

Dive deeper: Deep Semantics for Sentiment Analysis

Nikhil Kumar Jadhav

Masters Student

Computer Science & Engineering Dept.
IIT Bombay

nikhilkumar@cse.iitb.ac.in

Pushpak Bhattacharyya

Professor

Computer Science & Engineering Dept.
IIT Bombay

pb@cse.iitb.ac.in

Abstract

Sentiment Analysis of online reviews is difficult due to the different opinions of different opinion holders with an added notion of time. The 5-tuple definition of sentiment analysis reiterates the same fact in a more formal way. Most of the previous methods for sentiment analysis make use of supervised approaches which fail to capture this crucial fact. They merely try to identify subjective words and or phrases. Due to this, the fact that not all of these words and phrases actually contribute to the overall sentiment of the *text* is ignored. We propose to use deep semantic processing to identify only relevant subjective terms. We generate a UNL (Universal Networking Language) graph for the input *text*. Rules are applied on the graph to extract relevant terms. The sentiment expressed in these terms is used to figure out the overall sentiment of the *text*. Results on binary sentiment classification have shown promising results.

1 Introduction

Many works in sentiment analysis try to make use of shallow processing techniques. The common thing in all these works is that they merely try to identify sentiment-bearing expressions as shown by Ruppenhofer and Rehbein (2012). No effort has been made to identify which expression actually contributes to the overall sentiment of the text. In Mukherjee and Bhattacharyya (2012) these expressions are given weight-age according to their position w.r.t. the discourse elements in the *text*. But it still takes into account each expression.

Semantic analysis is essential to understand the exact meaning conveyed in the *text*. Some words tend to mislead the meaning of a given piece of *text*

as shown in the previous example. WSD (Word Sense Disambiguation) is a technique which can be used to get the right sense of the word. Balamurali et al., (2012) have made use of WordNet synsets for a supervised sentiment classification task. Tamare (2010) and Rentoumi (2009) have also shown a performance improvement by using WSD as compared to word-based features for a supervised sentiment classification task. In Hasan et al., (2012), semantic concepts have been used as additional features in addition to word-based features to show a performance improvement. Syntagmatic or structural properties of text are used in many NLP applications like machine translation, speech recognition, named entity recognition, etc. A clustering based approach which makes use of syntactic features of text has been shown to improve performance in Kashyap et al., (2013). Another approach can be found in Mukherjee and Bhattacharyya (2012) which makes use of lightweight discourse for sentiment analysis. In general, approaches using semantic analysis are expensive than syntax-based approaches due to the shallow processing involved in the latter. As pointed out earlier, all these works incorporate all the sentiment-bearing expressions to evaluate the overall sentiment of the *text*. The fact that not all expressions contribute to the overall sentiment is completely ignored due to this. Our approach tries to resolve this issue. To do this, we create a UNL graph for each piece of *text* and include only the relevant expressions to predict the sentiment. Relevant expressions are those which satisfy the rules/conditions. After getting these expressions, we use a simple dictionary lookup along with attributes of words in a UNL graph to calculate the sentiment.

The rest of the paper is organized as follows. Section 2 discusses related work. Section 3 explains our approach in detail. The experimental setup is explained in Section 4. Results of the ex-

periments are presented in Section 5. Section 6 discusses these results followed by conclusion in Section 7. Section 8 hints at some future work.

2 Related Work

There has been a lot of work on using semantics in sentiment analysis. Hasan et al., (2012) have made use of semantic concepts as additional features in a word-based supervised sentiment classifier. Each entity is treated as a semantic concept e.g. *iPhone, Apple, Microsoft, MacBook, iPad, etc.*. Using these concepts as features, they try to measure their correlation with positive and negative sentiments. In Verma et al., (2009), effort has been made to construct document feature vectors that are sentiment-sensitive and use world knowledge. This has been achieved by incorporating sentiment-bearing words as features in document vectors. The use of WordNet synsets is found in Balamurali et al., (2012), Rentoumi (2009) and Tamara (2010). The one thing common with these approaches is that they make use of shallow semantics. An argument has been made in Choi and Carde (2008) for determining the polarity of a sentiment-bearing expression that words or constituents within the expression can interact with each other to yield a particular overall polarity. Structural inference motivated by compositional semantics has been used in this work. This work shows use of deep semantic information for the task of sentiment classification. A novel use of semantic frames is found in Ruppenhofer and Rehbein (2012). As a step towards making use of deep semantics, they propose SentiFrameNet which is an extension to FrameNet. A semantic frame can be thought of as a conceptual structure describing an event, relation, or object and the participants in it. It has been shown that potential and relevant sentiment bearing expressions can be easily pulled out from the sentence using the SentiFrameNet. All these works try to bridge the gap between rule-based and machine-learning based approaches but except the work in Ruppenhofer and Rehbein (2012), all the other approaches consider all the sentiment-bearing expressions in the text.

