

# Data Intensive Review Mining for Sentiment Classification across Heterogeneous Domains

Federica Bisio, Paolo Gastaldo, Chiara Peretti, and  
Rodolfo Zunino  
DITEN, University of Genoa  
Via Opera Pia 11a, 16145, Genoa  
Italy  
{paolo.gastaldo, chiara.peretti, rodolfo.zunino}@unige.it  
federica.bisio@edu.unige.it

Erik Cambria  
Temasek Laboratories  
National University of Singapore  
5A Engineering Drive 1, 117411  
Singapore  
cambria@nus.edu.sg

**Abstract**— The automatic detection of orientation and emotions in texts is becoming increasingly important in the Web 2.0 scenario. There is a considerable need for innovative techniques and tools capable of identifying and detecting the attitude of unstructured text. The paper tackles two crucial aspects of the sentiment classification problem: first, the computational complexity of the deployed framework; second, the ability of the framework itself to operate effectively in heterogeneous commercial domains. The proposed approach adopts empirical learning to implement the sentiment-classification technology, and uses a distance-based predictive model to combine computational efficiency and modularity. A suitably designed semantic-based metric is the cognitive core that measures the distance between two user reviews, according to the sentiment they communicate. The framework ultimately nullifies the training process; at the same time, it takes advantage of a classification procedure whose computational cost increases linearly when the training corpus increases. To attain an objective measurement of the actual accuracy of the sentiment classification method, a campaign of tests involved a pair of complex, real-world scoring domains; the goal was to compare the predicted sentiment scores with actual scores provided by human assessors. Experimental results confirmed that the overall approach attained satisfactory performances in terms of both cross-domain classification accuracy and computational efficiency.

**Keywords**— *sentiment analysis; cross-domain sentiment classification*

## I. INTRODUCTION

To make computers able to analyze people's opinions, sentiments, evaluations, attitudes, and emotions from written language is a challenging task, which has recently been addressed by several communities dealing with natural language processing, data mining, text mining, and computational intelligence. The automatic detection of emotions in texts is becoming increasingly important in the fight against terrorism and crime since social media are increasingly becoming a vector of malicious activities through the web.

The research areas of opinion mining and sentiment analysis deal with the extraction of people's attitudes from text.

The (heterogeneous) group of related works [1] can be split into clusters according to the specific applications, the adopted technologies, and the problem formulation (e.g., classification, extraction). When considering the approach adopted to map texts into sentiment labels, Cambria et al. [11] showed that existing research constitutes four main groups. The approaches in the first group, keyword spotting, ascribe texts to categories depending on the occurrences of predefined, unambiguous terms. The works in the second category, lexical affinity, endow each term with a probabilistic affinity to a specific concept. The frameworks in the third group, statistical methods, adopt machine-learning approaches such as support vector machines, k-means clustering, latent semantic analysis, and use a training corpus. Finally, concept-based approaches rely on large semantic knowledge bases or ontologies in order to leverage on implicit features of natural language concepts.

Among these, sentic computing [2] adopts a multi-disciplinary approach to opinion mining and sentiment analysis by integrating computer and social sciences to recognize, interpret, and process sentiments expressed in natural language text. In particular, semantic multi-dimensional scaling and artificial neural networks [12] are employed on a common-sense knowledge base to assess opinions that are implicitly conveyed through context- and domain-dependent concepts, thus improving on conventional approaches that exploit a word-level analysis of texts.

The present research tackles sentiment classification of user reviews, and therefore covers a specific field in the opinion-mining landscape. The underlying problem plays a central role in the Web 2.0 scenario, where the collection of user reviews is a common practice by social media, considered as a source of knowledge for illegal activities detection. The ability to assess the sentiment expressed by the author of a review in an automated fashion may therefore prove crucial.

In sentiment classification of user reviews, technology faces challenging requirements about computational efficiency, since one expects to process huge amounts of data and, at the same time, to ensure real-time response. In addition, the technology should be able to operate across heterogeneous commercial domains, since reviews may concern a wide

variety of commodities and services (e.g., hotels, films, books, consumer electronics devices).

