Multi-view Learning for Text Subjectivity Classification

Dinko Lambov[1] and **Gaël Dias**[2] and **João V. Graça** [3]

Abstract. In this paper we consider the problem of building models that have high sentiment classification accuracy across domains. For that purpose, we present and evaluate a method based on co-training using both high-level and low-level features. In particular, we show that multi-view learning combining high-level and low-level features with adapted classifiers can lead to improved results over text subjectivity classification. Our experimental results present accuracy levels across domains of 86.4% combining LDA learning models over high-level features and SVM over bigrams.

1 Introduction

Over the past few years, there have been an increasing number of publications focused on the detection and classification of sentiment and subjectivity in texts. However, as stated in ([1], [2],[4], [6]), most research have focused on the construction of models within particular domains and have shown difficulties in crossing domains. In this paper, we propose to use multi-view learning to maximize classification accuracy across topics. For that purpose, we combine high-level features (e.g. level of affective words, level of abstraction of nouns) and low-level features (e.g. unigrams, bigrams) as different views to learn models of subjectivity which may apply to different domains such as movie reviews or newspaper articles. As stated in [15], SVM classifiers have usually been adopted for sentiment classification based on unigrams and bigrams. However, improvements over high-level features have been reached using LDA (Linear Discriminant Analysis) classifiers. So, our approach combines both SVM and LDA classifiers in the co-training algorithm [3] to obtain maximum performance over two views (high-level and low-level features). Experimental results show that the proposed approach outperforms over 9.2% the methodology proposed by [7] i.e. the SAR (Stochastic Agreement Regularization) algorithm and reaches 86.4% accuracy on average over four different data sets embodying different domains.

2 Related Work

The subjectivity and polarity[4] of language has been investigated at some length. Many features have been used to characterize opinionated texts at different levels: words [8], sentences [10] and texts ([8], [20], [25], [5]). In this section, we will only enumerate research works which focus on cross-domain classification.
One possible approach is to train a classifier on a domain-mixed set of data instead of training it on one specific domain as it is proposed in ([1], [6], [4]). Another possibility is to propose high-level features which do not depend so much on topics such as part-of-speech statistics or other semantic resources as in ([6], [15]). In this case, higher level representations do not reflect the topic of the document, but rather the type of text used. Just by looking at high-level features statistics, improved results can be obtained comparatively to unigram or bigram models (low-level models) when trying to cross domains.
Another approach is to find anchor terms which cross domains and evaluate the correlation between those words and words which are specific to the domain [2]. In this case, pivot features are discovered based on domain mutual information to relate training and target domains. The overall approach extends to sentiment classification the SCL (Structural Correspondence Learning) algorithm. Then, they identify a measure of domain similarity that correlates well with the potential for adaptation of a classifier from one domain to another. Best results across domains reach 82.1% accuracy. Finally, over the past few years, semi-supervised and multi-view learning proposals have emerged. [7] propose a co-regularization framework for learning across multiple related tasks with different output spaces. They present a new algorithm for probabilistic multi-view learning which uses the idea of stochastic agreement between views as regularization. Their algorithm called SAR (Stochastic Agreement Regularization) works on structured and unstructured problems and generalizes to partial agreement scenarios. For the full agreement case, their algorithm minimizes the Bhattacharyya distance between the models of each of the two views. [24] proposes a co-training approach to improve the classification accuracy of polarity identification of Chinese product reviews. First, machine translation services are used to translate English training reviews into Chinese reviews and also translate Chinese test reviews and additional unlabeled reviews into English reviews. Then, the classification problem can be viewed as two independent views: Chinese view with only Chinese features and English view with only English features. They then use the co-training approach to make full use of the two redundant views of features. An SVM classifier is adopted as a basic classifier in the proposed approach. Experimental results show that the proposed approach can outperform the baseline inductive classifiers and more advanced transductive classifiers.
Unlike all proposed methods so far, our approach aims at taking advantage of different view levels. We propose to combine high-level features (e.g. level of affective words, level of abstraction of nouns) and low-level features (e.g. unigrams, bigrams) to learn models of subjectivity which may apply to different domains. For that purpose, we propose a new scheme based on the classical co-training algorithm over two views [3] and join two different classifiers LDA and SVM to maximize the optimality of the approach.

