

ESTIMATING THE OVERALL SENTIMENT SCORE BY INFERRING MODUS PONENS LAW

R.Nithya¹ and Dr.D.Maheswari²

Abstract- Growing business intelligence drives new era of the mining sentiment behind the users' comments posted on the web. Many researchers focus on the supervised and unsupervised technique, but still it is challenging to choose the prompt one. The proposed model discusses about the application of rules of inference for developing sentiment analysis based application. Till now there is no research work made on identifying the sentiment and to calculate the sentiment score by inferring the Modus Ponens Law from Discrete Mathematics. With this hope the analysis is made and its evaluation metrics shows better result.

Keywords- Discrete Mathematics, Modus Ponens Law, Natural Language Processing, Sentiment Analysis, Pattern Mining

I. INTRODUCTION

Sentiment Analysis is playing a vital role in building new opportunities in world markets. It is one of the major sub-discipline of Natural Language Processing, that leads to build a model or system that can mine and classify the text into positive or negative. It is also possible to identify the apt features and aspects of the products available for sale online. More number of web consumers have a tendency to post their reviews on the 4P's namely People, Product, Policy or Process. These kinds of analysis are also gaining more potential to plan for innovative business strategies. In a more generic context, opinions are gathered in two ways. In maximum questionnaires are used to fetch the opinion about the quality facts about the product. While the deep structured analysis needs to follow scraping, organizing and analyzing the unstructured text posted in reputed social media.

Different levels of analysis that can be performed are named document, sentence and entity level. Although they are of three types, the second and third levels are very specific towards mining sentiments. They also have tended to evolve with different dimensional subtask like detecting subjectivity, predicting sentiments, entity ranking, and opinion retrieval, extracting features, summarizing the sentiment at aspect level and summarizing the contrasting viewpoint. There are various types of techniques available for performing sentiment analysis, but each one has its own merits and demerits at different aspects. The lexical approach needs a dictionary to undergo analysis, whereas the learning method needs a training model for classifying the document. Furthermore, to analyze sentiments in text data, it is necessary to find some set of rules for categorizing phrases of text into either positive, negative. The sentiment analysis models dictate on the statement that maps to positivity or negativity in the text. Most of the research papers deal with machine learning approaches, as they deduct this logic from learning examples. And most of the machine learning model seldom reframed into algorithms like Naïve Bayes, Support Vector Machine and Decision tree classifier that produces predicted output.

Thus, it is clear that when a model is applied to the dataset, it results with most promising outputs. To be specific, these sets of rules contain logic to classify the phrases. Discrete mathematics is underlying as the basis for formal approaches. It deals with mathematical reasoning on the given issue, examine their nature and analyze the relationship exists between them. The semantic of propositional logic define by specifying the interpretation of propositional symbols and constants and specifying the meaning of the logical connectives. So, this proposed paper applies such mathematical model to identify the sentence pattern for undergoing the sentiment analysis followed by feature identification.

¹ Assistant Professor & Ph.D Scholar, School of Computer Studies(UG), RVS College of Arts and Science, Sulur, Coimbatore, India.

² Assistant Professor, Head of the Department, School of Computer Studies(PG), RVS College of Arts and Science, Sulur, Coimbatore, India.

II. RELATED WORKS

Sentiment analysis for the emotional preference of online comments has gained great achievement since it was raised up by Pang et al. and studied in-depth [3]. Hu and Liu et al attempt to find features by performing POS tagging and generate n-grams, for sentiment prediction [4]. To the contrast of Alekh Agarwal's research, [8] Lei Zhang et al identified noun and noun phrases from objective sentences and determined their polarity which is considered as challenging research phase. Popescu and Etzioni et al investigated on problem of extracting the product feature [5]. The Base Noun Phrase (BNP) and its combinations are proposed as a hybrid dependency pattern by Khairullah khan et al [1,2]. Anuj Sharma et al tried to use five feature selection methods on movie dataset [6]. Farah Benamara et al proposes [12] an AAC-based sentiment analysis technique that uses a linguistic analysis of adverbs of degree. The performance of the rule based mining algorithm to that of identifying feature using log likelihood test on five product specific document collections regarding consumer electronic devices by [7] Liliana Ferreira et al.