In the considerable literature dealing with the review-mining area, only a few works tackled the cross-domain problem in the past years [3], and the computational complexity of the proposed technologies did not seem to represent a central issue. This issue, in fact, may prove crucial when considering that a sentiment-classification model needs to be retrained periodically to properly tackle cross-domain applications, typically in the presence of huge corpora of samples.

The approach presented in this paper has been specifically designed to combine effectiveness at sentiment classification with computational efficiency, and includes two main technologies. An integrated strategy combines semantic networks and contextual valence shifters [4] to define a feature space where reviews eventually lie. The resulting descriptions define the input space for Computational Intelligence (CI) tools, which support the classification problem. The adoption of machine-learning models is not new in the sentiment-classification area [5], and empirical learning is used to map reviews into sentiment labels. The proposed solution takes advantage of an augmented k-Nearest Neighbor (kNN) approach [6], which virtually nullifies the computational cost of the training process. This aspect sets a novelty point of the present research with respect to other methods in the literature, which usually exploit machine learning algorithms that may prove computationally expensive.

The performance of the proposed approach has been evaluated experimentally by involving a pair of real, large datasets of reviews: the first one collects product reviews from Amazon™ [3], whereas the other holds hotel reviews from TripAdvisor™ [7]. The presented datasets are publicly available, allowing the reproduction of the showed tests. These corpora have been chosen to meet two main experimental requirements: first, the test reviews for assessing cross-domain abilities should pertain to areas that are quite distant from each other; secondly, the performance of the proposed framework should be assessed on large datasets. Experimental results show that the framework attains satisfactory performances in terms of both cross-domain classification accuracy and computational efficiency.

The paper is organized as follows. After formalizing the review-mining problem setup, specific Sections present the approach to classification and cognitive metric. The practical deployment of the framework is then illustrated. The presentation of experimental results proves the approach effectiveness, and some conclusions are eventually drawn.

## II. A FRAMEWORK FOR CROSS-DOMAIN SENTIMENT CLASSIFICATION

### A. Problem setup

Cross-domain sentiment classification involves two main design issues.

The first issue concerns the training mechanism for assessing the sentiment embedded in a user review. Let  $(\mathcal{R}s)$

denote a (training) instance that associates a text review,  $\mathcal{R}$ , with a sentiment label,  $s$ . In general  $s \in \Sigma$ , where  $\Sigma$  is a set of sentiment labels; a typical choice in the literature is  $\Sigma = \{-1, +1\}$ , and the classification problem focuses on the dichotomy separating positive/negative attitudes. Thus a sentiment-classification system is trained on a corpus,  $\Omega = \{(\mathcal{R}_i, s_i); i = 1, \dots, N_i\}$ , where  $N_i$  is the number of available instances.

The second issue involves the generalization ability of the resulting classification system. The term ‘cross-domain’ implies the assumption that reviews belong to different class of entities [3]. This is the case, for example, of reviews on books as opposed to reviews on computers, since one assumes that the two classes are fairly uncorrelated from a lexical point of view. The notation  $\Omega^{\mathcal{D}}$  will associate a review corpus,  $\Omega$ , with a reminder of the underlying domain,  $\mathcal{D}$ . Thus a system for cross-domain sentiment classification is trained on a corpus,  $\Omega^{\mathcal{D}}$ , and can predict the sentiment labels of reviews drawn from a corpus,  $\Omega^{\mathcal{D}'}$ , where  $\mathcal{D}' \neq \mathcal{D}$ .

An effective framework for such a classification task should take into account the training and the generalization issues at the same time. A statistical approach addresses this problem by building a model that maps the space,  $\Xi$ , of test reviews into sentiment labels as

$$\phi(p_1, p_2, \dots, p_n) : \Xi \rightarrow \Sigma \quad (1)$$

where  $\mathcal{R} \in \Xi$ , and the terms  $p_i$  are the parameters to be adjusted by the training process.

