561	0.6430
Noun + verb + adv	794	480	561	790	0.5860	0.6232	0.6040	0.6034
Adj + verb + adv	856	418	519	832	0.6225	0.6719	0.6463	0.6430
Adj + noun + verb + adv	981	293	627	724	0.6101	0.7700	0.6808	0.6495

Table 6
Best results achieved with the three classifiers individually.

Model	Precision	Recall	F1	Accuracy
MC_SVM	**0.8771**	**0.8763**	**0.8767**	**0.8766**
MCE_SVM	0.8704	0.8693	0.8698	0.8697
MCE_SWN	0.5669	0.8744	0.6879	0.6149

(positive or negative), single-winner voting systems are more appropriate. Single-winner systems can be classified based on their ballot type. In one-vote systems, a voter picks one choice at a time. In ranked voting systems, each voter ranks the candidates in order of preference. In our case, each voter (system) can vote for one candidate (positive or negative), so this is the reason why we apply the majority rule principle: each voter votes for one choice, and the choice that receives the most votes wins.

From a classification point of view, majority rule can be seen as plurality rule when the total number of classes is two ($c = 2$). Therefore, according to Kuncheva (2004), the majority rule could be represented mathematically as follows:

$$\sum_{i=1}^{L} d_{i,k} = \max_{j=1}^{c} \sum_{i=1}^{L} d_{i,j},$$

where it is assumed that the label outputs of the classifiers are given as c-dimensional binary vectors (for majority rule only two classes, i.e. $[d_{i,1}, d_{i,2}]^T \in \{0, 1\}^c$, $i = 1, \ldots, L$), and where $d_{i,j} = 1$ if D_i labels x in w_j, and 0 otherwise.

Taking into account the evaluation of the individual experiments carried out using MC and MCE, we proposed two experiments combining such results in a voting system using the majority rule. We considered the best individual configurations. For each review there were several voters (individual configurations) and two possible candidates (positive or negative). Therefore, two different experiments called *ExpVS1* and *ExpVS2* were proposed:

- **ExpVS1** (two voters): MC_SVM + MCE_SVM
- **ExpVS2** (three voters): MC_SVM + MCE_SVM + MCE_SWN

where MC_SVM represents the polarity predicted for the review using the supervised approach over MC, MCE_SVM is the same as MC_SVM but over MCE, and MCE_SWN means the polarity predicted for the review using the approach based on semantic orientation integrating SentiWordNet (SWN) over MCE. Therefore, ExpSV1 only combines the best results obtained using the supervised approach over both corpora individually and ExpSV2 combines the best results obtained using the supervised approach over the MC corpus, the same approach over the MCE corpus and the semantic orientation approach over the MCE corpus.

Due to the fact that the number of voters in ExpVS2 is odd, the application of the voting system in this experiment always returns a single-winner. However, in the experiment ExpVS1 it is possible to obtain a draw because the predicted class for the MC_SVM voter may be different from that obtained by the MCE_SVM voter. In order to resolve this issue we considered two possible heuristics:

- **ExpVS1-both:** assign a final positive prediction only if both voters return a positive prediction (otherwise negative prediction), or
- **ExpVS1-one:** assign a final positive prediction if at least one of the voters returns a positive prediction (negative prediction only when both voters return a negative prediction)

Table 7 shows some examples of the predicted class assigned by the voting system when only two voters (MC_SVM and MCE_SVM) are involved. Table 8 shows the results obtained using the voting system approach. As can be seen in Table 8, the experiment that combines the results obtained using the supervised approach over both corpora (ExpVS1-one) achieves the best F1 score, although the difference from the other experiments is not very relevant: +1.25% and +0.03% regarding the ExpVS1-both and ExpVS2 experiments, respectively. Therefore, the use of the unsupervised approach in combination with the supervised one under a voting system using the majority rule does not result in an improvement, although it could be used because the difference with the experiment that combines only the supervised approach over both corpora is irrelevant.