---

[1] University of Beira Interior, Portugal, email: d.lambov@gmail.com
[2] University of Beira Interior, Portugal, email: ddg@di.ubi.pt
[3] L2F INESC-ID INESC-ID,Portugal
[4] Most papers deal with polarity as the essence of subjectivity. However, subjectivity can be expressed in different ways. In this paper, we will focus on subjectivity classification and not just polarity.

3 Characterizing Subjectivity

Many works to date have been concerned with the less ambitious goal of identifying the polarity of sentiment in texts. However, subjectivity can be expressed in different ways as summarized in [4] who identify the following dimensions: evaluation (positive or negative), potency (powerful or unpowerful), proximity (near or far), specificity (clear or vague), certainty (confident or doubtful) and identifiers (more or less), direct expressions, elements of actions and remarks. Based on these assumptions, our methodology aims at classifying texts at the subjectivity level (i.e. subjective vs. objective and not, (positive, negative) vs. objective) taking into account both high-level features which cross domains easily [15] as well as low-level features (unigrams or bigrams) which evidence high precision results within domains [20].

3.1 High-Level Features

Intensity of Affective Words: sentiment expressions mainly depend on some words which can express subjective sentiment orientation. [22] use words from the WordNet Affect lexicon [23] to annotate the emotions. For example horror and hysteria express negative fear, enthusiastic expresses positive emotion, glad expresses joy, and so on and so forth. So, we propose to evaluate the level of affective words in texts as shown in Equation 1.

$$K_1 = \frac{total\ affective\ words\ in\ text}{total\ words\ in\ text} \tag{1}$$

Dynamic and Semantically Oriented Adjectives: [9] consider two features for the identification of opinionated sentences: (1) semantic orientation, which represents an evaluative characterization of word deviation from its semantic group and (2) dynamic adjectives which characterize word ability to express a property in varying degrees. For the present study, we use the set of all adjectives automatically identified in a reference corpus i.e. the set of dynamic adjectives manually identified by [9] and the set of semantic orientation labels assigned as in [8]. So, we propose to evaluate the level of these adjectives in texts as shown in Equation 2.

$$K_2 = \frac{total\ specific\ adjectives\ in\ text}{total\ adjectives\ in\ text} \tag{2}$$

Classes of Verbs: [5] present a method using verb class information. The verb classes they use express objectivity and polarity. To obtain relevant verb classes, they use InfoXtract [21], an automatic text analyzer which groups verbs according to classes that often correspond to their polarity. As InfoXtract is not freely available, we reproduce their methodology by using the classification of verbs available in Levins English Verb Classes and Alternations [17]. So, we propose to evaluate the level of each class of verbs (i.e. conjecture, marvel, see and positive) in texts as in Equation 3.

$$K_3 = \frac{total\ specific\ verbs\ in\ text}{total\ verbs\ in\ text} \tag{3}$$

Level of Abstraction of Nouns: There is linguistic evidence that level of generality is a characteristic of opinionated texts, i.e. subjectivity is usually expressed in more abstract terms than objectivity [15]. Indeed, descriptive texts tend to be more precise and more objective and as a consequence more specific. In other words, a word is abstract when it has few distinctive features and few attributes that can be pictured in the mind. One way of measuring the abstractness of a word is by the hypernym relation in WordNet [19]. In particular, a hypernym metric can be the number of levels in a conceptual taxonomic hierarchy above a word (i.e. superordinate to). For example, chair (as a seat) has 7 hypernym levels: $chair \Rightarrow furniture \Rightarrow furnishings \Rightarrow instrumentality \Rightarrow artifact \Rightarrow object \Rightarrow entity$. So, a word having more hypernym levels is more concrete than one with fewer levels. So, we propose to evaluate the hypernym levels of all the nouns in texts as shown in Equation 4.

$$K_4 = \frac{total\ hypernym\ levels\ for\ nouns\ in\ text}{total\ nouns\ in\ text} \tag{4}$$

Calculating the level of abstraction of nouns should be preceded by word sense disambiguation. Indeed, it is important that the correct sense is taken as a seed for the calculation of the hypernym level in WordNet. However, in practice, taking the most common sense of each word gives similar results as taking all the senses on average [15].

