Sarcasm Detection Of Online Comments Using Emotion Detection

Shubham Rendalkar

Student, Department of Information Technology, MIT, Pune, India
rendalkarshubham@gmail.com

Chaitali Chandankhede

Asst. Prof, Department of Information Technology, MIT, Pune, India
chaitali.chandankhede@mitpune.edu.in

Abstract— **Sarcasm is a sophisticated form of sentiment expression where speaker express their opinions opposite of what they mean. Sarcasm detection and Emotion detection from social net- working sites has been a great field of study. With the growth of e-services such as e-commerce, e-tourism and e-business, the companies are very keen on exploiting emotion and sarcasm analysis for their marketing strategies in order to evaluate the public attitudes towards their brand. Thus efficient emotion and sarcasm modeling system can be a good solution to the above problem. This work aims at developing a system that groups posts based on emotions, sentiment and find sarcastic posts, if present. The proposed system is to develop a prototype that help to come to an inference about the emotions of the posts namely anger, surprise, happy, fear, sorrow, trust, anticipation and disgust with three sentic levels in each. This helps in better understanding of the posts when compared to the approaches which senses the polarity of the posts and gives just their sentiments i.e., positive, negative or neutral. The posts handling these emotions might be sarcastic too. The Sentiment & emotion identification module identifies the sentiment or emotion of the post by evaluating score of each word in the comment which is used by different sarcasm detection methods to detect sarcasm. The emotion identification module uses the lexical databases WordNet, SentiWordNet to find the right sentiment scores for the words with respect to each emotion. It also uses Sarcasm detection algorithms like Emoticon sarcasm detection, Hybrid sarcasm detection, Hashtag Processing, Interjection Word Start (IWT).**

Keywords— *WordNet, SentiWordNet, Hybrid sarcasm detection, Interjection Word Start (IWT)*

I. INTRODUCTION

There are different trends opening in the era of sentiment analysis, which analyze 'attitude and opinion people in social media, which including social sites like Facebook, Twitter, blogs, etc. The main aim of sentiment analysis is to identify the polarity (positive, negative or neutral) in a given text. Sarcasm is a special type of sentiment which have the ability to flip the polarity of the given text. Sarcasm is defined as 'the use of irony to mock or convey contempt'. Sarcasm is a sophisticated form of sentiment expression where speaker express their opinions opposite of what they mean. Sarcasm is a contrast between positive sentiment word and a negative situation [6]. What makes task of detecting sarcasm hard is

that even humans find it hard to understand them sometimes without prior knowledge of the topic. Example: "Oh! He out on a duck, what a legendry batsman". In this example, person use to express the positive sentiment (greatness) but overall tweet reflect negative sentiment toward the batsman. Unlike a simple negation, a sarcastic text, typically conveys a negative opinion using only positive words. Recognition of sarcasm is one of the most difficult tasks in natural language processing. The average human reader will have difficulty in recognition of sarcasm in twitter data, product review, blogs, online discussion forum, etc. Sarcasm is also very closer to lie in some context, making it more problematic and hard task. As user or author writes exactly opposite of what he means, this is similar in lying.

Emotion is a strong feeling deriving from one's circumstances, mood, or relationships with others. Any strong agitation of the feelings actuated by experiencing love, hate, fear, etc., and usually accompanied by certain physiological changes, as increased heartbeat or respiration, and often overt manifestation, as crying or shaking. Detecting emotion of comments or reviews of social sites is very important in different research areas. One of the application of Emotion detection is in sarcasm detection. It is also used to study point of views of customer of different businesses, employees of companies, opinion of online site user etc.

An emoticon/emoji is a pictorial representation of a facial expression using punctuation marks, numbers and letters, which is used to express a person's feelings or mood. Every emoji have its emotion, which is also used to express the feeling instead of writing the text. Emoticons also used to make sentence sarcastic.

There are different libraries used for the sentiment finding, for synset detection and to find set of cognitive synonyms. For Sarcasm detection and Emotion detection it uses lexical databases like WordNet, SentiWordNet, WordNet-Affect. WordNet is a large lexical database of English. Nouns, verbs, adjectives and adverbs are grouped into sets of cognitive synonyms, each expressing a distinct concept. Synsets are interlinked by means of conceptual-semantic and lexical relations [2]. SentiWordNet is a lexical resource used for opinion mining. SentiWordNet assigns to each synset of WordNet to get the sentiment scores which divide that word into three categories: positive, negative, neutral word [7].

