

A Vector Space Approach for Aspect Based Sentiment Analysis

Abdulaziz Alghunaim, Mitra Mohtarami, Scott Cyphers, James Glass

MIT Computer Science and Artificial Intelligence Laboratory

Cambridge, MA 02139, USA

azizkag@mit.edu, {mitra, cyphers, glass}@csail.mit.edu

Abstract

Vector representations for language has been shown to be useful in a number of Natural Language Processing tasks. In this paper, we aim to investigate the effectiveness of word vector representations for the problem of *Aspect Based Sentiment Analysis*. In particular, we target three sub-tasks namely *aspect term extraction*, *aspect category detection*, and *aspect sentiment prediction*. We investigate the effectiveness of vector representations over different text data and evaluate the quality of domain-dependent vectors. We utilize vector representations to compute various vector-based features and conduct extensive experiments to demonstrate their effectiveness. Using simple vector based features, we achieve F1 scores of 79.91% for aspect term extraction, 86.75% for category detection, and the accuracy 72.39% for aspect sentiment prediction.

1 Introduction

Natural language representation in continuous vector space has been successfully used in many NLP tasks (Al-Rfou et al., 2013; Bansal et al., 2014; Bowman et al., 2014; Boyd-Graber et al., 2012; Chen and Rudnicky, 2014; Guo et al., 2014; Iyyer et al., 2014; Levy and Goldberg, 2014; Mikolov et al., 2013c). Previous research attempted to employ vector representations to present the *syntactic* and *semantic* information in textual content. In this paper, we aim to investigate the effectiveness of vector space representations for *Aspect Based Sentiment*

Analysis in which we aim to capture both *semantic* and *sentiment* information encoded in user generated content such as product reviews.

Sentiment analysis or *opinion mining* deals with computational analysis of people’s opinions, sentiments, attitudes and emotions towards target entities such as products, organizations, individuals, topics and their attributes (Liu, 2012). The majority of early approaches to this research problem (Pang et al., 2002; Pang and Lee, 2005; Baccianella et al., 2009) attempted to detect the overall sentiment of a sentence, paragraph, or text span regardless of the entities (e.g., *restaurants*) and their aspects (e.g., *food*, *service*) expressed in context. However, only considering overall sentiments fails to capture the sentiments over the aspects on which an entity can be reviewed (Lu et al., 2011). For example, although the restaurant review shown in Table 1 might express an overall positive sentiment, it additionally expresses a positive sentiments toward the restaurant’s food and service, as well as negative sentiment toward the restaurant’s ambiance. To achieve this aim, aspect based sentiment analysis attempts to extract the aspects (or semantic labels) of given target entities and the sentiment expressed towards each aspect (Hu and Liu, 2004; Liu, 2012). For this purpose, three sub-tasks need to be addressed: (1) *aspect term extraction*, (2) *aspect category detection*, and (3) *aspect sentiment prediction*. We briefly describe these sub-tasks here:

Aspect term extraction identifies aspect terms (or semantic labels) appeared in a given text about a target entity. For instance, in the review in Table 1, the aspects are “orecchiette with sausage and

*Our agreed favorite is the **orecchiette with sausage and chicken** and usually the **waiters** are kind enough to split the **dish** in half so you get to sample both **meats**. But, the **music** which is sometimes a little too heavy for my taste.*

Table 1: An example of restaurant review.

chicken”, “waiters”, “dish”, “meats” and “music”, and the target entity is “restaurant”. Multi-word aspect terms are treated as single aspect, like “orecchiette with sausage and chicken” in the example.

Aspect category detection identifies (latent) aspect categories available in a given text. Aspect categories are coarser than aspect terms, and they do not necessarily occur as terms in the text. For example, the review in Table 1 contains the latent aspect categories “food”, “service”, and “ambiance”. Aspect categories are often considered as predefined categories (e.g., “price”, “food”) with respect to the target entities.