3 Use of Deep Semantics

Before devising any solution to a problem, it is advisable to have a concise definition of the problem. Let us look at the formal definition of the

sentiment analysis problem as given in Liu (2010). Before we do that, let us consider the following review for a movie, "1) I went to watch the new James Bond flick, *Skyfall* at IMAX which is the best theater in Mumbai with my brother a month ago. 2) I really liked the seating arrangement over there. 3) The screenplay was superb and kept me guessing till the end. 4) My brother doesn't like the hospitality in the theater even now. 5) The movie is really good and the best bond flick ever." This is a snippet of the review for a movie named *Skyfall*. There are many entities and opinions expressed in it. 1) is an objective statement. 2) is subjective but is intended for the theater and not the movie. 3) is a positive statement about the screenplay which is an important aspect of the movie. 4) is a subjective statement but is made by the author's brother and also it is about the hospitality in the theater and not the movie or any of its aspects. 5) reflects a positive view of the movie for the author. We can see from this example that not only the opinion but the opinion holder and the entity about which the opinion has been expressed are also very important for sentiment analysis. Also, as can be seen from 1), 4) and 5) there is also a notion of time associated with every sentiment expressed. Now, let us define the sentiment analysis problem formally as given in Liu (2010).

A direct opinion about the object is a quintuple $\langle o_j, f_{jk}, oo_{ijkl}, h_i, t_l \rangle$, where o_j is the object, f_{jk} is the feature of the object o_j , oo_{ijkl} is the orientation or polarity of the opinion on feature f_{jk} of object o_j , h_i is the opinion holder and t_i is the time when the opinion is expressed by h_i .

As can be seen from the formal definition of sentiment analysis and the motivating example, not all sentiment-bearing expressions contribute to the overall sentiment of the *text*. To solve this problem, we can make use of semantic roles in the *text*. Semantic role is the underlying relationship that the underlying participant has with the main verb. To identify the semantic roles, we make use of UNL in our approach.

UNL (Universal Networking Language)

UNL is declarative formal language specifically designed to represent semantic data extracted from natural language texts. In UNL, the information is represented by a semantic network, also called UNL graph. UNL graph is made up of three discrete semantic entities, Universal Words, Univer-

sal Relations, and Universal Attributes. Universal Words are nodes in the semantic network, Universal Relations are arcs linking UWs, and Universal attributes are properties of UWs. To understand UNL better, let us consider an example. UNL graph for "I like that bad boy" is as shown in Figure 1

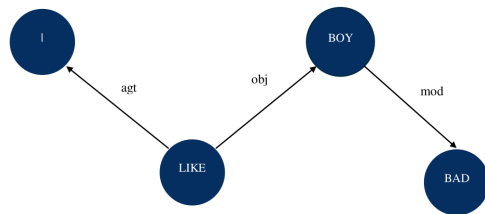


Figure 1: UNL graph for "I like that bad boy"

Here, "I", "like", "bad", and "boy" are the UWs. "agt" (agent), "obj" (patient), and "mod" (modifier) are the Universal Relations. Universal attributes are the properties associated with UWs which will be explained as and when necessary with the rules of our algorithm.

UNL relations

Syntax of a UNL relation is as shown below,

$\langle rel \rangle : \langle scope \rangle \langle source \rangle ; \langle target \rangle$

Where, $\langle rel \rangle$ is the name of the relation, $\langle scope \rangle$ is the scope of the relation, $\langle source \rangle$ is the UW that assigns the relation, and $\langle target \rangle$ is the UW that receives the relation

We have considered the following Universal relations in our approach,

1) agt relation : *agt* stands for agent. An agent is a participant in action that provokes a change of state or location. The *agt* relation for the sentence "John killed Mary" is *agt(killed , John)*. This means that the action of *killing* was performed by *John*.