WordNet-Affect is an extension of WordNet Domains, including a subset of synsets suitable to represent affective concepts correlated with affective words. The affective concepts representing emotional state and also represents moods, situations eliciting emotions, or emotional responses [7].

II. RELATED WORK

In recent times, research interest grew rapidly towards sarcasm detection in text. Many researchers have investigated sarcasm on the data collected from various sources such as tweets on Twitter, Facebook, Amazon product reviews, website comments, etc.

The identification of emotion on social media has gained lot of attention in recent years. Researchers have programmed the analysis of emotions in the text. It is done by using six basic emotions namely anger, fear, disgust, joy, surprise and sadness [1]. SenticNet is used for opinion mining which is built using common sense reasoning techniques along with emotion categorization model. Researchers used the combination of SentiWordNet and WordNetAffect to find the emotion in web based Content [3]. WordNet is used to find the similarity of text and also find emotion synsets. Words have different meaning depending on the context they appear which is not handled. Emoticons are also used for the sarcasm detection. One of the work states that Emoji's are not always a direct labeling of emotional content. For instance, a positive emoji may serve to disambiguate an ambiguous sentence or to complement an otherwise relatively negative text [2]. Latent Semantic Analysis (LSA) gives a vector space model that allows for a homogeneous representation of words, word sets, sentences and texts. The LSA space represents an emotion in different ways like: the vector of the specific word denoting the emotion (e.g. love), the vector representing the synset of the emotion e.g. (choler, ire), and the vector of all the words in the synsets labeled with the emotion [3]. Sequence labeling is uses as a learning mechanism for sarcasm detection in dialogue. Based on information available in our dataset it present new features. Comparison is done between two sequence labelers (SEARN and SVMHMM) with three classifiers (SVM with oversampled and under-sampled data, and Naıve Bayes) [4]. Another approach is to use of Novel Bootstrapping Algorithm that automatically learns lists of positive sentiment phrases and negative situation phrases from sarcastic tweets. SVM classifier is used for classifying tweets [5].

III. SYSTEM ARCHITECTURE

The block diagram of the entire system is shown in Figure.1. Facebook posts, Tweeter posts etc. are retrieved using graph API. Posts which contain non-English words are filtered out. The posts are tagged using Stanford POS tagger and features pertaining to the post such as hashtags, special phrases, PS tags, comments data are extracted from the posts. The emotion identification dictionary identifies the major sentiment of the post by finding the sentiment score for each

word in the post. The emotion identification module uses the lexical databases WordNet, SentiWordNet to find the right sentiment scores for the words with respect to sentence. The fuzzy union of all the sentiment score for the each keywords is taken to obtain the measure for the entire sentence and emotion of the sentence is find out. The emotion score of the sentence is normalized based on the number of words in the sentence. The same methodology is applied to each of the sentences to find the emotion of the whole paragraph and then to the entire post text.

The sentiment score found are provided to sarcasm detection module. This module has several sub modules namely Emoticon based sarcasm detection approach, Post-Comments based approach, Hybrid approach, Positive sentiment and Negative sentiment sarcasm detection & Pattern text match sarcasm detection. Every module has their significance to detect the sarcasm.
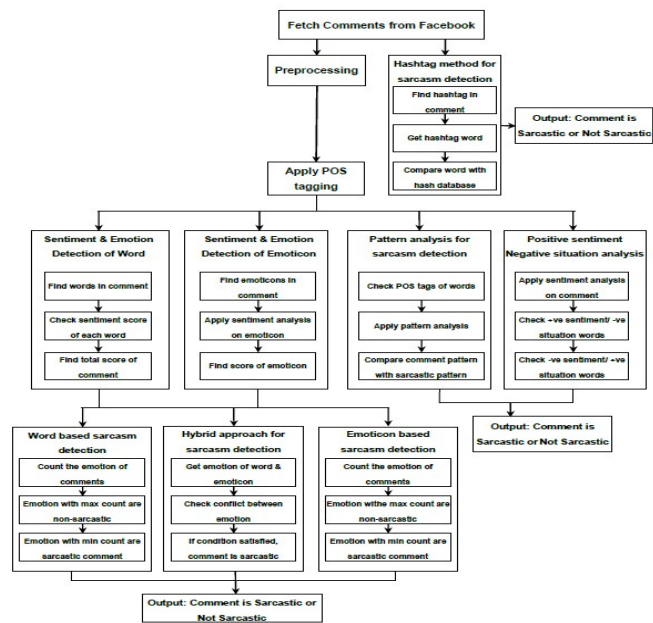

Fig. 1. System Architecture

IV. MODULE DESIGN

It explains detailed description of modules used in the system architecture.

