# SENTIMENT ANALYSIS

**A Seminar Report**

Submitted in partial fulfillment of requirements
for the degree of
Master of Technology (M. Tech)
by
**Vaibhav Tripathi**
**Roll No. 143050009**

Under the guidance of
**Prof. Pushpak Bhattacharyya**

Department of Computer Science and Engineering
Indian Institute of Technology, Bombay

# ABSTRACT

Sentiment analysis is the art of detecting sentiment, opinion or emotion from a piece of text. With the proliferation of social networks, there is now a large amount of opinionated text available on the internet. In recent years, opinionated postings in social media have helped reshape businesses, and sway public sentiments and emotions, impacting our social and political systems. However, sentiment analysis on social media is a challenging task. This report discusses the problems in sentiment analysis in general and sentiment analysis on social media in particular and suggests some solutions.

# CONTENTS

# 1 INTRODUCTION

Natural language processing deals with extracting the subjective information from natural language texts. One such information is the sentiment behind the text. This has also been referred to as opinion/-mood/attitude/appraisal/evaluation in the literature. The simplest form of sentiment analysis involves identifying the polarity of a text, which can be positive, negative or neutral.

An analysis of this form is very useful when the text is large and manual examination is impossible. For instance, a company can easily obtain a feedback on its newly released by performing such an analysis on its review page. Note that this kind of feedback is more natural since the customers/reviewers may write more freely when writing a review in natural language rather than when rating a product numerically (say, on a one to five stars scale) or when filling up feedback forms. More importantly, companies don't have to ask their customers for feedback. The customers post feedbacks at their own leisure in their own writing styles. What is needed is a way to automatically analyse all these reviews and generate some result (may be a numerical score or labels like positive or negative) that may be indicative of the overall customer experience. Sentiment analysis seeks to provide that way.

Putting it simply, if we perform sentiment analysis on a movie review page, and there are more posts like

**"Loved the movie! Too good. :)"**

and less posts like

**"Boring! Wasted time and money."**,

we should be able to conclude that the movie review document has a positive sentiment. This is the minimal setting for a sentiment analysis problem.

Sentiment analysis can be defined as the process of computationally identifying and categorizing sentiments expressed in a piece of text to determine whether the attitude of writer (rather, the sentiment holder) towards a particular topic, product, etc. is positive, negative, or neutral.

### 1.1.1 *Sentiment analysis research*

Sentiment analysis is a Natural Language Processing (NLP) problem. It touches several aspects of NLP, e.g., coreference resolution, negation handling, and word sense disambiguation, which makes it a difficult task since these are not solved problems in NLP. However, sentiment analysis is a highly restricted NLP problem because the system need not fully understand the semantics the text but only needs to understand some aspects of it, i.e., positive or negative sentiments and their target entities or topics.

Sentiment analysis is a relatively new branch of research, having evolved mainly after 2000 AD. However, the research has taken a serious shape since, and there is a heavy ongoing research in this area. There are several reasons for this.

(a) It has a wide arrange of applications, almost in every domain.

(b) Commercial applications have proliferated.

(c) It offers many challenging research problems, which had never been studied before.

(d) Ever since the evolution of the likes of social networks and blogs, data of unprecedented scale is available for analysis.

Sentiment can be defined as a quintuple:

$$\boxed{(e_i, a_{ij}, s_{ijkl}, h_k, t_l)}$$

Here,

- $e_i$ is the name of an entity
- $a_{ij}$ is an aspect of $e_i$

- $s_{ijkl}$ is the sentiment on aspect $a_{ij}$ of entity $e_i$
- $h_k$ is the opinion holder
- $t_l$ is the time when the opinion is expressed by $h_k$

Thus, the task of setiment analysis can be stated as: Given a sentiment document d, discover all sentiment quintuples $(e_i, a_{ij}, s_{ijkl}, h_k, t_l)$ in d. However, it is unlikely that we will ever use all of the components of the quintuple in a particular problem. [1]

### 1.1.2 *Levels of sentiment analysis*

Sentiment analysis can be performed at many levels:

(a) **Document Level:** In this level, we assume that the entire document talks about one particular entity only. In this type of a setting, the opinion holder and time are usually irrelevant. Since we are concerned with just one target throughout the text, the aspect is GENERAL. The opinion tuple thus becomes:

$$\boxed{(e_i, \mathsf{GENERAL}, s_{ijkl}, -, -)}$$

(b) **Sentence Level:** This level of analysis goes to the sentences to determine whether each sentence is positive, negative or neutral.

(c) **Aspect Level:** It is based on the idea that an opinion/sentiment consists of a sentiment (positive or negative) and a target. This is a fine-grained analysis where we identify the aspects of the entity being talked about and find the sentiment about each one of them.

For instance, **"In spite of having a great cast, the movie lacks at story."** is positive about the cast but negative about the story.

(d) **User Level:** This is a relatively less discussed level of sentiment analysis. It is different in the sense that in this approach, we don't seek information about the text but the user producing the text. This kind of a problem is more relevant to social media domain since we have more information about the users there (available from the social graph).

### 1.1.3 *The obvious approach and why it doesn't work*

Not surprisingly, the most important indicators of sentiments are sentiment words, also called opinion words. These are words that are commonly used to express positive or negative sentiments. For example, good, wonderful, and amazing are positive sentiment words, and bad, poor, and terrible are negative sentiment words. Apart from individual words, there could be idioms and phrases that are indicative of a particular sentiment.

The naive approach of sentiment analysis involves identifying sentiment words from the input text, find their polarity scores and them somehow, combine these scores to generate a final sentiment score for the document. Over the years, researchers have designed various sentiment lexicons that contain a list of sentiment words and phrases and their sentiment scores.

While it is established that identifying sentiment words and phrases is crucial to sentiment analysis, using only these for sentiment analysis in a practical setting is far from sufficient. This is due to the follwing issues:

(a) A positive or negative sentiment word may have opposite orientations in different application domains.
For example,

**"The plot of the movie is unpredictable"**

is positive (keeps the viewers engaged) whereas,

**"He is a unpredictable player."**

is likely to be negative since we would like players to be consistent and not unpredictable.

(b) A sentence containing sentiment words may not express any sentiment.