2) obj relation : *obj* stands for patient. A patient is a participant in action that undergoes a change of state or location. The *obj* relation for the sentence "John killed Mary" is *obj(killed , Mary)*. This means that the patient/object of *killing* is *Mary*.

3) aoj relation : *aoj* stands for object of an attribute. In the sentence "John is happy", the *aoj* relation is *aoj(happy , John)*.

4) mod relation : *mod* stands for modifier of an object. In the sentence "a beautiful book", the *mod* relation is *mod(book , beautiful)*.

5) man relation : *man* relation stands for manner. It is used to indicate how the action, experience or process of an event is carried out. In the sentence "The scenery is beautifully shot", the *man* relation is *man(beautifully , shot)*.

6) and relation : *and* relation is used to state a conjunction between two entities. In the sentence "Happy and cheerful", the *and* relation is *and(Happy , cheerful)*.

Architecture

As show in Figure 1, the modifier "bad" is associated with the object of the main verb. It shouldn't affect the sentiment of the main agent. Therefore, we can ignore the modifier relation of the main object in such cases. After doing that, the sentiment of this sentence can be inferred to be positive. The approach followed in the project is to first generate a UNL graph for the given input sentence. Then a set of rules is applied and used to infer the sentiment of the sentence. The process is shown in Figure 2. The UNL generator shown in the Figure 2 has been developed at CFILT.¹ Before, the given piece of text is passed on to the UNL generator, it goes through a number of pre-processing stages. Removal of redundant punctuations, special characters, emoticons, etc. are part of this process. This is extremely important because the UNL generator is not able to handle special characters at the moment. We can see that, the performance of the overall system is limited by this. A more robust version of the UNL generator will certainly allow the system to infer the sentiment more accurately.

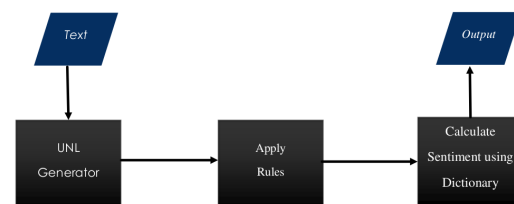


Figure 2: System Architecture

Rules

There is a separate rule for each relation. For each UW (Universal word) considered, if it has a @not

¹<http://www.cfilt.iitb.ac.in/>

attribute then its polarity is reversed. Rules used by the system are as follows,

1) If a given UW is source of the *agt* relation, then its polarity is added to the overall polarity of the text. e.g., "*I like her*". Here, the agt relation will be *agt (like , I)*. The polarity of like being positive, the overall polarity of the text is positive. e.g., "*I don't like her*". Here the agt relation will be *agt (like@not , I)*. The polarity of like is positive but it has an attribute *@not* so its polarity is negative. The overall polarity of the text is negative in this case.

2) If a given UW is source or target of the *obj* relation and has the attribute *@entry* then its polarity is added to the overall polarity of the text. This rule merely takes into account the main verb of the sentence into account, and the it's is polarity considered. e.g., "*I like her*", here the obj relation will be *obj (like@entry , her)*. The polarity of like being positive, the overall polarity of the text is positive

3) If a given UW is the source of the *aoj* relation and has the attribute *@entry* then its polarity is added to the overall polarity of the text. e.g., "*Going downtown tonight it will be amazing on the waterfront with the snow*". Here, the *aoj* relation is *aoj (amazing@entry , it)*. *amazing* has a positive polarity and therefore overall polarity is positive in this case.

4) If a given UW is the target of the *mod* relation and the source UW has the attribute *@entry* or has the attribute *@indef* then polarity of the target UW is added to the overall polarity of the text. e.g., "*I like that bad boy*". Here, the *aoj* relation is *mod (boy , bad)*. *bad* has a negative polarity but the source UW, boy does not have an *@entry* attribute. So, in this case negative polarity of bad is not considered as should be the case. e.g., "*She has a gorgeous face*". Here, the *mod* relation is *mod (face@indef , gorgeous)*. *gorgeous* has a positive polarity and face has an attribute *@indef*. So, polarity of gorgeous should be considered.