3.2 Low-Level Features

The most common set of features used for text classification is information regarding the occurrences of words or word ngrams in texts. Most of text classification systems treat documents as simple bags-of-words and use the word counts as features. Here, we consider texts as bags-of-words of lemmatized unigrams or lemmatized bigrams for which we compute their TF.IDF weights as in Equation 5 where w_{ij} is the weight of term j in document i, tf_{ij} is the normalized frequency of term j in document i, N is the total number of documents in the collection, and n is number of documents where the term j occurs at least once.

$$w_{ij} = tf_{ij} * \log_2 \frac{N}{n} \tag{5}$$

4 The Multi-View Approach

4.1 Co-Training

The co-training algorithm [3] is a typical bootstrapping method, which starts with a set of labeled data, and increases the amount of annotated data using some amounts of unlabeled data in an incremental way. One important aspect of co-training is that two conditional independent views are required for co-training to work, but the independence assumption can be relaxed. The co-training algorithm is illustrated in Figure 1. In the algorithm, the class distribution in the labeled data is maintained by balancing the parameter values of p and n at each iteration (e.g. positive (resp. negative) examples will be subjective (resp. objective) texts). The intuition of the co-training algorithm is that if one classifier can confidently predict the class of an example, which is very similar to some labeled ones, it can provide one more training example for the other classifier. But, of course, if this example happens to be easy to be classified by the first classifier, it does not mean that this example will be easy to be classified by the second classifier, so the second classifier will get useful information to improve itself and vice versa [13].

In the context of cross-domain sentiment classification, each labeled or unlabeled text has two views of features: high-level features (V1) and low-level features (V2). A basic classification algorithm is also required to construct both models H1 and H2. Typical sentiment classifiers include Support Vector Machines and Maximum Entropy. In this study, we adopt the widely used SVM classifier [11] as well as the LDA classifier which has proved to provide better results than

Given a set L of labeled examples
Given a set U of unlabeled examples
Loop for k iterations

- Train a classifier H1 on view V1 of L
- Train a classifier H2 on view V2 of L
- Allow H1 and H2 to label U
- Add the p positive and n negative most confidently predicted textsto L
- Retrain H1 and H2 on L

Figure 1. The co-training algorithm.

SVM for high-level features [15]. So, we will present results both with SVM or LDA classifiers for the view V1 while only SVM will be applied to the view V2 due to its huge number of features. Moreover, it is important to notice that the unlabeled set of examples U will be from a different domain than the labeled set of examples U. Indeed, the overall idea is that each classifier gets useful information from the other view to improve itself to cross domains.

4.2 SAR Algorithm

[7] propose the SAR (Stochastic Agreement Regularization) algorithm. It models a probabilistic agreement framework based on minimizing the Bhattacharyya distance [12] between models trained using two different views. They regularize the models from each view by constraining the amount by which they permit them to disagree on unlabeled instances from a theoretical model. Their co-regularized objective which has to be minimized is defined in Equation 6 where Li for i = 1,2 are the standard regularized loglikelihood losses of the models p1 and p2, Eu[B(p1,p2)] is the expected Bhattacharyya distance between the predictions of the two models on the unlabeled data, and c is a constant defining the relative weight of the unlabeled data.

$$MinL_1(\theta_1) + L_2(\theta_2) + cE_u[B(p_1(\theta_1), p_2(\theta_1))] \quad (6)$$

In the context of sentiment classification and multi-view learning, [7] is certainly the best reference up-to-date, reaching accuracy levels of 82.8% for polarity detection upon reviews from the kitchen and the dvd domains using random views of unigrams. In this work, we will test SAR on our dataset both on random views of unigrams and random views of bigrams and take its results as baselines[5].