Aspect sentiment prediction identifies the sentiment toward aspect terms as *positive*, *negative*, *neutral*, or *conflict* (i.e., both positive and negative) for a given set of aspect terms in a text. For example, in the review in Table 1, the aspects “orecchiette with sausage and chicken” and “waiters” are positive, while “music” is negative, and “dish” and “meats” are neutral.

To tackle the above problems and investigate the utility of vector representation models for aspect based sentiment analysis, we present a supervised approach in which vector representations of aspect terms and categories are effectively utilized for aspect based sentiment analysis. Our approach outperforms the baselines and provides significant performance using the simple vector-based features as compared to previous approaches using different text-based features (Pontiki et al., 2014).

The remainder of this paper describes our approach (Section 2), followed by experimental results and analysis (Section 3), and finally conclusion.

2 Method

Distributed vector representations, described by Schütze (Schütze, 1992a; Schütze, 1992b), associate similar vectors with similar words and phrases.

These vectors provide useful information for the learning algorithms to achieve better performance in Natural Language Processing tasks (Mikolov et al., 2013c). Most approaches to computing vector representations use the observation that similar words appear in similar contexts (Firth, 1957; Sahlgren, 2006; Mikolov, 2012; Socher, 2014).

To compute the vector representations of words, we use the skip-gram model of Word2Vec (Mikolov, 2014; Mikolov et al., 2013a; Mikolov et al., 2013b; Mikolov et al., 2013d). The Skip-gram model aims to find word representations that are useful for predicting the surrounding words in a sentence or document (Mikolov et al., 2013b). The model needs a large amount of unstructured text data for training the word vector representations.

When training the skip-gram model we use the GoogleNews dataset (Mikolov, 2014) that contains 3 million unique words and about 100 billion tokens. In addition, to account for the effect of domain information on the quality of word representations, we employ a dataset of restaurant reviews¹ from Yelp that contains 131,778 unique words and about 200 million tokens. We constructed 300-dimensional word vectors for all these words.

We propose to utilize word vector representations to compute vector-based features for the three sub-tasks of aspect based sentiment analysis. We employ these features in a supervised learning setting to address the above tasks. Our data (reviews) are first analyzed by Stanford tokenizer (Manning et al., 2010), POS-tagger (Toutanova et al., 2003) and dependency-tree extractor (de Marneffe and Manning, 2008). Then, the pre-processed data and word representations are used to compute task-specific features as explained in the following subsections.

2.1 Aspect Term Extraction

The objective of this sub-task is to extract aspect terms from reviews with respect to a target entity (e.g. restaurant) as explained in Section 1. This task can be considered as part of Semantic Role Labeling (SRL). Previous research has shown that Conditional Random Fields (CRFs) (Lafferty et al., 2001)

¹This dataset is available on: http://www.yelp.com/dataset_challenge.

and sequence tagging with Structural Support Vector Machines (SVM-HMM) (Altun et al., 2003) are effective for the SRL task (Cohn and Blunsom, 2005). As such, we employ CRFsuite (Okazaki, 2007) and SVM-HMM (Altun et al., 2003) with word vector representations as features to label the token sequence with respect to two possible tags: “Aspect” and “Not-Aspect”, where an aspect can be multi-word. To the best of our knowledge, this is the first attempt of solving the problem of aspect term extraction using CRFsuite or SVM-HMM with vector representations as features. Furthermore, in addition to the above vector features, we employ the POS-tags information as an extra feature. This is mainly because “nouns” are strong candidates to be aspects (Blinov and Kotelnikov, 2014; Pontiki et al., 2014). However, this feature is more effective for the single term aspects as we will discuss in Section 3.

2.2 Aspect Category Detection

The objective of this sub-task is to detect the aspect categories expressed in a sentence with respect to a given set of categories (e.g., *food*, *service*, *price*, *ambience*, *anecdotes/miscellaneous*) as explained in Section 1. Since a sentence can contain several categories, we employ multi-label one-vs-all Support Vector Machines (SVMs) in conjunction with the following vector-based features for a given sentence:

Normalized Average Vector (NAV) is obtained by averaging the vector representations of the words in the sentence. That is, given a sequence of words $S = w_1, w_2, \dots, w_n$, the normalized average vector is computed as follows:

$$NAV = \frac{\frac{1}{N} \sum_{i=1}^N v_i}{\left| \frac{1}{N} \sum_{i=1}^N v_i \right|} \quad (1)$$

where N is the number of words, v_i is the vector representation of w_i in the sentence, and $|x|$ means the $L2$ - norm of x . In addition, we only consider adjectives, adverbs, nouns, and verbs to compute the NAV. This is because these word types capture most semantic and sentiment information in a sentence.

