

SENTENCE-LEVEL SENTIMENT POLARITY CALCULATION FOR CUSTOMER REVIEWS BY CONSIDERING COMPLEX SENTENTIAL STRUCTURES

N.Lalithamani¹, Leela Sravanthi Thati², Rakesh Adhikesavan³

¹Assistant Professor (SG), Department of Computer Science & Engineering, Amrita School of Engineering, Amrita Vishwa Vidyapeetham(University), Tamil Nadu, India

²Student 4th year B.Tech, Computer Science & Engineering, Amrita School of Engineering, Amrita Vishwa Vidyapeetham (University), Tamil Nadu, India

³Student 4th year B.Tech, Computer Science & Engineering, Amrita School of Engineering, Amrita Vishwa Vidyapeetham (University), Tamil Nadu, India

Abstract

People often find themselves in a situation where they need to make simple decisions, for example, what book to read, or which movie to watch, or which smart phone to buy. In these kinds of situations, one often seeks the opinion of others in the form of IMDB (International Movie Database) rating or magazine reviews or online product reviews etc. The general opinion about a product or a publication or a service or a company is usually expressed in the form of customer reviews and the web has facilitated sharing and expressing opinions in the form of free style texts which is unstructured or semi structured. In this paper, we introduce a sentence level, sentiment polarity calculation that identifies complex sentential structures and modifies the sentence polarity accordingly. The sentiment polarity of a text is given by a score that lies in the range [-1,1] and denotes the positivity of the tone of the text's author. Thus, the sentiment polarity or the sentiment score of a review can be used to perceive the opinion of the reviewer. The average polarity of all the words in a sentence gives the polarity of a sentence. The polarity of all the common words in the English dictionary is retrieved from CLiPS Pattern module (BSD license) for Python. After obtaining the polarity of a sentence, it is modified based on the structure of the sentence which is identified using a set of heuristic rules that use POS (Parts of Speech) tagging to identify relationships between words. Then the average polarity of all the sentences in a review, gives the overall polarity of the review. We have compared the results against the inbuilt function `pattern.en.sentiment (sentence)` and based on empirical evidence the result are more accurate in many cases and in other cases the results are the same.

Keywords: Sentimental analysis, Sentiment polarity, Opinion mining, and Sentiment mining

-----***-----

1. INTRODUCTION

The World Wide Web is an enormous collection of user generated data. Social networking, blogs, forums, Twitter, micro-blogs, etc has facilitated and encouraged people to share their personal experiences or viewpoints / opinions about a product, service or a company and many users of the web are seeking these personal experiences or opinions of others because these opinions are important due to reasons stated by Bing Liu [1], which are:

- Opinions are key influences of our behavior
- Our perception and beliefs of reality are based on how others see the world
- Whenever we need to make a decision we seek opinion from others, individuals need to make decisions to buy a product or avail a service and they seek opinions from friends and family, while organizations try to benchmark their products and

services and use consultants, surveys, opinion polls etc to find consumer opinions.

With ever increasing amount of data and diverse viewpoints that is presented on the web, it is impossible for one to discern an overall opinion or a general viewpoint of a product, service or a company by just reading through reviews, also it is not humanly possible for one to read more than a particular number of reviews, which makes automating the process of extracting opinions from raw data a very important and pertinent task.

In order to address this problem of Opinion Mining or Sentiment Mining, in this paper we introduce a method to modify prior assigned sentiment polarity by considering complex sentential structures.

Generally, as observed by Polanyi and Zaenen [6] and Wilson et al. [7], there exist complex relationships between words in a sentence that influences its sentiment polarity. For example, consider negation (eg. “not” in the sentence “The screen resolution is not good” or “never” in the sentence-“The products of this series has never performed well”), intensifier terms (eg. “very” intensifies the sentiment of its adjacent word in “The performance is very good”), Un certainty (eg. “The mileage has to be good”) etc.

In our approach, we use prior assigned sentiment scores of words in the English dictionary from CLiPS pattern module for Python [2] and calculate sentence polarity and then identify such complex relationships between words with the use of heuristic rules and then identify its influence on the sentiment polarity and then modify the polarity accordingly.

