International Conference on Computer, Engineering, Law, Education and Management 2017
[ICCELEM 2017]

ISBN	978-81-933584-7-4		**VOL**	01
Website	www.iccelem.com		**eMail**	iccelem@asdf.res.in
Received	05-June-2017		**Accepted**	20-July-2017
Article ID	ICCELEM002		**eAID**	ICCELEM.2017.002

Sentiment Analysis on Service Complaints using Sarcasm and Emoticon Detection

Ria A Sagum[1], Clowie Fel Y Bingco[2], J Wesley P Florentino[3]
[1,2,3]Department of Computer Science, College of Computer and Information Sciences,
Polytechnic University of the Philippines, Sta. Mesa, Manila

Abstract- *This study developed a sentiment analyzer that utilizes sarcasm detection. A multilingual language model was used to classify complaints with the use of Probabilistic Model with Kneser Ney smoothing to improve the system's accuracy. Emoticons were also detected to include in the sarcasm detection of a service complaint. The researchers made used of dataset from servicing company or government agency's service complaint to ensure that the research would be based on real life data. The system gained precision of 89%, recall of 92 % and F1-Score of 90%. The study shows that the accuracy of the Sarcasm Detection increased upon the integration of Emoticon Detection and resulted to the increase in sentiment analysis of the system.*

Keywords: Emoticon Detection, Language Model, Sentiment Analysis, Sarcasm Detection, Sarcasm and Emoticon Detection

I. Introduction

Sentiment analysis of review sites and online forums has been a popular subject for several years in the field of natural language processing. [1] Before the internet awareness became widespread, many of people used to ask their friends or neighbors for opinion of a good electronic products or a food before buying it or going for it. With the growing availability and popularity of opinion-rich resources such as online review websites and personal blogs, new opportunities and challenges arise as people now can, and do, actively use information technologies to seek out and understand the opinions of others. Unfortunately, these opinion rich resources are available in unstructured format. It has encouraged the analysts to develop an intelligent system that can automatically categorize or classify these text documents.

Sarcasm is a form of art that is marked using sarcastic language and is intended to make its victim the buff on contempt of ridicule. In text mining, automatic detection of sarcasm is considered a difficult problem [2] and has been addressed in only a few studies. Sarcasm can be used to transform the polarity of an apparently positive or negative utterance into opposite [3]. It was suggested by the study of (Sagum et.al.), that sarcasm can be used to increase the accuracy of a sentiment analyzer. Detecting sarcasm and emoticon in text is a complex process. To recognize sarcasm, tone recognition must also be considered since people express their feelings with high and low pitches [4]. To recognize emoticon, we must know the following: First, emoticons represent body language, which is nonverbal. Second, there has been a lack of sufficient methods for the analysis of emoticons and need to recognize the pattern or identify rather what is the emoticon that has been used [5].

Considering these, the researchers developed a system that will recognize and able to analyze emoticons and sarcastic statements. The researchers strongly agree that it will help to accurately detect emotions, emoticons that were used were based on the article Smiley Face and Text Emoticon Symbols by Beal [6]. The system will be able to classify the polarity of a complaint whether it is positive, negative or neutral. The sentiment analyzer will be using sarcasm detection and emoticon detection as its feature to improve the accuracy of the analyzer, it will analyze service complaints of customers.

II. Related Works

The study of Wang etal. [3], explains that automatically detecting sarcasm in twitter is a challenging task since sarcasm transforms the polarity of an apparently positive or negative utterance into its opposite. Previous work focuses on feature modelling of the single tweet, which limit the performance of the task. The sarcasm detection problem is modelled as a sequential classification task over a tweet and his contextual information.

The first study to addresses the use of emoticons to recognize sarcasm was done by Walther et.al [7]. Their participants were asked to read emails including positive or negative messages, followed by a smiley face :-), a sad face :-(, a wink face ;-) or no emoticon. Messages were ambiguous as to whether they were intended literally or sarcastically. Furthermore, the most sarcastic condition was a positive verbal message with a wink. However, this message–emoticon combination was not significantly more sarcastic than a positive message with a smile, a sad face, or nothing at all. Therefore, the researchers concluded that winks do not actually connote greater sarcasm than other emoticons.