A. Preprocessing:

This module to the processes like fetching Facebook data, classify that data, remove unwanted part and take only required one.

1) Fetching of Social data: The Graph API is used to get Facebook's data. It's a low-level HTTP-based API used to query data, post new stories, upload photos and a variety of other tasks. The application then uses the Graph API to fetch

the data from Facebook database. The whole data is analyzed and the required fields are extracted. The language in which the posts are written is analyzed and non-English posts are clean by using parser.

2) POS Tagger: POS tagger which takes a word from text as input and assign part-of-speech to each word as output. Stanford NLP is used for this with parsing process. Parsing is a process of analyzing grammatical structure, identifying its parts of speech and syntactic relations of words in sentences. The post text is analyzed and the Part-of-speech of every word in the text is found out using Stanford POS tagger. The POS is used to get verbs, nouns, adjectives, etc. When a sentence is passed through a parser, the parser divides the sentence into words and identifies the POS tag information.

3) Extracting Required Data: The dataset consists of a lot of noise such as http links, elongated words etc. The words which doesn't use in emotion or sarcasm detection, such as prepositions, pronouns, conjunctions are removed. From each keyword the Unicode characters are removed and stemming is performed.

B. Emotion & Sentiment Identifier:

It finds emotion of words and emoticons present in the comment.

1) Finding the Sentiment score of words: The first step in this phase is to find sentiment score of each word in the comment. To find sentiment score it uses SentiWordNet dictionary. The dictionary finds score for each word in the comment with the help of POS tagging. It only take only those word which are helpful for sentiment analysis. For that POS tagging is helpful because it identifies the part of speech words from comment. After providing score to each word addition is done to get final output that is negative score or positive score. According to that emotion of the sentence is provided to that comment that is positive emotion or negative emotion.

2) Finding Sentiment score & Emotion of Emoticons: In this step sentiment score and emotion of emoticon is find out. To do so, it uses emoticon library and SentiWordNet dictionary. There is a emoticon library which stores the all information about emoticon like, emoticon id, meaning, emoticon score (with the help of SentiWordNet dictionary), etc. For every emoticon there is a unique id called as emoticon id. Emoticon id is checked in that library to get its score and according to that sentiment and emotion of emoticon is find out.

C. Sarcasm Detection:

After finding the emotions, the sarcasm in the posts under each of these emotions are also detected using the following

methodologies which also makes use of the emotion model. In this work, we have designed four methodologies for identifying sarcasm:

1) Word Based Detection: In this method, first it gets the emotion of the comment with the help of SentiWordNet dictionary that is positive, negative or neutral. Then it will check the count of each emotion of comment. If the count of one of the emotion is maximum then other comments are sarcastic.

For ex. Consider 25 comments of one feed of a Facebook page. If we find 16 positive comments, 5 negative comments and 4 neutral comments. Here max count is 16 of positive comments then the sarcastic comments are 9, which are 5 negative comments and 4 neutral comments.

2) Emoticon Based Detection: Every post may or may contain emoticon but if it is present in the comment then it is used to detect the sarcasm. In emoticon based detection, it gets the emotion of the emoticon that is positive, negative or neutral. Then it will check the count of each emotion of comment. If the count of one of the emotion is maximum then other comments are sarcastic.

For ex. Consider 25 comments of one feed of a Facebook page. If we find 17 positive comments, 5 negative comments and 3 neutral comments. Here max count is 15 of positive comments then the sarcastic comments are 8, which are 5 negative comments and 3 neutral comments.

3) Hybrid sarcasm detection approach: This method combines the two approaches that is emoticon based and comment based to get the output. Text part of the post is extracted separately and its emotion is found out. Like that emoticons are extracted separately and its emotion is found out. If there is valid conflict between the emotion of sentence and emotion of emoticon gives the possibility of sarcasm [1].