Expression encompasses the above-mentioned design issues: the set up of a feature space,  $\Xi$ , and the set up of a predictive model,  $\phi$ . The former aspect is crucial because review features feed the predictive model. The feature space should be carefully designed to provide cross-domain abilities; for example, an approach only considering lexical elements would probably fail at characterizing reviews drawn from heterogeneous domains. In addition, one should pick out the appropriate machine-learning model that can support the overall statistical approach; the model,  $\phi$ , must render the complex cognitive mechanisms that correlate a text review with a sentiment label (mostly expressed by a numerical score). At the same time, that model should prove flexible enough to adjust to the various applicative situations.

### B. The proposed framework

To take into account these requirements, the present research adopts a distance-based predictive model, which combines simplicity and modularity, and delegates domain-dependent aspects to the definition of semantic-based metrics. A k-nearest neighbor (kNN) machine [6] supports the model  $\phi$ , thus implementing the simplest empirical-learning strategy in the design of the review-sentiment mapping function. The assignment of a semantic-orientation label to an unlabeled, ‘test’ review is ruled by the  $k$  closest texts, chosen among those included in the training corpus.

A proper metric measures the distance between a pair of reviews, and evaluates how ‘close’ the comparisons are in terms of the sentiments that they express. To pick out those concepts that are expected to convey affective bias in a review, the distance metric relies on a semantic associative network. In addition, the metric takes into account the influence of contextual valence shifters [4]; the research presented here focuses on negatives, as they often flip the valence of a term. The metric combines the information provided by these factors and yields a distance function that improves over a conventional bag-of-words approach.

Figure 1 schematizes the three steps involved in the sentiment classification of a review. The first step requires the definition of a training corpus and a distance metric; in this case, it is assumed that the training corpus includes only positive and negative reviews (two different gray levels in the figure). In the second step, a new review is going to be classified; the distance metric is used to identify in the training corpus the three closest reviews (i.e.,  $k = 3$ ). Finally, in the third step the unlabeled review is tagged according to a majority-rule strategy.

The two main components of the proposed framework will be detailed in the following Section.

### III. K-NEAREST NEIGHBOR

The  $k$ NN algorithm supports an intuitive strategy for classification problems. Let  $TG = \{(r_i, s_i); i = 1, \dots, N_p\}$  be a training set, where  $r_i \in \mathcal{R}^m$  is a review (viewed as a pattern in the feature space), and  $s_i \in \Sigma$  is the corresponding semantic-orientation label. An unlabeled test pattern  $\bar{x}$  is assigned to a semantic category,  $\bar{s}$ , by applying the following procedure:

- Select in TG the set, K, holding the  $k$  patterns that are closest to  $\bar{x}$ ;
- Identify the class,  $\bar{s}$ , that appears most frequently among the elements in the subset S;
- Assign label  $\bar{s}$  to  $\bar{x}$ .

The implementation of this procedure requires a strategy to set the consensus amplitude,  $k$ . The value of  $k$  determines the number of reviews that contribute to the label-prediction process. In principle, one obtains a smoother classification rule as long as  $k$  increases; however, by increasing  $k$  one decreases the locality of the estimation, since the prediction process involves patterns that are more and more distant from the test

point,  $\bar{x}$ . Model selection strategies [6] usually trim the value of  $k$  that is expected to yield the best performance in terms of generalization ability (i.e., the classification error on those patterns that are not in the training set).

Some specific aspects favor the use of the  $k$ NN algorithm in a framework for cross-domain sentiment classification:

- The  $k$ NN algorithm supports empirical learning. This feature seems reasonable when considering that 1) a huge amount of reviews is available on the Internet, and 2) modeling explicitly the function that maps reviews into sentiment labels is a complex task;
- The  $k$ NN algorithm simplifies the learning process, as training a model just implies to acquire a reference corpus. This aspect provides an important advantage over other machine learning algorithms, which exhibit complex training algorithms and that must be re-run after any updating of the training corpus;
- The classification rule adopted by the  $k$ NN may avoid highly non-linear separating functions. At first sight, this feature might compromise the model's ability to represent complex mappings. At the same time, however, the classifier can embed a custom metric in the review-distance measurement. As a result, the supported metric algorithm can explicitly take into account the lexical and semantic rules that one wants to emphasize in the natural language text.