The study of Derks etal. , [8] ran a similar study to Walther etal., examining the same set of emoticons, but included a neutral message condition (in addition to positive and negative), and the participants in the study were the recipients of the emails. In contrast to Walther etal. , the work od Derks etal. showed that emoticons enhanced the valence of a message. Additionally, and again in contrast to Walther etal. messages with a wink face were rated as significantly more sarcastic than those without an emoticon.

Hogenboom established that in order to exploit emoticons in automated sentiment analysis, the researchers first need to analyze how emoticons are typically related to the sentiment of the text they occur in. Interestingly, this positioning of emoticons suggests that it is typically not a single word. [8]

Lexical feature-based classification as first of the types to detect sarcasm is text properties such as unigram, bigram, n-grams, etc. are classified as lexical features of a text. Authors used these features to identify sarcasm, introduced this concept for the first time and they observed that lexical features play a vital role in detecting irony and sarcasm in text. [9] Riloff, et. al. [10] used a well-constructed lexicon-based approach to detect sarcasm and for lexicon generation they used unigram, bigram and trigram features. Barbieri et. al. considered seven lexical features to detect sarcasm through its inner structure such as the intensity of the terms. [11]

Pragmatic feature-based classification as the second type to detect sarcasm uses symbolic and figurative text in tweets is frequent due to the limitations in message length of a tweet. These symbolic and figurative texts are called pragmatic features (such as smiles, emoticons, replies, @user, etc.). It is one of the powerful features to identify sarcasm in tweets as several authors have used this feature in their work to detect sarcasm. Pragmatic features are one of the key features used by Kreuz & Caucci [9] to detect sarcasm in text. The study of Carvalho et. al. [12] used pragmatic features like emoticons and special punctuations to detect irony from newspaper text data.

After thorough understanding on the different studies on sentiment analysis it was then highlighted that sarcasm can help with the accuracy of sentiment analysis, likewise the emoticons can give impact in the accuracy of sarcasm detection. The researchers made used of these features to look on to the accuracy of a sentiment analyzer once sarcasm and emoticon detection were taken into consideration.

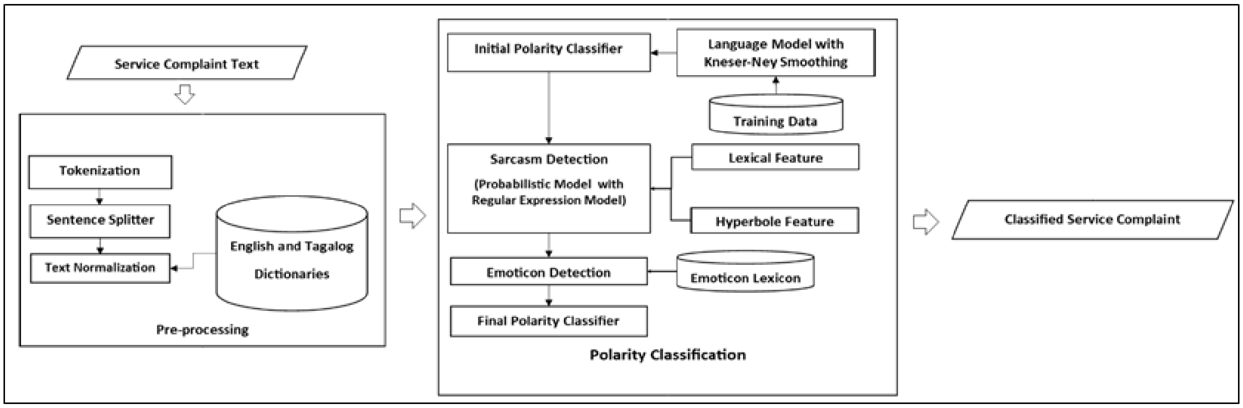
III. Discussion of the System's Design

The process of the system starts with pre-processing activities (see Figure 1). These tasks include Tokenization, Sentence Splitter and Text Normalization. The researchers used Python's NLTK for this module.