Token Numbers (TN) is number of words in sentence that used to compute NAV. Although NAV is effective for this task, some information like TN is missing during the averaging process.

Category Similarities (CS) are computed for each predefined aspect category. To compute CS, we first identify a set of words (called *seeds*) for each category by selecting top 20 nearest word vectors to the vector of category name. Then, for each category, we compute the cosine similarity between its seed vectors and the word vectors of the input sentence. We consider the maximum cosine similarity as a feature representing the similarity between the category and the input sentence.

2.3 Aspect Sentiment Prediction

The objective of this task is to predict the sentiments for a given set of aspects in a sentence as positive, negative, neutral and conflict (i.e., both positive and negative) as explained in Section 1. For this task, we apply one-vs-all SVM and the following vector-based features for a given aspect:

Average Dependency Vector (ADV) is obtained by averaging the vector representations of the *dependency words* (DW) for the aspect. We define dependency words for an aspect as the words that modify or modified by the aspect in dependency tree of the input sentence.

Rating Vectors (RV) are the same as ADV features but they are computed using the vector representations trained on different subsets of our data. We have five subsets, each subset contains only reviews with a specific review rating. Ratings range from 1 (strong negative) to 5 (strong positive). Previous researches showed the impact of the word (w) distribution over different ratings (r) to compute the sentiment of the word (i.e., $P(r|w)$) (de Marneffe et al., 2010) and construct opinion lexicon (Amiri and Chua, 2012). Using this feature, we can investigate the distribution of words and their vector representations in different ratings.

Positive/Negative Similarities (PNS) are obtained by computing the highest cosine similarity between DW vectors and the vectors of a set of positive/negative sentiment words. The sentiment words are automatically computed by selecting top 20 of nearest neighbor word vectors to the vectors of the word “excellent” for positive and “poor” for negative seeds. Furthermore, the difference between the positive and negative similarities is used as an additional feature.

Categories	food	srvc.	price	amb.	misc.	Total
Train	1,232	597	321	431	1,132	3,713
Test	418	172	83	118	234	1,025

Table 2: Category distributions over the dataset.

3 Evaluation and Results

We evaluate our vector-based approach on the aspect term extraction, aspect category detection, and aspect sentiment prediction tasks. We use the restaurant review dataset provided by (Pontiki et al., 2014; Ganu et al., 2009) that contains 3,041 training and 800 test sentences. The training dataset contains 3,693 aspects and 3,713 categories, and test dataset contains 1,134 aspects and 1,025 categories. In the dataset, the predefined aspect categories are *food*, *service*, *price*, *ambiance*, *anecdotes/miscellaneous*, and Table 2 shows the distributions of these categories over the dataset.

Aspect Term Extraction The results of our vector-based approach for this task are shown in Table 3. As explained in Section 2.1, we employ CRFs and SVM-HMM for this task. As features we utilize POS-tags of aspect terms and their vector representations computed by `word2vec` trained on Yelp (Y) or GoogleNews (G) data. Their corresponding results are shown in the fourth and fifth rows of the table. These results indicate the vector representations trained on Yelp data leads to a high performance in both SVM and CRF. This is while the GoogleNews dataset contains a larger vocabulary of around 3M words as compared to the Yelp data with around 100K words. This implies the effectiveness of the domain in the quality of word representations.

To evaluate the effectiveness of the vector-based features, we repeat our experiments with only POS-tags of the aspect terms. The performance is significantly dropped, as shown in the third row of the table. Although the nouns can be strong candidates for aspects (Blinov and Kotelnikov, 2014), the majority of aspects, like multi-word aspects cannot be captured by only considering their POS-tags.