Table 7
Working example of the voting system using two voters for each review (ExpVS1).

Review	Real class	MC_SVM	MCE_SVM	Count POS	Count NEG	ExpVS1	
						ExpVS1-both	ExpVS1-one
1255.xml	POS	POS	POS	2	0	POS	POS
1261.xml	NEG	NEG	POS	1	1	NEG	POS
1268.xml	NEG	NEG	NEG	0	2	NEG	NEG
1608.xml	NEG	POS	NEG	1	1	NEG	POS
347.xml	POS	POS	NEG	1	1	NEG	POS
3973.xml	POS	NEG	POS	1	1	NEG	POS

5.2.2. Stacking

Stacking (Wolpert, 1992) is a simple meta-classifier based on the combination of several models generated by different learning algorithms. Thus, several models must be combined in the best way in order to improve the results. For this end, the Stacking approach adds a new classifier that takes as inputs the outputs of the first classifiers, and learns the best way to combine the first classifiers. We have adapted the general Stacking approach to our needs by taking the outputs of the supervised and unsupervised models developed over MC and MCE as inputs of the feature vectors of this combination algorithm. We have proved the performance of three different feature vectors:

1. **Classes (CL)**: The features are the class predicted by the base algorithms. In this scenario there are three features per opinion or document.
2. **Confidences (CF):** The features are the confidence values calculated by the classifiers for each of the possible classes. The MC_SVM and MCE_SVM returned a confidence value for the positive class, and another one for the negative class. The MCE_SWN returned three confidence values. The first one measures the positive value of the opinion, the second one the negative value and the third one the neutral value. Therefore, the number of features in the second configuration is seven.
3. **Classes and confidences (CL_CF)**: The feature vector of the third configuration is the combination of the CL and CF features, so the number of features in this experiment is ten.

In addition to the different configurations employed to generate the feature vectors, we also tested the performance of four machine learning algorithms for the Stacking experiments: Support

Table 8
Results obtained using the voting system approach.

Voting system experiments	Precision	Recall	F1	Accuracy
ExpVS1-both	0.8551	0.8893	0.8719	0.8731
ExpVS1-one	0.8003	0.9843	**0.8828**	0.8731
ExpVS2	0.8160	0.9608	0.8825	0.8758

Table 9
Results obtained using the Stacking approach.

Meta-classifier	F. vectors	Precision	Recall	F1	Accuracy
SVM	CL	0.8771	0.8764	0.8768	0.8766
	CF	**0.8833**	**0.8828**	**0.8831**	**0.8831**
	CL_CF	0.8771	0.8764	0.8768	0.8766
NB	CL	0.8781	0.8782	0.8782	0.8781
	CF	0.8836	0.8834	0.8835	0.8834
	CL_CF	**0.8858**	**0.8857**	**0.8856**	**0.8857**
C4.5	CL	**0.8771**	**0.8764**	**0.8766**	**0.8766**
	CF	0.8654	0.8630	0.8642	0.8632
	CL_CF	0.8654	0.8630	0.8642	0.8632
BBR	CL	0.8770	0.8755	0.8763	0.8758
	CF	0.8814	0.8813	0.8814	0.8815
	CL_CF	**0.8829**	**0.8824**	**0.8827**	**0.8827**

Vector Machine (SVM), Naïve Bayes (NB), C4.5 and BBR (Bayesian Logistic Regression). The three first algorithms are widely known by the research community. BBR (Genkin, Lewis, & Madigan, 2007) is a Bayesian implementation of the logistic regression that avoids overfitting of the training data. The algorithm is based on the calculation of the following conditional likelihood:

$$p(y|\beta, x_i) = \psi(\beta^T, x_i) = \psi(\beta_j x_{i,j}),$$

where $y \in \{+1, -1\}$ are the document classes, each document is represented by a vector (x_i) of values, β_j are the predictor variables, and ψ is a logistic link function.