5 Multi-Domain Corpora

To perform our experiments, we used three manually annotated standard corpora and built one corpus based on Web resources which could be automatically annotated as objective or subjective.
The Multi-Perspective Question Answering (Mpqa) Opinion Corpus[6] contains 10.657 sentences in 535 documents from the world press on a variety of topics. All documents in the collection are marked with expression-level opinion annotations. The documents are from 187 different news sources in a variety of countries and date from June 2001 to May 2002. The corpus corpus has been collected and manually annotated with respect to subjectivity as part of the summer 2002 NRRC Workshop on Multi-Perspective Question Answering. Based on the work done by [20] who propose to classify texts based only on their subjective/objective parts, we built a corpus of 100 objective texts and 100 subjective texts by randomly selecting sentences containing only subjective or objective phrases. This case represents the ideal case where all the sentences in texts are either subjective or objective.
The second corpus (Rotten/Imdb) is the subjectivity dataset v1.0[7] which contains 5000 subjective and 5000 objective sentences collected from movie reviews data [20]. To gather subjective sentences, [20] collected 5000 movie review snippets from http://www.rottentomatoes.com. To obtain (mostly) objective data, they took 5000 sentences from plot summaries available from the Internet Movie Database http://www.imdb.com. Similarly to what we did for the Mpqa corpus, we built a corpus of 100 objective texts and 100 subjective texts by randomly selecting only subjective or objective sentences.
The third corpus (Chesley) has been developed by [5] who manually annotated a dataset of objective and subjective documents[8] . It contains 496 subjective and 580 objective documents. Objective feeds are from sites providing content such as world and national news (e.g. CNN, NPR), local news (e.g. Atlanta Journal and Constitution, Seattle Post-Intelligencer), and various sites focused on topics such as health, science, business, and technology. Subjective feeds include content from newspaper columns (e.g. Charles Krauthammer, E.J. Dionne), letters to the editor (e.g. Washington Post, Boston Globe), reviews (e.g. dvdver-dict.com, rottentomatoes.com), and political blogs (e.g. Powerline, Huffington Post). For our purpose, we randomly selected 100 objective texts and 100 subjective texts.
The fourth corpus is based on the idea that Wikipedia conveys objective contents whereas Web Blogs provide subjective contents to its audience [16]. As a consequence, [15] built the automatically annotated Wiki/Blog[9] corpus. They downloaded part of the static Wikipedia dump archive[10] and automatically spidered Web Blogs from different domains. The final corpus contains 200 Mb of downloaded articles from Wikipedia and 100 Mb of downloaded texts from different Web Blogs. These texts are in English and cover many different topics. Due to their characteristics, Wikipedia texts were automatically labeled as objective and Web Blogs automatically labeled as subjective. From this data set, we finally randomly selected 100 objective texts and 100 subjective texts.

6 Experiments

In order to evaluate the difference between high-level features with low-level features, [15] performed a comparative study on the four data sets presented in the previous section. For the high-level features, they took into account 7 features: affective words, semantically oriented adjectives, dynamic adjectives, conjecture verbs, marvel verbs, see verbs and level of abstraction of nouns. For the unigram and bigram models, they used all the lemmas inside the corpora withdrawing their stop words. In Table 1, we summarize the results obtained for the single view classification task using high-level or low-level features.

[5] The SAR package has been implemented for unigrams and bigrams only [7]. Future work will aim at adapting the SAR to other views.
[6] http://www.cs.pitt.edu/mpqa/
[7] http://www.cs.cornell.edu/People/pabo/movie-review-data/
[8] http://www.tc.umn.edu/ ches0045/data/
[9] The corpus is available on the web (url omitted for anonymity)
[10] http://download.wikimedia.org/enwiki/

Table 1. Accuracy for high-level features (HL) and low-level features (LL) across domains in %.

		MPQA	Rotten	Chesley	Wiki
HL	SVM	52.6	69.5	73.9	71.0
	LDA	67.6	70.9	73.6	74.5
LL	Unigram	53.8	63.9	59.9	61.1
	Bigram	54.4	67.1	55.0	57.5

All experiments were performed on a leave-one-out 5 cross validation basis combined with both SVM and LDA classifiers for high-level features and only SVM for low-level features due to the high level of features which does not suit to LDA classifiers. In particular, they used Joachims SVMlight package[11] [11] for training and testing with SVM and the implementation of LDA in the R[12] software for statistical computing. As part-of-speech tagger, they used the MontyTagger module of the free, common sense-enriched Natural Language Understander for English MontyLingua[13] [18]. In order to test models across domains, they proposed to train different models based on one domain only at each time and test the classifiers over all domains together. So, each percentage can be expressed as the average results over all data sets. Best results overall are obtained for high-level features with the Wiki/Blog data set as training set and the LDA classifier with an average accuracy of 74.5%. This result will represent our baseline for single view classification as we aim at showing that multi-view learning can lead to improved results to cross domains. In all our experiments, we will use the same process as in [15] to evaluate accuracy so that values are comparable.