Algorithm: Hybrid sarcasm detection
Input: Facebook comments
Output: Classification of comments as sarcastic or not sarcastic
Notation: C: Comment, CR: Corpus, E: Emoticon, EE: Emotion of emoticon, N: Negative, Ne: Neutral, P: Positive, W: Words, WE: Emotion of words

1. for C in CR
2. W= Find words
3. E= find emoticons
4. end for
5. for W in C
6. Score= find score for each word
7. Total score= score + score
8. end for
9. WE= sense get from the score
10. for E in C
11. Check Unicode of each emoticon
12. Score= find score for each Unicode

13. Total score= score + score
14. end for
15. EE= sense get from score
16. for Emotion in C
17. if WE= P && EE= N || WE= N && EE= P || WE= NE && EE= P || WE= NE && EE= N then
18. Comment is sarcastic
19. else if WE= P && EE= P || WE= N && EE= N || WE= NE && EE= NE || WE= P && EE= NE || NE || WE= N && EE= NE then
20. Comment is not sarcastic
21. end if
22. end for

4) Positive Sentiment & Negative Situation: Some sarcastic sentences have a common structure that creates a positive/negative contrast between a sentiment and a situation. Specifically, sarcastic sentence often express a positive sentiment in reference to a negative activity or state.

The PBLGA method uses for this approach which uses Sentiment and Situation phrases. It exploits four files, namely Positive Sentiment, Negative Sentiment, Positive Situation and Negative Situation files. It then checks if feedback to be tested matches with any Positive Sentiment and then it subsequently checks for any matches with Negative Situation. If both phrases match, then feedback is sarcastic, otherwise not sarcastic. Similarly, it checks for sarcasm with Negative Sentiment and Positive Situation.

For example: "My team lost the match, let's *celebrate this special occasion*." In this example, where the negative situation terms is underlined and the positive activity is italicized. The sarcasm in these tweets arises from the juxtaposition of a positive sentiment word (e.g., love, enjoyed, happy, pleased etc.) with a negative activity or state (e.g., being ignored etc.).

5) Pattern & Text Match Based Detection: The occurrence of part of speech in the sentence gives possibility of sarcasm. It is possible to detect sarcasm in the sentence by identifying pattern of part of speech such as adjective, noun, prepositions etc. in the sentence. It uses IWS algorithm.

6) Interjection Word Start (IWS): Sometimes sarcastic sentences start with an interjection words such as ahh, aha, wow, nah, uh, etc. And also these interjection words are followed by Adjectives or adverbs, which signaled the sarcasm. Example: "Wow! you are generous, you failed to score simple goal." IWS algorithm uses Part-of-Speech tagging technique to find all possible tags in the sentence such as Noun, Pronoun, Adjective, adverb, etc. After finding all the possible tags, IWS checks for first tag as an Interjection word and then subsequently checks for second-tag as an Adjective/Adverb. If it matches, then sentence is Sarcastic, otherwise not.

V.   ANALYSIS AND EXPERIMENTAL RESULTS

A. Dataset:

Posts are extracted from various public pages, groups and profiles using the GRAPH API. The extracted posts are in JSON format. The JSON format of a post comprises of several attributes such as 'message' of the post, 'user id' of the post which uniquely identifies the user who posted it, 'post id' which uniquely identifies the post, 'comments' , 'type' etc. These posts are then processed to filter out non English posts. The remaining posts are input to the system. We selected 1000 comments from different post of different Facebook pages.

B. Experiment with Examples:

1) Word based sarcasm detection:

Do females ever sit back and think 'I was wrong, I am gonna apologise to him'.
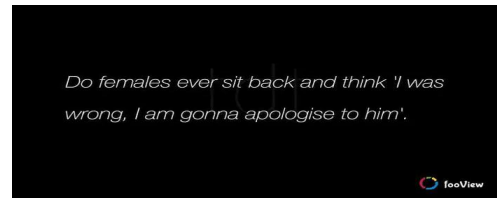
fooView

Fig. 3. Facebook post- Food for thought

a) Do sentiment analysis of comment which provide score to each word.

b) Then calculate all score to get total score of comment.

c) According to the score positive, negative, neutral, comment gives emotion, i.e. positive, negative, neutral respectively.

d) Check the count of emotion of the comments in the post.

e) *Now emotion with maximum count having the probability of non-sarcastic comments while other than maximum count emotion comments are sarcastic comment:* For the above post, Count of Negative comment: 16, Count of Positive comment: 8, Count of Neutral comment: 1