The first cell of Table 3 shows the F1 performance of 47.15% produced by our baseline (Pontiki et al., 2014). The baseline creates a dictionary of aspect terms from the training data, and then a given sequence of words are tagged as aspects by looking up

Baseline-F1 = 65.65		SVM (C = 0.1)	
Features	Precision	Recall	F1
NAV (Y)	89.02	80.68	84.64
NAV + TN (Y)	90.42	81.07	85.49
NAV + TN + CS (Y)	91.18	82.73	86.75
NAV + TN + CS (G)	91.51	81.07	85.98

Table 4: Results for the aspect category detection task.

the dictionary. This approach cannot handle the out of vocabulary aspects.

Aspect Category Detection The results of our vector-based approach for this task are shown in Table 4. As explained in Section 2.2, SVMs are applied to this task with a combination of *Normalized Average Vector* (NAV), *Token Numbers* (TN) and *Category Similarities* (CS) features for a given sentence. These features employ the `word2vec` trained on Yelp (Y) or GoogleNews (G) to obtain the vector representations. Their corresponding results are shown in the fifth and sixth rows of the table. The results imply the impact of our vector-based features that lead to the highest performance using the Yelp data.

To evaluate the effectiveness of above vector-based features, we repeat our experiments with different combination of them. It leads to the lower performances by using NAV and TN and ignoring the CS, as shown in the forth row of Table 4., and by using NAV and ignoring both CS and TN, as shown in the third row of the table.

The first cell of Table 4 shows an F1 performance of 65.65% obtained by the baseline (Pontiki et al., 2014). Given a sentence, the baseline first retrieves a set of K similar sentences from the training data. The similarity of two sentences is then determined by computing the Dice Coefficient between the sets of distinct words in the two sentences (Pontiki et al., 2014). Finally, the input sentence is tagged by the most frequent aspect categories appeared in the K retrieved sentences. The limitation of this approach is that it employs the text-based similarity measure to measure the semantic similarity between the sentences. However, the results in the table shows that the vector-based features can better capture the semantic similarity between the sentences as compared to the text-based features.

Baseline-F1 = 47.15	CRF Suite			SVM HMM		
Features	Precision	Recall	F1	Precision	Recall	F1
POS-tags	44.09	9.87	16.13	44.58	9.43	15.57
POS-tags + word2vec (Y)	82.38	72.57	77.16	77.33	78.83	78.08
POS-tags + word2vec (G)	82.69	74.16	78.19	76.88	74.51	75.68

Table 3: Results for the aspect term extraction task.

Baseline-Accuracy = 64.28	SVM (C = 0.1)			
Features	Pos-F1	Neg-F1	Neu-F1	Accuracy
ADV (Y)	82.70	52.30	31.39	71.34
RV (Y)	83.26	51.79	32.85	71.95
RV + PNS (Y)	83.48	53.29	32.97	72.39

Table 5: Results for the aspect sentiment prediction task.

Aspect Sentiment Prediction The results of our approach for this task are shown in Table 5. The SVMs are applied to this task and the parameter C for SVMs is optimized through cross-validation on training data. The third row of the table shows the results when we use the *Average Dependency Vector* (ADV) computed based on word2vec trained on the whole Yelp (Y) data. As explained in Section 2.3, to investigate the distribution of words (Amiri and Chua, 2012) and their vector representations over different ratings, we present *Rating Vectors* (RV). RV features include 4 ADVs in which four vector representations for a word are computed on Yelp reviews with ratings 1, 2, 4, and 5, respectively. Reviews with the rating 3 are not considered, because they are mostly of neutral or conflict orientation. Using RV results in a better performance, as shown in the fourth row of Table 5. However, there is not a significant difference between the results of experiments with RV and ADV. The reason is that most of the reviews in the Yelp data have positive ratings (i.e., ratings 4 and 5) and as such the distributions of words does not dramatically changed as compared to the whole review data.