The simplest examples of these would be conditional sentences or interrogative sentences. For example,

**"Can anyone suggest me a good quality camera?"** and **"If you find any fault with this mobile phone, kindly post here."** don't contain sentiment;

whereas,

**"Does anyone know how to fix this stupid camera?"** and **"If you are planning to buy this phone, please change your plan."** do.

(c) Sarcastic sentences with or without sentiment words are the nightmare of sentiment analyzers. Even humans find it non-trivial to understand sarcasm very often. The main challenge is that in most of the cases, sarcasm detection requires domain-specific world knowledge.

Consider this.

**"Had an amazing movie-time. I didn't sleep that good in years!"** Now, to actually find out that this review is negative, the system would have to have the knowledge that sleeping during a movie is a bad indicator for the movie. This makes the problem extremely challenging.

(d) Many sentences without sentiment words can also imply opinions. Sentiment can be expressed by objective sentences also, containing just factual information.

**"My iphone 6 has bending problems."** doesn't contain an opinion as such. Nevertheless, it still has a sentiment.

## 1.2 Applications of sentiment analysis

There are interesting applications to sentiment analysis. With the explosive growth of social media (e.g., reviews, forum discussions, blogs, micro-blogs, Twitter, comments, and postings in social network sites) on the Web, there is a large amount of user-generated content available to access at all times. Sentiment analysis on this data can yield interesting and useful results. Some of the most common applications of sentiment analysis are:

### 1.2.1 *Decision Making*

People have always depended on opinions or advice for making important decisions. This was true even before the digital age. But then, you had to reach out to the people you wanted an opinion from, and there were a limited number of people who could be trusted. After that, there was a time of survey forms where a large group of people were expected to voice their opinions and then, the decision would be taken based on the majority. However, people felt bored and tired of filling forms. This was to an extent that the organizations who were interested in those feedbacks started paying the reviewers for filling out the surveys. This was obviously, not a great strategy as the customers were being forced into writing reviews rather than stipulating their own thoughts.

Now, with the advancement in NLP technologies, organizations can mine the web for relevant data and then analyze it to come up with the information essential for making appropriate decisions. This kind of decision making approach is now being used frequently by producers of goods to identify acceptance of their products by the masses and take production decisions accordingly. Similar approach can also be used to make investment decisions. This is the reason why there have been at least 40-60 start-up companies in the space in the USA alone. Many big corporations have also built their own sentiment systems, e.g., Microsoft, Google, Hewlett-Packard, SAP, and SAS.

### 1.2.2 *Prediction*

Here is the most interesting application of sentiment analyzers. Public inclination is responsible for a lot of important happenings. This ranges from political elections to box office collections. If we can understand the sentiment of the masses, we may be able to predict successfully the outcome of such public-influenced events.

Sentiment analyzers have been used to successfully predict electoral results, stock markets and movie revenues in the past. Some researchers have also used the setiments behind the text to predict the gender of the person generating the text. A more recent example comes from the UK, where Tweview attempted to predict the winner of the 2013 X Factor using sentiment, volume, and a lot of other social factors. While they predicted Nicholas McDonald as the winner of 2013 X Factor, he ended up second, with Sam Bailey winning. However, the predictions made all throughout the show were very close to the actual results, revealing the potential of sentiment systems. [2]

## 1.3 THWARTING - A PARTICULARLY IMPORTANT PROBLEM

Thwarting is defined by Pang et al., (2008) as follows:

*Thwarted expectations basically refer to the phenomenon wherein the author of the text first builds up certain expectations for the topic, only to produce a deliberate contrast to the earlier discussion.*

Consider this example:

**"The movie didn't have a story at all. The music was dull. Also, the direction could have been a lot better. But I lovvved the movie because Caprio is in it! I love Caprio!"**

Notice that the above movie review is negative for most part. But as we move to the last segment of the sentence, there is a shift in polarity from negative to positive and the positive dominates. So, if we are not dealing with the aspect level analysis, the overall polarity of this review must be positive.

For computational purpose, thwarting is defined as:

*The phenomenon wherein the overall polarity of the document is in contrast with the polarity of majority of the document.*

The problem at hand is detecting the correct polarity of such thwarted documents. This is easy if we can detect somehow whether the document is thwarted or not (then we can deal with it accordingly). This leads to a simpler problem which is to identify whether a given document is thwarted or not. This is called as the problem of detecting turnarounds in sentiment.

To handle this problem, an aspect level of analysis is required. If we treat out target as a combination of subtargets arranged in a hierarchical fashion (with the original entity as root) , thwarting can be considered as the phenomenon of polarity reversal at a higher level of the hierarchy compared to the polarity at lower level. Thus, the

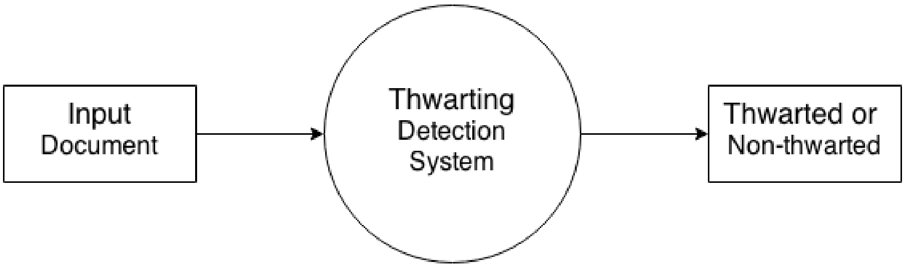

Figure 1: Thwarting detection system

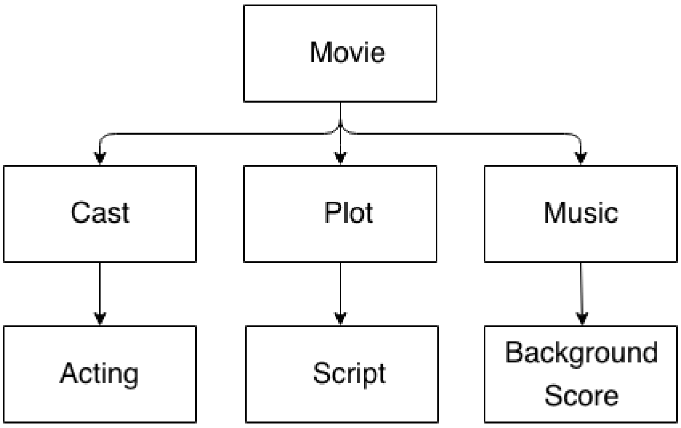

Figure 2: Ontology creation for a movie - an example

proposed solution starts with constructing a domain ontology of the system under consideration.