III. PROBLEM STATEMENT

Unfortunately, identifying the right product [4] with prominent features is complicated with the available functionality. The opinion words are very important features both in features extraction and categorization. Our problem is to extract he evaluated expression [5] from a sequence of words from the given sentences.

Problem Statement:

- 1) Identifying the substantial portion of sentiment component is one of the challenging phases in Sentiment Analysis. It is also very critical to restrict [1,2] or confine with the particular pattern for identifying features.
- 2) Most often, only noun and adjectival phrase [3,4,8] were concentrated to extract feature and its opinion, whereas very few on verb and adverb phrase, and when feature based sentiment analysis is undergone, most of the authors considered Term Frequency[6] as one of the major criteria for identifying feature set. Most of the paper focus on presence or absence of feature word in a given dataset.

IV. MODEL DESCRIPTION AND DATASET USED

The proposed model is portrayed in Fig 1. using Open source diagrammatic tool named as 'Dia'.

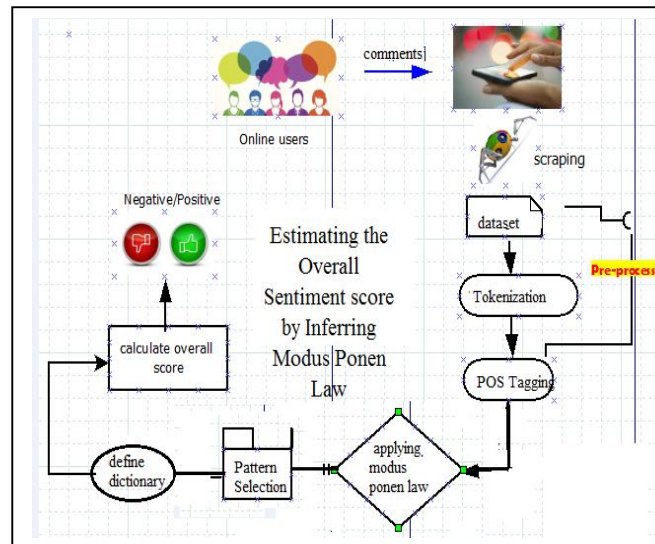


Fig 1. Model Description Using Dia Tool

A. Steps in Pre-processing:

The Benchmark data sets about five different products that are collected [5] from Amazon product review site by Hu and Liu for product feature extraction and opinion summarization. The same data set has been re-annotated by [13] due their focus study on feature extraction only. The difference between these two annotations is that the Hu and Liu

consider only those features about which opinion is expressed while the later one considers all the features related to the product.

- <http://www.cs.uic.edu/~liub> - contains annotated customer reviews of 5 products from amazon.com:-
- digital camera: Canon G3,
- digital camera: Nikon coolpix 4300,
- cellular phone: Nokia 6610,
- mp3 player: Creative Labs Nomad Jukebox Zen Xtra 40GB,
- dvd player: Apex AD2600 Progressive-scan DVD player
- Annotated by: Minqing Hu and Bing Liu, Department of Computer Science, University of Illinois at Chicago,2004.

TABLE 1. Summary of five products data sets with manually tagged features

Datasets	No of Sentences	Manually Tag Features by [5]		Manually Tag Features by [13]	
		Distinct	Total	Distinct	Total
APEX	739	110	347	166	519
Canon	597	100	257	161	594
Creative	1716	180	736	231	1031
Nikon	346	74	185	120	340
Nokia	546	109	310	140	470

The sentences or comments from the above dataset are already stored in a .txt file for preprocessing. All the pre-processing activities like stemming, parsing and tokenizing are carried out using nltk - an efficient tool for working with computational linguistics in Python. As they are mostly unstructured, re module is used to remove the html tags. Spell check and tokenization process were carried out, next to stopwords removal. Next the grammatical tagging of sentence (POS) is performed based on the word and its corresponding position. As they are mostly unstructured, re module is used to remove the html tags. Spell check and tokenization process were carried out, next to stopwords removal. Next the grammatical tagging of sentence (POS) is performed based on the word and its corresponding position.