The highest performance is achieved when we use the combination of RV and *Positive/Negative Similarities* (PNS) features, as shown in the fifth row of the Table 5. Since the vector representations for some positive and negative words (e.g., good and bad) are similar, PNS feature provides more information for a classifier to distinguish between these

vectors by defining a set of positive and negative vectors, as explained in Section 2.3.

The first cell of Table 5 shows a performance of 64.28% obtained by our baseline (Pontiki et al., 2014). The baseline tags a given aspect in a test sentence by the most frequent sentiment for the aspect in top K similar training sentences to the test sentence. In addition, for the out of vocabulary aspects, the majority sentiment over all aspects in training data will be assigned.

4 Related Work

Previous works on aspect based sentiment analysis (Liu, 2012; Pang and Lee, 2008) attempted to tackle sentiment and semantic labeling using different approaches such as sequence labeling (Choi and Cardie, 2010; Yang and Cardie, 2013), syntactic patterns (Zhao et al., 2012; Xu et al., 2013; Zhou et al., 2013), topic models (Lu et al., 2011). While some works first separate the semantic and sentiment information and then label them (Mei et al., 2007; Zhao et al., 2010), some other previous works presented joint models for joint semantic and sentiment labeling (Lin and He, 2009; Jo and Oh, 2011).

Vector representations for words and phrases has been found useful for many NLP tasks (Al-Rfou et al., 2013; Bansal et al., 2014; Bowman et al., 2014; Boyd-Graber et al., 2012; Chen and Rudnicky, 2014; Guo et al., 2014; Iyyer et al., 2014; Levy and Goldberg, 2014). Given the success of the previous works the effectiveness of recursive neural networks in re-

lating semantically similar words, in this research, we investigated the impact of word representations techniques for aspect based sentiment analysis. In particular, we aimed to employ vector-based features using word representations to capture both semantic and sentiment information.

5 Conclusion

In summary, we employed vector representations of words to tackle the problem of *Aspect Based Sentiment Analysis*. We introduced several effective vector-based features and showed their utility in addressing the aspect term extraction, aspect category detection, and aspect sentiment prediction sub-tasks. Our vector space approach using these features performed well compared to the baselines. To further improvement, these vector-based features can be combined with text-based features used typically for each sub-task.

Acknowledgments

This research was supported in part by the Qatar Computing Research Institute (QCRI). In addition, we would like to thank Maria Pontiki from the Institute for Language and Speech Processing for providing us access to the database.