Here max count=16, so other than max count i.e. 8+1= 9 comments are Sarcastic

2) Emoticon based sarcasm detection: Refer fig.2
Comment: "A woman forgives you when she is at fault!! 😂😂😂😂 "

a) Check for emoticons in the comment: check for id of the emoticon

Here id is: ud83d%ude02

b) Check that id in the emoticon database and get the emotion of the emoticon: It is laughing emoticon & emotion of the emoticon is Positive

c) Do sentiment analysis of emoticon which provide score to each emoticon

d) Then calculate all score to get total score of emoticon.

e) Check the count of emotion of the comments in the post.

d) Now emotion with maximum count having the probability of non-sarcastic comments while other than maximum count emotion comments are sarcastic comment.

Count of Negative comment: 14, Count of Positive comment: 8, Count of Neutral comment: 3
Here max count=14, so other than max count i.e. 8+3= 11 comments are Sarcastic

3) Hybrid based sarcasm detection: Refer fig.2
Comment: "A woman forgives you when she is at fault!! 😆😆😆😆 "

a) Get the emotion of the comment from Word based emotion detection: In this case emotion of the comment text is Negative

b) Get the emotion of the emotion if present in the comment from Emoticon based emotion detection: In this case emotion of the emoticon is Positive

c) Use Hybrid Algorithm to check the conflict condition between comment text and emoticon then provide the result as comment is sarcastic or not sarcastic: According to the algorithm if the emotion of the word is negative and emotion of emoticon is positive then conflict is happen and the comment is Sarcastic.

4) Hashtag Sarcasm Detection: Refer fig.3

Fig. 3. Facebook Post-these guys are real heroes!

Comment: "Heroes without doing anything #sarcasm"

a) Find the Hashtag in the comment: "Heroes without doing anything #sarcasm"

b) Check for the word after hashtag (#): "Heroes without doing anything #sarcasm"

c) Check that word in the Hashtag database

d) If that word is present in the hashtag database, the comment is Sarcastic else comment is Not-Sarcastic: Word found in the database so comment is Sarcastic.

5) Pattern analysis for Sarcasm Detection: Refer fig.2

Comment: "Oh dear! I don't know about this. GROW UP".

a) Do POS tagging to the comment which provide part of speech to each word of the comment: Oh_UH dear_RB !_. I_PRP do_VBP n't_RB know_VB about_IN this_DT ._. GROW_VB UP_RP

b) Apply Interjection Word Start (IWS) algorithm on comments

c) Check the pattern of Comment with IWS pattern

d) If the pattern match is found comment is sarcastic else not sarcastic: According to IWS pattern if, after the interjection word adverb is present then comment is sarcastic: In the example after interjection word (Oh), adverb is present (dear) so comment is Sarcastic

Oh_UH dear_RB !_. I_PRP do_VBP n't_RB know_VB about_IN this_DT ._. GROW_VB UP_RP

C. Statistical Evaluation Metrics:

There are four statistical parameters namely accuracy, precision, recall and F-score, which are used to evaluate our proposed approaches.

$$Recall = \frac{True\ Positives}{True\ Positives + False\ negatives}$$

i.e, Recall = No. of correctly classified posts/No. of posts manually classified

Precision is calculated as follows:

$$Precision = \frac{True\ Positives}{True\ Positives + False\ Positives}$$

i.e, precision = No. of correctly identified posts/Total no. of posts identified.

F-Measure is calculated as follows:

$$F\ Measure = \frac{2 * Precision * Recall}{Precision + Recall}$$

$$Accuracy = \frac{True\ Positives}{True\ Positives + False\ Positives}$$

TABLE I. Comparison of proposed methods with state-of-art

Method	Precision	Recall	F-Score
Word based method	0.8841	0.9177	0.9005
Emoticon based method	0.8412	0.8617	0.8513
Hybrid method	0.8857	0.9323	0.9084
Hashtag method	0.833	0.854	0.846
Pattern Analysis	0.831	0.736	0.774
Positive Sentiment & Negative Situation	0.801	0.515	0.639

According to the table we can say that Hybrid method gives more accuracy according to their precision, recall & F-score value because on Facebook people probably use emoticon & the confliction pattern mentioned in the algorithm, when they want to make sarcastic statement. Positive Sentiment & Negative Situation method gives lowest accuracy because on the Facebook people generally doesn't follow the pattern which provided in the PBLGA algorithm. Another reason is this method requires full-fledged database which contain all the positive & Negative situation phrases present in the English literature.