IV. PROPOSED METHODOLOGY

A. Building Lexicon Dictionary:

Fig.2. composed of 2006 positive and 4783 negative words was built to undergo synchronization with the chunked sentences. The LBSA approach follows only the lexical method of calculating the score by just checking the presence of positive and negative word in the dictionary.

```
>>> dic2={'a+':'pos',
'abound':'pos',
'abounds':'pos',
'abundance':'pos',
'abundant':'pos',
'accessible':'pos',
'accessible':'pos',
'acclaim':'pos',
'acclaimed':'pos',
'acclamation':'pos',
'accolade':'pos',
'accolades':'pos',
'accommodative':'pos',
```

Fig 2. Lexicon dictionary built using Python 2.7

B. Defining Algorithm to find overall Sentiment score using LPSA:

The Lexical Pattern Sentiment Analysis (LPSA) algorithm undergoes combinatorial technique of the Lexicon and Pattern mining approach in undergoing Sentiment Analysis. First, Use the sentence pattern [For example, look at the commented sentence 'excellent camera'- it follows the Base Noun Phrase Pattern - JJNN] gets selected and then the algorithm adds +1 as the word 'excellent' is in positive list (*Use of Lexicon*). Likewise, the algorithm calculates the score overall for the sentences that fix with a particular pattern and display the result in '- value', if the product is overall negative, and vice versa. So, with this constraint the interpretation of final score must take place.

```

>>> file=open('D:\works\motogclean.txt','r')
>>> score=0
>>> for line in file.readlines():
    if re.search('\S', line):
        s=line.split();
        tokenized=nltk.word_tokenize(line)
        for words in tokenized:
            if words in dic2:
                d=dic2.get(words)
                p=dic2.has_key(words)
                print p,words,d
            if dic2.get(words)=='pos':
                score=score+1
            else:
                score=score-1
            print score

        file.close()

True best pos
1
True superior pos
2
    
```

```

Python Shell
File Edit Shell Debug Options Windows Help
True great pos
145
True excellent pos
146
True great pos
147
True awesome pos
148
True excellent pos
149
True impressed pos
150
True best pos
151
True smart pos
152
True great pos
153
True perfect pos
154
True great pos
155
    
```

Fig 3. Snapshot of running LPSA algorithm

Fig 4. Score generation on execution of algorithm

In Fig 3, the dictionary words are synchronized towards chunked pattern. By the default score is set to zero. Now, the chunked data get tokenized and mapped with the proposed lexicon dictionary. If the word/key matches with a positive value / pair, the score increases by 1 or else vice versa as shown in Fig 4. Modifiers set with a value as '-1'.

TABLE 2.Comparison of LBSA (Lexicon Based SA) to LPSA (Lexicon + BNP Pattern based SA) Approach

Dataset	APEX		Canon		Creative		Nikon		Nokia	
	LBSA	LPSA	LBSA	LPSA	LBSA	LPSA	LBSA	LPSA	LBSA	LPSA
No of sentence	739	254	597	313	1716	1537	346	313	546	327
No of Positive prediction	522	185	327	110	980	772	327	165	334	198
No of Negative prediction	106	16	112	98	532	320	112	18	178	95
Number of prediction	628	201	439	208	1512	1092	439	183	512	293
Overall score	416	165	256	91	980	539	256	113	189	78

To conclude with, the syntactic pattern based (LPSA) approach has given a better result compared to lexical analysis (LBSA) as shown in Table 2. The first problem stated above is resolved with the fact that on combining lexicon with the pattern based method, it is possible to restrict or confine with the particular pattern for identifying features.