### IV. DEFINING THE METRIC

The main goal of this research is to design a core metric that gives the distance between two reviews according to the sentiment they communicate.

In principle, a review can be represented by using the conventional bag-of-words approach. Thus, given a dictionary (vocabulary),  $\mathcal{V}$ , a review  $\mathcal{R}$  is an ordered sequence of sentences:  $\mathcal{R} = \{s_k; k = 1, \dots, N_s\}$ ; each sentence,  $s_k$ , is a list of terms  $s_k = \{t_j \in \mathcal{V}; j = 1, \dots, N_w\}$ . Common pre-processing steps of the review text lead to that representation: first, the text is split into sentences; then, stop-word removal takes out those terms that either are shorter than three characters or appear in a language-specific list of terms (conjunctions, articles, etc). Finally, a lemmatization process reduces each token to its root term; the WordNet stemmer is adopted toward that end [8].

The distance between two reviews is worked out in the

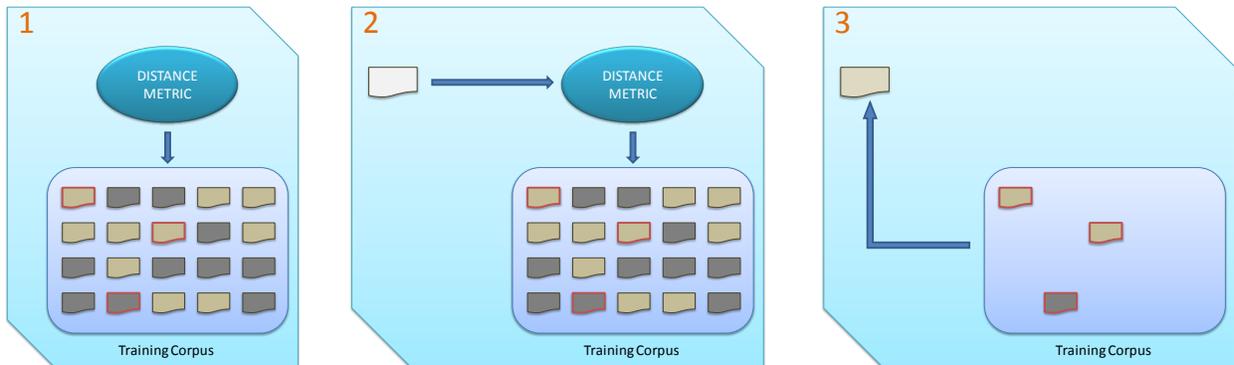


Fig. 1. The three steps involved in the sentiment classification of a review.

feature space that covers the terms in the dictionary. The sentiment-oriented metric distance modulates the relative weights assigned to the various terms, and therefore supports a semantic-driven Mahalanobis distance in the feature space. The modulation strategy considers the presence of contextual valence shifters and the emotional valence of each term.

Contextual valence shifters have been defined in [4]; that work analyzed the lexical phenomena that can cause a shift in the original sentiment communicated by individual lexical items. Valence shifters include negatives (e.g.: ‘not,’ ‘never,’ ‘none,’ etc.), intensifiers (e.g.: ‘rather,’ ‘deeply’), presuppositional items (e.g., ‘barely’), and others. In the research presented here, the semantic distance takes into account and highlights negatives shifters, mostly because they can actually flip the valence of either a term or an entire sentence.