6.1 Results for the SAR algorithm

We first propose to show the results obtained with SAR [7] which represents the state-of-the-art in multi-view learning to cross domains in the field of sentiment analysis. To perform SAR experiments, we used two views generated from a random split of low-level features together with the maximum entropy classifiers with a unit variance Gaussian prior. Indeed, the actual implementation of SAR does not allow to testing it with different views but only with random subsets of views (e.g. unigrams are divided into two subsets: unigrams1 and unigrams2), nor with different classifiers. The results are illustrated in Table 2 exactly in the same way they have been processed in [15].

Table 2. SAR accuracy for low-level features across domains in %.

	MPQA	Rotten	Chesley	Wiki
Unigram	65.3	73.5	72.2	59.2
Bigram	71.6	75.2	77.2	65.1

The results show indeed interesting properties. Models built upon bigrams constantly outperform models based on unigrams. Higher accuracy compared to [15] is reached with less knowledge. Indeed, the baseline with single view classification is 74.5% while 77.2% can be obtained with the SAR algorithm upon a random split of bigrams. One great advantage of only using low-level features is the ability to reproduce such experiments on different languages without further resources than just texts. However, a good training data set will have to be produced as the best results are obtained from the manually annotated corpus Chesley.

6.2 Results for Co-Training

In this subsection, we propose to use the co-training algorithm to combine a first view which contains 7 high-level features (7F) and a second view which contains low-level features (unigrams or bigrams). As a consequence, we expect that the low-level classifier will gain from the decisions of the high-level classifier and will self-adapt to different domains based on the high results of high-level features for crossing domains. In Table 3, we show the results obtained using two SVM classifiers i.e. one for each view. In Table 4, we show the results obtained using an SVM classifier for the low-level view and an LDA classifier for the high-level classifier as we know that LDA outperforms SVM for high-level features.

Table 3. Co-training accuracy with two SVM classifiers across domains in %.

		MPQA	Rotten	Chesley	Wiki
7F	Unigram	61.0	72.3	78.8	62.75
7F	Bigram	66.4	78.1	75.3	85.6

Table 4. Co-training accuracy with one SVM and one LDA classifiers across domains in %.

		MPQA	Rotten	Chesley	Wiki
7F	Unigram	63.3	74.9	79.0	63.5
7F	Bigram	67.4	78.1	68.5	86.4

The benefit from the high-level features is clear based on the results of Tables 3 and 4. The best result is obtained by the combination of high-level features with the LDA classifier and bigram low-level features with the SVM classifier trained over the automatically annotated corpus Wiki/Blogs. In this case, the average accuracy across domains is 86.4% outperforming SAR best performance 77.2%. It is interesting to notice that in almost all cases, bigram low-level features provide better results than only unigrams. The only exception is the Chesley training set. But, it is especially evident for the Wiki/Blog training data set that bigrams drastically improve the performance of the co-training as the difference between unigrams or bigrams as second views is huge. Accuracy results were obtained from the second view classifier, i.e. the low-level classifier. Indeed, while the high-level classifier accuracy remains steady iteration after iteration, the low-level classifier steadily improves its accuracy based on the correct guesses of the high-level classifier[14] . We illustrate the behavior of each classifier in Figure 2.

In order to better understand this situation, we propose a visual analysis of the distribution of the data sets in the space of high-level and low-level features. The goal of this study is to give a visual interpretation of the data distribution to assess how well co-training may perform using high-level and low-level features. If objective and subjective texts can be represented in a distinct way in a reduced space of features, one may expect good classification results. To perform this study, we use a MDS (Multidimensional Scaling) process which is a traditional data analysis technique. MDS [14] allows to displaying

[11] http://svmlight.joachims.org/
[12] http://www.r-project.org/
[13] http://web.media.mit.edu/ hugo/montylingua/

[14] After each iteration, 2 positive (subjective) examples and 2 negative (objective) examples from each classifier are added to the set of labeled data L.

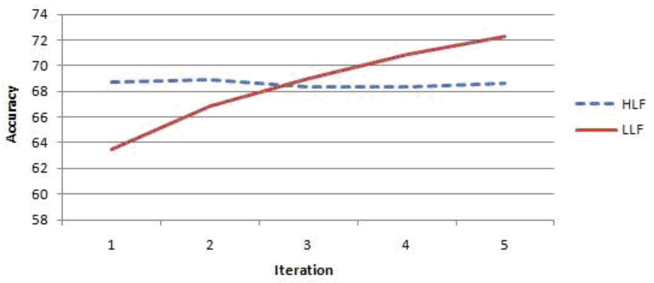

Figure 2. Low-level and high-level accuracies iteration after iteration for the Rotten/Imdb data set with LDA over 7F and SVM over bigrams.

the structure of distance-like data into an Euclidean space. In practice, the projection space we build with the MDS from such a distance is sufficient to have an idea about whether data are organized into classes or not. For our purpose, we performed the MDS process over pairs of corpora represented by low-level features and high-level features to try to visualize how texts evolve in the multidimensional space before and after co-training.