The next phase is the Polarity Classification Module, this module consists of the ff:

a. **Initial Polarity Classifier,** it distinguishes the polarity of the text whether it is positive, negative or neutral by using the classification of bag-of-words and Language Model. It sets out the polarity for each token in the sentences.

Figure 1: System Architecture



b. **Language Model,** assigns a probability to a set of string based on its occurrence in text prior processed. The model used N-Grams and a Kneser–Ney smoothing algorithm to improve the probabilities of each gram. The Training Data is a collection of classified complaint sentences based in polarity such as, positive, negative or neutral. The model used up to tri-gram. The reason is that the data that have been used are composed of Tagalog, English and Taglish complaints and the structure of its sentences can be processed correctly using this gram.

c. **Sarcasm Detection, this** will be implemented after setting the polarity in each text. The algorithm implemented in this process was Probabilistic Model and Regular Expression Model. Using the model, the probability of the word of being negative is from -1.0-0.4, for neutral -0.399-0.399, and for positive it was 0.4-1.0.

e.g. [‘Napakagaling’,0.642,Pos], [‘ninyo’,-0.132,Neu], [‘gumawa’, 0.539, Pos], [‘ng’, 0.539, Neu], [‘daan’, -0.205, Neu], [‘.’ , -0.132, Neu], [‘Lubak’, -0.742, Neg], [‘Lubak’, -0.742, Neg], [‘parin’, 0.143, Neu], [‘:D’, 0.053,Neu]

c.1. Lexical Feature Classifier, will be the first to detect sarcasm in text properties using unigram, bigram, trigram, n-grams, etc.each of which are classified as lexical features of a text.

c.2. Hyperbole Feature, composed of Intensifier will search for keywords that denote intensity or degree of a given text. These keywords will be used to increase/decrease the positivity or negativity of a certain word/s. The Regular Expression Model was used by the researcher to detect sarcasm by observing patterns that denote sarcastic complaints from a given text. Interjection such as: “wow”, “aha”, “yay” etc. has a higher chance of being sarcastic, and series of punctuation marks will be used as additional cues for sarcasm detection.

d. **Emoticon Detection,** will add a value to its polarity if an emoticon placed on a given text. It will add a positive, negative or neutral weight on its context. The emoticon lexicon contains all the emoticons included in the American English twitter corpus study that will be used in this research. In the example, the emoticon polarity’s detected shifted to a positive polarity.

e.g. [‘:D’, Neu] -> [‘:D’, Pos]

e. **Final Polarity Classifier,** computes for the overall polarity weights in the Sarcasm and Emoticon Detection. Adding to it, a rule based was implemented to check if a sentence’s sentiment was Positive and it was followed by a negative emoticon and if a sentence’s sentiment was Negative and was followed by a positive emoticon, it will have a corresponding result of being sarcastic. Lastly, it will output the classified service complaint whether it is sarcastic or not sarcastic and if an emoticon existed or not. In the example, the first sentence which resulted to a low probability of being positive and the second sentence scoring a higher probability of being negative, resulting to shift the total context of the sentence into a negative sentiment. Now the negative sentence is followed by a positive emoticon. Sarcasm is detected by applying the rule of negative sentence followed by a positive emoticon.

Sarcasm Detection

Napakagaling nyo gumawa ng daan. Lubak lubak parin = NEG

Emoticon Detection

:D = POS

NEGATIVE (Sentiment) + POSIsTIVE (Emoticon) = Sarcastic

IV. Presentation of Results

The performance of the system was measured using the F measure or F1 score (Eq. 1).

$$F1\ Score = 2 * \frac{precision * recall}{precision + recall} \quad (1)$$

F1 score refers to the harmonic mean of precision (Eq. 2) and recall (Eq. 3).

$$Precision = \frac{True\ Positives}{(True\ Positives + False\ Positives)} \quad (2)$$

$$Recall = \frac{True\ Positives}{(True\ Positives + False\ Negatives)} \quad (3)$$

The researchers made use of the table of Dan Jurafsky of Stanford University to classify the output of the stream. It uses True Positive (TP), the number of sentiments classified positively correct by both system and human, True Negative (TN), the number of sentiments that classified positively incorrect by both system and human, False Positive (FP), the number of sentiments classified correctly by system but Incorrect by human, and False Negative (FN), the number of sentiments classified correctly by human but incorrect or neutral by system.