C. Pattern Based Feature Identification

This approach do not depends on frequency of occurrence but depends on sequence or patterns of terms. The sequence depends on certain rules or regular expressions.

- **Base Noun Phases(BNP):** NN, NNNN, JJNN, NNNN NN, JJNNNN, JJJJNN. As it commonly observed that all noun phrases are not product features therefore it is required to restrict the patterns for feature extraction. [6,14]
- **Definite Base Noun Phrase (dBNP) :**This pattern restricts candidate feature terms to definite base noun phrases, which are noun phrases (BNP) preceded by the definite **article** 'the'" [30,13,1,2].

- **Beginning Definite Base Noun Phrase (bBNP)** :This pattern presents as sequence of **noun phrase followed by verb**” [30,13,1,2].
- **Combined Pattern Based Noun Phrases (cBNP)**: [1,2]
 - a) Linking verb based noun phrases (**vBNP**) BNP followed **Verb Followed by Adjective or adverb**. [1,2]
 - b) Definite based noun phrases (**dBNP**) Beginning definite base noun phrase **preceded by article “the”** . [1,2]
 - c) Preposition Based Noun Phrases (**iBNP**) The noun phrases **having prepositioned (“of/IN”)** [1,2]

We have proposed a new pattern by inferring Modus Ponens Law and combine it with the existing pattern to identify feature.

1) Identifying Sentiment through Mathematical Reasoning:

As all the commented sentences does not bear sentiment, it is necessary to identify the potential pattern of sentences that can reveal the real emotions of end-user. To substantiate on this, the model initiates with translation of the sentence into Boolean values as T or F.

Let B be the set { } containing the notion T, F denoting the truth and falsehood of semantic truth values.

$$B = \{T, F\}$$

(1)

2) Translating Sentences:

Translating commented sentences into expression needs propositional variables and logical connectives. Proposition can either be T or F, but they cannot be both. The alphabets used to represent the statements are called propositional variable.

For example, Let P and Q be the two propositional variables. As logic is the base for mathematical reasoning, the rules have to provide a detailed description of mathematical statements. The connectives are used for joining two logics.

\wedge, \vee, \square - Connectives (AND, OR, NOT)

$P \vee Q, P \wedge Q$ - Propositional Logic

The need for translating sentences into logic is to avoid more than one interpretation. This also involves in making set of reasonable notion based on the predefined meaning of the given sentences. After translating sentence from English into logics, it is necessary to examine the logical expressions and define their truth values. It is also possible to manipulate them and use rules of inference to reason about them.

3) Applying Rule of Inference:

A rule of inference is actually the syntactical transformation that when inferred, ends up with a valid conclusion. If the logical truth value is T for all the composition of propositions, then the logic is to be in tautology. Modus Ponens law or law of detachment is one among the rule of inference that is adopted here to select the interesting collection of patterns.

3.1. Modus Ponens Law towards Pattern Selection:

The Sentiment Analysis Model evolves with many syntactic pattern selection techniques.

Axiom: 1 The phrases or words that are succeeding or preceding the noun phrase (i.e., in the presence of feature word) always bear sentiment.

Example: If this is the commented sentence, then ‘beautiful display’ follows JJNN (BNP PATTERN)

To elucidate with this represent the propositions P and Q as,

Let Proposition P := { {PosJJ V \square PosJJ } }

(Interpretation: P be the presence of Positive or Non-Positive Adjective (PosJJ, \square PosJJ))

Proposition Q: = { NN V \square NN }

(Interpretation: Q be the presence or absence of Noun(NN, \square NN)).

With this simple logic, it is possible to generate four possible combinations of truth values.