A measure of the emotional valence of a term is obtained from the WordNet-Affect [9] semantic network, which stores the lexical representation of affective knowledge. WordNet-Affect is an extension of the WordNet knowledge base [9], which provides a semantic base for English and is inspired by the psycholinguistic theory of human lexical memory [9]. WordNet [9] connects nouns, verbs, adjectives and adverbs to sets of synonyms (synsets); each synset represents a lexical concept. The same term can appear in several synsets, as it can convey different senses (polysemy). WordNet-Affect relies on the subset of synsets that relate to affective concepts and evoke emotional states. In practice, WordNet-Affect assigns affective labels (a-labels) to those synsets that are expected to express affective states. In this work, only the a-labels designed to express emotional valence have been exploited: positive, negative, ambiguous, and neutral.

The assessment of emotional valence also integrates the information supplied by a specific semantic knowledge base [10]. That work tagged 15726 adjective synsets (taken from EuroWordNet) with a semantic orientation expressed by three tags: positive, negative, neutral. The cognitive knowledge had been collected by a web-based procedure involving several human assessors, each of whom was asked to tag a subset of the available synsets. Then, for each synset, scores were filtered and grouped to improve reliability.

#### A. The metric

The review-distance metric operates in the feature (term) space defined by the dictionary  $\mathcal{V} = \{t_j; j=1, \dots, N_T\}$ . A review  $\mathcal{R}$  is coded by a vector of real-valued weight terms,  $\mathbf{r} = \{w_j; j=1, \dots, N_T\}$ , lying in a space spanned by  $\mathcal{V}$ . The  $j$ -th component,  $w_j$ , of that  $N_T$ -dimensional vector is a non-negative weight, which measures the relevance of the associate term within the review  $\mathcal{R}$ .

The evaluation of such a relevance takes into account three factors: the frequency of term  $t_j$  in the review, the presence of term  $t_j$  in the set of negative shifters, and the emotional valence associated to the term. Given a review,  $\mathcal{R}$  and a dictionary,  $\mathcal{V}$ , the term weights are computed as follows:

#### I. Occurrences

$$\text{for } j = 1 \text{ to } N_T \\ w_j = o_j / N_e$$

where  $o_j$  is the number of occurrences of term  $t_j$  in  $\mathcal{R}$  and  $N_e$  is the total number of terms in  $\mathcal{R}$ .

#### II. Valence shifters

Let  $\Lambda$  be the set of terms that are considered negative shifters

$$\text{for } j = 1 \text{ to } N_T \\ \text{if } t_j \in \Lambda \\ w_j = \lambda \cdot w_j$$

#### III. Affect

Let  $\Pi$  be the set of terms that are tagged as *positive* or *negative*

$$\text{for } j = 1 \text{ to } N_T \\ \text{if } t_j \in \Pi \\ w_j = \pi \cdot w_j$$

The overall distance between two reviews is evaluated at the document level, and the metric computes the similarity between two reviews as follows:

$$\delta(\mathcal{R}_m, \mathcal{R}_n) = \sum_j (w_j^{(m)} - w_j^{(n)})^2 \quad (2)$$

Although a metric working at the sentence level might endow the model with a finer granularity, the intricacies and the ambiguities brought about by sentence-level analysis suggest the adoption of a document level analysis, especially in view of the related computational efforts. Moreover, one should consider that a document-level analysis could better fit cross-domain scenarios, where domain-dependent information plays a lesser role. The use of a coarse-grained approach seems to favor the extraction of useful patterns that are shared across heterogeneous domains.

## V. SETTING UP THE FRAMEWORK

The following pseudo-code outlines the steps required to implement sentiment classification of a review.

#### I. Initialization

- Define a corpus  $\Omega = \{(\mathcal{R}_i, s_i); i = 1, \dots, N_I\}$ , where  $s \in \Sigma$  is a sentiment label and  $\Sigma$  defines the set of labels that can be assigned to a review  $\mathcal{R}$

- Define a metric  $\delta$  that measures the distance between a pair of reviews ( $\mathcal{R}_i, \mathcal{R}_j$ ). Thus,  $\delta(\mathcal{R}_i, \mathcal{R}_j) = z$ , where  $z \in \mathbb{R}^+ \cup \{0\}$ .
- Set a value  $k$  for the kNN algorithm.

