Semi-Supervised Learning for Opinion Detection

Ning Yu, Sandra Kübler
Indiana University
Bloomington, Indiana
e-mail: {nyu, skuebler}@indiana.edu

***Abstract*—Research on opinion detection has shown that a large number of opinion-labeled data are necessary for capturing subtle opinions. However, opinion-labeled data, especially at the sub-document level, are often limited. This paper describes the application of Semi-Supervised Learning (SSL) to automatically produce more labeled data and explores the potential of SSL to improve transfer of labeled data to new domains. Preliminary results show that SSL performance is very close to a supervised system trained on the full data set and improves performance on out-of-domain data.**

Keywords-opinion detection; semi-supervised learning; domain transfer

I. INTRODUCTION

Opinion detection aims to identify opinion-bearing documents or opinion bearing portions of a document. In order to capture subtle opinion information, a large number of labeled data are always preferable for generating a great variety of opinion-bearing features.

Since an opinion document is usually a mixture of facts and opinions, opinion detection at sub-document level has more practical advantages over opinion detection at the document level. Unfortunately, opinion-labeled data at the sub-document level are often limited. We believe that insufficient labeled data have prevented researchers from effectively testing and evaluating opinion detection strategies.

While obtaining opinion-labeled data can be expensive and difficult, fetching unlabeled data from the Web is cheap and easy. Semi-Supervised Learning (SSL), a class of machine learning methods that require only a small number of labeled data to automatically learn and label unlabeled data, is therefore attractive to opinion detection. SSL methods have achieved satisfactory results for topical classification and many NLP problems, yet they have not been fully investigated for use in opinion detection.

SSL has potential for partial handling of the domain transfer problem: When labeled data are in a non-target data domain, an SSL method can reduce the bias of non-target data by gradually replenishing the labeled dataset with auto-labeled data from the target domain.

This paper focuses on two basic SSL methods, self-training and co-training, both of which are wrapper approaches that can be adapted by many existing opinion detection systems. Experiments for sentence-level opinion detection were designed to: 1) investigate the gain of SSL over Supervised Learning (SL), which uses labeled data only; 2) compare the effectiveness of self-training and co-training relative to the amount of labeled data needed for optimized performance; 3) explore different co-training strategies; and 4) examine the value of SSL in addressing the domain transfer problem.

II. SSL AND RELATED WORK

This section reviews a few SSL techniques and previous opinion detection studies that have used these techniques.

A. *Self-training*

Self-training is the simplest form of SSL: A classifier is repeatedly retrained on a labeled dataset that is extended by auto-labeled data from its own output. Self-training is intuitive and can be applied to any existing classifier that produces confidence scores for its predictions. Self-training treats a classifier as a black box and avoids dealing with any inner complexities. The downside of self-training, however, is that there is no guarantee of its performance: Errors are often reinforced, especially if they appear in early stages of the iterative process.

Some studies (e.g., [10]) have successfully applied self-training to expand an opinion lexicon. In every iteration, the opinion lexicon is augmented with similar terms that have been identified either by referencing an online dictionary (e.g., WordNet) or by analyzing partially labeled data.

Self-training has also been applied to sentence-level opinion detection. The initial classifier can be either an ad hoc rule-based classifier or a supervised classifier. With a self-trained Naïve Bayes classifier, Wiebe and Riloff [13] achieved satisfying recall and modest precision for classifying subjective sentences.

B. *Co-training*

The original co-training algorithm assumes that redundancy exists in the target data domain and thus more than one view can be used to represent and classify each example independently and successfully [3]. For example, an image can be naturally represented by its text description or by its visual attributes. Co-training differs from self-training in that it uses two classifiers with different biases. When labeling new examples, a final classifier is constructed by combining the predictions of the two updated classifiers.

However, a natural split in the feature set is not always available and the lack of natural feature splits has sometimes kept researchers from exploring co-training for opinion detection. Fortunately, studies have proven that the key to co-training is the existence of two largely different initial learners, regardless of whether they are built by using two feature sets or two learning algorithms. Jin, Ho and Sriharia

[7], for example, created disjoint training sets for building two initial classifiers and successfully identified opinion sentences in camera reviews by selecting auto-labeled sentences agreed upon by both classifiers.