TABLE 1: Logic values of Modus Ponens with JJ and NN combinations

P	Q	P->Q	Modus Ponens P^(P->Q)	Outcome (P->Q)^P->Q =: ⊨
T	T	T	T	T
Pos JJ	NN	Pos JJ follows NN	valid: bear sentiment	SELECTED pattern
T	F	F	F	T
JJ	□NN	JJ doesn't follow any NN	Invalid state: Absence of NN	REJECTED based on Axiom1
F	T	T	F	T
NonPos □JJ	NN	JJ follows NN	valid: bear sentiment	SELECTED pattern
F	F	T	F	T
□JJ	□NN	Any JJ doesn't follow any NN	Invalid state: Absence of NN	REJECTED based on Axiom1

Modus Ponens law states that if a conditional statement and the hypothesis of the conditional statement both are T, then the result must also be T.

$$(P \rightarrow Q) \wedge P \rightarrow Q =: \models \quad (2)$$

Interpretation: From Table 1. pos: JJNN and Non-Pos:JJNN were selected for further analysis.

When two propositions are taken, 2ⁿ equals to 4 combination of truth values gets generated. If four propositions are taken, then 2⁴=16 combination of truth values gets generated.

If four propositions are taken, then 2⁴=16 combination of truth values will be generated.

Let **Proposition A:** = { {PosJJ V □PosJJ } V {PosRB V □PosRB } V { {PosVB v □PosVB } }

(Interpretation: A be the presence of Positive or Non-Positive Adjective (PosJJ, □PosJJ) or presence of Positive or Non-Positive Adverb (PosRB V □PosRB) or presence of Positive or Non-Positive Verb (PosVB V □PosVB) Phrases)

Let **Proposition B:** = { NN } According to AXIOM-1

(Interpretation: B be the presence of Noun(NN)Phrase).

Let **Proposition C:** = { {PosRB V □PosRB } V {PosJJ V □PosJJ } V { {PosVB v □PosVB } }

(Interpretation: A be the presence of Positive or Non-Positive Adverb (PosRB, □PosRB) or presence of Positive or Non-Positive Adjective (PosJJ V □PosJJ) or presence of Positive or Non-Positive Verb (PosVB V □PosVB) Phrases)

Let **Proposition D:** = { {PosVB V □PosVB } V {PosJJ V □PosJJ } V { {PosRB v □PosRB } }

(Interpretation: A be the presence of Positive or Non-Positive Verb(PosVB, □PosVB) or presence of Positive or Non-Positive Adjective (PosJJ V □PosJJ) or presence of Positive or Non-Positive Adverb (PosRB V □PosRB)Phrases)

But out of 16 combinations only seven combinations are taken for interpretation based on Axiom:1.

TABLE 2: Logic values of Modus Ponens with JJ ,NN,RB and VB combinations

A	B	C	D	A->B	C->D	(A->B)->(C->D)	A^(A->B)->(C->D)	Outcome: Modus Ponens law A^ (A->B)->(C->D)- >B=: ⊢
T	T	T	T	T	T	T	T	T
T	T	T	F	T	F	F	F	T
T	T	F	T	T	T	T	T	T
F	T	T	T	T	T	T	F	T
F	T	F	T	T	T	T	F	T
F	T	T	F	T	F	F	F	T
T	T	F	F	T	F	F	F	T

Inference:

With reference to Table 2 by applying the Modus Ponens law or the law of detachment it is able to select only the following pattern to proceed with further analysis and they are JJNNRBVB, JJNNRB, JJNNVB, NNRBVB, NNVB, NNRB, JJNN, RBNNJJVB, RBNNJJ, RBNNVB, NNJJVB, NNVB, NNJJ, RBNN, VBNNRBJJ, VBNNRB, VBNNJJ, NNRBJJ, NNVB, NNRB, VBNN.

3.2. Normalization

Normalization is a systematic approach of eliminating data redundancy and it has various normal forms like 1-NF, 2-NF, BCNF etc. Based on NF-1 the redundant patterns are removed to avoid ambiguity. Here, NNVB and NNRB are repeated more than once. And hence it results with only eighteen patterns. They are JJNN, NNJJ, RBNN, NNRB, VBNN, NNVB, JJNNRB, JJNNVB, RBNNJJ, VBNNJJ, VBNNRB, RBNNVB, NNRBVB, NNJJVB, NNRBJJ, JJNNRBVB, RBNNJJVB, VBNNRBJJ.