2. RELATED WORK

Previous works have studied sentiment analysis at document level or sentence level to predict the overall sentiment polarity of the document. Hu and Liu [4] extracted and summarized customer reviews of Electronic goods such as smart phones, televisions etc. They extracted features of entities (products) such as resolution, battery life etc and predicted whether the opinion of the feature or aspect is positive or negative by checking for the presence of positive or negative opinion words. Wilson et al. [7] used a machine learning approach to predict sentiment polarity in the phrase level, while Shaikh et al [3] and Moilanen and Pulman [5] analyzed the words present within phrases to detect sentiment polarity.

Kim and Hovy [15] tried an alternate way of summarizing reviews by extracting information on why a reviewer liked or disliked a particular product, additional problem is use of comparative sentences [16, 17] like “ Brand X is better than brand Y because brand Y has poor performance” and are important source of information that influence the sentiment of the review.

3. PROPOSED MODEL

Part-of-speech information is very often exploited in sentiment analysis because POS tagging can be used for word sense disambiguation [9]. Researchers suggest that verbs (eg. “like”) can be strong influencers of sentiment. There have been many comparisons of the effectiveness of adverbs [14], verbs, adjectives in influencing sentiment polarity and sub categorization often plays a role too [10,11,12].

In our approach, sentence structures and complex relationships between words are identified using POS tagging which is achieved by using the tag() function that is imported from pattern.en package.

Some of the common tags are:

Table -1: POS tags

Tag	Description	Example
CC	conjunction, coordinating	and, or, but
CD	cardinal number	five, three, 13%
DT	determiner	the, a, these
EX	existential there	there were six boys
FW	foreign word	mais
IN	conjunction, subordinating preposition	of, on, before, unless or
JJ	adjective	nice, easy
JJR	adjective, comparative	nicer, easier
JJS	adjective, superlative	nicest, easiest
NN	noun, singular or mass	tiger, chair, laughter
NNS	noun, plural	tigers, chairs, insects
NNP	noun, proper singular	Germany, God, Alice
NNPS	noun, proper plural	we met two Christmases ago
POS	possessive ending	's
PRP	pronoun, personal	me, you, it
PRP\$	pronoun, possessive	my, your, our
RB	adverb	extremely, loudly, hard
RBR	adverb, comparative	better
RBS	adverb, superlative	best
RP	adverb, particle	about, off, up
SYM	symbol	%
VB	verb, base form	think
VBZ	verb, 3rd person singular present	she thinks

VBP	verb, non-3rd person	I think
VBD	verb, past tense	they thought
VBN	verb, past participle	a sunken ship
VBG	verb, gerund or present participle	thinking is fun

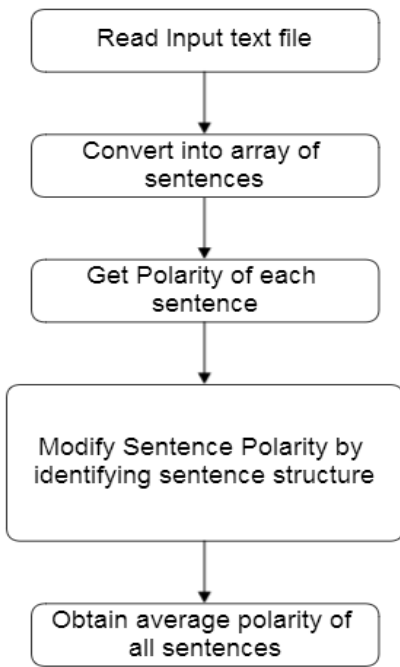


Fig - 1: System Design

Fig -1 is a block diagram that shows the basic design of the system and gives a crude idea of how the sentiment polarity is obtained; the input is read in the form of raw text which is converted into an array of sentences using NLTK (Natural language Toolkit) sentence tokenizer. (<http://www.nltk.org/api/nltk.tokenize.html>)

The polarity of words is retrieved from the package pattern and the sentence polarity is calculated using: Sum of polarity of all the words in a sentence divided by the total number of words in the sentence.

$$\frac{\sum(\text{polarity}(\text{word}))}{N(\text{words})}$$

Polarity(word) – polarity value returned by pattern.en.polarity(word)

N(words) – total number of words in the sentence

A set of heuristic rules is now used to modify the sentence polarity by identifying complex phrases and relationships between words. Each sentence is checked against each rule and if there is a match the existing sentence polarity is modified based on the rule. The more rules that can be identified and added to the system, more complex relationships between words or phrases can be identified and the accuracy of the end resultant polarity value of the sentence will be more accurate.