For example, if we are detecting turnarounds in movie reviews, we will have to find the ontology of a movie. This can be done manually as well as automated, using techniques like Latent Dirichlet Allocation (LDA). LDA can learn the key features of a movie (e.g. cast, acting, music etc.) from a movie review corpus. If LDA misses out on some features, human intervention may be required to provide the necessary components. The obtained list of all components should be then arranged into a proper hierarchy by a human annotator.

The naive approach to detect thwarting after this would be to check the sentiment score obtained at all nodes in the ontology tree. If different levels exhibit different polarity, the document is likely to be thwarted. In procedure, this is done in three distinct steps:

(a) **Dependency parse:** For every input document, it has to undergo a dependency parsing first. This is essential to identify all the adjective-noun relations.

(b) **Obtaining polarity at nodes:** Now, for all extracted nouns, we check if it is present in the domain ontology. If it is, the score for the corresponding node will be the determined by the associated adjective word.

(c) **Thwarting detection:** Now that we have sentiment scores for all nodes in the ontology, we can summarize the sentiment polarity at each level of the ontology. If there is a polarity reversal between levels, we can judge the document as thwarted.

However, this method fails to perform well in practical situations, since nodes at the same level may not be equally important for the sentiment of the document. For example, For example, say in a camera ontology, the body and video capability might be subjective whereas any fault in the lens or the battery will render the camera useless, hence they are more critical. Therefore, a relative weighing is required between all features of the ontology.

The idea used here is of percolated polarities. This means that polarity at a node is its polarity in the document along with the polarities of all its descendants. This is based on the intuition that every word contributes some polarity to its parent node in the domain ontology. We can also define controlled percolation, wherein the value added for a particular descendant is a function of its distance from the node.

For example,

$$\boxed{p(movie) = p(movie) + p(cast)/2 + p(plot)/2 + p(music)/2 + p(acting)/4 + p(script)/4 + p(background - score)/4}$$

A more complicated approach involves using the above as preprocessing to obtain features and then pose the problem of thwarting detection as a classification problem. Ramteke et al. (2013) have used SVM classifier on such features to achieve an AUC score as high as 81%. [4]

### 1.4 ROADMAP

So far in this report, the importance of sentiment analysis as a research area has been established and its numerous applications stipulated.

More importantly, it is shown why sentiment analysis is a challenging problem. In the chapters to follow, I have discussed some very useful approaches that may help in the direction of solving this problem of sentiment analysis on text in general and on social media, in particular. Chapter 2 focuses on posing sentiment analysis as a classification problem and working out a machine learning based solution to the problem at hand. Chapter 3 discusses the importance of social networks as a domain for applying sentiment analysis strategies and what are the special challenges there. Lastly in Chapter 4, solutions specific to social media domain are discussed and it is demonstrated how we can utilise the power of social networks for obtaining better solutions for our problem.

# 2 SENTIMENT ANALYSIS AS A CLASSIFICATION PROBLEM

The essence of sentiment analysis is to label a piece of text (or aspect) as positive, negative or neutral. In modern times, after the unprecedented level of development in the machine learning area, we can easily formulate the problem as a classification problem and solve it with machine learning techniques. This has proved to be an extremely successful approach in the past. The most important task here is to come up with the right features, and classification tools will do the rest.

However, one of the first attempts to solve sentiment analysis as a classification problem came as early as 2002 by Turney. [1] This was an unsupervised approach.

## 2.1 UNSUPERVISED CLASSIFICATION

Turney identified that there are some fixed syntactic patterns that are more likely to express sentiments. The syntactic patterns are composed based on part-of-speech (POS) tags. Turney took the phrases that conformed to these patterns as his indicators of sentiment.

The algorithm consists of three steps:

### 2.1.1 *Extracting of sentiment phrases*

**I_PRP had_VBD a_DT long_JJ day_NN**

In this example, "long day" seems to contain a sentiment.

Turney handcrafted some POS tag sequences that should be extracted from the text to analyze for sentiment. The exhaustive list is given in the following table:

| First Word | Second Word | Third Word (Not Extracted) |
|---|---|---|
| JJ | NN or NNS | anything |
| RB or RBR or RBS | JJ | not NN or NNS |
| JJ | JJ | not NN or NNS |
| NN or NNS | JJ | not NN or NNS |
| RB or RBR or RBS | VB, VBD, VBN or VBG | anything |

And as it should be, "long day" conforms to the first pattern, and hence must be extracted.

### 2.1.2 *Calculating sentiment score of extracted phrases*

For calculating the sentiment scores of extracted phrases, a measure of closeness (or distance) between two phrases is required. We can then find the closeness of the extracted phrase from two reference phrases - one positive and one negative. The closer reference phrase will decide the sentiment of this new phrase.

A measure of distance is Pointwise Mutual Information (PMI). It is given as:

$$\boxed{\mathrm{PMI}(\mathrm{term}_1, \mathrm{term}_2) = \log_2\left(\frac{\Pr(\mathrm{term}_1, \mathrm{term}_2)}{\Pr(\mathrm{term}_1)\Pr(\mathrm{term}_2)}\right)}$$

As evident, this measure sees closeness between two phrases as the number of times they have occured together in some text. Now, to calculate the sentiment orientation of a given phrase, we calculate its PMI with two reference phrases - excellent and poor.

$$\boxed{\mathrm{SO}(\mathrm{phrase}) = \mathrm{PMI}(\mathrm{phrase}, \text{`excellent'}) - \mathrm{PMI}(\mathrm{phrase}, \text{`poor'})}$$

If the phrase is closer to "excellent" than to "poor", SO(phrase) will turn out to be a positive value. In such a case, the new phrase can be labelled as positive, otherwise negative.

### 2.1.3 *Assigning label to the text*

For the given text, the average SO score over all the extracted phrases is calculated. Then, depending on the final value, we can label the entire text as positive or negative.

This approach seems like a naive approach. However, it has shown classification accuracies on reviews from various domains ranging from 84% for automobile reviews to 66% for movie reviews. A little advancement over this approach uses sentiment lexicons to determine the SO score of the phrases. Rest of the procedure is exactly the same. Sentiment lexicons are precomputed dictionaries containing the sentiment scores of common phrases.

