ASPECT BASED OPINION MINING USING DEEP CNN FOR ASPECT EXTRACTION

by

C.CARA EVANGELINE 411517

C.KUSHALA 411518

A.SHIRISHA 411507

Under the guidance of

DR.SRILATHA CHEBROLU

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

NATIONAL INSTITUTE OF TECHNOLOGY, ANDHRA PRADESH

TADEPALLIGUDEM-534102, INDIA

May – 2019

DECLARATION

I declare that this written submission represents my ideas in my own words and where others' ideas or words have been included, I have adequately cited and referenced the original sources. I also declare that I have adhered to all principles of academic honesty and integrity and have not misrepresented or fabricated or falsified any idea/data/fact/source in my submission. I understand that any violation of the above will be cause for disciplinary action by the Institute and can also evoke penal action from the sources which have thus not been properly cited or from whom proper permission has not been taken when needed.

C.Cara Evangeline	C.Kushala	A.Shirisha
411517	411518	411507
Date: __________	Date: __________	Date: __________

CERTIFICATE

It is certified that the work contained in the thesis titled "**Aspect based opinion mining using deep CNN for aspect extraction**," by "C Cara Evangeline bearing Roll no 411517, C.Kushala bearing Roll no 411518 and A.Shirisha bearing Roll No 411507 " has been carried out under my supervision and that this work has not been submitted elsewhere for a degree.

Dr.Srilatha Chebrolu
Department of CSE
N.I.T. Andhra Pradesh
May, 2019

ACKNOWLEDGEMENT

We take this opportunity to express our profound gratitude and deep regards to our supervisor ***DR.SRILATHA CHEBROLU***, Assistant Professor, Department of Computer Science and Engineering, NIT Andhra Pradesh for her exemplary guidance, monitoring and constant encouragement through out the project.

We avail ourselves of this proud privilege to express our regard to all the faculty of the department of Computer Science and Engineering at NIT Andhra Pradesh for empathizing and providing all the necessary facilities throughout the work.

I thank all my friends and batch mates for being a significant part of my student life and making me ready for the future. They will always stay with me in the form of the conversations we shared and memories we had. Finally, I thank my parents for being a constant support in my life and helping me through my highs and lows.

C Cara Evangeline C.Kushala A.Shirisha

LIST OF FIGURES

ABSTRACT

Merchants selling products on the Web often ask their customers to review the products that they have purchased and the associated services. As e-commerce is becoming more and more popular, the number of customer reviews that a product receives grows rapidly. For a popular product, the number of reviews can be in hundreds or even thousands. This makes it difficult for a potential customer to read them to make an informed decision on whether to purchase the product. It also makes it difficult for the manufacturer of the product to keep track and to manage customer opinions. For the manufacturer, there are additional difficulties because many merchant sites may sell the same product and the manufacturer normally produces many kinds of products. In this research, we aim to mine and to summarize all the customer reviews of a product. This summarization task is different from traditional text summarization because we only mine the features of the product on which the customers have expressed their opinions and whether the opinions are positive or negative. We do not summarize the reviews by selecting a subset or rewrite some of the original sentences from the reviews to capture the main points as in the classic text summarization. Our task is performed in three steps: (1) mining product features that have been commented on by customers; (2) identifying opinion sentences in each review and deciding whether each opinion sentence is positive or negative; (3) summarizing the results.

TABLE OF CONTENTS

CHAPTER 1

INTRODUCTION

1.1 Background and Motivation

Opinion mining (also known as sentiment analysis) is the computational study of subjective information towards different entities. Entities usually refer to products, organizations, services or/and their aspects, functions, components and attributes. Opinion mining is a major task of Natural Language Processing (NLP) that studies methods for identifying and extracting opinions from written text, such as product reviews, discussion groups, forums and blogs. It makes the Web an extensive and excellent source of information to gather opinions about a specific object. With the undeniable growth of the Web, individuals and organizations are using online content for their buying and manufacturing decision-making. Every time someone attempts to discover what other people think about something on the Web, the response is an enormous amount of data, which makes it difficult to find useful information easily. For organizations, tracking customer feedback can help to measure the level of satisfaction and make optimal manufacturing and selling decisions. Due to human mental and physical limitations, it is difficult to manually gather and analyse the massive amount of information on the Web. Therefore, a system that can automatically summarize documents is increasingly desirable. Such a system extracts relevant information and presents it in a manner that is easy to read and understand in order to make informed decisions. To gather the initial data, most e-commerce web-pages provide a feedback or review area where individual customers can exchange their experience and opinions. The customer review content is often presented as a natural language text in an unstructured form. Hence, analysing and extracting meaningful information is a challenging task that needs specialized techniques.

Even though opinion mining at document and sentence levels is valuable, neither discover exactly what people like and dislike. An opinionated text on a particular entity does not mean that the author likes or dislikes every single aspect of the entity. “The Canon camera is amazing; it is better than the Samsung camera,” is an example of a product review that express positive opinion about one product and a negative opinion on another product. It is not valid to classify and generalize the sentiment on both products. To obtain fine-grained opinions, it is necessary to examine the aspect level Liu [2015].

The aspect level is also known as aspect-based opinion mining (ABOM), which identifies and extracts opinions and their targets. ABOM has three main tasks: first, identify and extract product aspects; then determine opinion words and their polarities; and finally determine the percentage of polarity for all general aspects.

1.2 Significance of the Study

Aspect-based opinion mining from customer reviews is a challenging problem for opinion mining and sentiment analysis. This research contributes to novel methods to identify and extract product aspects and sentiment from customer reviews by employing natural language processing (NLP) in both supervised and unsupervised learning techniques. This is performed through data mining, machine learning, deep learning, linguistics and statistical techniques. Finally, this research will produce a set of Dependency Patterns that mainly focuses on the grammatical structure of the sentences, which enhanced the extraction process. This research will also contribute to extracting all possible aspects, both implicit and explicit.

1.3 Project Scope

This project will be limited to studying techniques within aspect-based opinion mining from customer reviews. Only relevant topics to this project will be discussed and evaluated. The purpose is to develop, test and evaluate different methods and techniques for aspect-based opinion mining from customer reviews. This project focuses on deep learning algorithms for identifying and extracting product aspects and the classification of opinionated customer reviews. Note that the goal of this project is not to necessarily produce an opinion summary but to develop approaches that can identify and extract meaningful aspects.

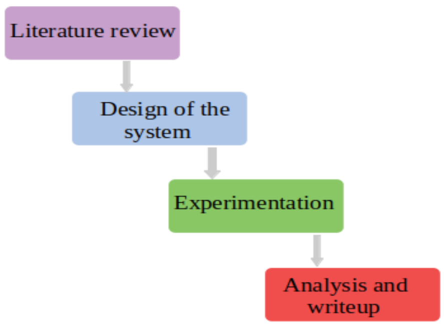


Fig 1.1 Project Scope

The scope of the project, as illustrated in Figure 1.1, includes a literature review Chapter 2, then system design and development, Chapters 3 and 4 and finally discussion and the experimentation results Chapter 5.

1.4 System Overview

Aspect-based opinion summarization is a systematic process that consists of three basic tasks shown in Figure 1.2. It aims to generate the aspect-based opinion summary through process outlined in this section.

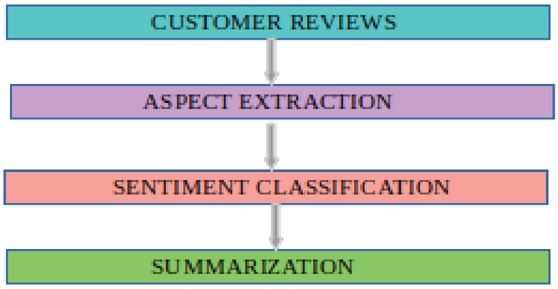


Fig 1.2: System Overview

1.4.1 Aspect/feature Extraction

This step identifies aspects/features of a specific entity. For example, if the aim is to generate an opinion summary about "iPhone 5", some common aspects of cellular phones will be listed, including battery life, camera resolution and size. Therefore, the main purpose is to find all possible aspects/features of a specific entity.

1.4.2 Sentiment classification

The sentiment prediction discovers the semantic orientation of each aspect/feature, e.g.positive or negative. The final sentiment predictions help in summarising the general sentiments of the product's aspect/feature.

1.4.3 SUMMARY GENERATION

The extracted information is used in a combination with the aspects/features and the semantic orientation to present the final opinion summary. The final summary can take the form of a structured summary (e.g., tables, XML, charts).

1.5 THESIS OUTLINE

This thesis consists of seven chapters. Chapter 1 describes the background, motivation and the framework. Chapter 2 discusses the important definitions and terminologies, then reviews the literature related to aspect-based opinion mining from customer reviews.Chapter 3 discusses the aspect extraction using deep CNN. Chapter 4 discusses the classification algorithms to determine the polarity based on the extracted aspects.Finally, Chapter5 discusses and analyses all results.

CHAPTER 2

LITERATURE REVIEW

This research aims to investigate the techniques for aspect and opinion extraction from customer reviews. Chapter 1 provided an overview of this research. This chapter will introduce related definitions and terminology, then review a number of existing techniques and applications related to aspect-based opinion mining. Finally, some of the challenges will be discussed. The knowledge provided in this chapter will be cited in later chapters.

2.1 Customer Review Mining

Increasingly large numbers of customers choose online shopping as it is more convenient, reliable, and easy to compare prices and get good feedback from other customers. Consequently, the number of products sold online is increasing rapidly. This makes it difficult for customers to make correct purchasing decisions only based on product description and some images or even videos of the product. Vast amounts of customer feedback data are available, such as blogs, product reviews, and forms. However, such data is typically unstructured, thus it needs a text mining approach to extract useful knowledge and interpret important information. There- fore, the mined information is re-structured and presented in an ultra-concise form. Applying sentiment analysis techniques to mine, analyse, and then summarise this type of data is called “customer review mining”.

In some cases, mining opinions at a sentence or document level is beneficial. However, such levels of information are not always enough for a complete decision-making process. For instance, a positive appraisal on a specific item does not mean that the reviewer likes each aspect of the item. Similarly, a negative review does not imply that the reviewer dislikes everything. In a typical review, the reviewer writes both negative and positive aspects of the item, while his overall opinion about the item may be negative or positive [Moghaddam, 2013]. Therefore, sentence-level and document-level opinions may not offer comprehensive information for decision-making. To acquire such information, finer level of granularity should be found. In the last decade, numerous methods have been suggested to solve the problem of aspect- based opinion mining. The previous works are frequency-based approaches wherein simple filters are applied on high frequency noun phrases to extract aspects. Although such methods are effective, they miss low frequency. To overcome such weakness, relation-based methods are suggested. Relation-based methods use NLP approaches to find certain relationships between related sentiments and aspects.

This section will investigate what kind of data is typically available in the form of customer reviews, how the data is structured and different sources of online reviews data. It then will illustrate a detailed example of an online review, followed by tasks of customer review mining.

2.2 Sources of Online Reviews

An online review is piece of text that is publicly available and written by a known or anonymous customer regarding a purchase of a product or service. Sources of online reviews are widely varied.

• A web log, or “blog” for short, is defined by the Oxford dictionary as “a regularly updated website or web page, typically one run by an individual or small group, that is written in an informal or conversational style.” It represents review comments by authors about newly purchased products (e.g. cameras) and provided services (e.g. hotel or restaurant).
• A review site is a website in which reviews are posted by authors about their experiences of purchased products or provided services. Typically, each product will have single or multiple pages where all reviews are gathered and published.
• An online shopping site is an electronic commerce (EC) site in which the selling and purchasing goods and services is done over the Internet. The process starts from the merchants, who advertise goods/services over the Internet to potential customers to purchase and provide feedback. The customer feedback can be provided on the product as a whole, parts of the product, the shipment, or anything related to the provided service. Amazon.com and ebay.com are the most popular examples of online shopping sites.

2.3 Formats of Online Reviews

Online reviews give authors the freedom to express their thoughts about provided products and/or services. They are considered to provide essential and informative data and can take different formats.

Format 1 - Free-format detailed review

This is a free-text box where the author can write the review in the form of free text and sometimes with no word limit. A very well-known example is amazon.com. Figure 2.1 shows an example from amazon.com.