References

- Rami Al-Rfou, Bryan Perozzi, and Steven Skiena. 2013. Polyglot: Distributed word representations for multilingual NLP. *CoRR*, abs/1307.1662.
- Yasemin Altun, Ioannis Tsochantaridis, and Thomas Hofmann. 2003. Hidden markov support vector machines.
- Hadi Amiri and Tat-Seng Chua. 2012. Mining slang and urban opinion words and phrases from cqa services: An optimization approach. In *WSDM*, pages 193–202, New York, NY, USA. ACM.
- Stefano Baccianella, Andrea Esuli, and Fabrizio Sebastiani. 2009. Multi-facet rating of product reviews. In *ECIR*, pages 461–472, Berlin, Heidelberg. Springer-Verlag.
- Mohit Bansal, Kevin Gimpel, and Karen Livescu. 2014. Tailoring continuous word representations for dependency parsing. In *ACL*, pages 809–815, Baltimore, MD, USA.
- Pavel Blinov and Eugeny Kotelnikov. 2014. Blinov: Distributed representations of words for aspect-based sentiment analysis at semeval 2014. In *Proceedings of the 8th International Workshop on Semantic Evaluation (SemEval 2014)*, pages 140–144. Association for Computational Linguistics.
- Samuel R. Bowman, Christopher Potts, and Christopher D. Manning. 2014. Recursive neural networks for learning logical semantics. *CoRR*, abs/1406.1827.
- Jordan L. Boyd-Graber, Brianna Satinoff, He He, and Hal Daumé III. 2012. Besting the quiz master: Crowdsourcing incremental classification games. In *EMNLP-CoNLL 2012, July 12-14, 2012, Jeju Island, Korea*, pages 1290–1301.
- Yun-Nung Chen and Alexander I. Rudnicky. 2014. Dynamically supporting unexplored domains in conversational interactions by enriching semantics with neural word embeddings. In *Proceedings of the 2014 Spoken Language Technology Workshop, December 7-10, 2014, South Lake Tahoe, Nevada, USA*, pages 590–595.
- Yejin Choi and Claire Cardie. 2010. Hierarchical sequential learning for extracting opinions and their attributes. In *Proceedings of the ACL 2010 Conference Short Papers*, pages 269–274, Stroudsburg, PA, USA.
- Trevor Cohn and Philip Blunsom. 2005. Semantic role labelling with tree conditional random fields. In *In Proceedings of CoNLL-2005*, pages 169–172.
- Marie-Catherine de Marneffe and Christopher D. Manning. 2008. The stanford typed dependencies representation. In *Coling 2008: Proceedings of the Workshop CrossParser*, pages 1–8, Stroudsburg, PA, USA.
- Marie-Catherine de Marneffe, Christopher D. Manning, and Christopher Potts. 2010. "was it good? it was provocative." learning the meaning of scalar adjectives. In *ACL*, pages 167–176, Stroudsburg, PA, USA.
- John Firth. 1957. A synopsis of linguistic theory, 1930–1955. *Studies in Linguistic Analysis*, pages 1–32.
- Gayatree Ganu, Nomie Elhadad, and Amlie Marian. 2009. Beyond the stars: Improving rating predictions using review text content.
- Daniel (Zhaohan) Guo, Gokhan Tur, Wen-tau Yih, and Geoffrey Zweig. 2014. Joint semantic utterance classification and slot filling with recursive neural networks. pages 554–559.
- Minqing Hu and Bing Liu. 2004. Mining and summarizing customer reviews. In *Proceedings of the Tenth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, KDD '04*, pages 168–177, New York, NY, USA. ACM.
- Mohit Iyyer, Jordan L. Boyd-Graber, Leonardo Max Batista Claudino, Richard Socher, and Hal Daumé III. 2014. A neural network for factoid question answering over paragraphs. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing, EMNLP*