VI. CONCLUSION AND FUTURE WORK

Using single approach for sarcasm detection is not sufficient. Our paper uses combined approach of different methods like emotion detection, use of emoticons, patterns, etc. identifies the social site comment is sarcastic or not. So it is required to use combined approach which take different methods and identify the comment is sarcastic or not.

The sarcasm identification model is a novel approach based on emotion model. The sarcasm identification model uses different algorithms, libraries and methods in emotion detection phase and its result is used for sarcasm detection. So it is too much dependent on the emotion identification module which poses risk at times. Most of the time Facebook comments are consists of words, hashtags and the emoticons. The system also considers hash tags and emoticons for sarcasm detection which are an important feature set of Facebook posts. The combined approach used in the system gives more accurate result than using individual methods. This system nullifies the disadvantages of former algorithms and methods which are mentioned in the paper. In future work we can find the sarcasm in images and videos.

REFERENCES

[1] Bjarke Felbo; Alan Mislove; Anders Søgaard; Iyad Rahwan; Sune Lehmann, "Using millions of emoji occurrences to learn any-domain representations for detecting sentiment, emotion and sarcasm", 27th Conference on Neural Information Processing Systems (NIPS), pp. 3111–3119, 2017.

[2] Raghavan V M; Mohana Kumar P; Sundara Raman R and Rajeswari Sridhar, "Emotion and sarcasm identification of Post from Facebook data using a Hybrid approach", ICTACT journal on soft computing, vol. 07, Issue. 02, 2017.

[3] Aditya Joshi; Vaibhav Tripathi; Pushpak Bhattacharyya; Mark Carman, "Harnessing Sequence Labeling for Sarcasm Detection in Dialogue from TV Series 'Friends'" Indian Institute of Technology Bombay, India Monash University, Australia IITB-Monash Research Academy, India, vol. 05, 2017.

[4] Dario Bertero and Pascale Fung, "A Long Short-Term Memory Framework for Predicting Humor in Dialogues", Association for Computational Linguistics, pp. 130–135, 2017.

[5] Aditya Joshi; Pushpak Bhattacharyya and Mark J. Carman, "Automatic Sarcasm Detection: A Survey", ACM Computing Surveys, vol. 50, pp.122-145, 2016.

[6] Prof. Nikita P. Desai; Anandkumar D. Dave, "A Comprehensive Study of Classification Techniques for Sarcasm Detection on Textual Data", International Conference on Electrical, Electronics, and Optimization Techniques (ICEEOT), pp. 1985-1991, 2016.

[7] Satoshi Hiai and Kazutaka Shimada, "A Sarcasm Extraction Method Based on Patterns of Evaluation Expressions", 5th IIAI International Congress on Advanced Applied Informatics, pp. 31-36, 2016

[8] Santosh Kumar Bharti; Korra Sathya Babu, "Parsing-based Sarcasm Sentiment Recognition in Twitter Data", IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining, pp. 1373-1380, 2015.

[9] Ellen R. and Prafulla S., "Sarcasm as Contrast between a Positive Sentiment and Negative Situation", in Proceedings the of 2013 Conference on Empirical Methods in Natural Language Processing (EMNLP), Association for Computational Linguistics (ACL), pp. 704-714, 2013.

[10] Li, W., & Xu, H., "Text-based emotion classification using emotion cause extraction", Published by Elsevier Ltd, Expert Systems with Applications, pp. 202-210, 2013.

[11] Elena Filatova, "Irony and sarcasm: Corpus generation and analysis using crowdsourcing", Proceedings of the Eight International Conference on Language Resources and Evaluation (LREC'12), pp. 392-398, 2012.

[12] Roberto González-Ibáñez; Smaranda Muresan and Nina Wacholder, "Identifying Sarcasm in Twitter: A Closer Look", Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: shortpapers, pp. 581–586, 2011.

[13] Ze-Jing Chuang and Chung-Hsien Wu, "Multi-Modal Emotion Recognition from Speech and Text", Computational Linguistics and Chinese Language Processing, vol. 9, pp. 45-62, 2011.

[14] C. Strapparava and R. Mihalcea, "Learning to Identify Emotions in Text", Proceedings of the 2008 ACM Symposium on Applied Computing, pp. 1556-1560, 2011.