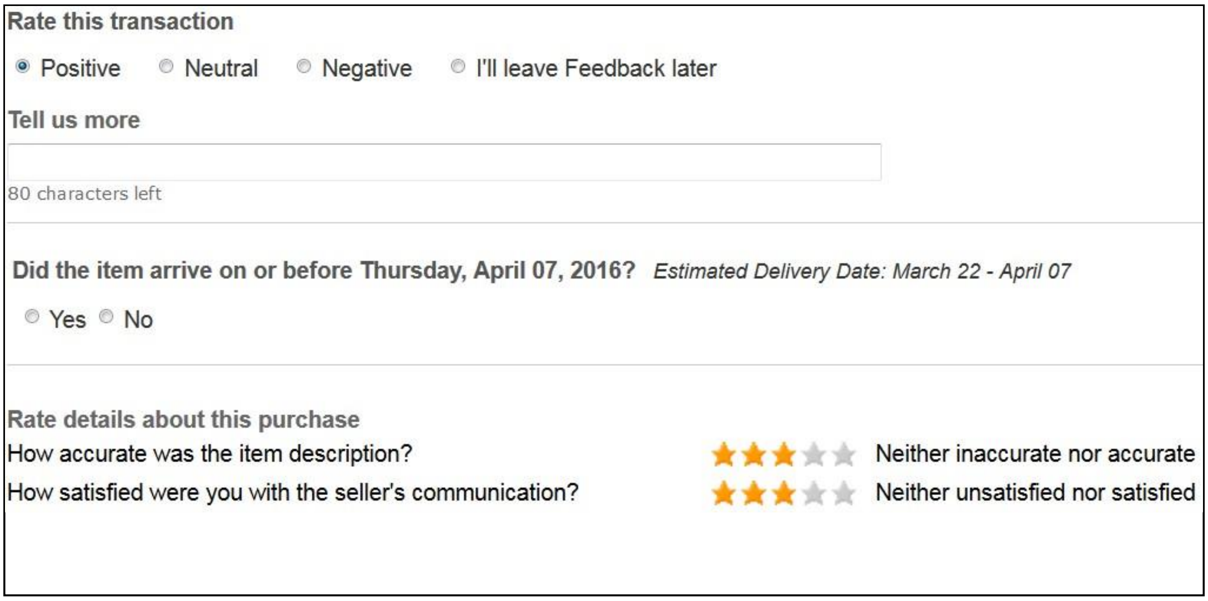


Figure 2.1: Example of Free-Format Customer Review from Amazon.com

Format 2 - Pros and Cons: This form of review is concise, and explicitly represents the positive and negative comments.

Format 3 - Pros, Cons, and the Detailed Review: This form of review combines both Formats 1 and 2 and allows the author to write comments and free text, and highlight the pros and cons, for example, ebay.com as shown in Figure 2.2.

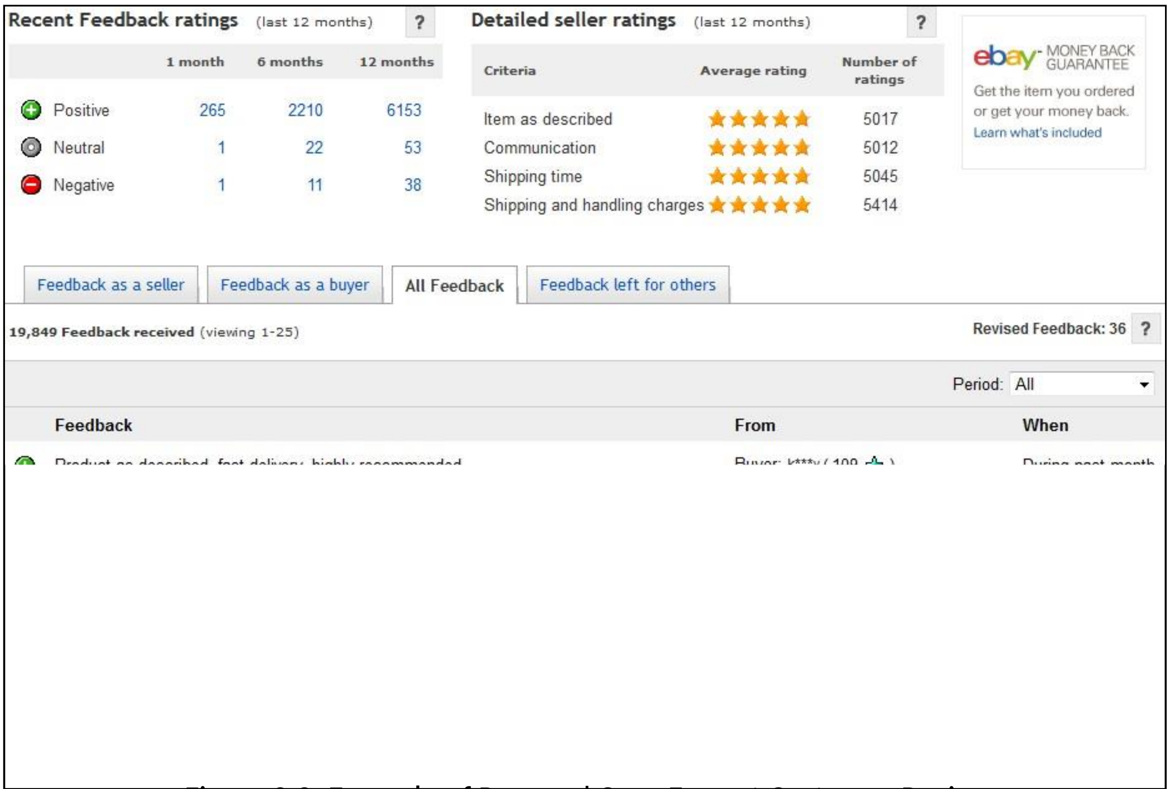


Figure 2.2: Example of Pros and Cons Format Customer Reviews

The entity and sentiment extraction process varies from free text format to the pros and cons format. [Liu et al., 2005] proposed a system called "opinion observer" to extract aspects and sentiments from pros and cons format reviews. It is based on sequential learning and relies on the two assumptions that all reviews are very short and each segment contains one aspect and its corresponding sentiment. Consequently, there remains a need for systems that can extract aspects.

2.4 Aspect-Based Opinion Mining

For many scenarios, document-level review classification is too coarse-grained and does not provide the desired information. Pure classification merely helps to gather information about how many customers are generally satisfied or unsatisfied. Based on these numbers where the trends in the customers' perceptions of a product can be found, but the exact reasons for satisfaction or dissatisfaction is not known We do not know what the customers like, and we do not know what they dislike. Aspect-oriented review mining goes one-step further and analyses the customers' sentiments with regard to individual product aspects. Whereas review classification considers only a single dimension (namely "sentiment polarity"), aspect-oriented review mining involves the joined analysis of two dimensions. The aim is to discover all relevant product aspects and the related expressions of sentiment needs to be identified and their polarity is determined. In contrast to review classification, the aspect-oriented task is better characterised as a problem in information extraction than a problem in text categorisation. It is transforming the unstructured information of a review text into a structured, aspect-oriented summary.

The main problem in the context of aspect extraction is to identify those text passages that refer to mentions of product aspects. Given a dictionary of relevant product aspects, the task would be relatively easy. However, if the relevant product aspects are not known a priori, therefore examining the provided collection of review documents is needed. Thus, the need to devise methods that automatically extracts a set of the most relevant product aspects from a corpus of reviews. To do so, notion of relevance need to be defined, and a desired level of granularity must be identified. Very fine-grained aspects are considered (e.g., "colour accuracy", "tone reproduction", "image noise", or "chromatic aberration") or more abstract concepts needs to be considered (e.g., "image quality", "ease of use", "battery", or "features"). Approaches to aspect extraction can be subdivided into three main classes: unsupervised, supervised, and topic modelling approaches.

Unsupervised approaches are typically frequency-based and often involve the use of pre-defined linguistic or syntactic patterns to detect candidate phrases. Most commonly, the goal is to automatically construct a dictionary of product aspects from a given review corpus.

2.4.1 Corpus-based Methods

Many studies of sentiment analysis (SA) have focused on extracting specific words, or given parts of speech (POS) [Liu, 2015]. Some of the POS tags (like adjectives) or sequences of POS (adjective- noun) have been shown to be more effective in opinion detection. The work of [Justeson and Katz, 1995] adopts an NLP-approach based on POS filtering [Toutanova and Manning, 2000].The words in the text are automatically processed and marked with appropriate POS tags. Afterwards, specific POS or given phrase patterns are filtered from the text, for example two adjectives in a row. [Hatzivassiloglou and McKeown, 1997] describe an approach based on the idea that the conjoined adjectives have the same orientation, apart from the ones used in the opposite orientation. They construct two clusters of adjectives using conjunction counts based on Wall Street Journal articles. Although they achieve quite high accuracy, it is important to note that they manually eliminated neutral adjectives on the first step. Other studies have focused on analysing single words and POS toautomatically deduce the polarity of a word from the data presented [Hatzivassiloglou and McKeown, 1997].

2.4.2 Machine Learning Methods

The field of machine learning has provided many models that are used to solve various text classification problems. Among them are Naïve Bayes (NB), Support Vector Machines (SVMs), decision trees, maximum entropy, and Hidden Markov Models (HMMs). The detailed overview of these and other algorithms can be found in the work of [Witten and Frank, 2005]. So far, the most popular machine learning approaches used as baselines are SVM and NB. [Pang et al., 2002] analysed several supervised machine learning algorithms on a movie reviews dataset, among them SVM, NB and maximum entropy. They also tested different feature selection techniques. Features are usually words, or bigrams of words, that could have been somehow pre-processed, for example, stemmed or lemmatised. The best performance was reported using the SVM method with unigram text representation. It has to be noted that the authors took into account just the presence of a feature, and did not count POS tagging information which may improve the effectiveness of NB and maximum entropy methods, but tends to decrease the performance for SVM. In a later study, [Pang and Lee, 2005] proposed to separate subjective sentences from the rest of the text at first. They assumed that two consecutive sentences would have similar subjectivity labels, as the author is inclined not to change sentence subjectivity too often. Thus, labelling all sentences as objective and subjective they reformulate the task of finding the minimum s-t cut in a graph [Kleinberg and Tardos, 2006].

2.4.3 Methods based on Conditional Random Fields (CRFs)

The accuracy level of the rule-based detection systems is approximately 50% as the review paper presented by [Rashid et al., 2013]. During error analysis, it has been identified that theme identification and subjectivity detection are deep semantic issues and it is nearly impossible to develop a complete set of definite rules. To overcome the limitations of a rule-based system, a machine-learning module has been developed with the already identified features along with a few additional ones. The Conditional Random Field (CRF) [Lafferty et al., 2001] machine- learning algorithm has been used. The CRF base subjectivity detection has achieved precision values of 76.08% and 79.90% for English news and movie review corpus, and 72.16% and 74.6% for Bengali news and blog domains respectively [Das and Bandyopadhyay, 2009]. CRF consists of a set of statistical modelling methods frequently used in pattern recognition and machine learning for structured prediction. This is very important for ABOM applications. Previous as well as ongoing studies have revealed that the sequence labelling approaches based on conditional relations increase the accuracy and performance of unstructured prediction tasks. Some noteworthy models for sequence labelling tasks are CRF [Lafferty et al., 2001], HMM [Eddy, 1996] and Max-Margin Markov Networks [Roller, 2004]. These models presented considerable improvement in several practical fields such as NLP, pattern recognition and infor- mation extraction. The models are applied to encode known relationships between reviewers' opinions and construct consistent interpretations of the reviews. With this scheme, CRF can predict the sequence of labels for a given input sequence where the reviews were considered as input sequences, and POS tags and opinion tags were used as output labels.

Generally, opinion extraction from customer reviews falls under the umbrella of phrase-level information mining. It aims to create a thorough sentiment analysis at the aspect level. Model-based approaches like HMM and CRF aim to overcome the limitations of the other approaches. HMM models assume that each feature is generated independently and ignore the underlying relationships between the actual words and labels, as well as the overlapping features [Qi and Chen, 2010]. CRF tackles those shortcomings since it is a discriminative model that instantiate the overlapping dependent features. [Choi et al., 2005] consider sentiment analysis as a hybrid task information extraction problem that combines CRF as a sequence tagging task and AutoSlog [Riloff, 1996] to learn the extraction patterns. Even though their system employs extraction learning with CRF, it resulted in a recall performance of 54% with exact match.