C. EM-based SSL

Expectation-Maximization (EM) refers to a class of iterative algorithms for maximum-likelihood estimation when dealing with incomplete data [4]. In [8], EM was combined with an NB classifier to resolve the problem of topical classification, where unlabeled data were treated as incomplete data. The EM-NB SSL algorithm yielded better performance than unsupervised lexicon-based approaches as well as supervised approaches for sentiment classification in different data domains, including blog data [2, 11].

D. S^3VM

Semi-Supervised Support Vector Machines (S^3VMs) are a natural extension of SVMs in the semi-supervised spectrum. They are designed to find the maximal margin decision boundary in a vector space containing both labeled and unlabeled examples. Although SVMs are the most favored supervised learning method for opinion detection, S^3VMs have not yet been applied for opinion detection or its related tasks.

III. Experimental Design

A series of SSL and SL runs were designed to explore and evaluate different strategies for employing SSL for opinion detection over different data domains. The Weka data mining software [6] was used for data processing and classification. LingPipe's [1] EM implementation was used for our EM-NB runs. S^3VMs implemented in SVM^{light} [5] and based on local search were adopted for our S^3VM runs.

A. Datasets and Pre-processing

Two benchmark datasets for sentiment analysis were chosen: the movie review dataset distributed by Pang and Lee [9] consisting of 5000 subjective sentences (or snippets) and 5000 objective sentences; and one news dataset containing 5297 subjective sentences and 5174 objective sentences extracted from the MPQA 2.0 corpus [8] according to the rules described in[10].

We removed only a few stop words. No stemming was conducted since the literature shows no clear gain from stemming in opinion detection. One reason for this is that stemming may actually erase some subtle opinion cues such as past tense verbs. Binary feature values were used, motivated by the brevity of the text unit when classifying opinions at sentence-level, as well as by the characteristics of opinion detection, where occurrence frequency is less influential. Feature selection was optional.

B. Data Split

For both the movie review and news datasets, 5% of the sentences were kept as the evaluation set and were not available during SSL and SL runs; 90% were treated as unlabeled data (U) for SSL runs and i% (i = 1, 2, 3, 4 or 5) as labeled data (L) for self- and co-training runs. For each SSL run, a baseline SL run was designed with the same number of labeled sentences only (i%) and a fully SL run was designed with all available sentences (90% + i%). If effective, an SSL run would significantly outperform its corresponding baseline SL run and approach the performance of a fully SL run.

C. General Settings for SSL

The Naïve Bayes classifier was selected as the initial classifier for self-training because of its ability to produce prediction scores and to work well with a small, labeled dataset.

Parameters for SSL include: (1) threshold k for k iterations. If k is not set, the stopping criterion is convergence; (2) number of unlabeled sentences available in each iteration u ($u \ll U$); (3) number of opinion and non-opinion sentences, p and n, to augment L during each iteration; (4) weighting parameter λ for auto-labeled data. When λ is set to 0, auto-labeled and labeled data are treated equally; when λ is set to 1, feature values in an auto-labeled sentence are multiplied by the prediction score assigned to the sentence.

Default settings were applied for EM-NB and S^3VM.

D. Special Settings for Co-training

For co-training, we investigated strategies for creating two initial classifiers and for selecting auto-labeled data in each iteration to augment L.

1) Initial classifiers construction

Two initial classifiers were generated by: (1) using unigrams and bigrams respectively to create two classifiers based on the assumption that there is enough redundancy in low-order n-grams and high-order n-grams, which also represented different views of an example: content and context; (2) randomly splitting the feature set into two; (3) applying two different learning algorithms (i.e., Naïve Bayes and SVM) that are based on different learning assumptions; and (4) randomly splitting the training set [7].