## 2.2 SUPERVISED CLASSIFICATION

Supervised machine learning techniques have been used for solving problems in NLP for a very long time now. They find application in almost all areas in Natural Language Processing, for example, topic-based text categorization. Thus, it was proposed that machine learning techniques should be useful in sentiment analysis too, which was found to be true.

The following technique basically models sentiment analysis as a topic-based categorization problem where positive and negative are two topics, and we have to find which topic a given text belongs to. For applying machine learning strategies, Then, each document $d$ is represented by the document vector $d := (n_1(d), n_2(d), ..., n_m(d))$. Here, $n_i(d)$ denotes the frequency of occurence of feature $f_i$ in document $d$. Note that we have considered $n$ different features.

### 2.2.1 *Naive Bayes*

The probabilistic approach to the problem goes like this:

Let $c*$ denote the actual class (positive or negative) of the given text document $d$. Then, we formulate the following argmax expression:

$$\boxed{c* = argmax_c P(c|d)}$$

By Bayes rule:

$$c* = argmax_c \frac{P(c)P(d|c)}{P(d)}$$

Since P(d) is common for all c, it plays no role in deciding the resulting c.

$$c* = argmax_c P(c)P(d|c)$$

Now, $P(d|c)$ can be determined easily by breaking it down into the feature representation.

$$P(c|d) = \frac{P(c) \prod_{i=1}^{m} P(f_i|c)^{n_{i(d)}}}{P(d)}$$

With unigram and POS tag features, Naive Bayes have reported upto 81.5 accuracy.

### 2.2.2 *Maximum Entropy*

The argmax expression is the same for maximum entropy classification but the probability $P(c|d)$ is given differently:

$$P(c|d) = \frac{1}{Z(d)} exp \sum_{i}^{m} \lambda_{i,c} F_{i,c}(d, c)$$

Note that $F_{i,c}(d, c')$ fires if and only if $c' = c$ and $n_i(d) > 0$. That is, when a particular feature $f_i$ is known to occur in the document and the documentâs sentiment is hypothesized to be c (negative or postive). Importantly, unlike Naive Bayes, MaxEnt makes no assumptions about the relationships between features, and so might potentially perform better when conditional independence assump- tions are not met.

The parameter values λ are set so as to maximize the entropy of the distribution subject to the constraint that the expected values of the feature/class functions with respect to the model are equal to their expected values with respect to the training data - that is the model should make the fewest assumptions about the data while being consistent with it.

### 2.2.3 *Support Vector Machines*

SVM has proved to be the most successful in other NLP applications already. It is also the most widely used classification algorithm used today. SVMs are large-margin classifiers. In the binary-classification case, the basic idea behind the training procedure is to find a hyperplane, that not only separates the document vectors in one class from those in the other, but for which the separation, or margin, is as large as possible.

$$\boxed{w = \sum_{j} \alpha_j c_j d_j, \alpha_j > 0}$$

The required hyperplane is given by vector $w$. Support vectors are those documents $d$ for which the $\alpha$ value turns out to be greater than zero.

These were the broad classification strategies that have proved very useful in sentiment analysis. Although the performance is nowhere near to perfect, it is still remarkable. Moreover, there are deep learning based techniques too, which have proved to be even more useful for sentiment analysis, achieving accuracy as high as 84%. [5]

## 2.3 INFORMATION RETRIEVAL BASED SENTIMENT ANALYSIS

This is also an unsupervised approach like Turney's. Remember we discussed that sentiment words can be ambiguous, meaning differently in different contexts. Determining contextual polarity of sentiment words is thus a challenging task. Here, we have discouraged the use of hand-crafted rules or sentiment lexicons. Rather, the problem is formulated as an information retrieval problem and solved using information retrieval approach. [6]

Consider this:

**Pizza is cold.** **Beer is cold.**

As we can see, the same word is used for two different sentiments and there are no other differences in the two sentences to tell the sentiments apart for the system. The solution is very simple and obvious. It goes like this:

For a sentiment word $a$, two vectors of different contexts need to be created, one each for the positive and the negative set. When word $a$ appears during test phase, the approach treats the current word context vector as query and the prepared context vectors as documents. The polarity of $a$ is decided by the document which is retrieved after providing the input query.

The context feature vector is created as follows:

First of all, just like Turney's unsupervised classification, the relevant phrases have to be extracted. For this, we concentrate on nouns. The first task is to locate all nouns that appear as governing words for a dependency relation. Here, a filter is applied to select only the nouns that interest us i.e. only the nouns belonging to that particular domain. The algorithm can be listed in the following steps:

1. For each governing word (noun), dependency triple of the form $\mathsf{DepRel}(n, a)$ is to be extracted. Dependency relation can be an adjectival modifier (amod), nominal subject (nsubj) or relative clause modifier (rcmod). For example, **nsubj(pizza,hot)**

2. For each adjective instance $a$ extracted in this way, all triples where $a$ appears as dependent must be extracted. For each adjective occurrence, the following information is recorded:

   a) negation (1, if adjective is negated; 0, if adjective is not negated)
   b) dependency relation of adjective with its governing noun (amod, nsubj or rcmod)
   c) adjective lemma

   These three pieces of information form a adjective pattern (AdjP), e.g., **0; amod; better**. A context feature vector is built for each of these patterns. The reason why building vectors is essential as opposed to just adjective lemmas is that, firstly, we want to differentiate between the negated and non-negated instances, and, secondly, we want to capture the syntactic usages of the adjective. This is important because adjectives occurring in a postmodifier position tend to be used more in evaluative manner, and are thus, more important for our analysis compared to those used in premodifier position (**drink is cold** vs **cold drink**).

3. For each adjective instance thus created, represented as **negation; dependency relation; lemma**, all dependency relations that contain it must be extracted. Each of them is transformed into a context feature f of the form: **lemma; Part Of Speech (POS); dependency relation**. For instance, if adjective **cold** occurs in dependency triple nsubj(drink, cold), the following feature is created - **drink; NN; nsubj**. For each feature, its frequency of co-occurrence with the adjective pattern is recorded.

