Aspect Term Extraction from Customer Reviews using Conditional Random Fields

Hardik Dalal
e-mail: hardik.dalal@dal.ca

Qigang Gao
e-mail: qggao@cs.dal.ca

Faculty of Computer Science
Dalhousie University
Halifax, NS Canada

Abstract — E-commerce customers generate a vast amount of information about services and products using comments and blogs. Customer reviews serve as one source of this information and they are a critical aspect of e-Business. Reviews are a vital source of feedback and they also help businesses to determine market trends, demographics, and develop knowledge about their competition. Collecting reviews from customers is only half of the challenge. The other half includes mining these reviews to gain insights. Sentiment Analysis techniques help to extract sentiments and determine the perceived product quality or level of customer satisfaction. Our work is focused on detecting product features from customer reviews which, is a part of Aspect Level Sentiment Analysis research. We address the task by expressing it as a sequence-labeling problem in which features are required to be labeled from sentences. The process is similar to that of Named Entity Extraction (NER). However, we are now targeting a different type of entity, i.e., product features. In comparison to NER, Aspect Term Extraction (ATE) poses unique challenges and we address them using Conditional Random Field (CRF), a conditional probability based model. Using dependency parsing, we have engineered a set of optimum features that allow for promising results.

Keywords – Aspect-based Sentiment Analysis; Aspect-term Extraction; Data Analytics; Conditional Random Fields

I. INTRODUCTION

The term 'Social Media' was coined in 1997 but it did not gain much traction in the real world until the Web 2.0 Summit in 2004. During the summit, Tim O'Reilly talked about the commercialization of Web 2.0 and he emphasized user-generated content and its usage. His speech was focused on creating platforms for users with the help of the Internet. Today, customer-generated review websites like Yelp and Amazon have also made major contributions in driving Social Media. Review websites are driven by users who post comments and share their experience about products and services. The content is rich in customer opinions and if used right, it can aid consumers and producers in many ways. However, to make informed decisions based on reviews, consumers have to read thousands of reviews. The overwhelming number of reviews will likely turn down the consumer from reading them. Based on these facts, we realized that the hidden value of customer reviews is never completely appreciated.

The rest of the paper is organized as follows. The remaining subsections of this section describes the research question we asked ourselves to address the problem and challenges faced in solving the problem. Section II talks about the major contributions in the field. Section III describes the data we used by our model. It also explains how we processed the data into feature vector which was fed to our model. Section IV depicts our model of choice and feature selection. Experiment plan, results and evaluation of our model are shown in Section V. Lastly, the paper concludes with the lesson learned and potential future enhancements.

A. Research Question

The research question that we ask ourselves in this paper focus on finding meaningful information from a large set of reviews. We intend to answer the following question: how can we extract product aspects from reviews? The answer to the question points to the core technique that is responsible for extracting aspects from reviews.

B. Challenges

ATE brings a very unique set of challenges when compared to NER. This is because ATE is different from NER on some fundamental levels. First, aspect terms are describing properties of a product. These aspect terms vary from product to product, e.g. a camera will have 'photo quality' as one of the many aspects, and, similarly 'screen size' for a laptop. An entity on the other hand, falls into one of the following categories: organization, person, location, or miscellaneous [1]. NER systems can detect percentages, ages of people, and dates [2].

Second, an entity usually follows certain characteristics throughout corpus such as capitalized first word, starting with 'the', membership to a group of words, etc. NER also tend to have explicit rules to detect certain type of entities, e.g. a date as an entity has a month and a value less than 32. ATE do not follow such traits because they are mostly

 ISBN: 978-1-61208-603-3

nouns. In order to detect aspect terms, other linguistic features are important.

II. SURVEY ON RELATED WORKS

There are several key techniques that have been proposed by researchers to solve ATE. For a better understanding, we divide the techniques into four major categories based on the properties of reviews that they exploit. The categories are:

1. Frequency-based
2. Relation-based
3. Supervised learning-based
4. Model-based

In the remaining part of this section, we will discuss notable work that is done in these categories.