Table 1 shows the result of the evaluation of the sentiment in every criterion. The precision of Complaints with Emoticon gained 100%, this is because the dataset that was fed in the system were correctly analyzed by the system. The system recorded 67% for recall due to a complaint that was not recognized by the system and 80% for F1–Score. For Complaints with Sarcasm and Emoticon, it gained 100% high in recall, due to 258 complaints that were correctly recognized, 100% in precision due to one complaint that was not recognize and 100% in F1-Score due to the computed precision and recall. The result of the system's evaluation under the criteria of Plain Complaints for precision, recall and F1-score were 84%, 80% and 82% respectively. And lastly for the Complaints with Sarcasms gained 34%, 42% and 38% respectively for precision, recall and F1 Score. This is due to unclean dataset that was fed in the system during evaluation phase, also the data set lacks complaints that includes sarcastic features.

Table 1: Summary of Results according to Criteria

Criteria	TP	TN	FP	FN	PRECISION	RECALL	F1SCORE
Complaints with Sarcasm	38	552	73	52	34%	42%	38%
Complaints with Emoticon	2	712	0	1	100%	67%	80%
Complaints with Sarcasm and Emoticon	258	456	1	0	100%	100%	100%
Plain Complaints	290	298	54	73	84%	80%	82%

Table 2: Overall Systems Performance in Recognizing Sentiments

PRECISION	89%
RECALL	92%
F1 SCORE	90%

By using the sarcasm detection with integration of emoticon detection, in the developed system, the results shows in Table 2 was the Overall Performance of the system for sentiment analysis regardless of the criteria. Based on the study of Ebola etal [13], that generated a rating system for the parameters: precision, recall and F1 score, the system's accuracy rate is "Very Good".

V. Conclusion

In this study, language model was utilized to determine the sentiment of service complaints. Sarcasm detection was employed through pattern extraction, to improve the overall accuracy of the system. Lastly, the researchers included emoticon detection to gain much higher accuracy in detecting sarcasm in service complaints.

The system was able to serve its purpose: to detect sarcasm and integrate it to the system to accurately analyze a sentiment. If it will be compared to the previous study (Sagum, De Vera, Lansang, Narciso, & Respeto, 2015) Ref [12], where smoothing algorithm was used for the Language Model, the system's precision and recall were quite higher than the previous one. However, there are numerous words that the program was not able to recognize correctly. Human raters typically agree 70% is a Pass rating when it comes to sentiment analyzers. [14] [13] [4] Thus, a sentiment analyzer that has 82% accuracy rate is quite doing well as humans do in analyzing sentiments but still has a lot to learn for improvement. The results of the study showed that the system was able to correctly determine the detection in service complaints. It was rated "Very Good" in performing sentiment analysis.

The results of the research furthermore showed a marginally dense F1-Score. F1-Score ranged from 50-100%, which is considerably satisfactory to excellent rate, but can be improved through a larger set of implementation data. The researchers observed that the F1-Score and Accuracy of the criteria Complaints with Sarcasm and Emoticon is higher than the criteria of Complaints with Sarcasm. The researchers achieved their goal to integrate the emoticon detection to sarcasm detection for better recognition of sarcasm and improved the previous study of Sagum et. al. [4] Researchers found out that the emoticon detection is effective in achieving higher accuracy when integrated in sarcasm detection.

VI. Recommendations

The implementation set was a set of mixed complaints with different sentiments so the results were not that accurate. The researchers recommend testing the system using a different set of complaints for each sentiment: Positive, Negative and Neutral for much more accurate results of the capability of the sentiment analyzer to classify sarcastic complaints.

The researchers recommend using Chen and Goodman's Modified Kneser-Ney. This smoothing method is tested to be the best smoothing method for these kinds of problems. Probabilistic Model with Modified Kneser-Ney Smoothing needs a much large of training data.