- 2014, October 25-29, 2014, Doha, Qatar, A meeting of SIGDAT, a Special Interest Group of the ACL, pages 633–644.
- Yohan Jo and Alice H. Oh. 2011. Aspect and sentiment unification model for online review analysis. In *WSDM*, pages 815–824, New York, NY, USA. ACM.
- John D. Lafferty, Andrew McCallum, and Fernando C. N. Pereira. 2001. Conditional random fields: Probabilistic models for segmenting and labeling sequence data. In *ICML*, pages 282–289, San Francisco, CA, USA. Morgan Kaufmann Publishers Inc.
- Omer Levy and Yoav Goldberg. 2014. Dependency-based word embeddings. In *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics, ACL 2014, June 22-27, 2014, Baltimore, MD, USA, Volume 2: Short Papers*, pages 302–308.
- Chenghua Lin and Yulan He. 2009. Joint sentiment/topic model for sentiment analysis. In *CIKM*, pages 375–384, New York, NY, USA. ACM.
- Bing Liu. 2012. Sentiment analysis and opinion mining.
- Bin Lu, Myle Ott, Claire Cardie, and Benjamin K. Tsou. 2011. Multi-aspect sentiment analysis with topic models. In *ICDMW*, pages 81–88, Washington, DC, USA. IEEE Computer Society.
- Christopher Manning, Tim Grow, Teg Grenager, Jenny Finkel, and John Bauer. 2010. StanfordTokenizer. <http://nlp.stanford.edu/software/tokenizer.shtml>.
- Qiaozhu Mei, Xu Ling, Matthew Wondra, Hang Su, and ChengXiang Zhai. 2007. Topic sentiment mixture: Modeling facets and opinions in weblogs. In *WWW*, pages 171–180, New York, NY, USA. ACM.
- Tomas Mikolov, Kai Chen, Greg Corrado, and Jeffrey Dean. 2013a. Efficient estimation of word representations in vector space. *CoRR*, abs/1301.3781.
- Tomas Mikolov, Ilya Sutskever, Kai Chen, Greg Corrado, and Jeffrey Dean. 2013b. Distributed representations of words and phrases and their compositionality. *CoRR*, abs/1310.4546.
- Tomas Mikolov, Ilya Sutskever, Kai Chen, Gregory S. Corrado, and Jeffrey Dean. 2013c. Distributed representations of words and phrases and their compositionality. In *Advances in Neural Information Processing Systems 26: 27th Annual Conference on Neural Information Processing Systems 2013. Proceedings of a meeting held December 5-8, 2013, Lake Tahoe, Nevada, United States.*, pages 3111–3119.
- Tomas Mikolov, Wen-tau Yih, and Geoffrey Zweig. 2013d. Linguistic regularities in continuous space word representations. In *NAACL*, pages 746–751.
- Tomáš Mikolov. 2012. *Statistical Language Models Based on Neural Networks*. Ph.D. thesis, Brno University of Technology.
- Tomas Mikolov. 2014. Word2Vec. <https://code.google.com/p/word2vec/>.
- Naoaki Okazaki. 2007. Crfsuite: a fast implementation of conditional random fields (crfs).
- Bo Pang and Lillian Lee. 2005. Seeing stars: Exploiting class relationships for sentiment categorization with respect to rating scales. In *ACL*, pages 115–124, Stroudsburg, PA, USA. Association for Computational Linguistics.
- Bo Pang and Lillian Lee. 2008. Opinion mining and sentiment analysis. *Found. Trends Inf. Retr.*, 2(1-2):1–135, January.
- Bo Pang, Lillian Lee, and Shivakumar Vaithyanathan. 2002. Thumbs up?: Sentiment classification using machine learning techniques. In *EMNLP*, pages 79–86, Stroudsburg, PA, USA.
- Maria Pontiki, Dimitris Galanis, John Pavlopoulos, Harris Papageorgiou, Ion Androutsopoulos, and Suresh Manandhar. 2014. Semeval-2014 task 4: Aspect based sentiment analysis. In *SemEval*, pages 27–35. Association for Computational Linguistics.
- Magnus Sahlgren. 2006. *The Word-Space Model*. Ph.D. thesis, Stockholm University.
- H. Schütze. 1992a. Dimensions of meaning. In *Proceedings of Supercomputing '92*, pages 787–796.
- Hinrich Schütze. 1992b. Word space. In *NIPS*, pages 895–902.
- Richard Socher. 2014. *Recursive Deep Learning for Natural Language Processing and Computer Vision*. Ph.D. thesis, Stanford University.
- Kristina Toutanova, Dan Klein, Christopher D. Manning, and Yoram Singer. 2003. Feature-rich part-of-speech tagging with a cyclic dependency network. In *NAACL-HLT*, pages 173–180, Stroudsburg, PA, USA. Association for Computational Linguistics.
- Liheng Xu, Kang Liu, Siwei Lai, Yubo Chen, and Jun Zhao. 2013. Mining opinion words and opinion targets in a two-stage framework. In *ACL*, pages 1764–1773, Sofia, Bulgaria.
- Bishan Yang and Claire Cardie. 2013. Joint inference for fine-grained opinion extraction. In *ACL*, pages 1640–1649, Sofia, Bulgaria.
- Wayne Xin Zhao, Jing Jiang, Hongfei Yan, and Xiaoming Li. 2010. Jointly modeling aspects and opinions with a maxent-lda hybrid. In *Proceedings of EMNLP*, pages 56–65, Stroudsburg, PA, USA.
- Yanyan Zhao, Bing Qin, and Ting Liu. 2012. Collocation polarity disambiguation using web-based pseudo contexts. In *EMNLP-CoNLL*, pages 160–170, Stroudsburg, PA, USA.
- Xinjie Zhou, Xiaojun Wan, and Jianguo Xiao. 2013. Collective opinion target extraction in Chinese microblogs. In *EMNLP*, pages 1840–1850, Seattle, Washington, USA, October.