A. Frequency-based methods

Frequency-based methods are based on the statistic that most of the aspects are nouns and noun phrases. According to a study by Liu et al. [3], around 60 to 70% of aspects are nouns. This fact is used to find frequent aspects from reviews. There are several techniques proposed, such as Hu et al. [4], which extract aspects by finding frequent nouns. Noun and noun phrases are determined from Part-Of-Speech (POS) tags and a threshold is decided experimentally. It is interesting to note that implicit features only account for 15-20% of the total aspects.

A method proposed by [5] uses a Web-based information retrieval system that used a Pointwise Mutual Information (PMI) score to evaluate the associations between phrases. A score that was estimated from Web search hit counts and the most frequent aspects after applying a threshold was retained.

[6] uses one of the few methods under the frequency-based category that do not rely on external sources like a Web search. The authors devised an unsupervised aspect-related term learning method using linguistic and statistical information. Their method is theoretically domain independent.

B. Relation-based methods

Relation-based approaches exploit syntactic relations among sentences to extract aspects and sentiments. One of these milestones is proposed in [3] and it explores opportunities in Pros, Cons and Review-type formats. The extraction is carried out using a supervise rule discovery, which involves labeling the dataset manually and feeding it to the association rule mining algorithm. The labelled dataset is used to derive an association rule in the form of X->Y with some confidence percentage. One of the key aspects of this work was that they could extract implicit aspects, those that are not specific and possess a hidden reference to an aspect.

[7] proposed a very interesting method to extract aspects by detecting sentiment-laden sentences in reviews and they used only those sentences to extract aspects. The motivation here is that most sentences that express some opinion are likely to target an aspect of products. [8] and [9] used a dependency parser to identify aspects and the sentiments that are associated with them. In [10], the authors developed a better technique to detect aspects and opinions, which was called Double-Propagation. The idea here lies in iteratively going through the syntactic relationships between aspects and sentiment words. Each iteration generates an aspect or sentiment word, which is added to the respective list and it is used in next iteration. This goes on until there are no additions to the list. The sentiment words are mostly adjectives, while the aspect words are nouns or noun phrases.

C. Supervised Learning-based methods

The current state-of-the-art techniques for aspect-based sentiment analysis, under the supervised learning category, are based on the Hidden Markov Model (HMM) and Conditional Random Field (CRF).

In [11], the authors have proposed a supervised learning technique that naturally integrates HMM with linguistic features to extract product feature-opinion pairs. The technique is partially adapted from a very common problem in Information Retrieval (IR), called Named Entity Recognition (NER). The problem of NER is to detect the names of people, places and organizations from text using POS tags. The proposed technique uses POS tags to identify product features and categorize them under components, functions, features and opinions. Based on the position of entities (beginning, middle or end) and the respective category, two tag sets are defined: a basic and pattern tag set. A basic tag set determines the category and a pattern tag set determines the position of the entity. These two sets together form a hybrid tag set that is integrated in HMM to determine the sequence of hybrid tags with higher probability.

A benchmark work using CRF is [12], which allows the extraction of opinion targets (aspects) from a cross-domain scenario. The authors proposed 5 features for their CFR-based approach namely token, POS, short dependency path, word distance and opinion sentence.

D. Model-based methods

Topic modeling in Machine Learning is to learn abstract concepts about available topics from large textual corpuses. There are mainly two models under this category that are used by researchers to detect aspects from product features, Latent Dirichlet Allocation (LDA) and Probabilistic Latent Semantic Indexing (PLSI). The authors in [13] used the extended probabilistic model to extract 'topic-sentiment' pairs from Web logs. The basic assumption was that every blog post is generated from sampling words from a model, which is a combination of a background language model, topic language model, positive sentiment model and negative sentiment model. The authors could extract topics/subtopics, correlations between the topics and relate the sentiments to their respective topics/subtopics.

Our contribution is based on a supervised learning algorithm/model which is independent of Web or outside source. Our model uses a combination of syntactic relations in sentences and probability theory to extract

aspect terms. It strikes a good balance between the available techniques in Relation, Supervised Learning and Model-based approaches.