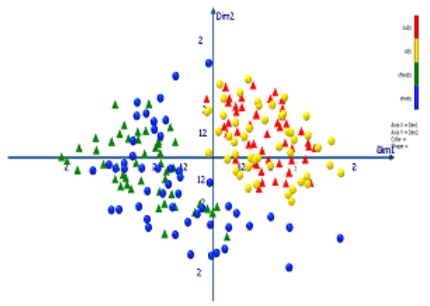

Figure 3. Low-level feature representation of subjective (red and green triangles) and objective (blue and yellow dots) texts before co-training.

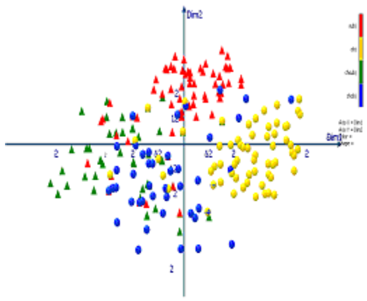

Figure 4. Low-level feature representation of subjective (red and green triangles) and objective (blue and yellow dots) texts after co-training.

In Figures 3 and 4, we graphically represent texts of Rotten/Imdb and Chesley in a reduced space of the low-level features space. Red and green triangles represent subjective texts from Rotten/Imdb and Chesley respectively. Yellow and Blue dots represent objective texts from Rotten/Imdb and Chesley respectively. This visualization clearly shows that after co-training subjective and objective texts from different domains tend to approximate. Comparatively, in Figures 5 and 6, we graphically represent the same texts in a reduced space of the high-level features space. In this experiment, we clearly see that texts do not tend to approximate and remain difficult to separate, as such comforting us in the choice of using low-level classifiers for our classification task using the co-training approach.

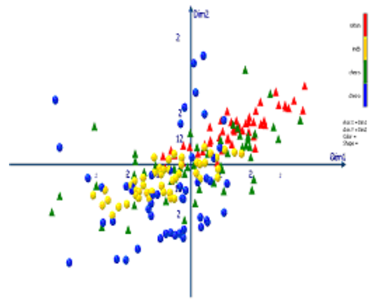

Figure 5. High-level feature representation of subjective (red and green triangles) and objective (blue and yellow dots) texts before co-training.

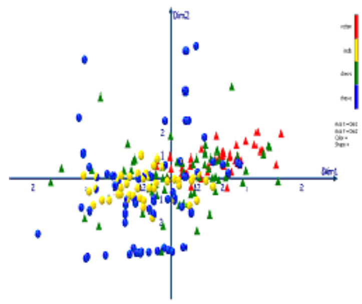

Figure 6. High-level feature representation of subjective (red and green triangles) and objective (blue and yellow dots) texts after co-training.

7 CONCLUSION

Sentiment classification is a domain specific problem i.e. classifiers trained in one domain do not perform well in others. At the same time, sentiment classifiers need to be customizable to new domains in order to be useful in practice. In this paper, we proposed to use the co-training approach to address the problem of cross-domain sentiment classification. For that purpose, we presented different experiments based on multi-view learning algorithms using high-level and low-level features to learn subjective language across domains. The experimental results showed the effectiveness of the proposed approach. Best results showed accuracy of 86.4% across domains compared to 77.2% for the SAR algorithm proposed by [7] and 74.5% for single view classification with LDA proposed by [15]. In future work, we plan to improve the subjectivity classification accuracy by using more than two views as well as customizing the SAR algorithm to receive different types and numbers of views.

REFERENCES

[1] A. Aue and M. Gamon, 'Customizing sentiment classifiers to new domains: a case study', in *Proceedings of International Conference on Recent Advances in Natural Language Processing (RANLP)*, (2005).