4. After we have generated vectors for all $AdjP$ patterns extracted from the positive set and the negative set separately, we can now determine the polarity of a specific adjective occurence. In this approach, the sentence (containing the adjective) only is used to generate the vector in a similar manner as before. This vector is called $EVal_{AdjP}$. The pairwise similarity of $Eval_{AdjP}$ with $posV_{AdjP}$ (vector constructed from positive examples) and $Eval_{AdjP}$ with $negV_{AdjP}$ (vector constructed from negative examples) is computed. If similarity with $posV_{AdjP}$ is higher, it is categorized as positive, and as negative if similarity with $negV_{AdjP}$ is higher.

Computing similarity is where the role of information retrieval comes in. The vector $Eval_{VAdjP}$ of a specific adjective occurrence AdjP, whose polarity we want to determine, is treated as the query, while the two vectors $posV_{AdjP}$ and $neg_{VAdjP}$ created from the positive and negative training sets respectively, are treated as documents. For the purpose of computing similarity we use Query Adjusted Combined Weight (QACW) document retrieval function:

$$\mathit{Sim}(\mathit{EvalV}_{\mathrm{AdjP}}, V_{\mathrm{AdjP}}) = \sum_{f=1} F \frac{TF(k_1+1)}{K+TF} \times QTF \times IDF_f$$

where,

F : the number of features that $\mathit{EvalV}_{\mathrm{AdjP}}$ and $V_{\mathrm{AdjP}}$ have in common
TF : frequency of feature f in $V_{\mathrm{AdjP}}$
QTF : frequency of feature f in $\mathit{EvalV}_{\mathrm{AdjP}}$
K : $k_1 \times (1-b) + b \times \frac{DL}{AVDL}$
k1 : feature frequency normalization factor
b : $V_{\mathrm{AdjP}}$ length normalization factor
DL : number of features in VAdjP
AVDL : average number of features in the vectors V for all AdjP patterns in the training set (positive or negative)

Frequency) of the feature f is calculated as $IDF_f = \mathit{log}(\frac{N}{n_f})$, where, $n_f$: number of vectors V in the training set (positive or negative) containing feature f; N: total number of vectors V in the training set.

A polarity score of AdjP is calculated for both positive and negative sets:

$$\mathit{PolarityScore} = \alpha \times \mathit{Sim}(\mathit{EvalV}_{\mathrm{AdjP}}, V_{\mathrm{AdjP}}) + (1-\alpha) \times P(\mathrm{AdjP})$$

where P(AdjP) is calculated as number of occurrences of AdjP in the set (positive or negative) divided by the total number of occurrences of all AdjP patterns in this set. If Polarity Score is higher for the positive set, the polarity is positive, and if lower, it is negative.

# 3 SENTIMENT ANALYSIS ON SOCIAL MEDIA

In the past few years, there has been a huge growth in the use of microblogging platforms such as Twitter and social networks like Facebook. People now spend more time on social networks than any other thing on the internet. Social networks have provided them the platform that not only allows them to connect with friends but also to express themselves, voice their opinions, broadcast themselves over the internet and build reputation. Thus, the content generated on these social networks is highly diverse. In the current scenario, it will not be an overstatement to say that people post about anything and everything. According to a report from Business Insider, Facebook has 1.2 billion monthly active users. With such an amount of activity, there is hardly any information about the people that the social networks are missing out on. Spurred by the growth of social networks, companies and media organizations are increasingly seeking ways to mine social media for information about what people think and feel about their products and services.

While we establish that social media analysis is crucial for latest trends and feedbacks, it is to be realized that sentiment analysis on social media comes with its own complexities. The expression of sentiments is different in social media than say, movie reviews. This is due to the informal language, inexplicit context and message-length constraints (esp. Twitter) of social network domain. As a result, features such as automatic part-of-speech tags and resources such as sentiment lexicons are less useful for sentiment analysis in social networks. Another challenge of microblogging and social media posting is the incredible breadth of topic that is covered. In social media, hardly ever will we find people adhering to a single issue. Usually they touch multiple areas and to recognize what is information and what is noise is the first challenge that sentiment systems have to face. To be able to build systems that mine the likes of Facebook, Twitter, LinkedIn etc, the conventional sentiment analysis approaches won't work. In this chapter we will see in detail why. In the next chapter, the special tools and solutions are discussed that will make sentiment analysis on social networks viable.

## 3.1 TAXONOMY OF SOCIAL NETWORKS

Social media has become the new corpus for sentiment analysis. However, different kinds of networks have their own characteristics and purpose, and may demand different treatment. Thus, it becomes important to understand the different types of social networks and the relationships among them.

### 3.1.1 *Social Connections*

These are the services that connect people with other people of similar interests and background, usually to the people they know outside the web also. Usually they consist of a user profile, where a user updates his latest activities. People from his/her social group can see them, and communicate back. Sometimes these networks also provide the ability to setup groups or personal chat, on case one wants to go secretive for a while. The most popular are Facebook, Google+ and MySpace.

LinkedIn also falls in the same domain although it differs from the given examples in the sense that the language here is more formal.

### 3.1.2 *Microblogging*

Services that focus on short updates that are pushed out to anyone subscribed to receive the updates. These networks are used primarily by celebrities and brands that seek to build a reputation by gathering followers. The most popular is Twitter.

### 3.1.3 *Social News*

Social news are collaborative content building networks where every user contributes to the generation and organization of information. Services like these allow people to post various articles of their own or otherwise or links to outside articles and then allows itâs users to vote on the items. The voting is the core social aspect as the items that get the most votes are displayed the most prominently. The community decides which news items get seen by more people. The most popular are Digg, Reddit and Quora.

### 3.1.4 *Media Sharing*

Media sharing services primarily provide a platform to share and distribute the likes of videos, photos and other forms of media on the internet. Even these kind of services are not bereft of social network structure. For example, in Instagram, you can share a photo with friends, just like on Facebook, and they can then like or comment or you can just broadcast your photo for everybody to see.

### 3.1.5 *Discussion Forums*

Discussion forums are more inclined towards a specific topic of discussion. They usually have threads of conversation about a particular subject or topic. If you have something different to say, a new thread should be created. For example, StackOverflow. However, not all such forums keep track of user information. In such cases, the social network structure doesn't apply.