III. DATASET DESCRIPTION AND PREPARATION

We used two datasets from the same source. One of them is freely available dataset from [14] consists of reviews of nine products and most of them are electronic products. Another dataset consisting of three products is also available on the same source. We merged the 12 dataset files into 6 based on the type of product. For example, the Canon PowerShot SD500 and Canon S100 dataset files are merged into Cameras file. The following table represents the number of reviews and aspects in merged files:

TABLE I. NUMBER OF REVIEWS AND ASPECTS IN EACH DATASET

Dataset	# of reviews	# of aspects
Antivirus (A)	380	250
Audio devices (B)	1220	632
Cameras (C)	530	387
Computers (D)	531	354
Mobile phones (E)	554	473
Routers (F)	1191	585

A. Dataset Description

Reviews used in this project are of the Free Format type. This format gives freedom to users to express their views and hence the name. This type offers the most challenging research problems as it has the most unstructured information compared to formats that, for example, include a pros and cons part.

A. Data Preparation

A text dataset is an unstructured form of information with knowledge that is hidden in a formed relationship among the occurrences of word, order, grammatical relations, etc., We need the data to be in a structured format that can be consumed by the model. The intrinsic relationships are converted to a vector (aka feature vector) that is fed to a model to learn those relationships. The features are chosen closely to address the problem of aspect extraction. Tagging schemes are adopted to encode this information. This is described in detail later in this section.

1) Tokenization

We have used the Stanford Tokenizer using Spark to ensure parallel processing. It uses Penn Treebank and a deterministic approach to tokenize a sentence [15].

2) Tagging

We used two types of tagging:

a) POS Tagging: We used the Stanford POS Tagger, a part of the Stanford CoreNLP Suite, to tag each review [16].

b) IO Tagging: IO (Inside, Outside) tagging is a very simple yet effective way to encode information to tokens. Each token is either labeled as 'I' if the token is a named entity under consideration and 'O' otherwise.

3) Dependency Parsing

We have used the Stanford Dependency Parser to extract relations and we have used head and dependent tokens.

IV. ASPECT TERM EXTRACTION USING CONDITIONAL RANDOM FIELDS

CRF is a generative sequence labelling model. According to Lafferty et al. [17], CRF specifies the probabilities of a sequence based on an observed sequence. The observed sequence, known as features, serves as an input to the model. CRF builds conditional probabilities based on these features. Possibility that a label will occur in a sequence is dependent on the current, previous and future sequence. Lafferty et al. [17], defines CRF as follows:

Let $G = (V, E)$ be a graph such that

$$\boldsymbol{Y} = (\boldsymbol{Y}_v)_{v \in V}$$

Then (X, Y) is a conditional random field in case, when conditioned on $\boldsymbol{X}$, the random variables $\boldsymbol{Y}_v$ obey the following property:

$$p(\boldsymbol{Y}_v | \boldsymbol{X}, \boldsymbol{Y}_w, w \neq v) = p(\boldsymbol{Y}_v | \boldsymbol{X}, \boldsymbol{Y}_w, w \sim v)$$

where $w \sim v$ means that w and v are neighbors in G.

Our aim was to handpick the features $\boldsymbol{X}$ so we can construct most accurate probability distribution for a given sequence of tokens. To achieve this aim, we planned to model the features that are capable of providing most accurate information.

A. Feature Selection

The selection of features is based on the observation that a word is an aspect not just because of itself but due to the neighboring words and the relationship it shares with those words. These relationships are based on lexical features aka grammar, which are semantic rules of any language. The goal is to learn these relationships using the conditional model. We model CRF so it can identify an aspect based on its POS tag, relative location and relationship to neighboring words and their POS tags.

The features that are chosen for this problem are commonly used in traditional NER systems that use features available from the text itself (also known as closed features) [18]. We are not using any external source to generate a feature vector. The following is the list of features that we use:

1) *Word:* Current (the phrase itself), previous and next word string in lowercase

2) *POS:* POS tag of current, previous and next word. POS tags provide important information about lexical category of words
3) *Head Word:* Syntactic head word of current word according to grammatical structure ('null' if current word does not have a head word)
4) *Head Word POS:* POS tag of head word ('null' if current word does not have a head word)
5) *Dependency Relations:* The dependency relation is identified from a dependency parsing of the review. The Stanford CoreNLP tool contains about 50 grammatical relations between two words [19]. This relationship is binary, meaning that a relation holds between a head (or governor) and a dependent. We derived two features: the relation that the current word shares as a dependent and governor. In other words, we derived the following two features:
    a. the relation when the current word as a governor, and
    b. the relation when the current word is a dependent.
6) *IO tag:* We are interested in modeling CRF so it correctly classifies each aspect as 'I' according to the IO tagging explained in the data preparation step.