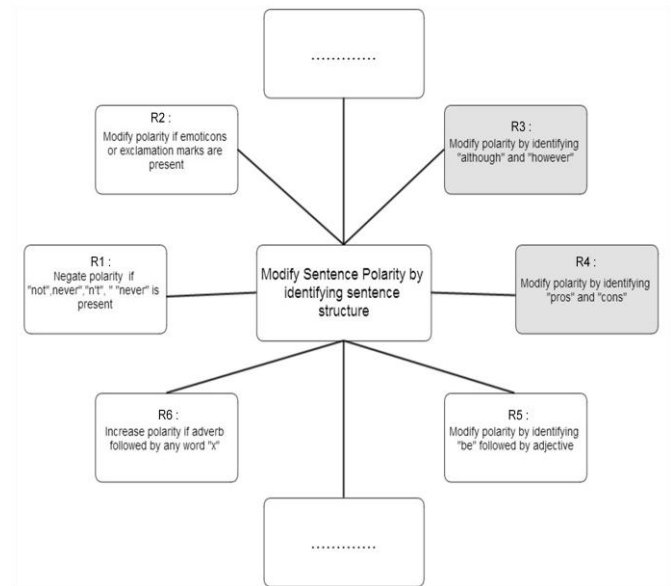


Fig - 2: System Design with defined Rules

Fig – 2 expands the fourth step of Fig – 1 (System Design) which is “Modify Sentence Polarity by identifying sentence structure” and displays the set of heuristic rules that are explained below. The dotted blocks imply that more rules can be added to the design to improve accuracy of the system. The briefings of rules are as follows:

3.1 Rule 1: Negation

Polarity of the word that follows the word “not” or “never” or a word ending with “n’t” is negated.

Consider the following example sentences from product reviews: (Note: All the following examples are sentences extracted from reviews that were taken from the web and the product name and model have not been mentioned)

- The battery life is **not** good.
- The lamp **isn't** good.
 - Negate polarity of “good”
- The models of this series have **never** performed great.
 - Negate polarity of “great”

However, one difficulty with handling negation is that it can often be expressed in subtle ways. Irony and sarcasm is very hard to detect. Wilson et al. [7] discusses some of the complex negation effects. Na et al. [13] attempted to model negation more accurately and observed a 3% increase in accuracy.

3.2 Rule 2: Symbols

The presence of emoticons or smiles and exclamation marks influences the sentiment polarity. Consider the following sentences:

- This Ultra Book is great!
 - Intensify sentence polarity
- The maps is very unreliable ☹️
 - Intensify sentence polarity

3.3 Rule 3: Adjective Intensifiers

There are some words that can be used to intensify adjectives such as, 'very' 'utterly' 'totally' 'completely' 'absolutely' 'entirely' 'really'.

Consider the following sentence:

- The design is very beautiful
 - Intensify polarity of beautiful

3.4 Rule 4: Uncertainty or Suggestion

Many a times, reviewers make a suggestion on how a particular feature of the entity or product or service has to be improved, the polarity of such sentences should rather be neutral than positive or negative. Consider the following examples:

- The picture clarity has to be made **better**
 - Neutralize polarity of "better"
- The graphic processor has to be **excellent** for Hi Def movies to play smoothly
 - Neutralize polarity of "excellent"

Sometimes, reviewer is uncertain about something, for example

- The mileage ought to be **good**
 - Neutralize polarity of "good"
- The product might be **expensive**
 - Neutralize polarity of "expensive"

This is achieved by looking for a comparative adjective that follows the word "be", In the case of superlative adjective the polarity is not modified, consider the example

- This has to be the **best** phone I have ever seen
 - Sentence Polarity is not modified

POS tags are identified using POS tagger that is imported from pattern.en package. (Refer to Table-1 for the POS tags).

3.5 Rule 5: Identify PROS and CONS

Some reviews have a PROS and CONS list, consider the example:

PROS

Battery life
Tons of apps
All slimmed up
It's fast
Dual Cameras

CONS

No standard slots or connectors
No Flash
Wi-Fi issues

Most of the words in the above list have zero polarity, but a PROS and CONS list definitely influences the sentiment of the review. So we could compare the number of items in each list and modify polarity according to which list has higher number of items.