2.4.4 Information Retrieval Methods

Representation of documents in most supervised approaches is based on the vector space model [Salton et al., 1975]. Every document is represented by a multi-dimensional vector, where each dimension corresponds to some feature (term) in a document. Thus, a collection of documents can be contained in a document matrix, where an element (x, y) means the number of times feature x was encountered in document y. The idea is that it is possible to separate two classes of documents shown as vectors in the feature space. The supervised model is said to be trained when a classifier, trained on the set of labelled documents, constructs a multi- plane that separates the two classes of documents with reasonable degree of error. The use of the vector space model for document sentiment classification was explored in the work of [Sarvabhotla et al., 2011]. They compose two vectors to represent each document; the first is based on a calculation of the average document frequency, while the second is built using the average subjective measure. They retain terms with higher than average document frequency and subjective measure. For the feature selection, they apply mutual information and Fisher discrimination ratio and then train the SVM model.
Experiments were carried out on different portions of the retaurant reviews corpus and show improvement amelioration in performance in comparison to other feature weighting techniques using the SVM classifier.There are various researchers who have carried out research on the subject of Opinion Mining .

2.5 Applications of Opinion Mining

Opinion Summarisation

The requirements of the end user are the driving force behind sentiment analysis research. The outcomes of these research endeavours should lead to the development of a real-time sentiment analysis system, which will successfully satisfy the needs of the end users. Let us have a look at some real-life needs of the end user. For example, a market surveyor from company A may want to know how public opinion about their product X has changed after the release of product Y by company B. The different aspects of product Y that the public consider better than product X are also points of interest. These aspects could typically be the durability of the product, power options, weight, colour and many other issues that depend on the particular product. In another scenario, a voter may be interested to study the change of public opinion about any leader or public event before and after an election. In this case, the aspect could be a social event, economic recession or may be other issues. The end users are not only looking for the binary (positive/negative) sentiment classification but they are also interested in aspectual sentiment analysis. Therefore, sentiment detection and classification is not enough to satisfy the needs of the end user.

A sentiment analysis system should be able to understand and extract the aspectual sentiments present in a natural language text. Previous research efforts have proposed various structures or components for sentiment extraction. The most widely used sentiment structures are Holder [Choi et al., 2005, Kim and Hovy, 2004], Topic [Zhou and Hovy, 2006] and other domain dependent attributes. However, real-life users are not always interested about all the aspects at a time, rather they look for opinion/sentiment changes of any "Who" during "When" and depending upon "What" or "Where" and "Why". With this hypothesis, the 5Ws (Who, What, When, Where and Why) constituent extraction technique for sentiment/opinion structuring process has been proposed.

The topic-opinion model is the most popular but end users may want to look into an at-a-glance presentation of opinion- oriented summaries. For example, a market surveyor from company A might be interested in the root cause for why their product X (e.g., a camera) is becoming less popular day by day. Company A may want look into the negative reviews only. Relatively few research efforts could be found in the literature on the polarity-wise summarisation compared to the popular topic-opinion model. Four important related works have been presented here which are significant in the aspects, problem definition and solution architecture.

Although there are plenty of research works available in the field of text summarisation [Hu and Liu, 2004a, Kabadjov et al., 2011, Kim and Hovy, 2004] and also in the area of sentiment analysis [Kim and Hovy, 2004, Pang and Lee, 2008, Turney, 2002], there is still a limited amount of research available that merges these two areas comprehensively. For the first time in 2008, there was a Summarization Opinion Pilot track at the Text Analysis Conference organised by the US National Institute of Standards and Technology (NIST). The techniques used by the participants were primarily based on the already prevailing summarisation methods. At the same time, as most participants added new features to account for the presence of positive opinions or negative ones, CLASSY [Conroy and Schlesinger, 2008], LIPN [Bossard et al., 2008], and IIITSum08 [Varma et al., 2008], efficient methods were proposed focusing on the retrieval and filtering stage, based on polarity DLSIUAES [Balahur et al., 2008] or on separating information rich clauses [Cruz et al., 2008]. Finally, fine-grained, feature-based opinion summarisation was defined by [Hu and Liu, 2004b].

The fact that opinion summarisation was presented in such an intricate setting, which additionally requires the determination of answers to opinion questions, the endeavour led to very poor results from the participating groups. After realising the immense challenge presented by end-to-end systems, research in this field has been split into two sub-divisions: opinion question answering and opinion summarisation.

Research Objectives	Research Questions	Methodology
The main objective of this thesis is to eliminate the manual analysis of a product reviews and replace with an automated system using some NLP Analysis techniques. The second major objective is to implement system that is capable of interpreting and understanding feeling as well as emotions of human based on previous experiments.	What is the effectiveness of manual analysis of product reviews from a reviews over an automated system? What are the better techniques of implementing an aspect-based opinion mining systems?	The methodology of this system involves the process of data analysis, preprocessing, aspects and opinion extraction, subjectivity and objectivity classification, orientation detection an finally aspect-based summary generating. Preprocessing techniques improve the accuracy of the process of opinion mining. Aspect extraction involves the use of identifying aspects which in most cases are nouns as well as noun phrases. Subjectivity classification is necessary because not all sentences in product reviews express an opinion. This step involves identification of orientation of an opinion on every aspect. The final aspect of this system would involve aspect-based summary. It involves aggregation of negative as well as positive scores thereby creating a sentiment profile of each product. Visualization tool may be applied for this purpose as future work. It will be useful for customers to understanding the opinion of various aspects of a product and also get a rough idea on each aspect. To determine the accuracy, the system can use precision, recall as well as F-measure.

Table 2.1: Methodology at a High Level

2.6 Challenges in Opinion Mining

While trying to identify expressions of sentiment in natural language text and to determine the conveyed polarity as well as considering the main sub-problem of extracting product aspects from customer reviews, the following challenges are primarily confronted:

2.6.1 Implicit Sentiment

Besides explicit expressions of sentiment (e.g., "the check-in process went fast"), sentiment may also be manifested implicitly ("needed to wait two hours to check in"). Whereas the first example includes a subjective assessment ("fast"), the second example merely expresses a fact (two-hour waiting time). To infer a negative evaluation of the check-in process in the second example, the common sense knowledge needed to be applied, in which a two-hour waiting time is normally inappropriate. In the literature this form of implicit sentiment is referred to as objective polar utterance [Wilson, 2008], evaluative fact [Nigam and Hurst, 2006], or polar fact [Toprak et al., 2010].

2.6.2 Contextual Polarity

The sentiment polarity of a phrase may be context dependent. For instance, consider the sentence "the hotel staff was not very friendly". The negation "not" flips the otherwise positive polarity of the word "friendly". Words, phrases, or syntactic constructions that affect the sentiment polarity or sentiment strength are commonly denoted as sentiment or valence shifters [Polanyi and Zaenen, 2006]. A detailed study of contextual polarity was conducted by [Wilson, 2008].

2.6.3 Target-Specific Polarity

The sentiment polarity of words and phrases may depend on the modified target. For instance, consider the adjective "long". If it modifies the product aspect "battery life", it refers to a positive evaluation. However, in the context of the aspect "flash recycle time", it would be interpreted as negative. Most sentiment lexicons ignore this phenomenon and only consider the prior polarity of words.

2.6.4 Relations Among Aspects

Product aspects are typically related among each other. For instance, part-of or type-of relations between different aspects can be observed (e.g., a lens cover is part of a camera lens and the landscape mode is a type of digital camera mode). Depending on the application scenario, these hierarchical relations needed to be made explicit and construct some sort of product aspect taxonomy. Another common relation is similarity. It is reasonable to group similar aspects and to detect synonyms. For instance, representing aspect references such as “image quality”, “picture quality”, or “quality of image” by the single canonical form “image quality”. Automatically grouping entities (e.g., product aspects) and determining relations between them can be considered as a problem of ontology learning [Maedche, 2002].

2.6.5 Implicit Aspect Mentions

Given a dictionary of relevant product aspects, it is relatively easy to identify the explicit mentions of those aspects in a review text. It is much more difficult to discover implicit mentions. For instance, the phrase “slept like rocks” can be considered. This phrase implicitly refers to the aspect “quality of sleep”. Another example would be “the camera is too heavy”. Without explicitly mentioning the term, the reviewer criticises the camera’s weight.

2.6.6 Comparative Reviews

Until now, the assumption is that a review only referred to a single product. It is, however, not always true. A reviewer may evaluate a product by comparing it to other, similar products. If this is the case, the different product entities mentioned in the text, needed to be determined. Then carefully need to inspect which expressions of sentiment relate to which product entity. This specific challenge is considered by [Ganapathibhotla and Liu, 2008]. Comparative evaluations are more common in expert reviews and occur relatively seldom in customer reviews.

Besides the main subtasks of identifying and extracting aspects, this research is confronted with further challenges such as relation among aspects and opinions, and identifying implicit aspects.

2.7 Approaches in aspect based sentiment analysis

2.7.1 Frequency –based Approach

In most of the reviews only a limited set of words is used often, these words are single nouns and compound nouns which are known to be aspects. The words that occur frequent ly are termed as frequent words or aspects. This approach was the most common method for aspect detection. The most familiar method using frequency based approach is [1]. It considers single and compound nouns for aspect extraction. The author in [2] followed an unsupervised information extraction system namely OPINE to extract important product features from reviews and later a classification technique is applied.

A most important shortcoming of the frequency based approach is the majorit y of the noun and noun phrases having a high frequency are mistakenly seen as aspects. To overcome this issue, in [3] Red Opal introduced a system, for a phrase to be considered as aspect, a word or bigram has to appear more often higher than the given baseline frequency .This method suggested that users prefer bigram features over unigram features .Th e baseline statistics method is used in [4], where it used to filter the list of high frequency noun phrases. Additionally in [5] have used part -of-speech (POS) pattern filter such that every aspect needs to be followed by adjective to identify frequent nouns and noun phrases from the reviews.

2.7.2 Syntax-Based Approach

As an alternative of focusing on frequencies to discover aspects, syntax -based methods and find aspects by way of the syntactical associations they are in. An extremely easy relation is the adjectival modifier relation between a sentiment word and an aspect, as in ,"fantastic food", where "fantastic" is an adjective modifying the aspect "food". A strong point of syntax-based methods is that low frequency aspects can be originated. Yet, to get good coverage, many syntactical relations need to be described.

To alleviate the low recall crisis, a generalization step for syntactic patterns using a tree kernel function is projected in this paper [6]. Given a labelled data set, the syntactic patterns of all the annotated aspects are extracted. Then, for the unseen facts, syntax trees of all sentences are obtained. As an alternative of straight trying to discover an accurate match between the aspect pattern and the syntax tree, both are split into several different substructures. Then the comparison between the pattern and a sentence can be calculated as the numeral of matching substructures. The common convolution tree kernel is used to figure similarity scores for each pair of substructures, with a threshold determining whether a pair is a match or not. In [7], [9] and [8], aspect detection and sentiment lexicon expansion are seen as interrelated problems for which a double propagation algorithm is projected, featuring parallel sentiment word expansion and aspect detection. With every extra known sentiment word, additional aspects can be originated, and with extra known aspect words, extra sentiment words can be found, etc.

The algorithm continues this method until no more extra sentiment words or targets can be initiated. To find sentiment words based on known aspect words, and the other way around, a set of rules based on grammatical relations from the employed dependency parser, is raised. A big advantage of this method is that it only needs a small seed set to work properly compared to the large corpus most trained classifiers require.

2.7.3 Supervised Machine Learning Approach

There are only quite a few supervised machine learning methods for aspect detection that are entirely machine learning methods. Since the power of supervised approaches lies in the features that are used, feature construction regularly consists of other methods (e.g., frequency-based methods) in order to generate more significant features that simplify better than plain bag of- words or part-of-speech features.

Aspect detection is cast as a labeling problem in [10], which is solved by linear chain Conditional Random Fie ld (CRF), familiar in natural language processing, for processing a entire sequence of words. This context of a word is automatically considered when assigning it a label. Mult iple features are used for determining the best label for a word, which also includes the actual word, its part-of-speech tag, and also the other factors like, whether a direct dependency relation exists between this word and a sentiment expression, whether this word is in the noun phrase that is close by to a sentiment expression, and whether this word is in a sentence that actually has a sentiment expression. The ground - truth from a subset of the used data sets [11] [12] [13] is used to train the model.