Thus, a feature vector for a given sentence will be as follows:

$$x_i = C_i, POS_i, C_{i-1}, C_{i+1}, POS_h, Dep_g, Dep_d, IO_i$$

For example, the feature vector for review "This camera is amazing" is shown below:

x_1 = This, DT, NULL, camera, NULL, det, NULL, O
x_2 = camera, NN, This, is, This, DT, nsubj, det, I
x_3 = is VBZ, camera, amazing, NULL, NULL, amazing, NULL, O
x_4 = amazing, JJ, amazing, NULL, Camera, NN, NULL, nsubj, O

B. Training and Test the Model

We used the Apache Spark based implementation of CRF available on GitHub by Intel Big Data group [20]. It is licensed under Apache 2.0 allowing free usage for everyone. We forked from the main branch on GitHub to create our own implementation around it. We preferred not to alter the library and used it as is for the purpose of comparison consistency.

We trained the model on each dataset individually. We used 5-fold cross validation to determine the accuracy of the model.

V. Experiment and Evaluation

The results of ATE are evaluated using following three metrics

$$Precision = \frac{Extracted\ Aspects \cap Gold\ Standard\ Aspects}{Extracted\ Aspects}$$

$$Recall = \frac{Extracted\ Aspects \cap Gold\ Standard\ Aspects}{Gold\ Standard\ Aspects}$$

$$F\text{-}measure = \frac{2 \times Recall \times Precision}{Recall + Precision}$$

The Gold Standard Aspects parameter is the number of aspects originally found in the dataset and the Extracted Aspect parameter represents the number of aspects detected by the model.

A. Experiment Plan

In our approach, we used hidden linguistic features that we extracted in the pre-processing step. These features contain a lot of information about the aspects and instead of leaning towards Web sources, we trained our model to use implicit features only. Our plan involves starting out with the bare minimum number of features and progressively adding more and recording the change in performance. This approach seems like trial-and-error, but, compared to NER problem, this problem is unique, and hence handcrafted features usually work best. We also believe that adding too much information might lead to over fitting the model so we wanted to keep the feature vector small.

As said, we are using features we can find from within the data. We found a total of 10 features that we could use to train our model. These features are:

- Current token and its POS, 2 features
- Previous and next token and their POS, 4 features
- Head token and its POS, 2 features
- Dependency relation the current token shares as governor and as dependent, 2 features. Each one is itself a varying size array of values. We also handpicked some of the dependency to improve the performance. These relations are named "selected dependency relations".

The next section shows the results that we recorded during the experiments in condensed tables. The sub columns A, B, C, etc. represents the merged dataset mentioned in Table I.

B. Experiment Results

The bare minimum number of features including current, previous and next token and their POS tags resulted in an F-measure of 0.41.

TABLE II. F-MEASURE OBSERVED ON EACH DATASET FROM CURRENT, PREVIOUS AND NEXT TOKEN AND POS TAG

F-measure						Avg
A	*B*	*C*	*D*	*E*	*F*	
0.437	0.572	0.471	0.414	0.414	0.576	0.411

The next major result was observed with similar features but replacing POS tags with selected dependency relations. The average F-measure improved to 0.57 (+0.16).

Table III. F-MEASURE OBSERVED ON EACH DATASET FROM CURRENT, PREVIOUS AND NEXT TOKEN AND SELECTED DEPENDENCY RELATION

F-measure						Avg
A	*B*	*C*	*D*	*E*	*F*	
0.526	0.648	0.548	0.516	0.529	0.689	0.576

A minor increase in F-measure (+0.01) was obtained when we added head token POS instead of selected dependency relationships.

TABLE IV. F-MEASURE OBSERVED ON EACH DATASET FROM CURRENT, PREVIOUS AND NEXT TOKEN AND HEAD TOKEN POS TAG

F-measure						Avg
A	*B*	*C*	*D*	*E*	*F*	
0.535	0.661	0.538	0.548	0.533	0.69	0.584

The best result was observed with head token POS and selected dependency relations. The observed F-measure with such features was 0.58 (+0.18).