3.6 Rule 6: Identify "however" and "although"

This rule has not been implemented and tested yet and is a mere suggestion or a proposal. Consider the following sentences:

- Although the phone is very expensive it's a great buy because of the excellent aesthetics
 - The overall sentiment is positive
- Although the printer is efficient it is very expensive
 - The overall sentiment is negative
- The boot space is good however the leg space in the front seat is very less
 - The overall sentiment is negative
- The CFL is expensive however it is very energy efficient
 - The overall sentiment is positive

In such cases the polarity can be modified by identifying the words "however" or "although" and identifying towards which side the sentiment is leaning towards.

4. RESULTS

A tool (named Sentinel) was built using Python programming language that follows the above proposed approach to modify sentence polarity based on the sentential structure.

Sentinel reads a text file as input and prints the Sentiment Score of the text. The polarity of all the sentences is obtained and modified based on the heuristic rules and the average polarity of all the sentences in the text file gives the Sentiment Score of the text which denotes the positivity of the tone of the text's author.

Sample input and output:(Note: Inputs are sentences extracted from product reviews obtained from cnet.reviews.com)

Input: Battery pack case plus extra flash storage: too good to be true? The Mophie Space Pack might be one of the best iPhone accessories ever seen

Output: 0.65

Input: Although the battery life is poor, the phone is a great buy because it is affordable

Output: 0.6

Input: The performance is good

Output: 0.7

Input: The performance is really good

Output: 1

Input: The case isn't attractive

Output: - 0.8

5. COMPARISON WITH BENCHMARK

CLiPS (Computational Linguistics & Psycholinguistics Research Center) patten.en module contains a fast part-of-speech tagger for English (identifies nouns, adjectives, verbs, etc. in a sentence), sentiment analysis, tools for English verb conjugation and noun singularization & pluralization, and a WordNet interface. Interested readers refer to [2] for more details on CLiPS pattern module.

Sentinel's results have been tested against the inbuilt function `patten.en.sentiment(sentence)` that returns (polarity, subjectivity) tuple, here we consider the sentiment polarity alone.

In: `patten.en.sentiment("The screen ought to be good")`[0]

Out: 0.7

Sentinel Output: 0

In: `patten.en.sentiment("The maps has never been good")`[0]

Out: 0.7

Sentinel Output: - 0.7

In: `patten.en.sentiment("The graphic processor has to be excellent for Hi Def movies to play smoothly")`[0]

Out: 0.5

Sentinel Output: 0.1

6. SUMMARY

To review, determining the overall Sentiment score of user/customer reviews of a product, publication, service or a company is an important research topic, it is especially useful for services that have a large number of reviews, to help an opinion seeker discern the public view point quickly and accurately. However I would like to add some cautionary

notes, human interpretation is still required to understand sarcasm, satire, etc. and a machine can easily misread the above, also the redundancy issue as mentioned by Tsaparas, Ntoulas and Terzi [8] is also a valid concern, In my opinion the distribution (in terms of negative and positive points) is important in determining the quality of a review and users tend to feel a review is helpful or not based on whether the review appears to be genuine and whether it expresses opinion about many features of the product or not. So developing the program to determine the genuineness and helpfulness of a review could be a useful and important addition.

7. CONCLUSIONS

Through this paper we have covered some very interesting techniques and approaches to calculate sentiment polarity and have proposed a way to modify pre assigned polarity values to increase accuracy of sentiment polarity calculation, and have explained the pertinent and intriguing task of sentimental analysis and we encourage the readers to take up some of the many open challenges that still remain.

ACKNOWLEDGEMENTS

We would like to express our sincere thanks and gratitude to Ms.N.Lalithamani, Assistant Professor(SG), Dept. of CSE, Amrita Vishwa Vidyapeetham for her guidance and we would also like to express our greatest gratitude to Mr.Sai Sundarakrishna, Research Scientist, R&D Dept., Caterpillar India Pvt. Ltd for his motivation and mentoring.
Rakesh & Sravanthi