Given the above classification, it is to be understood that there are non-trivial overlaps between most of the mentioned categories. For instance, Facebook provides a kind of microblogging type feature in the form of statuses. [11]

## 3.2 SOCIAL NETWORK ANALYSIS APPLICATION IN INDUSTRY

Awareness is one of the key ingredients to success in industry. If one needs to survive in the market, he/she must know the demands of its consumer as well as the response to its products or services at all times. Ever since sentiment analysis on social media has been possible, companies have become very enthusiastic to use this for themselves. The advantages are multifarous. One, they get to connect to their customers directly. Second, they are always getting an immediate feedback on whatever they do or fail to do. Moreover, since no data on social networks is hidden whatsoever, they can also find out about their rival companies and plan accordingly.

### 3.2.1 *Brand Assessment*

For the past few years, a lot of startups emerged offering sentiment analysis services to companies. Also, the companies are adopting these technologies at a great pace. Many of the established firms (e.g. Google, Adobe) have developed their own systems to do the same.

The idea is simple. The more informed you are, the better decisions you can take.

### 3.2.2 *Social Media Marketing*

Facebook conducted a massive psychological experiment on 689,003 users, manipulating their news feeds to assess the effects on their emotions. The details of the experiment were published in an article entitled "Experimental Evidence Of Massive-Scale Emotional Contagion Through Social Networks"published in the journal Proceedings of the National Academy of Sciences of the United States of America. In simple words, Facebook has the ability to make you feel good or bad, just by tweaking what shows up in your news feed.

However, this was not the first time something of this kind was seen. The idea of emotional contagion is much older, which can be stated simply as "The sentiment of a social media author influences the sentiment of people in his social media network". Now, from a company's prespective this can be very useful. If they can sway the emotion of people towards their side, it can do wonders for them. This is what has given rise to a new form of marketing called social media marketing. Hence, there is immense competition between companies to mark their influence on the social media.

## 3.3 CHALLENGES IN SOCIAL MEDIA SENTIMENT ANALYSIS

Sentiment analysis on social media is not straightforward as say, on movie reviews. There is a whole set of problems that researchers have faced. Some of them are:

### 3.3.1 *Lack of Specificity in Context*

People don't conform to any standards or guidelines while writing on social networks. They just go by the famous phrase - "What's on your mind?". Since there is no control whatsoever, it usually becomes very difficult to extract the required and leave the rest. Technically, the miners have to deal with a lot of noise here.

For example:

**I haven't studied a word since I brought my new Sony Playstation home. My mom hates it!**

Here, even if the author of the post is talking about Sony Playstation, this is not a valid review for the same.

### 3.3.2 *Highly informal language*

Language on the social networks is not something we can call English , or any other language for that matter. People post according to their own comfort, at their own leisure. Also, there is something called intensifiers, which is a technical name for purposefully misspelling a word to increase its intensity.

For instance,

**I totallllyyyy looovve my new Playstation!**

Note that these kind of things are difficult to handle. Still, they cannot be ignored since they are major carriers of sentiment.

### 3.3.3 *Social network specific entities*

Consider this example,

**I support #ProteaFire.**

Now, without the knowledge that #ProteaFire was a hashtag circulated to support Republic of South Africa in Cricket World Cup'15, this statement hardly makes sense. But as we have this information, this post suddenly becomes meaningful. When we are working in the social media domain, these kind of entities are our primary target, in fact.

## 3.4 OPINION SPAMMING

A key feature of social media is that it enables anyone from anywhere in the world to freely express his/her views and opinions without disclosing his/her true identity. However, this has a price to pay. All our effort in analyzing the sentiment from social media assumes that the opinions or posts expressed are genuine and actually belong to a person. Unfortunately, people with malicious intents can game the system give and post fake opinions to promote or to discredit target products, services, organizations, or individuals. The worst part is that there are commercial companies that are in the business of opinion spamming. Detecting these spams is extremely challenging and out the scope of NLP itself. [1]

However researchers have tried to find reviewer-specific features that may help in classifying a user as trustworthy or not. Simply stating, looking at the behaviour of a reviewer, we can figure out whether the author is a spammer. However, we refrain ourselves from going into details here.

# 4 SOCIAL MEDIA SPECIFIC APPROACHES

Since social networks posed special problems in sentiment analysis, it became important to come up with special approaches to deal with them. Ever since the sentiment analysis research on social media started, researchers have been trying to combat these problems by suggesting approaches and coming up with the tools that are more suited to this domain. We know that, in a classification-problem setting of sentiment analysis problem, the choice of features play the most important role. Since the characteristics of social media text are very much different from regular plain English (or any other) text, the features would have to be re-engineered. This is only one of the many adaptations that our sentiment analysis systems have to undergo if they have to be applied to social networks.

Redesign of features, however, is not enough. We know that NLP tools have always been of great help when it comes to sentiment analysis. Tools like lexicons, wordnets, parsers etc. have greatly contributed to the design of sentiment analysis systems. The problem is, these tools fail miserably when applied to social networks. What we need is a redesigning of these tools as well, so as to make it suitable to our domain of interest, or to come up with alternative approaches.

## 4.1 CLASSIFICATION FEATURES

To understand what features are important for sentiment classification in social media domain, one needs to know about the practices and conventions of modern social networking. One cannot afford to be ignorant to the hieroglyphs and the abbreviations that the users are generating everyday if he/she really wants to understand their sentiments. The state-of-the-art system in Tweet Sentiment Analysis, NRC-Canada have used the following features in their system: [7]

1. **Word ngrams and Character ngrams:** Character n-grams are added specially because word n-grams are not always useful to capture the language (because of frequent misspelling)

2. **All caps, Elongated words and Punctuations:** These have special meaning in social networks. All caps are used to denote

’yelling’ or abbreviations, both of which are useful for sentiment analysis. Elongated words (also called intensifiers) are diligent misspellings of known words to emphasize on their emotions. Punctuations also perform the same function. They are not cameos of language grammar here in this context. A lot of repeated punctuations (question or exclamation) is a very often sight in social media, which denotes intensity of emotion. For example, **OMG I am soooo happy!!!!!!!!**

3. **POS tags:** Using POS tags as a feature is not new to social media. However, the regular POS tagger can’t be used here, rather a special tool created by Carnegie Mellon is used.

4. **Negation:** The number of negated contexts in the text. It has been shown that if we handle negation well, we can increase the sentiment accuracy to a great extent.