## II. Classification

0. *Input*: a test review  $\mathcal{R}^*$

1. *Distance computation*: generate the set  $Z := \{(z_i, i); i = 1, \dots, N_i\}$  where the elements  $z_i$  are computed as

$$\begin{aligned} \text{for } i = 1 \text{ to } N_i \\ z_i := \delta(\mathcal{R}_i, \mathcal{R}^*) \end{aligned}$$

2. *Sorting*: sort the elements in  $Z$  into the set  $Z^*$  in increasing order according to  $z_i$ :

$$Z^* := \{(z_i, i)_m; z_{i,m} \leq z_{i,m+1} \forall m = 1, \dots, N_i\}$$

3. *Nearest neighbors detection*: extract from  $Z^*$  the subset  $N := \{(z_i, i)_q \in Z^*; q = 1, \dots, k\}$

4. *Scoring*: assemble the set  $S := \{s_q; q = 1, \dots, k\}$ , where each element  $s_q$  is obtained as

$$\begin{aligned} \text{derive} \quad & j := i_q; \\ \text{locate} \quad & (\mathcal{R}_j, s_j) \in \Omega; \\ \text{set} \quad & s_q := s_j \end{aligned}$$

5. *Labeling*: assign to  $\mathcal{R}^*$  the sentiment label that appears in  $S$  with the highest frequency; (unlikely) tie cases are marked as neutral.

The adoption of the  $k$ NN algorithm ultimately avoids any training procedure. Furthermore, the computational cost of the classification procedure is linear in the size of the training corpus. This approach actually applies to an arbitrary number of labels in  $\Sigma$ , as opposed to other sophisticated machine-learning models (e.g., SVMs) that are inherently binary classifiers.

On the other hand, the  $k$ NN algorithm is based on memory-based learning, since the classification procedure requires the access to every pattern included in the training corpus. However, memory usage may not represent a crucial issue in modern data mining technologies.

## VI. EXPERIMENTAL RESULTS

The experimental evaluation of the cross-domain approach imposed that training reviews and test reviews belonged to heterogeneous domains. The tests involved two publicly available datasets of reviews, relating to quite distant commercial areas. The first dataset collected reviews from Amazon [3] related to four product categories: books, DVDs, electronics and kitchen appliances. For each product type, a total of 2000 reviews were provided. The second dataset collected 246398 hotel reviews obtained from TripAdvisor [7].

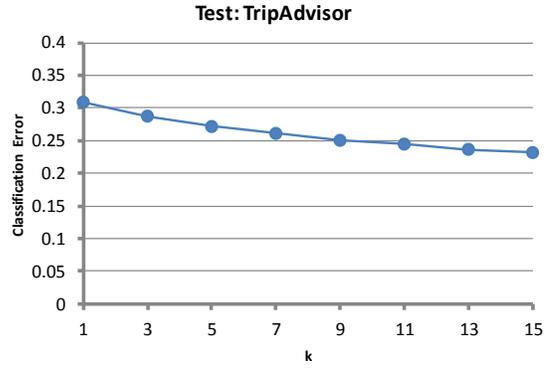


Fig. 2. Classification error on the TripAdvisor dataset.

In both cases, each review had been assigned a discrete rating in the range [1, 5].

The classification framework could directly apply to that multiclass problem; however, to allow comparisons with several works on sentiment classification, the experimental session was remapped into a digital classification problem. Therefore, each review was assigned to one of two categories ("positive" and "negative" sentiment). The reviews having an original rating larger than 3 were labeled as positive; reviews with an original rating smaller than 3 were labeled as negative. All the reviews scoring a rating of 3 were discarded, under the assumption that they expressed a somewhat ambiguous sentiment. After this preprocessing, the Amazon dataset provided 1000 positive reviews and 1000 negative reviews for each product type; the TripAdvisor dataset provided 34158 negative reviews and 170338 positive reviews.