2.7.4 Unsupervised machine learning

Unsupervised machine learning approach towards aspect based sentiment classification can resolve the problem of domain dependency and reduce the need for labeled training data. However, a large amount of data is usually needed to successfully train these type of data. The most common approach used is LDA, which is a topic model proposed in [14].LDA is similar to probabilist ic latent semantic analysis which uses a dirichlet pior for topic distribution instead of a uniform topic distribution [15]. The disadvantage of LDA is that the generated topics are unlabeled, that shows no direct correspondence between topics and specific aspects or entities.

Since LDA was designed to operate on the document level, implementing it for the much finer-grained aspect-level sentiment analysis is complicated. The first LDA-based approach for aspect-level is discussed in [16]. To overcome this problem an extension to LDA is proposed known as Multi-grain LDA which has two model level namely global and local. This model approach [17] increases the accuracy of the loc al topics that should represent the sought aspects. In [18] a similar notion is described where a distinction is made between global and local topics. In [19] LDA is combined with a Hidden Markov Model (HMM) to distinguish between aspect -words and background words.

In [20] another way of another way of adding syntactic dependencies is shown, where the topic model employs two vocabularies to pick words from. In [21] so -called cold start problem is proposed which incorporates product categories and the reviewers into the model. This distribut ion over the aspects with reviewer comments or with rating gives a more accurate predict ion for products with little or no data. In [22], a supervised joint aspect and sentiment model is proposed to determine the reviews on aspect level. It simultaneously models both aspect and sentiment words to improve the quality of found aspect topics.

2.7.5 Hybrid Approach

This approach is a combination of two methods that are used known as hybrid method which has two types: serial hybridization, where the output of one phase forms the input of next phase, and parallel hybridization, where two or more methods are used to find complementary sets of aspects. In [23] serial hybridization is used, where Pointwise Mutual Informat ion [24] is used to find possible aspects, which then imported into a naïve bayes classifier to out put a set of explicit aspects. Serial hybridizat ion also used in [25], where the Dice similarity measure [26] is used to cluster noun phrases that are about the same aspect, and [27] which targets pros and cons to find aspects using frequent nouns and noun phrases, feeding those into an SVM classifier to make the final decision whether it is an aspect or not.

The parallel hybridization can be found in [28], where a MaxEnt classifier is used to find the frequent aspects, for which there is ample data, and a rule-based method that uses frequency information and syntactic patterns to find the less frequent ones. In this way, available data is used to drive aspect detection, with a rule-based method that acts as back-up for cases where there is not enough data available.

2.8 Aspect based sentiment analysis task

The most important objective of Aspect Based Sent iment Analysis is to identify the aspects of the given target entities and sentiment expressed for each aspect. The objectives of Aspect Based Sentiment Analysis can be done through the following tasks. The first task is the extraction of aspect terms and grouping aspect terms into aspect categories. The second task is about identification of polarity of the aspect terms and polarity of the aspect categories of each sentence. The above tasks are divided into four sub tasks namely: Aspect Term Extraction (ATE), Aspect Term Polarity (ATP), Aspect Category Detection (ACD) and Aspect Category Polarity (ACP).

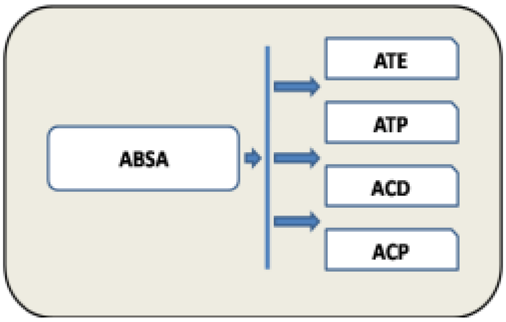


Fig 2.3 Aspect Based Sentiment Analysis Task

2.8.1 Aspect Term Extraction

The work of first sub-task Aspect Term Extraction (ATE) is also known as information extraction task is to identify all the aspect terms given in each review sentence. There can be multiple aspects, in a sentence and every aspect need to be extracted. The aspect in the aspect terms of the sentence can be expressed by a noun, verb, adverb and adject ive. It is well known that 60% - 70% of aspect terms are explicit nouns. The aspect terms can also consist of mult iword entities such as “screen size”. These multiword entit ies and their aspects are considered to be much critical than single word aspects. The researchers have used various processes for extracting aspect terms, like Word N-grams, Bigrams, Word cluster, Casting, POS tagging, Parse dependencies, Relations and Punctuation marks. The various Methods used for extracting aspect terms , such as Condit ional Random Fields (CRF), Support Vector Machines (SVM), Random trees and Random Forest.

The researchers have proposed various methods for Aspect Extraction Task. They are FREQ baseline, H&L method, H&L+W2V method and FREQ+W2V method. All these methods come under unsupervised learning and can be used across domains with minimum changes. The FREQ baseline is considered to be the most efficient method because it returns the most dist inct nouns and noun phrases from the reviews in each dataset by decreasing the sentence frequency.

Researchers have also used LDA (Latent Dirichlet Allocat ion) based methods for Aspect extraction. The method of Hu and Liu (2004) extracts all the different nouns and noun phrases from the reviews of each dataset and consider them as candidate distinct aspect terms. In a co -occurrence based method [30] for category discovery a dictionary based sentiment classification algorithm is used through which aspects can be identified by annotation process. On the other hand, by using the training set to count how frequently each word appears within an aspect, a simple probability should be computed , which specifies the chance that this word is an aspect word or not.

The above probability is also used for filtering a set of noun phrases, such that the noun phrases remaining will have at least one word. The aspect probability for the noun phrases holding the value greater than or equal to 0.05 and the noun phrases with the probability below 0.05 are removed. This process will remove some of the determiner words from the initial noun phrase, as those are excluded. A modelled Aspect extraction [31] as sequential labelling task and extract features is used for CRF training. In addition to the common features used in Named Entity Recognition (NER) systems, it also uses the available external resources for building different name lists and word clusters.

A supervised machine learning algorithm [31] is used to extract the aspect term. An aspect can be expressed by a noun, adjective, verb or Input Text adverb. But the recent research in (Liu, 2007) shows that 70% of the aspect terms are explicit nouns and the aspect terms can contain of multiword entities. In [33] Aspect term can be extracted using casting as a sequence tagging task, in which each token in a candidate sentence is denoted as either Beginning, Inside or Outside (BIO). Conditional Random Fields [35] (CRF) is used for extracting aspect terms and BIO model for representing aspect terms.

The conditional random fields (CRF) and a liner chain CRF are used for determining the conditional probability. The authors employ a graph co-ranking approach [34], to model aspect terms and opinion words as graph nodes, and then they generate three different sub-graphs defining their bond between the nodes. To obtain a list of dependable aspect term, the candidates rank the nodes using a combined random walk on the given three sub graphs.

2.8.2 Aspect Term Polarity

The second sub-task is aspect term polarity is that, within a sentence for a given set of aspect terms, the task is to determine the polarity of each aspect term: positive, negative, neutral or conflict (i.e., both positive and negative). Here in the identification of Aspect term polarity different features like Word N -grams, Polarity of neighboring adjectives, Neighboring POS tags and Parse dependencies and relations have been widely used by researchers.

The sentiment of aspect in [30] is computed by using sentiment value of each n-gram and distance between the n-gram and the aspect. In [31] Aspect lexicon based on additional informat ion such as POS for polarity identification was developed by the author. In [32] a new class called conflict has been introduced along with they developed a method called RFC (Random Forest Classification). They have used many futures in this classification like local context, POS, chunk, prefix and suffix. In [33] aspect term polarity, they have extracted it by using various features like word N-grams, polarity of neighboring adjectives, neighboring POS tags and parse dependencies and relations.

In [34] the author reusing the generated Word2Vec model, developed a polarity lexicon for the corresponding domain with the perception that a polarity word in a domain should be more "similar" to a set of "very posi ive" words than to a set of "very negative" words, and vice versa. This is engaged the in -domain generated Word2Vec models since the polarity of words may differ between domains and wanted to detain the polarity for each particular domain. In [35] the words that affect the sentiment of the aspect term are assumed to be close in most of cases and thus used a context window of 10 words in both directions around the target aspect term.

2.8.3 Aspect Category Detection

The third sub-task is Aspect Category Detection, in which the task is to identify the majority of categories that are discussed in each sentence. Aspect categories are usually difficult to find than the aspect terms as defined in Aspect Term Extraction, and at times they do not even occur as terms in the sentence. Aspect category detection [35] is based on a set of binary Maximum Entropy classifiers. The final decision is merely calculated from decisions of various individual classifiers.

Aspect category classification [29] is based on a set of available binary classifiers, one classifier for each category found in the training set. To create a training example each sentence in the training set, the e xtracted features from all words in the sentence is taken. The co -occurrence based algorithm [30] is used for category detection. The algorithm is a co-occurrence matrix that captures the frequency ofthe co-occurrences between words in the sentence and the annotated aspect category, gives mapping from words to aspect categories.

Aspect category detection [33] is considered as multi label classification problem. In a given instance, it should predict all labels that instance fit into. In [36] Aspect category detection the authors have used supervised classification approach and each task is done by identifying every entity E and attribute A pair E#A towards which an opinion is expressed.

2.8.4 Aspect Category Polarity

The final sub-task is Aspect Category Polarity is which it takes the information from the previous task (Aspect Category Detection) to determine the polarity of each aspect category discussed in review sentence. The sentiment of aspect category [30] is computed by calculating the distanc e between n-gram and the corresponding aspect. The aspect category polarity has been detected using just unigram and bigram features in [33]. In aspect category polarity detection [35], the whole sentence is taken into account, and maximum entropy classifier is used to distinguish one category with other. In [36] for sentiment polarity classification, authors have extracted Bag of Words and Wordnet Synset features from both train and test data and implemented them on variety of classifiers (like Stochastic Gradient Descent, SVM, Adaboost) multiple times and stored the confidence scores obtained from decision functions of each of these classifiers.

CHAPTER 3

ASPECT EXTRACTION USING DEEP CNN

This chapter discusses the first step, Aspect Extraction using deep CNN. First, it briefly introduces about CNN for sentence classification, and then Section 3.2 discusses the methodology, starting with the problem statement and ending by describing the method in detail. Section 3.3 shows the experiment tools used. The chapter concludes with a discussion and a summary. All the results, analysis and further discussions are Chapter 5.

3.1 Convolutional Neural Networks for NLP

CNNs were responsible for major breakthroughs in Image Classification and are the core of most Computer Vision systems today, from Facebook's automated photo tagging to self-driving cars. More recently CNNs are used to address problems in Natural Language Processing and gotten some interesting results. This section summarizes the use and working of CNN in NLP.

A Convolutional Neural Network typically involves two operations, which can be though of as feature extractors: convolution and pooling.

The output of this sequence of operations is then typically connected to a fully connected layer which is in principle the same as the traditional multi-layer perceptron neural network (MLP).

3.1.1 Definitions and Terminologies

- convolution : an operation which applies a filter to a fixed size window.
- convolution filter or kernel: a template matrix which is used in the convolution operation.
- pooling: combines the vectors resulting from different convolution windows into a singlel-dimensional vector.
- feature_maps : the number of feature maps directly controls capacity and depends on the number of available examples and task complexity.

3.1.2 CNN Architecure Overview

The input to most NLP tasks are sentences or documents represented as a matrix. Each row of the matrix corresponds to one token, typically a word, but it could be a character. That is, each row is vector that represents a word. Typically, these vectors are word embeddings (low-dimensional representations) like word2vecor GloVe, but they could also be one-hot vectors that index the word into a vocabulary. For a 10 word sentence using a 100-dimensional embedding we would have a 10×100 matrix as our input.

In NLP we typically use filters that slide over full rows of the matrix (words). Thus, the "width" of our filters is usually the same as the width of the input matrix. The height, or region size, may vary, but sliding windows over 2-5 words at a time is typical. Putting all the above together, a Convolutional Neural Network for NLP may look like as shown in Fig 3.1.