5. **Hashtags and Emoticons:** These are the most powerful features that social networks have to offer. Emoticons, most of the times, very much sum up the emotion of the text they are contained in (of course, only if the author knows how to use them correctly) whereas hashtags act as predefined labels that automatically put the text in some class of tweets. We can then, find the polarity of the hashtag and it is very much likely that this will be the polarity of the given text.

6. **Lexicon features:** Lexicon features make use of a pre-built lexicon that contains sentiment scores of various phrases. Here, we look for things like the number of words in the text with a positive score in the lexicon, or the average score of the text, as given by summation of individual scores of all phrases obtained from the lexicon etc.

These kind of features, on an SVM classfier have successfully given an accuracy as high as 88.93% over tweets.

## 4.2 DEPENDENCY PARSER FOR TWEETS

As already discussed, NLP tools have alwyas been very helpful in sentiment analysis. One such tool is the dependency parser. As is true for most of NLP techniques, this also cannot be used directly in social network domain. In this section, we will see why. Also, the adaptations that need to be made are discussed. [9]

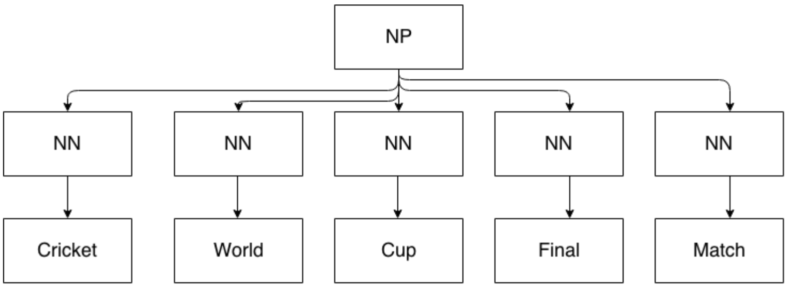

Figure 3: A Constituency Parse

### 4.2.1 *Challenges*

1. Tokens may or may not have syntactic functions. So, their identification is important. Consider this:

   **Another fine effort from @IamRaina**

   The token @IamRaina is a part of the sentence structure and hence has a syntactic function. On the other hand, in the example to follow, the same token does not have a syntactic function.

   **Suresh Raina has always been agile on the field and hard hitting with the bat. @IamRaina**

2. Internal analysis of Multi-word Expressions [MWE] is not very important for the task at hand. Annotators should not expend energy on developing and respecting conventions (or making arbitrary decisions) within syntactically opaque units. e.g. World Cup 2015 (Proper Nouns), as well as (Connectives), out of (Prepositions) etc.

3. Tweets need not be a single sentence. They may contain multiple utterances, each with its syntactic root disconnected from the others.

4. Noun phrase internal structures are usually poorly represented by most parsers. Richer treatment of such structures is required in the dependency parser we seek to design. For example, the phrase "Cricket World Cup Final Match"will have the constituency parse tree as shown above.

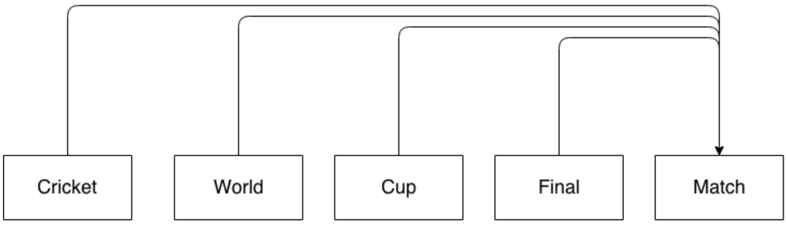

Figure 4: Equivalent Dependency Parse

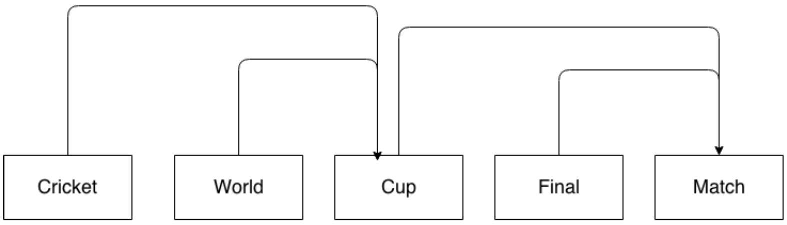

Figure 5: The richer dependency parse

This when converted to a dependency parse will generate something like Figure 4.

However the accurate representation of the dependency parse should be as given in Figure 5.

### 4.2.2 *Design of the dependency parser*

This section deals with how the traditional dependency parsing algorithm is adapted for tweet dependency parsing. The required parse tree is given by the following formula:

$$Parse^{\star}(x) = argmax_{y \epsilon Y_x} W^T g(x, y)$$

where,

- x : input sentence
- $Y_x$ : Set of all possible dependency parses of x
- g : feature vector representation

- W : parameter vector of weights, learnt by data

Decomposition of features into parts is critical to the performance of the parser. Let us consider there are $n$ different tokens in our sentence. Then, the set of all possible dependency arcs is given by $A = (h, m) : h\epsilon(0, n), m\epsilon(1, n)$

Then, Y, the set of all dependency parses is given as $Y \subset (0, 1)^{|A|}$. $g$ can be decomposed into features over each branch in the dependency parse. For example, consider:

$$very \leftarrow good \leftarrow food$$

$g(x, y)$ can be calculated as $score(good \rightarrow very) + score(food \rightarrow good)$. Now, $score(w_h, w_m)$ is given as a weighted sum of score over individual features.

$$score(w_h \rightarrow w_m) = \phi(x, < h, m >)\theta$$

Here $\phi$ is a feature vector and $\theta$ is the vector of weights.

### 4.3 EXPLOITING THE SOCIAL GRAPH

So far in this report, all problems in sentiment analysis were tackled at the document level or aspect level (thearting). However, user level sentiment analysis is equally important and interesting. Moreover, it is easier to perform since we are making a bold assumption that a user always tweets with the same sentiment. This is not true in practice. But when we restrict our domain to a specific topic, this type of behaviour is known to emerge. For example, if a person is inclined towards or associated with an organization, he will mostly (if not always) talk positively about that particular organisation. Thus, user level sentiment analysis becomes valid.