The performance of the framework was evaluated in two different experiments. In the first test, the Amazon dataset provided the training corpus, while classification accuracy was measured on the reviews held in the TripAdvisor dataset. Conversely, in the second experiment the training corpus covered the TripAdvisor dataset, and classification accuracy was evaluated on the reviews from the Amazon dataset. Since the original TripAdvisor dataset conveyed an unbalanced corpus (positive reviews outnumbered negative reviews), the experiment involved a corpus comprising 34000 negative reviews and 34000 positive reviews randomly selected from the dataset.

In both experiments, parameters  $\lambda$  and  $\pi$  were always set as  $\lambda := 2$ ;  $\pi := 2$ . This implied that the relevancies of valence shifters and affective terms doubled in the distance metric. Such set up seemed reasonable in terms of cognitive adequacy, and was confirmed by subsequent empirical evidence.

Figure 2 presents the results obtained in the first experiment (i.e., training on Amazon, test on TripAdvisor). The graph gives the classification error attained versus increasing values of the consensus amplitude,  $k$ , hence the x axis includes the odd values in the range [1, 15]. Empirical evidence points out that the proposed approach attained interesting results, scoring a classification error inferior to 25% with  $k > 11$ .

Figure 3 presents, by using the same format as above, the results obtained in the second experiment (i.e., training on TripAdvisor, test on Amazon). Results show that this experiment involved a more challenging task: the sentiment classifier could score lower error rates than 30%. The fact that the two domains are distant at both the lexical and the semantic level only partially explains the lesser results obtained in the second experiment. The two experiments seem to indicate that the TripAdvisor dataset did not provide a robust, effective training corpus for cross-domain sentiment classification in the Amazon domain, whereas the opposite is not true. From the point of view of empirical learning, this may suggest that the two dataset differs in the ability to provide a representative sample of the data population that one wants to generalize.

Table 1 and Table 2 give the confusion matrix ( $k = 15$ ) for the Amazon test and the TripAdvisor test, respectively. Empirical results show that the errors always distributed over the two classes almost equally. This confirms the robustness of the proposed solution.

The performance obtained in the two experiments compares favorably with previous research in the literature [3]. In fact, experimental validations of cross-domain sentiment classification only involved the product categories of the Amazon dataset. In that situation, the approach recently described in Bollegala et al. [3] featured a classification error ranging between 15% to 24%. Although this research scored higher classification errors, one should notice that the experimental set up adopted in this work was aimed at a harder cross-domain scenario, as the training/test comparison reviews belonged to more distant domains. From this perspective, the performance achieved by the proposed framework can be considered promising.

The technology solution performed satisfactorily in terms of computational efficiency. The overhead was assessed on an Intel Xeon processor (clock frequency @ 2.4 GHz) with a

TABLE I. Confusion matrix for the TripAdvisor test.

		prediction	
		positive	negative
truth	positive	133909	36429
	negative	10971	23187

TABLE II. Confusion matrix for the Amazon test.

		prediction	
		positive	negative
truth	positive	2736	1264
	negative	1166	2834

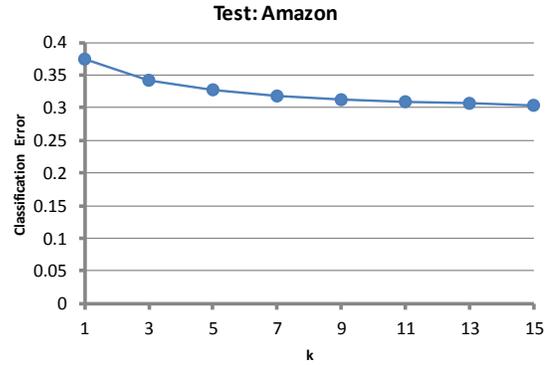


Fig. 3. Classification error on the Amazon dataset.