TABLE V. F-MEASURE OBSERVED ON EACH DATASET FROM CURRENT, PREVIOUS AND NEXT TOKEN AND HEAD TOKEN POS TAG AND SELECTED DEPENDENCY RELATION

F-measure						Avg
A	*B*	*C*	*D*	*E*	*F*	
0.537	0.657	0.564	0.511	0.556	0.695	0.586

1) Optimal Feature Set

After the experiments shown before, we found the optimal set of features and they are as follows:

- Current token
- Head token POS
- The dependency relation that the current token shares with another token as governor and as dependent

We narrowed down the relationships that the current token shares as governor to only adjectival modifier (amod), nominal subject (nsubj) and dependent (dep) [10]. This is based on the study by Hu et al. [4] indicating that about 60 to 70% of aspects are nouns. So, instead of feeding the model all of the relations, we handpicked a few of them to increase the likelihood of extracting correct aspects. Similarly, we determined the three relations namely nsubj, direct object (dobj), and dep are useful when the current token is dependent as it suggests that the dependent word is likely a noun phrase and hence a potential aspect.

2) SemEval-2014 ABSA Dataset

We also tested our model against datasets provided in Aspect-Based Sentiment Analysis task in the International Workshop on Semantic Evaluation (SemEval-2014) [21]. The conference focuses on evaluating computational semantic analysis systems and falls under the Special Interest Group of the Association for Computational Linguistics. The conference has several tasks and Aspect Based Sentiment Analysis is one of them. The results are shown in Table VI below.

TABLE VI. RESULT OBSERVED ON SEMEVAL DATASET FROM CURRENT, PREVIOUS AND NEXT TOKEN AND HEAD TOKEN POS TAG AND SELECTED DEPENDENCY RELATION

	F-measure	
	Laptop Dataset	*Restaurant Dataset*
SemEval-Baseline	0.356	0.471
Our model	0.507	0.522
SemEval-Best	0.744	0.84

Some of the best works at the conference used SVM like [22] and [23] and CRF like [18] and [24]. [25] shows a very unique contribution by using a combination of SVM and HMM.

C. Experiment Observations

It was interesting to see that certain features like token's POS tags and neighboring token's POS tags proved to be bad for our model. After investigating, we found that this happened due to the tagging scheme that we used. IO tagging shares very limited knowledge of current tokens' neighboring words, meaning that it does not indicate what the next and previous tokens (and its POS tag) are. This also caused issues when tagging multi-word aspects. Multi-word aspects require more information about current tokens such as whether the previous token and current token together represent an aspect or the current token and the next one together represent an aspect. Since our tagging scheme was unable to generate such information, our experiments only tagged one-word

 ISBN: 978-1-61208-603-3

aspects during training, which ultimately affected the results.

The head token did not help the model to predict the current token but the head tokens' POS tag did help the model. This is because the head token for aspects varies between train and test data but the POS tag is usually consistent.

We also noticed that some datasets contain words along with symbols that Stanford Tokenizer cannot tokenize. For example, we found "-LRB-" and "-RRB-" in Computer reviews. Such phrases caused the Stanford CoreNLP toolkit to misinterpret relationships and the structure of the sentence. In other words, POS tag and dependency relations associated to tokens were incorrect and the result was a faulty feature vector.

We compared our model with some of the best works submitted in SemEval-2014 ABSA task and found that our features were very limiting. For example, [18] used WordNet, name list and word clusters. When comparing with other type of models such as SVM, we found that results of NER system were fed into the model as a feature like in [23]. The authors of [25] used only lexical features to detect aspects like we did. The difference between our and their feature set was that their feature set included all the features we used and a few more, such as prefixes, suffixes, and POS bigrams and trigrams. In [26], the authors used frequency based information such as PMI and Term Frequency – Inverse Document Frequency (TF-IDF) as features. These features helped them identify commonly occurring aspect terms. Our model did miss some of the commonly occurring aspects because of the lack of frequency based information. They also used NER-based information to identify whether a token is person, place or organization, which we did not.

We did not find any significant work carried our using LDA or other topic-based algorithms. The reason participants did not use LDA for this problem was because of the lack of data available to train the model. LDA suffers from a very common problem called cold start problem. LDA requires substantial amount of information to achieve decent results. A breakthrough research to address the cold start problem using Factorized LDA was published in [27]. The authors modelled reviewers rating along with the reviews to achieve impressive results. Unfortunately, SemEval dataset did not contain reviewers' information.