The approach of using the social graph to determine user sentiments is based on a psychological principle called "homophily". There is a popular phrase in English that defines the underlying idea - "Birds of a feather flock together."In sentiment analysis, this would

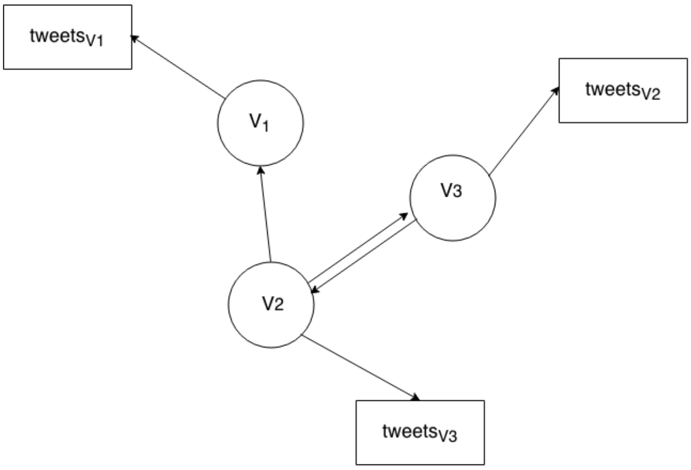

Figure 6: A sample directed follow graph

mean that people who interact more on a social network among each other are likely to hold a similar sentiment on a particular topic. Of course, for this approach to work, we need to start with some seed set of people whose sentiments are already known. [10]

In a Twitter environment, the relationship between users can be extracted in the form of four different graphs:

1. Directed follow graph: user $v_i$ follows $vj$ ( $v_j$ may or may not follow $v_i$ in return )

2. Mutual follow graph: user $v_i$ follows $v_j$ and user $v_j$ follows $v_i$

3. Directed @ graph: $v_i$ has mentioned $v_j$ via a tweet containing @$v_j$ ( $v_j$ may or may not mention $v_i$ in return )

4. Mutual @ graph: $v_i$ has mentioned $v_j$ via a tweet containing @$v_j$ and vice versa

Let us assume the number of users be V. Let $y_{v_i}$ be the label for user $v_i$, and Y be the vector of labels for all users. We make the Markov assumption that the user sentiment $y_{v_i}$ is influenced only by the sentiment labels of tweets belonging to that particular user and the sentiment labels of the immediate user neighbors.

Probability of a particular Y vector will be given as:

$$\log P(Y) = \sum_{v_i \epsilon V} \left( \sum_{t \epsilon tweets_{v_i}} \mu_{k,l} f_{k,l}(y_{v_i}, y_t) + \sum_{v_j \epsilon neighbours_{v_i}} \lambda_{k,l} h_{k,l}(y_{v_i}, y) \right)$$

where,

indices k, l range over the set of sentiment labels (0,1).
$f_{k,l}$ and $h_{k,l}$ are feature functions, and $\mu_{k,l}$ and $\lambda_{k,l}$ are parameters representing impact.
Z is the normalization factor

We can keep changing f,h,$\mu$ and $\lambda$ to meet different kind of adaptation needs. For example, $f_{k,l}(y_{v_i}, y_t)$ will fire only if $y_{v_i} = k$ and $y_t = l$. Even after that, we can have different values of f depending upon whether $v_i$ is a labeled user or not. We can also tweak $\mu$ as $\mu_{1,0} = 0$. This means that we are giving no weightage to the case where a positively labeled user posts a negative tweet. (Assuming 1 as positive and 0 as negative.)

# CONCLUSIONS

Sentiment analysis is an emerging field with a thrilling potential. It, along with the power of social networks, can harness a great deal of unseen and useful information. Since this information is coming from people, it allows the providers to be in touch with their consumers at all times, understanding their demands and grievances. Sentiment analysis systems, thus, seek to create a world where the voice of the people is given a status as never before and their needs are always understood, wherever and howsoever, they may express it.

It being established that sentiment analysis systems can achieve a lot, some doubt the future of research in this direction, saying that sentiment systems can never achieve an accuracy at which they can reliable enough to be instrumental to global change. The realization to be made here is that the gold standard here is not a full 100%.

Sentiment is humanly. So, the performance of sentiment analysis systems has to be measured against a human's ability to do the same. Incidentally, there are studies which show that there is an agreement of about 70% among humans regarding sentiment on any subject. This is called as human concordance. So, this is the accuracy that sentiment systems target, which is very much achievable.

# BIBLIOGRAPHY

[1] Bing Liu, *Sentiment Analysis and Opinion Mining*. 2012.

[2] *"Social data gets the X-Factor."* http://www.brandwatch.com/2013/12/social-data-gets-the-x-factor/. Online; accessed 17-Apr-2015.

[3] *"Deeply moving: Deep learning for sentiment analysis."* http://nlp.stanford.edu/sentiment/. Online; accessed 24-March-2015.

[4] A. Ramteke, A. Malu, P. Bhattacharyya, and J. S. Nath, "Detecting turnarounds in sentiment analysis: Thwarting.," in *ACL (2)*, pp. 860–865, The Association for Computer Linguistics, 2013.

[5] B. Pang, L. Lee, and S. Vaithyanathan, "Thumbs up? sentiment classification using machine learning techniques," in *Proceedings of EMNLP*, pp. 79–86, 2002.

[6] O. Vectomova, K. Suleman, and J. Thomas, "An information retrieval-based approach to determining contextual opinion polarity of words," in *Proceedings of the 36th European Conference on Information Retrieval (ECIR), Amsterdam*, 2014.

[7] S. M. Mohammad, S. Kiritchenko, and X. Zhu, "NRC-Canada: Building the state-of-the-art in sentiment analysis of tweets.," *CoRR*, vol. abs/1308.6242, 2013.

[8] E. Kouloumpis, T. Wilson, and J. Moore, "Twitter sentiment analysis: The good the bad and the OMG!," in *ICWSM* (L. A. Adamic, R. A. Baeza-Yates, and S. Counts, eds.), The AAAI Press, 2011.

[9] L. Kong, N. Schneider, S. Swayamdipta, A. Bhatia, C. Dyer, and N. A. Smith, "A dependency parser for tweets," in *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, 2014.

[10] C. Tan, L. Lee, J. Tang, L. Jiang, M. Zhou, and P. L. 0001, "User-level sentiment analysis incorporating social networks," *CoRR*, vol. abs/1109.6018, 2011.

[11] *"The six types of social media."* http://timgrahl.com/the-6-types-of-social-media/. Online; accessed 17-Apr-2015.