[2] J. Blitzer, K. Crammer, A. Kulesza, F. Pereira, and J. Wortman, 'Biographies, bollywood, boom-boxes, and blenders: Domain adaptation for sentiment classification', in *Proceedings of Association for Computational Linguistics*, (2007).
[3] A. Blum and T. Mitchell, 'Combining labeled and unlabeled data with co-training', in *Proceedings of Conference on Computational Learning Theory*, (1998).
[4] E. Boiy, P. Hens, K. Deschacht, and M-F. Moens, 'Automatic sentiment analysis of on-line text', in *Proceedings of the 11th International Conference on Electronic Publishing, Openness in Digital Publishing: Awareness, Discovery & Access*, (2007).
[5] P. Chesley, B. Vincent, L. Xu, and R. Srihari, 'Using verbs and adjectives to automatically classify blog sentiment', in *Proceedings of AAAI Spring Symposium*, (2006).
[6] A. Finn and N. Kushmerick, 'Learning to classify documents according to genre', *Journal of American Society for Information Science and Technology, Special issue on Computational Analysis of Style*, **57**, (2006).
[7] K. Ganchev, J. Graca, J. Blitzer, and B. Taskar, 'Multi-view learning over structured and non-identical outputs', in *In Uncertainty in Artifitial Intelligence*, (2008).
[8] V. Hatzivassiloglou and K.R. McKeown, 'Predicting the semantic orientation of adjectives', in *In Proceedings of the 35th Annual Meeting of the Association for Computational Linguistics and the 8th Conference of the European Chapter of the Association for Computational Linguistics*, (1997).
[9] V. Hatzivassiloglou and J. Wiebe, 'Effects of adjective orientation and gradability on sentence subjectivity', in *In Proceedings of International Conference on Computational Linguistics*, (2000).
[10] Y. Hong and V. Hatzivassiloglou, 'Towards answering opinion questions: Separating facts from opinions and identifying the polarity of opinion sentences', in *In proceedings of Conference on Empirical Methods on Natural Language Processing*, (2003).
[11] T. Joachims, *Learning to Classify Text Using Support Vector Machines*, Springer, 2002.
[12] T. Kailath, 'The divergence and bhattacharyya distance measures in signal selection', *IEEE Transactions on Communications*, **15**, 5260, (1967).
[13] S. Kiritchenko and S. Matwin, 'Email classification with co-training', in *In Proceedings of the 2001 Conference of the Centre for Advanced Studies on Collaborative Research*, (2001).
[14] J. B. Kruskal and M. Wish, *Multidimensional Scaling*, Sage Publications, Beverly Hills, 1977.
[15] D. Lambov, G. Dias, and V. Noncheva, 'High-level features for learning subjective language across domains', in *Proceedings of International AAAI Conference on Weblogs and Social Media ICWSM*, (2009).
[16] D. Lambov, G. Dias, and V. Noncheva, 'Sentiment classification across domains', in *14th Portuguese Conference on Artificial Intelligence EPIA*, (2009).
[17] B. Levin, *English Verb Classes and Alternations*, University of Chicago Press, 1993.
[18] Hugo Liu. Montylingua: An end-to-end natural language processor with common sense. available at: http://web.media.mit.edu/˜hugo/montylingua, 2004.
[19] G.A. Miller, *Wordnet: A Lexical Database*, In Communications of the ACM 38, 1995.
[20] P. Pang and L. Lee, 'A sentimental education: Sentiment analysis using subjectivity summarization based on minimum cuts', in *In Proceedings of the 42nd Annual Meeting of the Association for Computational Linguistics*, (2004).
[21] R. Srihari, W. Li, C. Niu, and T. Cornell, 'Infoxtract: A customizable intermediate level information extraction engine', in *In Proceedings of the HLT-NAACL 2003 Workshop on Software Engineering and Architecture of Language Technology Systems*, (2003).
[22] C. Strapparava and R. Mihalcea, 'Learning to identify emotions in text', in *In Proceedings of the Symposium on Applied Computing*, (2008).
[23] C. Strapparava and A. Valitutti, 'Wordnet-affect: An affective extension of wordnet', in *In Proceedings of the Language Resources and Evaluation International Conference*, (2004).
[24] X. Wan, 'Co-training for cross-lingual sentiment classification', in *In Proceedings of the 47th Annual Meeting of the Association for Computational Linguistics*, (2009).
[25] J. Wiebe, T. Wilson, R. Bruce, M. Bell, and M. Martin, 'Learning subjective language', *Computational Linguistics*, **30**, 277308, (2004).