VI. CONCLUSION

CRF has its pitfalls, such as the fact that it requires highly accurate labelled training data. However, it is a very good candidate for a sequence labeling problem such as ATE. We recommend using advanced tagging scheme like Before-Inside-Outside (BIO) for labeling tokens. BIO is also advantageous for multi-word aspects. According to [28], there are about 27% aspects in restaurant domain are multi-word aspects, 44% aspects in laptop reviews. BIO is used in many state-of-the-art NER systems and it has proven to be better at tagging schemes then IO. That is because BIO has the potential to carry more informational value then IO.

Feature selection-wise, we strongly believe that a single source of information such as lexical information, is not enough to train high accuracy model. Combination of frequency-based information (e.g. TF-IDF) and open features such as word clusters can definitely help achieve better results. WordNet is an excellent source to form word clusters. Another good option is word2vec. It carries a lot of information in the form of a high dimensional vector. Moreover, the open feature makes the model portable on different datasets.

Computational wise, with the advent of Big Data frameworks like Spark and Hadoop, it is possible to optimize the algorithm. Our implementation is capable of running on a Spark cluster. However, for future work, we recommend bringing Cloud and Big Data Framework together for large scale data processing and effective resource utilization.

REFERENCES

[1] E. F. T. K. Sang and F. D. Meulder, "Introduction to the CoNLL-2003 Shared Task: Language-Independent Named Entity Recognition," *Proceeding CONLL '03 Proceedings of the seventh conference on Natural language learning at HLT-NAACL 2003 - Volume 4,* vol. 4, pp. 142-147, 2003.

[2] D. M. Bikel, M. Scott, S. Richard, and W. Ralph, "Nymble: a high-performance learning name-finder," *Proceedings of the fifth conference on Applied natural language processing,* pp. 194-201, 1997.

[3] B. Liu, M. Hu, and J. Cheng, "Opinion observer: analyzing and comparing opinions on the Web," *Proceedings of the 14th International conference on World Wide Web - WWW '05,* pp. 342-351, 2005.

[4] M. Hu and B. Liu, "Mining and summarizing customer reviews," *KDD '04 Proceedings of the tenth ACM SIGKDD international conference on Knowledge discovery and data mining,* pp. 168-177, 2004.

[5] O. Etzioni et al. "Web-scale information extraction in knowitall," *WWW '04 Proceedings of the 13th international conference on World Wide Web,* pp. 100-110, 2004.

[6] J. Zhu, H. Wang, B. K. Tsou, and M. Zhu, "Multi-aspect opinion polling from textual reviews," *CIKM '09 Proceedings of the 18th ACM conference on Information and knowledge management,* pp. 1799-1802, 2009.

[7] S. Blair-Goldensohn, K. Hannan, R. McDonald, T. Neylon, G. Reis and J. Reynar, "Building a sentiment summarizer for local service reviews," *Proceedings of the WWW2008 Workshop: NLP in the Information Explosion Era (NLPIX 2008),* vol. 14, pp. 339-348, April 2008.

 ISBN: 978-1-61208-603-3

[8] N. Kobayashi, R. Iida, K. Inui, and Y. Matsumoto, "Opinion Mining on the Web by Extracting Subject-Aspect-Evaluation Relations," *AAAI Spring Symposium: Computational Approaches to Analyzing Weblogs,* pp. 86-91, 2006.

[9] S. Somasundaran, G. Namata, L. Getoor, and J. Weibe, "Opinion graphs for polarity and discourse classification," *TextGraphs-4 Proceedings of the 2009 Workshop on Graph-based Methods for Natural Language Processing,* pp. 66-74, 2009.

[10] G. Qiu, B. Liu, J. Bu, and C. Chen, "Opinion word expansion and target extraction through double propagation," *Computational Linguistics,* vol. 37, no. 1, pp. 9-27, 2011.

[11] W. Jin and H. H. Ho, "A novel lexicalized HMM-based learning framework for web opinion mining," *ICML '09 Proceedings of the 26th Annual International Conference on Machine Learning,* pp. 465-472, 2009.