5) If a given UW is the target of the *man* relation and the source UW has the attribute *@entry* then polarity of the target UW is added to the overall polarity of the text. Or if the target UW has the attribute *@entry* then also we can consider polarity of the target UW. e.g., "*He always runs fast*". Here, the *aoj* relation is *mod (run@entry , fast)*. *fast* has a positive polarity and the source UW, run has the *@entry* attribute. So, in this case positive

polarity of fast is added to the overall polarity of the sentence. Polarities of both the source and target UW of the *and* relation are considered.

6) In "*Happy and Cheerful*", the *and* relation is *and(Happy, Cheerful)*. *Happy* and *Cheerful*, both have a positive polarity, which gives this sentence an overall positive polarity.

The polarity value of each individual word is looked up in a dictionary of positive or negative words used is Liu (2010) After all the rules are applied, summation of all the calculated polarity values is done. If this sum is greater than 0 then it is considered as positive, and negative otherwise. This system is negative biased due to the fact that people often tend to express negative sentiment indirectly or by comparison with something good. A more detailed discussion on negative texts is provided in section 6.

4 Experimental Setup

Analysis was performed for monolingual binary sentiment classification task. The language used in this case was *English*. The comparison was done between 5 systems viz. System using words as features, WordNet sense based system as given in Balamurali et al., (2012), Clusters based system as described in Kashyap et al., (2013), Discourse rules based system as given in Mukherjee and Bhattacharyya (2012), UNL rule based system. Two polarity datasets were used to perform the experiments.

1. EN-TD: English Tourism corpus as used in Ye et al., (2009). It consists of 594 positive and 593 negative reviews.
2. EN-PD: English Product (music albums) review corpus Blitzer et al., (2007). It consists of 702 positive and 702 negative reviews.

For the WordNet sense, and Clusters based systems, a manually sense tagged version of the (EN-PD) has been used. Also, a automatically sense tagged version of (EN-TD) was used on these systems. The tagging in the later case was using an automated WSD engine, trained on a tourism domain Khapra et al., (2013). The results reported for supervised systems are based on 10-fold cross validation.

5 Results

System	EN-TD	EN-PD
Bag of Words	85.53	73.24
Synset-based	88.47	71.58
Cluster-based	95.20	79.36
Discourse-based	71.52	64.81
UNL rule-based	86.08	79.55

Table 1: Classification accuracy (in %) for monolingual sentiment analysis

System	EN-TD		EN-PD	
	Pos	Neg	Pos	Neg
Discourse rules	94.94	48.06	92.73	36.89
UNL rules	95.72	76.44	90.75	68.35

Table 2: Classification accuracy (in %) for positive and negative reviews

The results for monolingual binary sentiment classification task are shown in Table 1. The results reported are the best results obtained in case of supervised systems. The cluster based system performs the best in both cases. The UNL rule-based system performs better only than the bag of words and discourse rule based system. For EN-PD (music album reviews) dataset, the UNL based system outperforms every other system . These results are very promising for a rule-based system. The difference between accuracy for positive and negative reviews for the rule-based systems viz. Discourse rules based and UNL rules based is shown in Table 2. It can be seen that the Discourse rules based system performs slightly better than the UNL based system for positive reviews. On the other hand, the UNL rules based system outperforms it in case of negative reviews by a huge margin.

6 Discussion

The UNL generator used in this case is the bottleneck in terms of performance due to it's speed. It can take a long time to generate UNL graphs for large sentences. Also, it makes use of the standard NLP tools viz. parsing, co-reference resolution, etc. to assign the proper semantic roles in the given *text*. It is well known fact that these techniques work properly only on structured data. The language used in the reviews present in both the datasets is unstructured in considerable number of cases. The UNL generator is still in its infancy

and cannot handle *text* involving special characters. Due to these reasons, a proper UNL graph is not generated in some cases. Also, it is not able to generate proper UNL graphs for even well structured sentences. As a result of these things, the classification accuracy is low. Negative reviews are difficult to classify due to comparative statements and presence of positive words. Also there are some sarcastic sentences which are difficult to classify. Sarcasm is a very difficult problem to tackle. Some related works can be found in Carvalho et al., (2009) and Muresan et al., (2011). In some cases, the reviewers make use of their native language and expressions. This is a big problem for the task of monolingual sentiment classification.