2) Auto-labeled data selection

Auto-labeled sentences were selected if they were: (1) most confidently labeled by either classifier; (2) confidently labeled by one classifier but more uncertain and thus more useful to the other classifier; (3) assigned a label that both classifiers agreed on; and (4) a fusion of (1) and (2) or (2) and (3).

E. Settings forDomain Transfer

All sentences in the source domain (i.e., movie reviews) were treated as labeled data. 90% of the sentences in the target domain (i.e., news) were treated as unlabeled data, 5% as evaluation data, and 5% were reserved for future experiments. The domain transfer self-training run used labeled data from the source domain and unlabeled data from the target domain.

IV. Results and Discussion

Our preliminary results suggest that SSL is promising for opinion detection, although the contribution of SSL varies in different data domains.

A. SSL vs. SL

Tables I and II report the performance of SSL runs over different numbers of initial labeled sentences and the performance of baseline and fully SL runs.

For movie reviews, Table I shows that SSL, except for S^3VM, always outperformed the corresponding SL baseline: When self-training converge, it achieved improvements in the range of 8% to 34% over baseline SL. The less initial labeled data, the more benefits an SSL run gained from using unlabeled data. For example, using 100 labeled sentences, self-training achieved classification accuracy of 85.2% and outperformed the baseline SL by 33.5%. Although this SSL run was surpassed by the fully SL run using all labeled data by 4.9%, a great amount of effort was saved by labeling 9,400 fewer sentences. Furthermore, although the best SL run was surpassed by its fully SSL run by 2.6%, it needed 9,000 less labeled sentences.

For news articles, however, the advantage of SSL was not as significant. As indicated in Table II, self-training actually hurt the performance of the corresponding baseline SL while co-training only slightly outperformed baseline SL. We suspect that this is due to low baseline accuracy, which decreases the quality of auto-labeled data

For both movie reviews and news articles, EM-NB runs consistently yielded top SSL results. Actually, with only 32 labeled movie review sentences, EM-NB was able to achieve 88% classification accuracy, which is close to the best performance of the simple NB self-training using 300 labeled sentences. This implies that the problem space of opinion detection can be successfully described by the mixture model assumption of EM.

The preliminary exploration of different parameter settings for both self- and co-training found no significant benefit gained by setting the weight parameter λ or applying feature selection. A larger number of u was also found unhelpful. Further investigations are needed for an in-depth explanation. The poor performance of S^3VMs also needs to be further investigated.

B. Co-training vs. Self-training

The best co-training runs reported here used unigrams to train one classifier and the union of unigrams and bigrams to train the other classifier. Bigrams were not used alone because they are weak features when extracted from limited labeled data. The auto-labeled sentences were selected based on a fusion strategy of (1) and (2), as described in auto-labeled data selection strategies in experimental design. Among different auto-labeled data selection strategies, this fusion strategy was found to be most helpful. One possible explanation is that, because our initial classifiers violated the original co-training assumptions, forcing agreement between confident predictions helped to maintain the relatively high precision. Adding an SVM classifier for co-training did not work well because, even with the logistic model to output probabilistic scores for the SVM classifier, the difference in probabilities was too small for selecting a small number of top predictions.

TABLE I. SSL AND SL AVERAGE PERFORMANCE ON MOVIE REVIEWS (CLASSIFICATION ACCURACY %)

Run Type	# Labeled Examples					
	100	***200***	***300***	***400***	***500***	***all***
Self-training*	85.2	86.6	87.0	87.2	86.6	89.4
SL_base	63.8	73.6	77.2	79.4	80.2	
Co-training*	85.0	87.2	88.0	87.6	88.2	90.4
SL_base**	67.6	75.2	80.2	81.8	84.6	
EM-NB	88.1	88.7	88.6	88.4	89.0	91.6
SL_base	73.5	78.7	81.3	82.8	83.9	
S^3VM	59.0	68.4	67.8	67.0	75.2	90.0
SL_base	70.0	72.8	75.6	76.2	80.0	

* Settings: Naïve Bayes classifier, k=0, u=20, p=2, n=2, λ =0, no feature selection.