[12] N. Jakob and I. Gurevych, "Extracting opinion targets in a single-and cross-domain setting with conditional random fields," *EMNLP '10 Proceedings of the 2010 Conference on Empirical Methods in Natural Language Processing,* pp. 1035-1045, 2010.

[13] Q. Mei, X. Ling, M. Wondra, H. Su, and C. Zhai, "Topic sentiment mixture: modeling facets and opinions in weblogs," *WWW '07 Proceedings of the 16th International conference on World Wide Web,* pp. 171-180, 2007.

[14] B. Liu and M. Hu, "Opinion Mining, Sentiment Analysis, and Opinion Spam Detection Dataset," University of Illinois at Chicago (UIC), [Online]. Available: https://www.cs.uic.edu/~liub/FBS/sentiment-analysis.html#datasets. [Accessed 1 November 2017].

[15] C. D. Manning, P. Raghavan and H. Schutze, Introduction to Information Retrieval, Cambridge University Press, 2008.

[16] D. Jurafsky and J. H. Martin, "Chapter 8 - Word Classes and Part-of-Speech Tagging," in *Speech and Language Processing*, Prentice-Hall Inc., 2000, pp. 310-319.

[17] J. Lafferty, M. Andrew, and P. Fernando CN, "Conditional random fields: Probabilistic models for segmenting and labeling sequence data," *Proceedings of the 18th International Conference on Machine Learning 2001 (ICML 2001),* pp. 282-289, June 2001.

[18] Z. Toh and W. Wang, "DLIREC: Aspect Term Extraction and Term Polarity Classification System," *Proceedings of the 8th International Workshop on Semantic Evaluation (SemEval 2014),* pp. 235-240, August 2014.

[19] d. M. Marie-Catherine and C. D. Manning, "Stanford typed dependencies manual," ResearchGate, 2008.

[20] P. Meng, H. Cheng, and Q. Huang, "CRF-Spark," Intel, 14 November 2016. [Online]. Available: https://github.com/Intel-bigdata/CRF-Spark. [Accessed 1 November 2017].

[21] M. Pontiki et al. "SemEval-2014 Task 4: Aspect Based Sentiment Analysis," *Proceedings of the 8th International Workshop on Semantic Evaluation (SemEval 2014),* pp. 27-35, 5 December 2014.

[22] J. Wagner et al. "DCU: Aspect-based Polarity Classification for SemEval Task 4," *Proceedings of the 8th International Workshop on Semantic Evaluation (SemEval 2014),* pp. 223–229, 23-24 August 2014.

[23] S. Kiritchenko, X. Zhu, C. Cherry, and S. M. Mohammad, "NRC-Canada-2014: Detecting Aspects and Sentiment in Customer Reviews," *The 8th International Workshop on Semantic Evaluation (SemEval 2014),* pp. 437–442, 23024 August 2014.

[24] T. &. K. Brychcín and J. Michal & Steinberger, "UWB: Machine Learning Approach to Aspect-Based Sentiment Analysis," *The 8th International Workshop on Semantic Evaluation (SemEval 2014),* pp. 817-822, 23-24 August 2014.

[25] G. Castellucci, S. Filice, D. Croce, and R. Basili, "UNITOR: Aspect Based Sentiment Analysis with Structured Learning," *The 8th International Workshop on Semantic Evaluation (SemEval 2014),* pp. 761–767, 23-24 August 2014.

[26] M. Chernyshevich, "IHS R&D Belarus: Cross-domain Extraction of Product Features using Conditional Random Fields," *Proceedings of the 8th International Workshop on Semantic Evaluation (SemEval 2014),* pp. 309-313, 23-24 August 2014.

[27] S. Moghaddam and M. Ester, "The FLDA Model for Aspect-based Opinion Mining: Addressing the Cold Start Problem," *Proceedings of the 22nd international conference on World Wide Web,* pp. 909-918, 2013.

[28] P. Blinov and E. Kotelnikov, "Blinov: Distributed Representations of Words for Aspect-Based Sentiment Analysis at SemEval 2014," *Proceedings of the 8th International Workshop on Semantic Evaluation (SemEval 2014),* pp. 140–144, 23-24 August 2014.

 ISBN: 978-1-61208-603-3