7 Conclusion

This paper made use of deep semantics to tackle the the problem of sentiment analysis. A semantic role labeling method through generation of a UNL graph was used to do this. The main motivation behind this research was the fact that not all sentiment bearing expressions contribute to the overall sentiment of the *text*. The approach was evaluated on two datasets and compared with successful previous approaches which don't make use of deep semantics. The system underperformed all the supervised systems but showed promise by yielding better results than the other rule-based approach. Also, in some cases the performance was very close to the other supervised systems. The system works well on sentences where are inherently complex and difficult for sentiment analysis as it makes use of semantic role labeling. Any rule based system can never be exhaustive in terms of rules. We always need to add new rules to improve on it. In some case, adding new rules might cause side-effects. In this case, as the rules are intuitive, adding of new rules will be easy. Also, analysis of the results hints at some ways to tackle specific problems effectively.

8 Future Work

In some cases, the sentiment was ignored unintentionally due to the lack of rules to capture it. The system certainly needs more rules to capture information effectively. As pointed out in the discussion, language gets updated almost daily. Adding these new words and expressions to the dictionary is critical for improving accuracy.

References

- Ruppenhofer, Josef and Rehbein, Ines. 2012. *Semantic frames as an anchor representation for sentiment analysis*. Proceedings of the 3rd Workshop in Computational Approaches to Subjectivity and Sentiment Analysis
- Mukherjee, Subhabrata and Bhattacharyya, Pushpak. 2012. *Sentiment Analysis in Twitter with Lightweight Discourse Analysis*. COLING
- Balamurali, AR and Joshi, Aditya and Bhattacharyya, Pushpak. 2011. *Harnessing wordnet senses for supervised sentiment classification*. Proceedings of the Conference on Empirical Methods in Natural Language Processing
- Rentoumi, Vassiliki and Giannakopoulos, George and Karkaletsis, Vangelis and Vouros, George A. 2009. *Sentiment analysis of figurative language using a word sense disambiguation approach*. Proceedings of the International Conference RANLP
- Martin-Wanton, Tamara and Balahur-Dobrescu, Alexandra and Montoyo-Guijarro, Andres and Pons-Porrata, Aurora. 2010. *Word sense disambiguation in opinion mining: Pros and cons*. Special issue: Natural Language Processing and its Applications
- Kashyap Popat, Balamurali A.R, Pushpak Bhattacharyya and Gholamreza Haffari. 2013. *The Haves and the Have-Nots: Leveraging Unlabelled Corpora for Sentiment Analysis*. The Association for Computational Linguistics
- Saif, Hassan and He, Yulan and Alani, Harith. 2012. *Semantic sentiment analysis of twitter*. The Semantic Web-ISWC 2012
- Verma, Shitanshu and Bhattacharyya, Pushpak. 2009. *Incorporating semantic knowledge for sentiment analysis*. Proceedings of ICON
- Choi, Yejin and Cardie, Claire. 2008. *Learning with compositional semantics as structural inference for subsentential sentiment analysis*. Proceedings of the Conference on Empirical Methods in Natural Language Processing
- Liu, Bing. 2010. *Sentiment analysis and subjectivity*. Handbook of natural language processing
- Ye, Qiang and Zhang, Ziqiong and Law, Rob. 2009. *Sentiment classification of online reviews to travel destinations by supervised machine learning approaches*. Expert Systems with Applications
- Balamurali, AR and Khapra, Mitesh M and Bhattacharyya, Pushpak. 2013. *Lost in translation: viability of machine translation for cross language sentiment analysis*. Computational Linguistics and Intelligent Text Processing
- Blitzer, John and Dredze, Mark and Pereira, Fernando. 2007. *Biographies, bollywood, boom-boxes and blenders: Domain adaptation for sentiment classification*. ACL
- González-Ibáñez, Roberto and Muresan, Smaranda and Wacholder, Nina. 2011. *Identifying Sarcasm in Twitter: A Closer Look*. ACL
- Carvalho, Paula and Sarmiento, Luís and Silva, Mário J and de Oliveira, Eugénio. 2009. *Clues for detecting irony in user-generated contents: oh...!! it's so easy;-)*. ACM