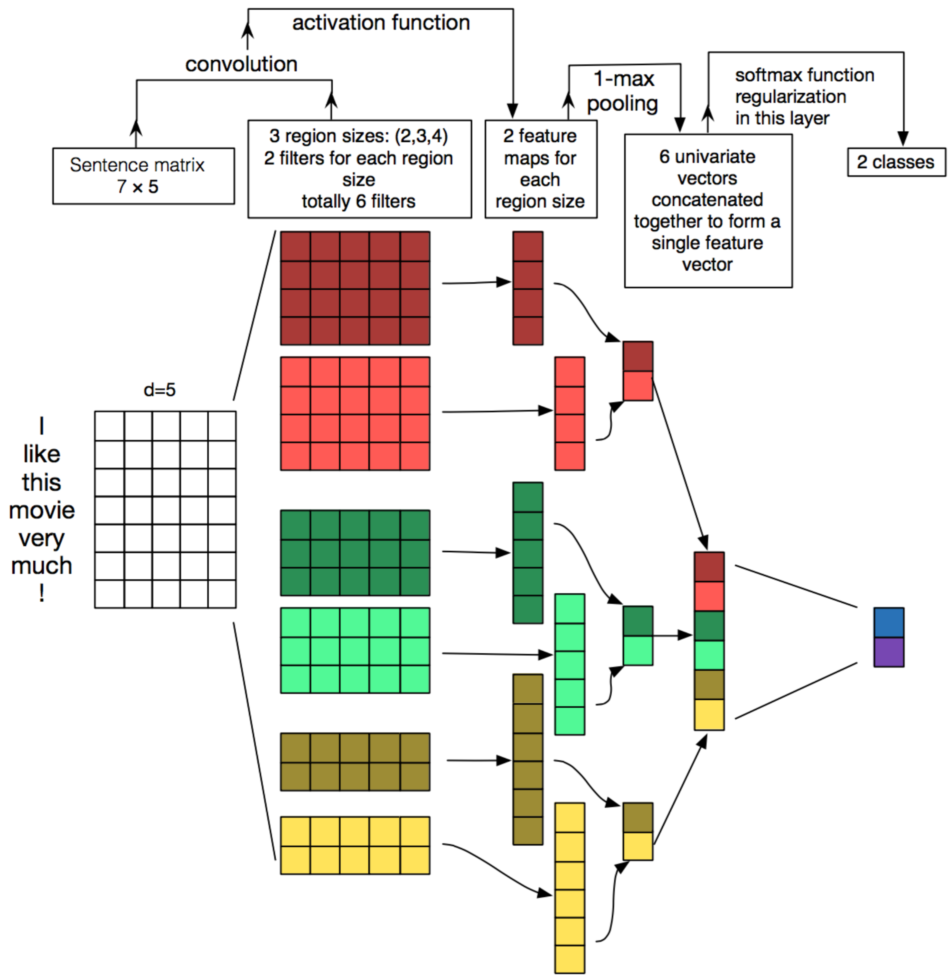


Fig 3.1 Working of CNN For NLP

Img:http://www.wildml.com/2015/11/understanding-convolutional-neural-networks-for-nlp

3.1.3 1-D Convolutions over Text

We have a 1 dimensional array representing the text. Given a sequence of words $w_{1:n} = w_1, \ldots, w_n$, where each is associated with an embedding vector of dimension. A 1D convolution of width-k is the result of moving a sliding-window of sizek over the sentence, and applying the same convolution filter or kernel to each window in the sequence, i.e., a dot-product between the concatenation of the embedding vectors in a given window and a weight vector, which is then often followed by a non-linear activation functiong.

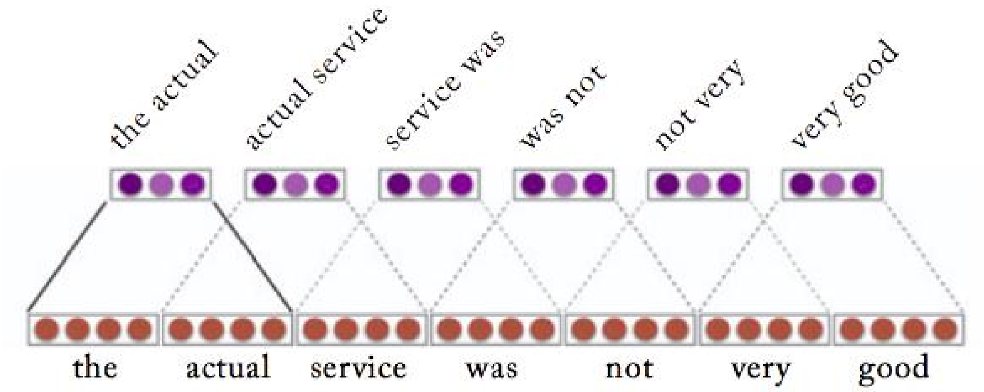


Fig 3.2 Example of a sentence convolution with k=2 and dimensional output l=3.
(Image adapted from Yoav Goldbergbook "Neural Network Methods for NLP")

Fig 3.2 represents An example of a sentence convolution in a vector-concatenation notation.

3.1.4 Pooling

The pooling operation is used to combine the vectors resulting from different convolution windows into a single l-dimensional vector. This is done again by taking the max or the average value observed in resulting vector from the convolutions. Ideally this vector will capture the most relevant features of the sentence/document.

This vector is then fed further down in the network - hence, the idea that ConvNet itself is just a feature extractor - most probably to a full connected layer to perform prediction.

3.2 METHOD

3.2.1 Problem Statement

Most of the previous works in aspect term extraction have either used conditional random fields (CRFs) or linguistic patterns . Both of these approaches have their own limitations: CRF is a linear model, so it needs a large number of features to work well; linguistic patterns need to be crafted by hand, and they crucially depend on the grammatical accuracy of the sentences. In this paper, we overcome both limitations by using a convolutional neural network (CNN), a non-linear supervised classifier that can more easily fit the data. In addition, we use linguistic patterns to further improve the performance of the method, though in this case the above-mentioned issues inherent in linguistic patterns affect the framework.

3.2.2 Architecture for Aspect Extraction

The features of an aspect term depend on its surrounding words. Thus, we used a window of 5 words around each word in a sentence, i.e., ± 2 words. We formed the local features of that window and considered them to be features of the middle word. Then, the feature vector was fed to a CNN.

To extend the deep neural network to a deep CNN, one simply partitions the hidden layer into Z groups. Each of the Z groups is associated with an $n_x \times n_y$ filter, where n_x is the height of the kernel and n_y is the width of the kernel. Assume that the input has dimensions $L_x \times L_y$, which in our case is given by L_x words in the sentence and L_y features, such as word embedding, of each word. Then the convolution will result in a hidden layer of Z groups, each of dimension $(L_x - n_x + 1) \times (L_y - n_y + 1)$.

The network contained one input layer, two convolution layers, two max-pool layers, and a fully connected layer with softmax output. The first convolution layer consisted of 100 feature maps with filter size 2. The second convolution layer had 50 feature maps with filter size 3. The stride in each convolution layer is 1 as we wanted to tag each word. A max-pooling layer followed each convolution layer. The pool size we use in the max-pool layers was 2. We used regularization with dropout on the penultimate layer with a constraint on L2-norms of the weight vectors, with 30 epochs. The output of each convolution layer was computed using a non-linear function; in our case we used the hyperbolic tangent.

As features, we used word embeddings trained on two different corpora. The CNN produces local features around each word in a sentence and then combines these features into a global feature vector. Since the kernel size for the two convolution layers was different, the dimensionality $L_x \times L_y$ was 3 × 300 and 2 × 300, respectively. The input layer was 65 × 300, where 65 was the maximum number of words in a sentence, and 300 the dimensionality of the word embeddings used, per each word. The process was performed for each word in a sentence. Unlike traditional max-likelihood leaning scheme, we trained the system using propagation after convolving all tokens in the sentence. Namely, we stored the weights, biases, and features for each token after convolution and only back-propagated the error in order to correct them once all tokens were processed using the training scheme.

3.2.3 Dataset used

1.) SemEval 2014 dataset - The dataset consists of training and test sets from two domains, Laptop and Restaurant;

Eg: Sample data

```
<Review rid="en_BlueRibbonSushi_478218171">
<sentences>
<sentence id="en_BlueRibbonSushi_478218171:0">
<text>Yum!</text>
</sentence>
<sentence id="en_BlueRibbonSushi_478218171:1">
<text>Serves really good sushi.</text>
</sentence>
<sentence id="en_BlueRibbonSushi_478218171:2">
<text>Not the biggest portions but adequate.</text>
</sentence>
</sentences>
</Review>
```

The annotations in this corpora were encoded according to IOB2, a widely used coding scheme for representing sequences. In this encoding, the first word of each chunk starts with a “B-Type” tag, “I-Type” is the continuation of the chunk and “O” is used to tag a word which is out of the chunk. In our case, we are interested to determine whether a word or chunk is an aspect, so we only have “B–A”, “I–A” and “O” tags for the words.

Here is an example of IOB2 tags:

also/O excellent/O operating/B-A system/I-A ,/O size/B-A and/O weight/B-A for/O optimal/O mobility/B-A excellent/O durability/B-A of/O the/O battery/B-A the/O functions/O provided/O by/O the/O trackpad/B-A is/O unmatched/O by/O any/O other/O brand/O

2.) Google word2vec embeddings:

This tool provides an efficient implementation of the continuous bag-of-words and skip-gram architectures for computing vector representations of words. These representations can be subsequently used in many natural language processing applications and for further research.

Word2vec is an algorithm invented at Google for training word embeddings. Word2vec relies on the distributional hypothesis to map semantically similar words to geometrically close embedding vectors. The distributional hypothesis states that words which often have the same neighboring words tend to be semantically similar. Both "dog" and "cat" frequently appear close to the word "vet", and this fact reflects their semantic similarity.

Word2Vec exploits contextual information like this by training a neural net to distinguish actually co-occurring groups of words from randomly grouped words. The input layer takes a sparse representation of a target word together with one or more context words. This input connects to a single, smaller hidden layer.
embedding can be reused in other classifiers.
In one version of the algorithm, the system makes a negative example by substituting a random noise word for the target word. Given the positive example "the plane flies", the system might swap in "jogging" to create the contrasting negative example "the jogging flies".

The other version of the algorithm creates negative examples by pairing the true target word with randomly chosen context words. So it might take the positive examples (the, plane), (flies, plane) and the negative examples (compiled, plane), (who, plane) and learn to identify which pairs actually appeared together in text.
embedding can be reused in other classifiers.The classifier is not the real goal for either version of the system, however. After the model
has been trained, you have an embedding. You can use the weights connecting the input layer with the hidden layer to map sparse representations of words to smaller vectors. This embedding can be reused in other classifiers.

3.3 SUMMARY

In summary, We have introduced the first deep learning-based approach to aspect extraction. As expected, this approach gave a significant improvement in performance over state-of-the-art approaches. We proposed a specific deep CNN architecture that comprises seven layers: the input layer, consisting of word embedding features for each word in the sentence; two convolution layers, each followed by a max-pooling layer; a fully connected layer; and, finally, the output layer, which contained one neuron per each word.

CHAPTER 4

Sentiment polarity detection from aspects extracted

4.1 Objective

Given an aspect term (also called opinion target) in a sentence, predict the sentiment label for the aspect term in the sentence.

4.2 Definitions and terminologies

Opinion

Opinion, in general, is “a view or a judgement formed about something that is not necessarily based on a fact or existing knowledge” [Soanes and Stevenson, 2006]. In the problem of sentiment analysis, Liu indicates that opinion is “a quintuple of Entity, Aspect, Orientation, Opinion Holder and Time” [Liu, 2010]. The entity is the name of an entity, which could refer to a product. The aspect can be a feature, component or function of the entity. The orientation is the opinion provided about the entity and/or the aspect that was provided by the opinion holder at a specific time.

Sentiments

A sentiment is a view or opinion that is held or expressed.Sentiments may be narrated as opinions, ideas or as judgements manifested by emotions Human sentiment knowledge grows by day-to-day cognitive interactions. Sentiment is not a direct property of languages. An intelligent system should need some prior knowledge to act properly. Sentiment knowledge is generally wrapped into computational lexicon, technically called sentiment lexicon.