REFERENCES

- [1]. Bing Liu, "Sentiment Analysis and Opinion Mining", Morgan and Claypool Publishers, May 2012.
- [2]. Tom De Smedt "Pattern For Python", International Journal of Machine Learning Research 13, pp. 2063 – 2067, 2012.
- [3]. Shaikh, M.A.M., Prendinger, H., Ishizuka, M.: "Sentiment Assessment of Text by Analyzing Linguistic Features and Contextual Valence Assignment", Applied Artificial Intelligence, Vol. 22, No.6, pp. 558–601, 2008.
- [4]. Hu M, Liu B. "Mining and Summarizing Customer reviews". In Proceedings of the 10th Special Interest Group on Knowledge Discovery and Data Mining (SGKDD), Seattle, WA, USA, pp 168-177, August 2004.
- [5]. Moilanen, K., Pulman, S.: "Sentiment Composition", "International Conference on Recent Advances in Natural Language Processing", pp. 378–382, 2007
- [6]. Polanyi, L., Zaenen, A. "Preface: Contextual Valence Shifters", Computing Attitude and Affect in Text: Theory and Applications, Springer Publications, Vol. 20, pp. 1–10, 2006
- [7]. Wilson, T., Wiebe, J., Hoffmann, P., "Recognizing contextual polarity: An exploration of features for phrase-level sentiment analysis", Journal of Computational Linguistics, Vol. 35, No.3, pp. 399–433, 2009

- [8]. P. Tsaparas, A. Ntoulas, E. Terzi, "Selecting a Comprehensive Set of Reviews", Proceedings of International Conference on Knowledge Discovery and Data Mining (KDD), August 2011
- [9]. Yorick Wilks and Mark Stevenson. "The grammar of sense: Using part-of-speech tags as a first step in semantic disambiguation", Journal of Natural Language Engineering, Vol. 4, No.2, pp. 135–144, 1998.
- [10]. Farah Benamara, Carmine Cesarano, Antonio Picariello, Diego Reforgiato, and V. S. Subrahmanian. "Sentiment analysis: Adjectives and adverbs are better than adjectives alone", Short Paper: Proceedings of the International Conference on Weblogs and Social Media (ICWSM), 2007.
- [11]. J. Kamps and M. Marx and R.J. Mokken and M. De Rijke, "Using WordNet to measure semantic orientation of Adjectives", Proceedings of the International Conference on Language Resources and Evaluation (LREC-04), Vol. 4, 2004, pp. 1115 - 1118, Lisbon, Portugal.
- [12]. Janyce M. Wiebe, Theresa Wilson, Rebecca Bruce, Matthew Bell, and Melanie Martin. "Learning subjective language", Journal of Computational Linguistics, Vol.30, No.3, pp. 277–308, September 2004.
- [13]. Jin-Cheon Na, Haiyang Sui, Christopher Khoo, Syin Chan, and Yunyun Zhou. "Effectiveness of simple linguistic processing in automatic sentiment classification of product reviews", Proceedings of the International Conference of the International Society for Knowledge Organization (ISKO)", pp. 49–54, 2004.
- [14]. Vasileios Hatzivassiloglou and Kathleen McKeown. "Predicting the semantic orientation of adjectives", Proceedings of the International Joint Conference on ACL/EACL (Association for Computational Linguistics/ European Association for Computational Linguistics)", pp.174–181, 1997.
- [15]. Soo-Min Kim and Eduard Hovy. Crystal: "Analyzing predictive opinions on the web", Proceedings of the International Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning (EMNLP-CoNLL), 2007.
- [16]. Nitin Jindal and Bing Liu. "Identifying comparative sentences in text documents", Proceedings of the ACM Special Interest Group on Information Retrieval (SIGIR)", 2006.
- [17]. Nitin Jindal and Bing Liu. "Mining comparative sentences and relations", Proceedings of the International Conference on Association for the Advancement of Artificial Intelligence (AAAI)", 2006.

BIOGRAPHIES



Ms. N. Lalithamani was born in 1975. She received her Master's degree in 1998. She holds a University rank as a part of her post graduate programme. Her fields of interests include (active) databases, automated

biometrics, Information retrieval etc. She is currently an Assistant Professor at Amrita School of Engineering, Amrita Vishwa Vidyapeetham. In the past she has worked on several projects concerning E-learning.



Leela Sravanthi That is the student of Amrita School of Engineering, Coimbatore, India. She was awarded Prathibha Award in her 10th Standard from Andhra Pradesh State government and she also stood among the top qualified students in International Maths Olympiad in the year 2009-2010.



Rakesh Adhikesavan, born in 1992 is now pursuing his B.Tech degree in Computer Science and Engineering from Amrita Vishwa Vidyapeetham and is currently participating in a Global Study Program at University of California Davis. His interests in the field of Computer Science include Information Retrieval and Information Visualization.