** The final prediction is decided by the highest predict score generated by two NB classifiers.

TABLE II. SSL AND SL AVERAGE PERFORMANCE ON NEWS ARTICLES (CLASSIFICATION ACCURACY %)

Run Type	# Labeled Examples					
	100	***200***	***300***	***400***	***500***	***all***
Self-training	60.1	65.8	66.4	67.7	67.6	76.9
SL_base	60.5	64.3	69.5	69.5	71.4	
Co-training	61.6	71.4	69.3	69.8	69.7	76.0
SL_base	60.7	64.3	69.3	66.8	66.4	
EM-NB	68.8	69.5	69.5	70.0	70.4	78.6
SL_base	63.5	66.2	67.3	69.0	70.3	
S^3VM	60.4	61.5	62.3	61.2	65.6	76.5
SL_base	61.2	64.4	66.9	68.3	70.0	

In order to examine if co-training made better use of labeled data than self-training, Fig. 1 illustrates their performance over time. The straight lines correspond to fully SL; asymptotic curved lines sketch the SSL runs. Overall, co-training runs had better performance than self-training runs since their curved lines are much closer to the corresponding straight line. Co-training runs also reached optimized performance faster since their curved lines approach the straight line more quickly. For instance, with 500 labeled sentences, a self-training run reached an optimized classification accuracy of 88.2% after labeling around 4,828 sentences, while the co-training run reached its optimized performance of 89.4% after labeling only 2,588 sentences.

C. Domain Transfer

Even without any explicit domain adaptation methods, simple self-training alone is promising for dealing with the domain transfer problem.

9,500 labeled movie review sentences were used to train a Naïve Bayes classifier C_m. Although C_m produced a fairly good classification accuracy of 89.2% on movie review data,

it generated a poor accuracy of 64.14% on news data. This shows the severity of the domain transfer problem. A self-training run with C_m as the initial classifier using unlabeled data from the news domain was able to make this problem less severe: It achieved a classification accuracy of 75.04% and thus surpassed the SL run using C_m by 17%.

To further understand how well SSL handled the domain transfer problem, a fully SL run that used all labeled news sentences was performed. This fully SL run achieved 3% higher classification accuracy in the news domain than the SSL run, which did not use any labeled news data.

V. Conclusion and Future Work

This paper has explored SSL for tackling the challenge of insufficient labeled data in sentence-level opinion detection. Self-training and co-training were mainly investigated because of their broad application. EM-NB and S^3VMs were also examined. Various co-training strategies that had not yet been studied for opinion detection were proposed and partially tested. To our knowledge, it is the first study using SSL for handling the domain transfer problem.

Based on the results shown here, we conclude that: (1) With limited labeled data, SSL runs significantly outperform corresponding baseline SL runs and approach the performance of fully SL runs for movie reviews; (2) EM-NB consistently makes a positive contribution to opinion detection while S^3VMs always hurt the performance; (3) Even when the original co-training assumptions are violated, carefully designed co-training runs are more effective than self-training runs relative to the amount of labeled data needed for optimized performance; and (4) SSL is valuable in resolving the domain transfer problem.

In the future, we will test more data domains. We are particularly interested in blog data, which we believe to be more challenging for opinion detection than reviews and news data because of the spontaneous speech characteristics of blogs. Since a high-precision classifier is critical for generating auto-labeled data, we will make efforts to increase the precision of the initial classifier(s) for SSL. For example, we plan to integrate high-quality opinion lexicon(s) into the initial classifier.

Other interesting directions can also be explored for co-training: using rule learner RIPPER and Naïve Bayes classifiers, which are significantly different, and using lexicon- and machine learning-based classifiers.

References

[1] Alias-i. 2008. LingPipe 4.0.0. http://alias-i.com/lingpipe (accessed October 1, 2008)

[2] A. Aue and M. Gamon (2005). Customizing sentiment classifiers to new domains: A case study. In *Proceedings of International Conference Recent Advances in Natural Language Processing (RANLP-2005)*, September 21-23, 2005, Borovets, Bulgaria.