Polarity classification

Classifying the polarity of a given text is a basic but important task in sentiment analysis: whether the expressed attitude in a document, a sentence, an entity, or an aspect is positive, negative or neutral. A large body of work related to reviews falls into this category. After all, the most interesting factor in reviews is how products or services are received by customers. In this section, we discuss opinion polarity classification at different levels. In most existing work, “sentiment analysis” or “sentiment classification” is used as synonyms of opinion polarity classification, although sentiment analysis actually includes many other subtasks. In this chapter, we use the name “polarity classification”. In terms of granularity, it

can be categorized at document, sentence, entity, or aspect level [YSZ17a]. At document level, the target of polarity classification is the whole review document. The review could be treated as a atomic unit of input or a unit bearing hierarchical structures 42 that consists of paragraphs, sentences, and words. People usually hold different attitudes in different sentences and paragraphs. It may be beneficial to summarize the sentiment polarities from bottom to top according to the internal structure of the given document.

At sentence level, the goal of this task is to determine the polarity of the given sentence. Small structure such as clauses, and the syntax dependencies can be exploited. One of challenges is that user-generated reviews often have grammar mistakes, misspelling errors, or emoticons.

At aspect level, there are two different tasks. One is to predict the sentiment polarity with regard to the specific aspect of a product or service, which belongs to a set of predefined classes. The words of aspect categories may or may not appear in the text. The other is to identify the polarity concerning the interesting entities which appear in text. Most aspect entities are phrases and the size of the vocabulary of aspect phrases could be more than a thousand. For instance, in the sentence “Average to good Thai food, but terrible delivery.”, the reviewer expresses positive towards the entity Thai food, but negative toward the aspect SERVICE, which actually does not show in the given text.

Opinion polarity classification is usually defined as a supervised learning problem. One critical step is how to represent text data. For general-purpose machine learning models such as support vector machine and naive Bayes [PLV02], people have to manually construct useful features to encode text into a sparse high-dimension vector. However, neural network based learning models are often based on distributed representation of words, each of which is embedded into a low dimension space. In the next following sections, we classify these neural networks into two categories: neural networks for general text classification and neural networks for sentiment polarity classification on reviews.

4.3 Sentiment analysis

Sentiment analysis aims to estimate the sentiment polarity of a body of text based solely on its content. The sentiment polarity of text can be defined as a value that says whether the expressed opinion is positive (polarity=1), negative (polarity=1), or neutral. In this tutorial, we will assume that texts are either positive or negative, but that they can’t be neutral. Under this assumption, Sentiment Analysis can be expressed as the following classification problem:

- Feature: the string representing the input text

- Target: the text's polarity (1 or -1)

But there is something unusual about this task, which is that the only feature we are working with is non-numerical. And in order to be able to train a machine/deep learning classifier, we need numerical features.

Unfortunately, we can't even use one-hot encoding as we would do on a categorical feature (such as a color feature with values red, green, blue, etc.) because the texts aren't categories, and there is probably no text that is exactly the same as another. Using one-hot encoding in this case would simply result in learning "by heart" the sentiment polarity of each text in the training dataset.

At first glance, solving this problem may seem difficult — but actually, very simple methods can go a long way. We need to transform the main feature — i.e., a succession of words, spaces, punctuation and sometimes other things like emojis — into some numerical features that can be used in a learning algorithm. To achieve this, we will follow two basic steps:

- A pre-processing step to make the texts cleaner and easier to process
- And a vectorization step to transform these texts into numerical vectors.

4.3.1 Pre-Processing

A simple approach is to assume that the smallest unit of information in a text is the word (as opposed to the character). Therefore, we will be representing our texts as word sequences.
For instance:
Text:This is a cat. -->Word Sequence: [this, is, a, cat]

In this example, we removed the punctuation and made each word lowercase because we assume that punctuation and letter case don't influence the meaning of words. In fact, we want to avoid making distinctions between similar words such as This and this or cat. And cat.

Moreover, real life text is often "dirty." Because this text is usually automatically scraped rom the web, some HTML code can get mixed up with the actual text. So we also need to tidy up these texts a little bit to avoid having HTML code words in our word sequences. For example :
<div>This is not a sentence.<\div> --> [this, is, not, a, sentence]

Making these changes to our text before turning them into word sequences is called pre-processing. Despite being very simple, the pre-processing techniques we have seen so far work very well in practice. Depending on the kind of texts you may encounter, it may be relevant to include more complex pre-processing steps. But keep in mind that the more steps you add, the longer the pre-processing will take.

4.3.2 Vectorization

The simplest text vectorization technique is Bag Of Words (BOW). It starts with a list of words called the vocabulary (this is often all the words that occur in the training data). Then, given an input text, it outputs a numerical vector which is simply the vector of word counts for each word of the vocabulary.

For example:
Training texts: ["This is a good cat", "This is a bad day"]
=> vocabulary: [this, cat, day, is, good, a, bad]
New text:"This day is a good day" --> [1, 0, 2, 1, 1, 1, 0]

As we can see, the values for “cat” and “bad” are 0 because these words don’t appear in the original text.

Using BOW is making the assumption that the more a word appears in a text, the more it is representative of its meaning. Therefore, we assume that given a set of positive and negative text, a good classifier will be able to detect patterns in word distributions and learn to predict the sentiment of a text based on which words occur and how many times they do. To use BOW vectorization in Python, we can rely on CountVectorizer from the scikit-learn library. In addition to performing vectorization, it will also allow us to remove stop words (i.e., very common words that don’t have a lot of meaning, like this, that, or the). Scikit-learn has a built-in list of stop words that can be ignored by passing stop_words="english"to the vectorizer. Moreover, we can pass our custom pre-processing function from earlier to automatically clean the text before it’s vectorized.

4.3.3 Classification

Natural Language Processing (NLP) is a vast area of Computer Science that is concerned with the interaction between Computers and Human Language. Within NLP many tasks are – or can be reformulated as – classification tasks. In classification tasks we are trying to produce a classification function which can give the correlation between a certain ‘feature’ and a class .This Classifier first has to be trained with a training dataset, and then it can be used to actually classify documents. Training means that we have to determine its model parameters. If the set of training examples is chosen correctly, the Classifier should predict the class probabilities of the actual documents with a similar accuracy (as the training examples). After construction, such a Classifier could for example tell us that document containing the words “Bose-Einstein condensate” should be categorized as a Physics article, while documents containing the words “Arbitrage” and “Hedging” should be categorized as a Finance article.

Another Classifier could tell us that mails starting with “Dear Customer/Guest/Sir” (instead of your name) and containing words like “Great opportunity” or “one-time offer” can be classified as spam.

Here we can already see two uses of classification models: topic classification and spam filtering. For these purposes a Classifiers work quiet well and perform better than most trained professionals.

A third usage of Classifiers is Sentiment Analysis. Here the purpose is to determine the subjective value of a text-document, i.e. how positive or negative is the content of a text document. Unfortunately, for this purpose these Classifiers fail to achieve the same accuracy.

This is due to the subtleties of human language; sarcasm, irony, context interpretation, use of slang, cultural differences and the different ways in which opinion can be expressed (subjective vs comparative, explicit vs implicit).

4.4 Dataset Information

The training dataset folder contains two .csv training data files. The data-1_train.csv contains header+2313 lines (training examples) and data-2_train.csv contains header+3602 lines. You can open the .csv files in excel to view them. However, sometimes excel may drop few lines from showing (due to the presence of special char symbols in review text) based on its settings. To view full content, you may use any text editor (like Sublime Text, Notepad++ etc.).

The description of each column in each .csv file is as follows:
Column A: review sentence id (unique id for a training instance).
Column B: review sentence.
Column C: aspect term in the sentence.
Column D: aspect term location [format: start index -- End index].
Column E: sentiment label.
Note:- "," in review sentence (column B) is denoted as "[comma]" to separate it from the column delimiter (",") of .csv file. While parsing (data file reading), you can use "," to split the line into fields (coloums) values.

4.5 Some external library files

Numpy

Numpy is a general-purpose array-processing package. It provides a high-performance multidimensional array object, and tools for working with these arrays. It is the fundamental package for scientific computing with Python. It contains various features including these important ones:

- A powerful N-dimensional array object
- Sophisticated (broadcasting) functions
- Tools for integrating C/C++ and Fortran code
- Useful linear algebra, Fourier transform, and random number capabilities

Besides its obvious scientific uses, NumPy can also be used as an efficient multi-dimensional container of generic data. Arbitrary data-types can be defined using Numpy which allows NumPy to seamlessly and speedily integrate with a wide variety of databases.

MatplotLib

Matplotlib is a Python 2D plotting library which produces publication quality figures in a variety of hardcopy formats and interactive environments across platforms. Matplotlib can be used in Python scripts, the Python and Ipythonshells, the Jupyternotebook, web application servers, and four graphical user interface toolkits.

Matplotlib tries to make easy things easy and hard things possible. You can generate plots, histograms, power spectra, bar charts, errorcharts, scatterplots, etc., with just a few lines of code. For examples, see the sample plots and thumbnail gallery.

For simple plotting the pyplot module provides a MATLAB-like interface, particularly when combined with IPython. For the power user, you have full control of line styles, font properties, axes properties, etc, via an object oriented interface or via a set of functions familiar to MATLAB users.

Pandas

Pandas is an open source, BSD-licensed library providing high-performance, easy-to-use data structures and data analysis tools for the Python programming language.

NLTK

NLTK is a leading platform for building Python programs to work with human language data. It provides easy-to-use interfaces to over 50 corpora and lexical resourcessuch as WordNet, along with a suite of text processing libraries for classification, tokenization, stemming, tagging, parsing, and semantic reasoning, wrappers for industrial-strength NLP libraries, and an active discussion forum.

NLTK is available for Windows, Mac OS X, and Linux. Best of all, NLTK is a free, open source, community-driven project.NLTK has been called “a wonderful tool for teaching, and working in, computational linguistics using Python,” and “an amazing library to play with natural language.”

textblob

TextBlobis a Python (2 and 3) library for processing textual data. It provides a simple API for diving into common natural language processing (NLP) tasks such as part-of-speech tagging, noun phrase extraction, sentiment analysis, classification, translation, and more.

SKLearn

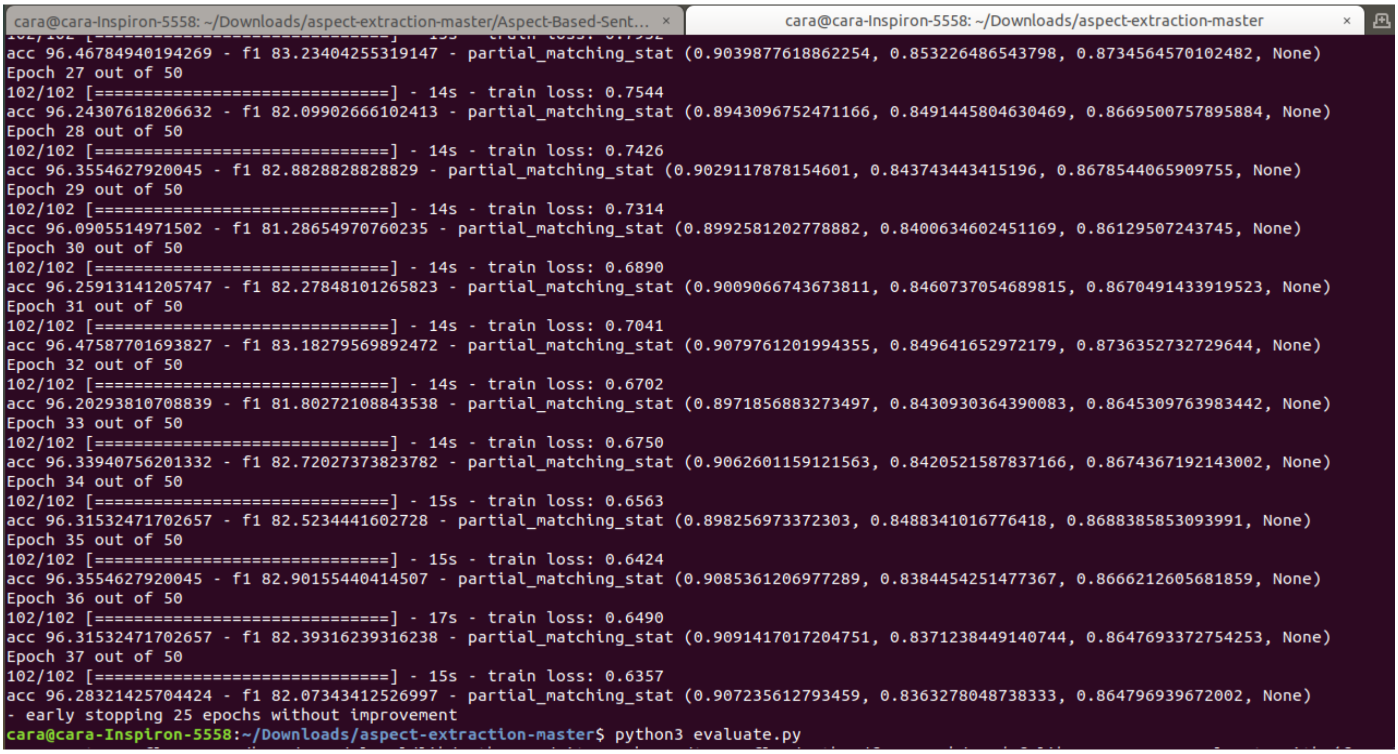
Scikit-learn is probably the most useful library for machine learning in Python. It is on NumPy, SciPy and matplotlib, this library contains a lot of effiecient tools for machine learning and statistical modeling including classification, regression, clustering and dimensionality reduction.