48GB RAM. To ensure reliability, all tests ran in single-thread mode. The first experiment involved a training corpus with 8000 reviews; the average processing time for classification was evaluated in 30 msec/review. The second experiment involved a larger training corpus, collecting 68000 reviews; the average processing time for classification was evaluated in 180 msec/review. These outcomes seem interesting when considering that the approach proposed in [3] is based on the use of L1 regularized logistic regression, a machine learning algorithm that faces computational burdens when applied to large scale problems.

## VII. CONCLUSIONS

Many factors contribute to complicate sentiment classification in user review mining. Linguistic nuances, first of all, such as the quality of lexicon and peculiarities in syntax, make classical text-mining approaches hardly applicable (from among the 23,628 terms used in one Amazon corpus, 12,030 terms appear once, mostly due to typos and grammar errors, and 15,202 appear only twice). In such a sparse scenario, drawing separating surfaces can be tricky for a sophisticated classifier. Another issue arises at the document level: the patterns lying closest to the separating surfaces draw class boundaries in SVM-related models; thus classification is ultimately ruled by the most uncertain reviews. Robust approaches, such as kernel logistic [3] or the kNN method, seem preferable. Cognitive issues arise from the need for an established model of the text-to-sentiment mapping; this supports holistic approaches, that integrate lexical and semantic information in a flexible manner.

## REFERENCES

- [1] B. Pang and L. Lee, "Opinion Mining and Sentiment Analysis," Foundations and Trend in Information Retrieval, vol. 2, no. 1-2, 2008, pp. 1-135
- [2] E. Cambria and A. Hussain, Sentic Computing: Techniques, Tools, and Applications. Heidelberg: Springer (2012)
- [3] D. Bollegala, D. Weir, and J. Carroll, "Cross-Domain Sentiment Classification using a Sentiment Sensitive Thesaurus," IEEE Trans. on Knowledge and Data Engineering, in press, DOI 10.1109/TKDE.2012.103

- [4] L. Polanyi and A. Zaenen, "Contextual Valence Shifters," *Computing Attitude and Affect in Text: Theory and Applications*, J. Shanahan, Yan Qu and J. Wiebe, eds., Springer, 2006, pp. 1-10
- [5] Hsinchun Chen and D. Zimbra, "AI and Opinion Mining," *IEEE Intelligent Systems*, vol. 25, no. 3, 2010, pp. 74-80
- [6] L. Devroye and T. Wagner, "Nearest neighbor methods in discrimination," *Classification, Pattern Recognition and Reduction of Dimensionality. Handbook of Statistics 2*, P. R. Krishnaiah and L. N. Kanal, eds., 1982, pp. 193-197
- [7] H. Wang, Y. Lu, and C. Zhai, "Latent aspect rating analysis on review text data: a rating regression approach," *Proc. of the 16th ACM SIGKDD Int. Conference on Knowledge Discovery and Data Mining*, 2010, pp. 783-792
- [8] G. A. Miller, "WordNet: A Lexical Database for English," *Communications of the ACM*, vol. 38, no. 11, 1995, pp. 39-41
- [9] C. Strapparava and A. Valitutti, "Wordnet-affect: an affective extension of wordnet," *Proc. of the 4th International Conference on Language Resources and Evaluation*, Lisbon, 2004, pp. 1083-1086
- [10] A. Leoncini, F. Sangiacomo, S. Decherchi, R. Zunino, "Semantic Oriented Clustering of Documents," *Int. Symp. Neural Networks ISNN 2011*, Guilin, China, Part III, Springer LNCS 6677, 2011, pp. 523-529
- [11] E. Cambria, B. Schuller, Y.Q. Xia, and C. Havasi, "New avenues in opinion mining and sentiment analysis," *IEEE Intelligent Systems*, vol. 28, no. 2, 2013, pp. 15-21
- [12] E. Cambria, T. Mazzocco, and A. Hussain, "Application of multi-dimensional scaling and artificial neural networks for biologically inspired opinion mining," *Biologically Inspired Cognitive Architectures*, no. 4, 2013, pp. 41-53