$$\psi(r) = \frac{e^{(r)}}{1 + e^{(r)}}.$$

Table 9 shows the results obtained using the Stacking algorithm with the different configurations explained above. The best results obtained for the polarity classification of the MC corpus individually were 0.8771 of F1 and 0.8769 of accuracy (see Table 3). Taking into account the results obtained using the Stacking approach, only two meta-classifiers performed better in all cases (SVM and NB). The major improvement over the MC classification was achieved using NB and CL_CF as the feature vector. These results mean that it is possible to enrich sentiment analysis performance following a machine learning approach with semantic resources, and even using a multilingual procedure.

6. Analysis of the results

This paper describes the approach followed in order to improve the polarity classification of Spanish texts by applying semantic orientation features. Firstly, we followed a supervised approach to classify the polarity of the original corpus in Spanish. Secondly, we translated this corpus into English, with the aim of applying linguistic resources for sentiment analysis. Then, supervised and unsupervised methods were applied to the English version of the corpus (MCE). As shown in Section 5.1, the results did not improve on those obtained using the original corpus, but they encourage us to attempt a methodology based on meta-classifiers. We assessed two ensemble methods, voting and stacking, that have been described in Section 5.2. A comparison of these results is shown in Table 10.

Taking as reference the F1 value the two ensemble methods improve the baseline classification (MC_SVM). The voting system achieves an improvement of +0.69%, while the stacking algorithm achieves +1.02%. In this manner, we have shown that the combination of several classifiers could be a good strategy to apply in sentiment analysis and specifically in a cross-language environment.

Table 10
Comparision of the Spanish polarity classification systems.

Experiment	Precision	Recall	F1	Accuracy
MC_SVM	0.8771	0.8763	0.8767	0.8766
Voting	0.8003	**0.9843**	0.8828	0.8731
Stacking	**0.8858**	0.8857	**0.88575**	**0.8857**

Table 11
Polarity classification results over MC corpus.

	Precision	Recall	F1	Accuracy
Cruz et al. (2008)	N/A	N/A	N/A	0.7705
del-Hoyo, Hupont, Lacueva, and Abadía (2009)	N/A	N/A	N/A	0.8086
Malvar-Fernández and Pichel-Campos (2011)	0.77	0.77	N/A	N/A
Martínez-Cámara, Martín-Valdivia, and Ureña-López (2011)	0.8684	0.8667	0.8675	0.8674
Our proposal	**0.8858**	**0.8857**	**0.88575**	**0.8857**

The main goal of the ensemble methods is to correct the misclassifications of the individual classifiers. In our case, we can confirm the benefits obtained by applying this strategy since both voting and stacking improve the final results for the Spanish polarity classification task.

Comparing our results with other studies that used the MC corpus for polarity classification purposes, the proposed approach improves all the systems, as can be seen in Table 11. The proposed stacking strategy outperforms the supervised approach over the Spanish corpus, and also improves the state of the art results with the MC corpus.

7. Conclusions and further work

In this paper we have presented an experimental study about polarity classification over a corpus of film reviews written in Spanish, the MuchoCine corpus (MC). Firstly, we translated the MC corpus into English in order to generate the parallel MCE corpus. Several experiments were carried out in order to build supervised and unsupervised models using these corpora. SentiWordNet was used as the linguistic resource for the unsupervised experiments over MCE. Finally, the individual systems were combined applying different approaches such as voting system and stacking algorithm. Although the results obtained with individual models were very promising, we have shown that the combination techniques improved on the performances achieved individually. These results encourage us to continue working along this line.

In addition, we would like to test performance using linguistic resources other than SentiWordNet, like for example WordNet-Affect (Valitutti, Strapparava, & Stock, 2004), General Inquirer (Stone, 1966), ANEW (Bradley & Lang, 1999) or Q-WordNet (Aguerri & García-Serrano, 2010). Moreover, we are working to generate our own list of Spanish affective words. Thus, we could apply a semantic orientation approach directly to the MC corpus and obtain a new model to consider in the meta-classifier architecture.