Chapter 5

Results and Discussion

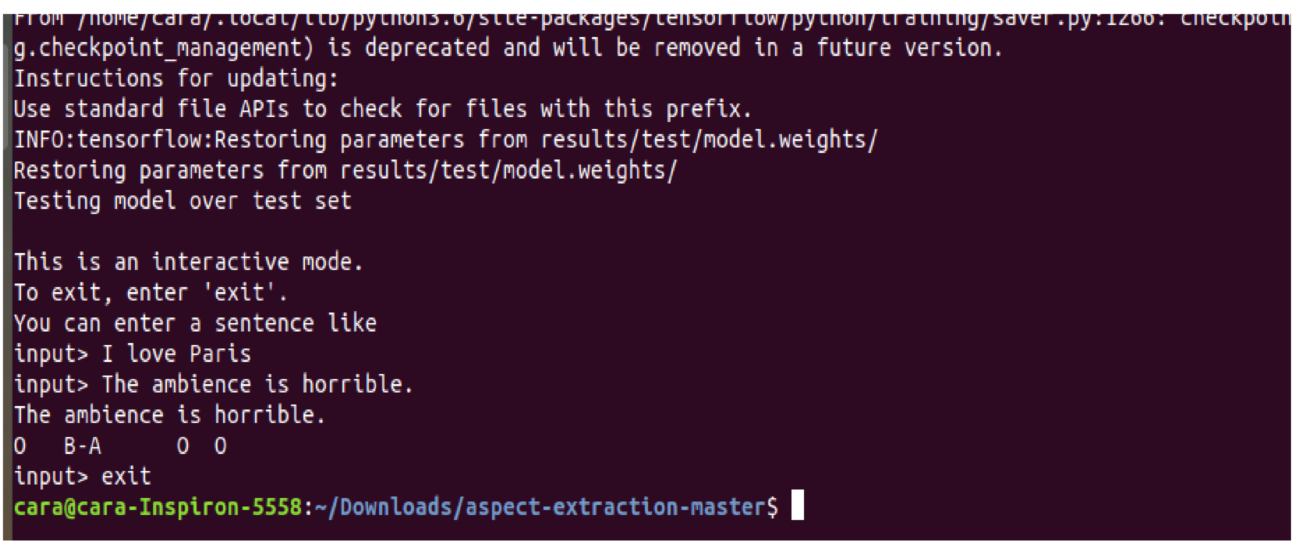
5.1 Aspect Extraction

Training and evaluation :



```
cara@cara-Inspiron-5558: ~/Downloads/aspect-extraction-master/Aspect-Based-Sent...    cara@cara-Inspiron-5558: ~/Downloads/aspect-extraction-master
acc 96.46784940194269 - f1 83.23404255319147 - partial_matching_stat (0.9039877618862254, 0.853226486543798, 0.8734564570102482, None)
Epoch 27 out of 50
102/102 [==============================] - 14s - train loss: 0.7544
acc 96.24307618206632 - f1 82.09902666102413 - partial_matching_stat (0.8943096752471166, 0.8491445804630469, 0.8669500757895884, None)
Epoch 28 out of 50
102/102 [==============================] - 14s - train loss: 0.7426
acc 96.3554627920045 - f1 82.8828828828829 - partial_matching_stat (0.9029117878154601, 0.843743443415196, 0.8678544065909755, None)
Epoch 29 out of 50
102/102 [==============================] - 14s - train loss: 0.7314
acc 96.0905514971502 - f1 81.28654970760235 - partial_matching_stat (0.8992581202778882, 0.8400634602451169, 0.86129507243745, None)
Epoch 30 out of 50
102/102 [==============================] - 14s - train loss: 0.6890
acc 96.25913141205747 - f1 82.27848101265823 - partial_matching_stat (0.9009066743673811, 0.8460737054689815, 0.8670491433919523, None)
Epoch 31 out of 50
102/102 [==============================] - 14s - train loss: 0.7041
acc 96.47587701693827 - f1 83.18279569892472 - partial_matching_stat (0.9079761201994355, 0.849641652972179, 0.8736352732729644, None)
Epoch 32 out of 50
102/102 [==============================] - 14s - train loss: 0.6702
acc 96.20293810708839 - f1 81.80272108843538 - partial_matching_stat (0.8971856883273497, 0.8430930364390083, 0.8645309763983442, None)
Epoch 33 out of 50
102/102 [==============================] - 14s - train loss: 0.6750
acc 96.33940756201332 - f1 82.72027373823782 - partial_matching_stat (0.9062601159121563, 0.8420521587837166, 0.8674367192143002, None)
Epoch 34 out of 50
102/102 [==============================] - 15s - train loss: 0.6563
acc 96.31532471702657 - f1 82.5234441602728 - partial_matching_stat (0.898256973372303, 0.8488341016776418, 0.8688385853093991, None)
Epoch 35 out of 50
102/102 [==============================] - 15s - train loss: 0.6424
acc 96.3554627920045 - f1 82.90155440414507 - partial_matching_stat (0.9085361206977289, 0.8384454251477367, 0.8666212605681859, None)
Epoch 36 out of 50
102/102 [==============================] - 17s - train loss: 0.6490
acc 96.31532471702657 - f1 82.39316239316238 - partial_matching_stat (0.9091417017204751, 0.8371238449140744, 0.8647693372754253, None)
Epoch 37 out of 50
102/102 [==============================] - 15s - train loss: 0.6357
acc 96.28321425704424 - f1 82.07343412526997 - partial_matching_stat (0.907235612793459, 0.8363278048738333, 0.864796939672002, None)
- early stopping 25 epochs without improvement
cara@cara-Inspiron-5558:~/Downloads/aspect-extraction-master$ python3 evaluate.py
```

Fig 5.1 Training of CNN for Aspect Extraction



```
g.checkpoint_management) is deprecated and will be removed in a future version.
Instructions for updating:
Use standard file APIs to check for files with this prefix.
INFO:tensorflow:Restoring parameters from results/test/model.weights/
Restoring parameters from results/test/model.weights/
Testing model over test set

This is an interactive mode.
To exit, enter 'exit'.
You can enter a sentence like
input> I love Paris
input> The ambience is horrible.
The ambience is horrible.
O   B-A      O  O
input> exit
cara@cara-Inspiron-5558:~/Downloads/aspect-extraction-master$
```

Figure 5.2 Interactive mode for aspect extraction

From the above results shown in Fig 5.1 we can infer that the model is trained for 37 epochs and the accuracy of the model is 96.283 %.

The model is trained with restaurant reviews, so when ever the user gives an input with respect to the aspects of the restaurant, the input is fed into the trained model and output is the extracted aspect.

5.2 Sentiment Polarity

```
1095 11651065#413029#1_2,The comments about fried foods is correct (below) but the other dishes[comma] including the lamb entree and many of the salads (avocado shrimp) were quite good.,lamb entree,86--
1096 11651310#581605#1_1,Our waiter was helpful and charming[comma] the food was perfect[comma] and the wine was good[comma] too.,food,41--45
1097 33071731#1007204#4_1,The unfortunate lady next to us thought she had ordered a salad (including asking for salad dressing) and was instead given a quesedilla.,salad dressing,86--100
1098 32857960#1320058#5_0,Not only was the waiter efficient and courteous[comma] but also extremely helpful.,waiter,17--23
1099 11313439#692431#0_2,Anytime and everytime I find myself in the neighborhood I will go to Sushi Rose for fresh sushi and great portions all at a reasonable price.,price,135--140
1100 32950007#1320096#2_0,Although small[comma] it has beautiful ambience[comma] excellent food (the catfish is delicious - if ya don't mind it a lil salty) and attentive service.,ambience,33--41
1101 15069510#648179#3_2,The food is average: breakfast food[comma] soups[comma] salads[comma] sandwiches[comma] etc.,soups,37--42
1102 15094232#1295193#0_0,We had a birthday party here recently and the food and service was amazing.,food,46--50
1103 35177381#521555#1_1,Our waitress had apparently never tried any of the food[comma] and there was no one to recommend any wine.,food,51--55
1104 11359790#823900#3_1,The manager then told us we could order from whatever menu we wanted but by that time we were so annoyed with the waiter and the resturant that we let and went some place else.
1105 32894966#1727613#3_0,While the smoothies are a little big for me[comma] the fresh juices are the best I have ever had!,smoothies,10--19
1106 32892130#435512#4_0,I definitely enjoyed the food as well.,food,25--29
1107 32894416#1193614#4_0,Another friend had to ask 3 times for parmesan cheese.,parmesan cheese,38--53
1108 11359767#969393#2_0,The food was great - sushi was good[comma] but the cooked food amazed us.,food,4--8
1109 35170181#0#3_1,"Dishes denoted as ""Roy's Classics"" (marked on the menu with asterisks) are tried-and-true recipes[comma] such as macadamia-crusted mahi mahi[comma] or subtly sweet honey-musta
1110 33069536#707145#3_1,My husband and I have been there at least 6 times and we've always been given the highest service and often free desserts.,desserts,113--121
1111 35084984#412657#0_0,I went to Common Stock for brunch and I was so impressed.,brunch,27--33
1112 11661949#1709112#3_0,We didn't know if we should order a drink or leave?,drink,36--41
1113 35177381#521555#0_2,I came to fresh expecting a great meal[comma] and all I got was marginally so-so food served in a restaurant that was just so freezing we couldn't enjoy eating.,served,80--86
1114 35938288#666236#2_0,Most importantly[comma] it is reasonably priced.,priced,35--41
1115 35668286#0#2_2, The menu includes pub fare--burgers[comma] steaks and shepherds pie--and even a portabella lasagna for those black sheep known as vegetarians.,steaks,39--45
1116 32936760#1397861#6_0,Result (red velvet): Great texture[comma] soft and velvety[comma] nice hint of cocoa.,texture,27--34
1117 35816071#637243#1_0,Chef Vincenzo[comma] always there if you need him[comma] is a real talent and a real Roman.,Chef,0--4
1118 35700989#574718#1_0,The waitstaff is solicitous and friendly and always seems glad to see us[comma] and the food is wonderful[comma] if not stunningly creative.,waitstaff,4--13
1119 11351762#644011#1_0,While it was large and a bit noisy[comma] the drinks were fantastic[comma] and the food was superb.,drinks,40--46
1120 33059678#631861#0_0,I went to Swiftys with some friends of the family and we had a very nice dinner[comma] but nothing amazing.,dinner,73--79
1121 11313392#560011#1_1,Pizzas were excellent in addition to appetizers and main courses.,appetizers,37--47
1122 7777777#77777#1_1,The ambience is horrible.,ambience,5–12
```

Figure 5.3 CSV file for the aspects extracted to find the sentiment polarity

```
11351628#404492#5_0;;1.0
11359797#469087#7_1;;-1.0
11359727#487554#5_3;;1.0
35171678#484493#3_0;;1.0
35938288#666236#1_1;;1.0
32894513#498296#0_0;;-1.0
32899282#556036#1_1;;1.0
11651065#413029#1_2;;1.0
11651310#581605#1_1;;1.0
33071731#1007204#4_1;;-1.0
32857960#1320058#5_0;;1.0
11313439#692431#0_2;;1.0
32950007#1320096#2_0;;1.0
15069510#648179#3_2;;-1.0
15094232#1295193#0_0;;1.0
35177381#521555#1_1;;1.0
11359790#823900#3_1;;-1.0
32894966#1727613#3_0;;1.0
32892130#435512#4_0;;1.0
32894416#1193614#4_0;;0.0
11359767#969393#2_0;;1.0
"35170181#0#3_1;;1.0
33069536#707145#3_1;;1.0
35084984#412657#0_0;;1.0
11661949#1709112#3_0;;0.0
35177381#521555#0_2;;1.0
35938288#666236#2_0;;1.0
35668286#0#2_2;;-1.0
32936760#1397861#6_0;;1.0
35816071#637243#1_0;;1.0
35700989#574718#1_0;;1.0
11351762#644011#1_0;;1.0
33059678#631861#0_0;;1.0
11313392#560011#1_1;;1.0
7777777#77777#1_1;;-1.0
cara@cara-Inspiron-5558:~/Downloads/Aspect-Based-Sentiment-Analysis-master$
```

Figure 5.4 Sentiment polarity

In figure 5.4, the first column is review_id and the last column indicates polarity where if polarity=1, it is positive and polarity= -1, it is a negative sentiment.