[3] A. Blum and T. Mitchell, "Combining labeled and unlabeled data with co-training," Proc. 11th Annual Conference on Computational Learning Theory, 1998, pp. 92-100.

[4] A. P. Dempster, N. M. Laird, and D. B. Rubin (1977). Maximum likelihood from incomplete data via the EM algorithm. *Journal of the Royal Statistical Society*, 39, 1-38.

[5] T. Joachims, "Making large-scale SVM learning practical," In B. Schölkopf, C. J. C. Burges & A. J. Smola (Eds.), Advances in Kernel Methods - Support Vector Learning: MIT Press, 1999.

[6] M. Hall, E. Frank, G. Holmes, B. Pfahringer, P. Reutemann, and I.H. Witten, "The WEKA data mining software: An update," SIGKDD Explorations, 2009, pp. 10-18.

[7] W. Jin, H. H. Ho, and R. K. Srihari, "OpinionMiner: A novel machine learning system for web opinion mining," Proc. 15th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, 2009, pp. 1195-1204.

[8] K. Nigam, A. K. Mccallum, S. Thrun, and T. Mitchell (1999). Text classification from labeled and unlabeled documents using EM. *Machine Learning*, 39, 103-134.

[9] B. Pang and L. Lee, "A sentimental education: Sentiment analysis using subjectivity summarization based on minimum cuts," Proc. 42nd Annual Meeting on Association for Computational Linguistics, 2004, pp. 271-278.

[10] E. Riloff, J. Wiebe, and T. Wilson, "Learning subjective nouns using extraction pattern bootstrapping," Proc. 7th Conference on Natural Language Learning at HLT-NAACL 2003, pp. 25-32.

[11] H. Takamura, T. Inui, and M. Okumura (2006). Latent variable models for semantic orientations of phrases. In *Proceedings of the 11th Conference of the European Chapter of the Association for Computational Linguistics (EACL)*, (pp. 201-208).

[12] J. Wiebe, T. Wilson, and C. Cardie, "Annotating expressions of opinions and emotions in language," Language Resources and Evaluation, 39(2), 2005, 165-210.

[13] J. Wiebe and E. Riloff, "Creating subjective and objective sentence classifiers from unannotated texts," Proc. 6th International Conference on Intelligent Text Processing and Computational Linguistics, 2005, pp. 486-497.

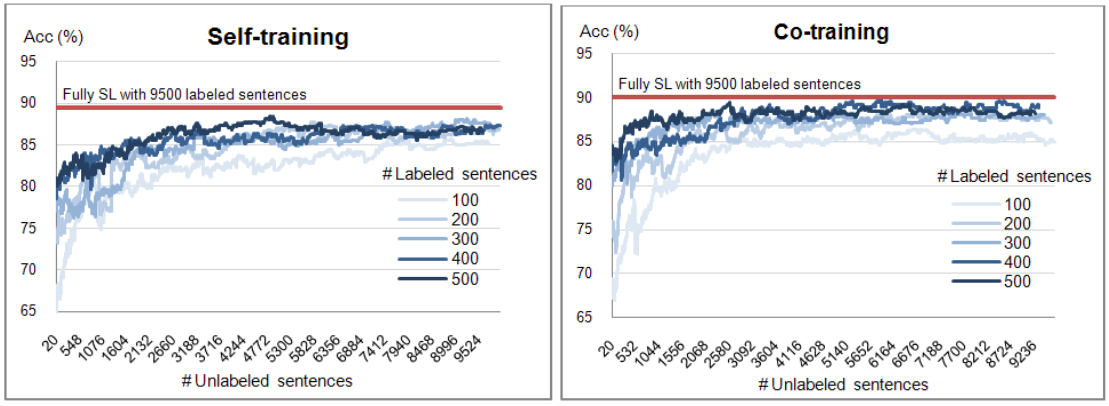

Figure 1. Performance of SSL runs over different numbers of labeled/unlabeled sentences.