Acknowledgement

This work has been partially supported by a grant from the Fondo Europeo de Desarrollo Regional (FEDER), TEXT-COOL 2.0 project (TIN2009-13391-C04-02) from the Spanish Government. Also, this paper is partially funded by the European Commission under the Seventh (FP7 – 2007–2013) Framework Programme for Research and Technological Development through the FIRST project

(FP7-287607). This publication reflects the views only of the authors, and the Commission cannot be held responsible for any use which may be made of the information contained therein.

References

Aguerri, R., & García-Serrano, A. (2010). Q-WordNet: Extracting Polarity from WordNet Senses. In *Proceedings of the Seventh conference on International, Language Resources and Evaluation (LREC'10)*.

Baccianella, S., Esuli, A., & Sebastiani, F. (2010). SentiWordNet 3.0: An enhanced lexical resource for sentiment analysis and opinion mining. In *Proceedings of the Seventh conference on International, Language Resources and Evaluation (LREC'10)*.

Balahur, A. & Turchi, M. (2012). Multilingual sentiment analysis using machine translation? In *Proceedings of the 3rd workshop on computational approaches to subjectivity and sentiment analysis workshop*, 52 Jeju, Republic of Korea.

Banea, C., Mihalcea, R., Wiebe, J. & Hassan, S. (2008). Multilingual subjectivity analysis using machine translation, In *'EMNLP', ACL* (pp. 127–135).

Barreiro, A., & Gonçalo, O. (2011). Cross-language semantic relations between English and Portuguese. In *Proceedings of the workshop on iberian cross-language natural language processing tasks (ICL 2011)* (pp. 49–58).

Bradley, M.M., & Lang, P.J. (1999). Affective norms for English words (ANEW): Stimuli, instruction manual, and affective ratings. Technical report, Center for Research in Psychophysiology, University of Florida.

Boldrini, E., Balahur, A., Martínez-Barco, P., & Montoyo, A. (2009). Emotiblog: an annotation scheme for emotion detection and analysis in non-traditional textual genres. In R. Stahlbock, S. F. Crone & S. Lessmann (Eds.), *DMIN* (pp. 491–497). CSREA Press.

Cruz, F., Troyano, J. A., Enriquez, F., & Ortega, J. (2008). Experiments in sentiment classification of movie reviews in Spanish. *Procesamiento de Lenguaje Natural (Sociedad Española para el Procesamiento de Lenguaje Natural), 41*, 73–80.

del-Hoyo, R., Hupont, I., Lacueva, F. J. & Abadía, D. (2009). Hybrid text affect sensing system for emotional language analysis. In *Proceedings of the international workshop on affective-aware virtual agents and social robots*. Boston, Massachusetts.

Denecke, K. (2008). Using SentiWordNet for multilingual sentiment analysis. *ICDE workshops* (pp. 507–512). IEEE Computer Society.

Dietterich, T. G. (2000). An experimental comparison of three methods for constructing ensembles of decision trees: bagging, boosting, and randomization. *Machine Learning* 40(2), 139–157.

Genkin, A., Lewis, D. D., & Madigan, D. (2007). Large-scale Bayesian logistic regression for text categorization. Technometrics (Taylor & Francis) (pp. 291–304).

Ghorbel, H., & Jacot, D. (2010). Sentiment analysis of French movie reviews. In *Proceedings of the 4th international workshop on distributed agent-based retrieval tools (DART 2010)*. Geneva, Italy.

Johnson, P. (2005). Voting systems. *A textbook-style overview of voting methods and their mathematical properties*.

Kim, S.-M. & Hovy, E. (2006). Identifying and Analyzing Judgment Opinions. In *Proceedings of the Joint Human Language Technology/North American Chapter of the ACL Conference (HLT-NAACL)*.