It is also recommended for the future researcher to use other combination of features in sarcasm detection such as Slang detection, Irony detection and Rant detection to improve the degree of accuracy of sarcasm detection in a hybrid manner, thereby increasing the accuracy of sentiment analyzer.

VII. References

1. Kawathekar, S. A., & Kshirsagar, M. M. (2012, January). Sentiments analysis using Hybrid Approach involving Rule-Based & Support Vector Machines methods. IOSR Journal of Engineering (IOSRJEN), 2(1), 55-58.
2. Nigam, K., & Hurst, M. (2006). Towards a Robust Metric Polarity. In J. G. Shanahan, Y. Qu, & J. Wiebe (Eds.), Computing Attitude and Affect in. Springer Netherlands. doi:10.1007/1-4020-4102-0_20
3. Wang, Z., Wu, Z., Wang, R., & Ren, Y. (2015). Twitter Sarcasm Detection Exploiting a Context-Based Model. Web Information Systems Engineering (WISE 2015), 9418, 77-91.
4. Sagum, R. A., De Vera, J. G., Lansang, P. J., Narciso, D. S., & Respeto, J. K. (2015, March). Application of Language Modelling in Sentiment Analysis for Faculty Comment Evaluation. Proceedings of the International MultiConference of Engineers and Computer Scientists, IMECS 2015, 1.
5. Ptaszynski, M., Maciejewski, J., Dybala, P., Rzepka, R., Araki, K., & Momouchi, Y. (2012). Science of Emoticons: Research Framework and State of the Art in Analysis of Kaomoji-Type Emoticons. In U. S. Tiwary, & T. J. Siddiqui (Eds.), Speech, Image, and Language Processing for Human Computer Interaction (pp. 234-257). Allahabad, India: IGI, USA.
6. Beal, V. (2016). Smiley Face and Text Emoticon Symbols. Retrieved September 11, 2016, from www.webopedia.com: http://www.webopedia.com/quick_ref/textmessageabbreviations_02.asp
7. Walther, J. B., & D'Addario, K. P. (2001). The Impacts of Emoticons on Message Interpretation in Computer-Mediated Communication. Social Science Computer Review, 19, 324–347.

8. Derks, D., Bos, A. E., & Von Grumbkow, J. (2008). Emoticons and Online Message Interpretation. Social Science Computer Review, 26, 379–388.
9. Hogenboom, A., Bal, D., & Frasincar, F. (2013, March). Exploiting Emoticons in Sentiment Analysis. SAC.
10. Kreuz, R., & Caucci, G. (2007). Lexical influences on the perception of sarcasm. (pp. 1–4). ACL: Proceedings of the Workshop on Computational Approaches to Figurative Language.
11. Barbieri, F., Saggion, H., & Ronzano, F. (2014). Modelling sarcasm in twitter a novel approach. (pp. 50–58). Proceedings of the 5th Workshop on Computational Approaches to Subjectivity, Sentiment and Social Media Analysis.
12. Carvalho, P., Sarmento, L., Silva, M., & De Oliveira, E. (2009). Clues for detecting irony in user-generated contents: oh...!! it's so easy;-). (pp. 53–56). ACM: Proceedings of the 1st International CIKM Workshop on Topic-Sentiment Analysis for Mass Opinion.
13. Eboña, K. M., O, S., Perez, G. P., Roldan, J. M., Sagum, R. A., & Domingo, I. V. (2013). Named Entity Recognizer (NER) for Filipino Novel Excerpts using Maximum Entropy Approach. Undergraduate Thesis, Polytechnic University of the Philippines, Department of Computer Science, Manila City.
14. Rafael, V. L. (2008). Taglish, or the phantom power of the lingua franca. In Philippine English. Riloff, E., Qadir, A., Surve, P., De Silva, L., Gilbert, N., & Huang, R. (2013). Sarcasm as contrast between a positive sentiment and negative situation. (pp. 704–714). Proceedings of the Conference on Empirical Methods in Natural Language Processing.