ACCURACY:

SVM: 0.741
Gaussian Naive Bayes: 0.668
Multi Layer Perceptron: 0.736

5.3 Summarization

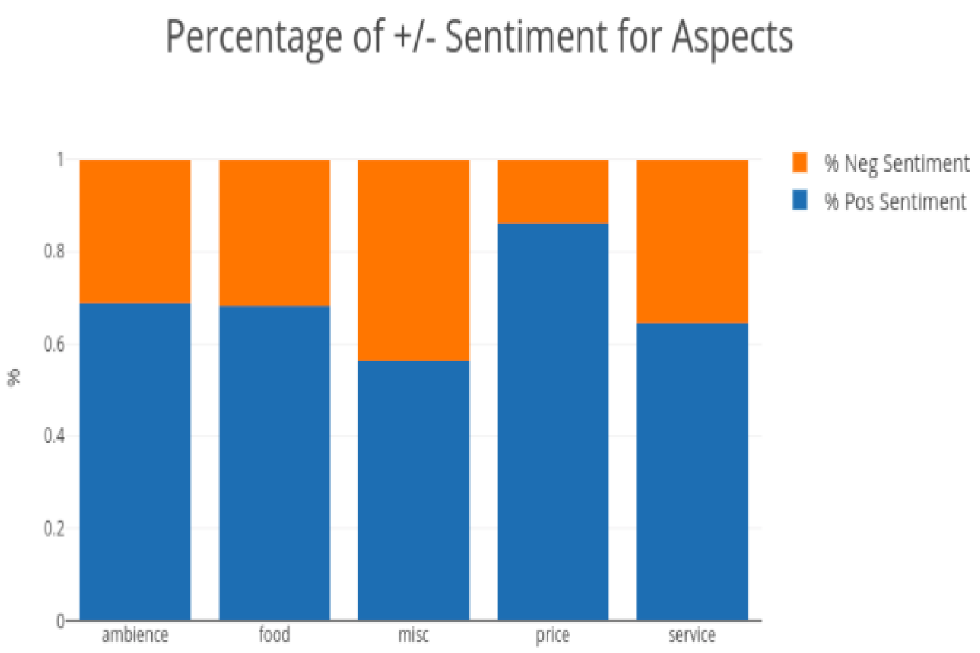


Fig 5.5 Percentage of postive and negative sentiments for Aspects

Figure 5.5 shows the percentage of postive and negative reviews under the aspects

- Ambience
- Food
- Miscellaneous
- Price
- Service

CHAPTER 6

CONCLUSION AND FUTURE WORK

Future work

The empirical research conducted throughout this thesis has laid the foundation for aspect and opinion extraction from customer reviews using data mining and natural language processing techniques. This research can be extended to overcome the limitations outlined above. The developed algorithms can be further refined and optimised in order to achieve a better performance. These motivate future work in the following areas.

i) Categorizing the aspect terms under a general and common aspects.
ii) Performing deep Learning techniques for sentiment Polarity analysis.

Conclusion

This thesis has explored an efficient approach for aspect extraction and predicted the polarity of opinions including machine learning, deep learning and natural language processing techniques. We have experimented the research with customer reviews about restaurants and laptop where both the implicit and explicit aspects are extracted and the polarity of every review in terms of every aspect is predicted.

REFERENCES

[1] M. Hu and B. Liu, “Mining Opinion Features in Customer Reviews,” in Proceedings of the 19th National Conference on Artifical Intelligence (AAAI 2004). AAAI, 2004, pp. 755–760. “Mining and Summarizing Customer Reviews,” in Proceedings of 10th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD 2004). ACM, 2004, pp. 168–177.

[2] C. Scaffidi, K. Bierhoff, E. Chang, M. Felker, H. Ng, and C. Jin, “Red Opal: Product-Feature Scoring from Reviews,” in Proceedings of the 8th ACM Conference on Electronic Commerce (EC 2007). ACM, 2007, pp. 182–191.

[3] Z. Li, M. Zhang, S. Ma, B. Zhou, and Y. Sun, “Automatic Extraction for Product Feature Words from Comments on the Web,” in Proceedings of the 5th Asia Information Retrieval Symposium on Information Retrieval Technology (AIRS 2009). Springer, 2009, pp. 112–123.

[4] M. Pontiki, D. Galanis, J. Pavlopoulos, H. Papageorgiou, I. Androutsopoulos, and S. Manandhar, “Semeval2014 task 4: Aspect based sentiment analysis,” in Proceedings of the 8thInternational Workshop on Semantic Evaluation (SemEval 2014). Association for Computational Linguistics and Dublin City University, 2014, pp. 27–35.

[5] Y. Zhao, B. Qin, S. Hu, and T. Liu, “Generalizing Syntactic Structures for Product Attribute Candidate Extraction,” in Proceedings of the Conference of the North American Chapter of the Association of Computational Linguistics: Human Language Technologies 2010 (HLT-NAACL 2010). ACL, 2010, pp. 377– 380.

[6] G. Qiu, B. Liu, J. Bu, and C. Chen, “Expanding Domain Sentiment Lexicon Through Double Propagation,” in Proceedings of the 21st International Jont Conference on Artifical Intelligence (IJCAI 2009). Morgan Kaufmann Publishers Inc., 2009, pp.1199–1204.

[7] L. Zhang, B. Liu, S. H. Lim, and E. O‟Brien-Strain, “Extracting and Ranking Product Features in Opinion Documents,” in Proceedings of the 23rd International Conference on Computational Linguistics (COLING 2010). ACL, 2010, pp. 1462–1470.

[8] 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.

[9] N. Jakob and I. Gurevych, “Extracting Opinion Targets in a Single- and Cross-Domain Setting with Conditional Random Fields,” in Proceedings of the 2010 Conference on

Empirical Methods in Natural Language Processing (EMNLP 2010). ACL, 2010, pp. 1035–1045.

[10] L. Zhuang, F. Jing, and X.-Y. Zhu, “Movie Review Mining and Summarization,” in Proceedings of the 15th ACM International Conference on Information and Knowledge Management (CIKM 2006), vol. 6, no. 11. ACM, 2006, pp. 43–50.

[11] C. Toprak, N. Jakob, and I. Gurevych, “Sentence and ExpressionLevel Annotation of Opinions in UserGenerated Discourse,” in Proceedings of the 48th Annual Meeting of the

[12] Association for Computational Linguistics (ACL 2010). ACL, 2010, pp. 575–584.

[13] J. S. Kessler and N. Nicolov, “Targeting Sentiment Expressions through Supervised Ranking of Linguistic Configurations,” in Proceedings of the 3rd International AAAI Conference on Weblogs and Social Media (ICWSM 2009). AAAI, 2009, pp. 90–97.

[14] D. M. Blei, A. Y. Ng, and M. I. Jordan, “Latent dirichlet allocation,” Journal of Machine Learning Research, vol. 3, pp.993–1022, 2003.

[15] T. Hofmann, “Learning the Similarity of Documents: An Information-Geometric Approach to Document Retrieval and Categorization,” in Advances in Neural Information Processing Systems (NIPS 2000. MIT Press, 2000, pp. 914–920.

[16] I. Titov and R. McDonald, “Modeling Online Reviews with Multi-Grain Topic Models,” in Proceedings of the 17th International Conference on World Wide Web (WWW 2009). ACM, 2008,pp. 111–120.

[17] Q. Mei, X. Ling, M. Wondra, H. Su, and C. Zhai, “Topic Sentiment Mixture: Modeling Facets and Opinions in Weblogs,” in Proceedings of the 16th International Conference on World WideWeb (WWW 2007). ACM, 2007, pp. 171–180.

[18] B. Lu, M. Ott, C. Cardie, and B. K. Tsou, “Multi-Aspect Sentiment Analysis with Topic Models,” in Proceedings of the 2011 IEEE 11th International Conference on Data Mining Workshops (ICDMW 2011). IEEE, 2011, pp. 81–88.

[19] H. Lakkaraju, C. Bhattacharyya, I. Bhattacharya, and S. Merugu, “Exploiting Coherence for the Simultaneous Discovery of Latent Facets and Associated Sentiments,” in SIAM International Conference on Data Mining 2011 (SDM 2011).SIAM, 2011, pp. 498–509.

[20] T.-J. Zhan and C.-H. Li, “Semantic Dependent Word Pairs Generative Model for Fine-Grained Product Feature Mining,”in Proceedings of the 15th Pacific-Asia Conference on Advances in Knowledge Discovery and Data Mining (PAKDD 2011). Springer,2011, pp.

460–475.[21] S. Moghaddam and M. Ester, “The FLDA Model for Aspect based Opinion Mining: Addressing the Cold Start Problem,”in Proceedings of the 22nd International Conference on World Wide Web (WWW 2013). ACM, 2013, pp. 909–918.

[22] Z. Hai, G. Cong, K. Chang, W. Liu, and P. Cheng, “Coarseto-fine Review Selection via Supervised Joint Aspect and Sentiment Model,” in Proceedings of the 37th International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR 2014). ACM, 2014, pp. 617–626.

[23] A.-M. Popescu and O. Etzioni, “Extracting Product Features and Opinions from Reviews,” in Proceedings of the Conference on Human Language Technology and Conference on Empirical Methods in Natural Language Processing 2005 (HLT/EMNLP 2005). ACL,2005, pp. 339–346.

[24] K. W. Church and P. Hanks, “Word Association Norms, Mutual Information, and Lexicography,” Computational Linguistics,vol. 16, no. 1, pp. 22–29, 1990.

[25] S. Raju, P. Pingali, and V. Varma, “An Unsupervised Approach to Product Attribute Extraction,” in Proceedings of the 31th European Conference on IR Research on Advances in Information Retrieval (ECIR 2009). Springer, 2009, pp. 796–800.

[26] L. R. Dice, “Measures of the Amount of Ecologic Association Between Species,” Ecology, vol. 26, no. 3, pp. 297–302, 1945.

[27] J. Yu, Z.-J. Zha, M. Wang, and T.-S. Chua, “Aspect Ranking: Identifying Important Product Aspects from Online Consumer Reviews,” in Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies (ACL 2011). ACL, 2011, pp. 1496–1505.

[28] S. Blair-Goldensohn, T. Neylon, K. Hannan, G. A. Reis, R. Mcdonald, and J. Reynar, “Building a Sentiment Summarizer for Local Service Reviews,” in Proceedings of WWW 2008 Workshop on NLP in the Information Explosion Era (NLPIX 2008). ACM,2008.

[29] Toh, Zhiqiang, and Jian Su. "NLANGP: Supervised Machine Learning System for Aspect Category Classification and Opinion Target Extraction."

[30] Schouten, Kim, Flavius Frasincar, and Franciska de Jong. "Commit-p1wp3: A co-occurrence based approach to aspect-level sentiment analysis." Proceedings of the 8th International Workshop on Semantic Evaluation (SemEval 2014). 2014.

[31] Zhiqiang, Toh, and Wang Wenting. "DLIREC: Aspect Term Extraction and Term Polarity Classification System." (2014).