Kuncheva, L. I. (2004). *Combining pattern classifiers: Methods and algorithms*. Wiley-Interscience.

Malvar-Fernández, P., & Pichel-Campos, J. R. (2011). Generación semiautomática de recursos de Opinion Mining para el gallego a partir del portugués y el español. In *Proceedings of the workshop on Iberian cross-language natural language processing tasks (ICL 2011)* (pp. 59–63).

Martínez-Cámara, E., Martín-Valdivia, M. T. & Ureña-López, L. A. (2011). Opinion classification techniques applied to a Spanish corpus. In *Proceedings of the 16th international conference on Natural language processing and information systems, NLDB'11* (pp. 169–176). Springer-Verlag.

Martínez Cámara, E., Martín-Valdivia, M. T., Perea-Ortega, J. M., & Ureña-López, L. A. (2011). Técnicas de clasificación de opiniones aplicadas a un corpus en español. *Procesamiento de Lenguaje Natural, 47*, 163–170.

Mihalcea, R., Banea, C., & Wiebe, J. 2007. Learning multilingual subjective language via cross-lingual projections. Proceedings of the Association for Computational Linguistics (ACL) (pp. 976–983). Prague, Czech Republic.

Pang, B., Lee, L., & Vaithyanathan, S. 2002. Thumbs up? Sentiment classification using machine learning. In *Proceedings of EMNLP* (pp. 79–86).

Pang, B., & Lee, L. (2008). *Opinion mining and sentiment analysis*. Now Publishers Inc.

Rushdi-Saleh, M., Martín-Valdivia, M. T., Ureña-López, L. A., & Perea-Ortega, J. M. (2011a). OCA: Opinion corpus for Arabic. *JASIST, 62*(10), 2045–2054.

Rushdi-Saleh, M., Martín-Valdivia, M. T., Ureña-López, L. A., & Perea-Ortega, J. M. (2011b). Bilingual Experiments with an Arabic-English Corpus for Opinion Mining. In Galia Angelova; Kalina Bontcheva; Ruslan Mitkov & Nicolas Nicolov (Eds.), *'RANLP'* (pp. 740–745).

Salton, G., & McGill, M. J. (1986). *Introduction to modern information retrieval*. McGraw-Hill, Inc..

Schmid, H. (1994). Probabilistic part-of-speech tagging using decision trees. In *Proceedings of the international conference on new methods in language processing, Manchester, United Kingdom*.

Stone, P. J. (1966). *The general inquierer: A computer approach to content analysis*. The MIT Press.

Taboada, M., Voll, K., & Brooke, J. (2008). Extracting sentiment as a function of discourse structure and topicality. Technical Report 20, Simon Fraser University. Available online, http://www.cs.sfu.ca/research/publications/techreports/#2008.

Tan, S., & Zhang, J. (2008). An empirical study of sentiment analysis for Chinese documents. *Expert Systems with Applications, 34*(4), 2622–2629.

Turney, P. D. (2002). Thumbs up or thumbs down?: Semantic orientation applied to unsupervised classification of reviews. ACL '02: In *Proceedings of the 40th annual meeting on association for computational linguistics* (pp. 417–424). Morristown, NJ, USA: Association for Computational Linguistics.

Valitutti, A., Strapparava, C., & Stock, O. (2004). Developing affective lexical resources. *PsychNology Journal, 2*(1), 61–83.

Wan, X. (2009). Co-Training for cross-lingual sentiment classification. In Keh-Yih Su, Jian Su & Janyce Wiebe (Eds.). In *Proceedings of the 47th annual meeting on association for computational linguistics* (pp. 235–243). ACL/AFNLP, Singapore.

Wolpert, D. H. (1992). Stacked generalization. *Neural Networks*, 241–259.

Zhang, C., Zeng, D., Li, J., Wang, F.-Y., & Zuo, W. (2009). Sentiment analysis of Chinese documents: From sentence to document level. *JASIST, 60*(12), 2474–2487.