Now, it is necessary to apply NF-2 as it deals with removing the partial dependency. The pattern JJNNRBVB is partially dependent on JJNN, and RBNNJJVB pattern is dependent on RBNNJJ. Thus the normalization ends with six independent patterns like JJNNVB, VBNNJJ, RBNNVB, JJNNRBVB, RBNNJJVB, VBNNRBJJ. - (vRBNP)

3.3. Relevancy Scoring

To eradicate irrelevant features the likelihood ratio test has been used in [7,11].

General formula for $G = 2 \sum O_i \cdot \ln(O_i / E_i)$

where O_i is the observed count, E_i is the expected count under the null hypothesis. The G-test should be preferred over Chi-squared test when for any cell in the table, $\frac{1}{2} O - \frac{1}{2} E > E$. Thus the number of annotated features in the later analysis is greater than the earlier annotation.

V. RESULTS AND DISCUSSIONS

A. Evaluation Metrics

To evaluate the effectiveness of our proposed features extraction algorithm we use standard evaluation measures i. e., precision, recall and f-score. The precision alternatively known as positive predictive value, is the fraction of retrieved instances that are relevant, while recall/sensitivity is the fraction of relevant instances that are retrieved.

TABLE 3: Results of pattern based approach using likelihood ratio test.

Patterns	BNP			cBNP			vRBNP			Combined pattern BNP+cBNP+vRBNP with [12]			Combined pattern BNP+cBNP+vRBNP with [8]		
	P	R	F	P	R	F	P	R	F	P	R	F	P	R	F
APEX	45.97	83.63	59.33	81.03	72.97	76.79	86.53	30.56	35.51	88.23	72.97	76.79	87.31	61.94	72.98
Canon	44.08	87.63	58.65	76.28	70.54	73.29	86.94	39.14	53.56	87.23	70.40	73.29	88.28	78.67	82.52
Creative	38.73	88.84	53.94	76.98	66.48	71.34	88.01	41.53	57.21	89.41	66.48	71.34	92.35	79.54	83.92
Nikon	46.70	90.76	61.66	79.33	74.32	76.75	85.72	36.25	50.63	85.89	74.32	76.75	87.83	73.84	79.85
Nokia	45.19	90.07	60.19	81.30	74.55	77.78	87.52	45.28	57.38	87.87	74.55	77.78	89.44	74.31	81.73
Average	44.13	88.19	58.75	78.98	71.77	75.19	86.94	38.55	50.86	87.73	71.74	75.19	89.04	73.66	80.20

From the TABLE 3, it is clear that the proposed pattern provides consistent results on both annotation schemes. The average precision of proposed patterns based on comparison with manual features of [10,6] are 87.73 and 89.04 respectively; which is comparatively improved from that of earlier combined pattern with precision of 78.98 and

77.96 respectively. Similarly, the recall and f-score are also identical on both schemes. The consistent results prove the validity of our proposed approach.

B. G-Measure

For comparison of the results, maximum likelihood statistical significance test or G-test used. LRT (Likelihood Ratio Test) approach was initially employed with f-score by [11,1] for product feature extraction and then extended by [5]. This approach employed dependency patterns with subsequent similarity. However, the F-Measure of cBNP outperform over the other two patterns. Therefore, for comparison, we have selected cBNP as shown in Table 8. Formula to calculate the G-Measure as $\sqrt{\text{Precision} \times \text{Recall}}$.

TABLE 4: Comparison of results using G-Measure

Dataset	G-Test			Combined Pattern		
	P	R	G-Measure	P	R	G-Measure
APEX	81.03	72.97	76.89	88.23	78.67	83.31
Canon	76.28	70.54	73.35	87.23	74.56	80.65
Creative	76.98	66.48	71.54	89.41	69.53	78.85
Nikon	79.33	74.32	76.78	85.89	78.94	82.34
Nokia	81.30	74.55	77.85	87.87	79.83	83.75
Average	78.98	71.77	75.28	87.73	76.31	81.78

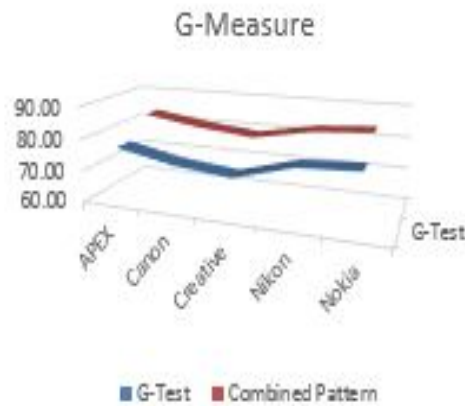


Fig 5. G-Measure

TABLE 4. and Fig 5, represents average G-Measure of our proposed Combined Pattern compared, which shows a significant improvement over maximum likelihood statistical significance or G-Test. Thus, our second problem has been resolved by proposing new hybrid pattern, i.e. by combining NN, JJ, VB, RB.

REFERENCES

[1] Khairullah khan, Baharum B. Baharudin, 'Analysis of Syntactic Patterns for Identification of Features from Unstructured Reviews', Intelligent and Advanced Systems (ICIAS), IEEE Xplore: 20 September 2012.

-
- [2] Khairullah khan, Baharum B. Baharudin and Aurangzeb Khan, 'Identifying Product Features from Customer Reviews Using Hybrid Patterns', The International Arab Journal of Information Technology, Vol. 11, No. 3, May 2014.
 - [3] Pang, B, L.Lee and S.Vaithyanathan, 'Thumbs up?: sentiment classification using machine Learning techniques'. In EMNLP' 02: Proceedings of the ACL-02 conference on Empirical methods in NLP. Association for Computational Linguistics, Morristown, NJ, USA, 79-86, 2002.
 - [4] Hu, M. and B. Liu, 'Mining and summarizing customer reviews'. In KDD' 04: Proceedings of the Tenth ACM SIGKDD international conference on KDD. ACM, New York, NY, USA,168-177,2004.
 - [5] Ana Mariz Popescu and Etzioni, 'Extracting product features and opinions from reviews', Proceedings of the conference on HLT and Empirical Methods in Natural Language Processing HLT '05, Pages 339-346 .
 - [6] Anuj sharma, Shubhamoy Dey, 'Performance Investigation of Feature Selection Methods and Sentiment Lexicons for Sentiment Analysis', Special Issue of International Journal of Computer Applications (0975-8887) on ACCTHPCA, June 2012.
 - [7] Liliana Ferreira, Niklas Jakob, Iryna Gurevych, 'A Comparative Study of Feature Extraction Algorithms in Customer Reviews', ICSC'08 Proceedings of the 2008 IEEE, pages 144-151.
 - [8] Lei Zhang, Bing Liu, 'Identifying Noun Product Features that Imply Opinions', Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics, page 575-580, Portland, Oregon, June 19-24,2011. ©2011 Association for Computational Linguistics.
 - [9] Lizhen Liu, Zhixin Lv, Hanshi Wang, 'Opinion Mining Based on Feature-Level', in 5th International Congress on Image and Signal Processing (CISP 2012).
 - [10] Farah Benamara et al, 'Sentiment Analysis: Adjectives and Adverbs are better than Adjectives Alone', ICWSM '2007 Boulder, CO USA.
 - [11] Yi.J., Nasukawa T., Bunescu R., and Biblack W., " Sentiment analyzer: extracting sentiments about a given topic using natural language processing techniques in Data Mining", 2003, ICDM 2003. Third IEEE International Conference on 2003.