Sentiment Detection

Loyiso Jiya
Department of Computer Science
University of the Western Cape
Cape Town, South Africa
3338868@myuwc.ac.za

I. Introduction

Sentiment detection, is a special niche of text classification, that emerged as a new research area at the beginning of the 2000s due to the increase of subjective texts in the social media, forums, and blog [1]. Sentiment detection is also commonly referred to as sentiment analysis, opinion mining, review mining, subjectivity extraction, and opinion extraction [2]. In this paper, we will continue referring to it as sentiment detection. The applications of sentiment detection are growing with the times, it has been applied for mental illness detection [3], sarcasm detection [4], movie review sentiment detection, etc. Sentiment detection can be performed on an variety of text sources whether it digital copies of books, documents, forum posts, social media posts.

Social media platforms contain a wealth of free, readily available data that can be gathered, analyzed and used to measure the public's perception of a company, the services it provides, or the products it produces. Twitter, is one such platform, that has become one of the biggest social media platforms. As of last year, Twitter had approximately 300 million active users per month. Millions of tweets are sent a day. It stands to reason that a large amount of data is generated by Twitter users. Among the daily tweets, are subjective reviews of companies, products, movies, and commentary on political issues. This generated data is used by companies and researchers for multiple sentiment detection problems. Companies require feedback from their customers to remain relevant and competitive. However, manually going through reviews or conducting focus groups can be rather cumbersome and costly. With this in mind, companies can use Twitter's publicly available *application programming interface* (API) to gather tweets about a company, the services it provides or products it produces and from those tweets analyze the general feeling of its customers towards them, their services, or the product. Large companies like Amazon and Dell are already leveraging this data to evaluate how their customers react to their service [5]. On the other hand, most of the research being done in the field is comparison research, where the performance of different machine learning techniques is compared. Also, much focus is being put on varying pre-processing and feature extraction and feature selection techniques to achieve different results.

The rest of the paper will contain a literature review in section two. Section three is about the implementation process of the pre-processing, feature extraction, and feature selection. The rest of the paper will contain the discussion, results, conclusions and future work.

II. Literature Review

This section gives a brief summary of the research that is currently being undertaken in the field of sentiment detection.

Contratres et al. attempted to remedy the problem of making recommendations to new users in recommender systems for e-commerce websites. The authors, used naive Bayes and a *support vector machine* (SVM), coupled with *term frequency-inverse document frequency* (TF-IDF) to perform sentiment analysis using the social media data that a new user of an e-commerce website volunteers to give to the system for the generation of product recommendations [6]. In their future work, they proposed using deep learning techniques to classify product categories and sentiment.

Zheng et al [7], focused on the feature extraction techniques that can be utilized in sentiment detection. The paper mainly focused on Chinese online reviews and used only a SVM to test the different techniques. For feature extraction N-Part-of-Speech-grams, TF-IDF. In the end, TF-IDF outperformed the different combinations of N-PoS-grams.

Jiaqiang and Xialoin (2017), observed that much focus is put on feature extraction in the field. In contrast, they studied the effects of pre-processing tweets by removing links, expanding acronyms and removing stop words just to mention a few. Their implementation used random forests, SVM, logistic regression and naive Bayes to perform sentiment detection. For feature extraction they used N-grams and prior polarity. They observed that the random forests and SVM algorithms are more sensitive to the different pre-processing techniques) [8].

On the other hand, the performance of different machine learning techniques was evaluated when applying feature selection using IG. They proposed that a *majority voting ensemble* (MVE) comprised of a SVM, linear regression, and naive Bayes algorithm. The results showed that using IG for feature selection improves performance and that the MVE outperformed the individual algorithms [9].

Likewise, Zainuddin et al, emphasized the importance of feature selection in their implementation of a hybrid sentiment classification for Twitter [10]. The research presented a comparison of the different feature selection methods. These methods included the *principal component analysis* (PCA), *latent semantic analysis* (LSA), and *random projection* (RP). To test their methods, they used a SVM. In the end, the PCA yielded the best results.

Lauren et al [11], expanded on their previous work where they proposed an ELM for word embeddings. In the aforementioned paper, they applied their previously designed ELM-based word embedding for sentiment detection and sequence labeling. Their implementation achieved better results than the standard word2vec word embedding and *global vectors* (GloVec) model in both sentiment detection and sequence labelling.

Furthermore, deep learning techniques that have been previously used in text classification tasks were compared to a SVM in sentiment detection for Arabic hotel reviews. In this study, the Arabic Natural Language Processing was used for pre-processing and N-grams, *parts-of-speech* tagging and word embeddings were used for feature extraction. The SVM outperformed the different deep learning techniques [12].

Similarly, in another study, a deep *convolutional neural network* (CNN) was used in an ensemble algorithm to achieve better results. The study compared the proposed (ensemble) model with other variations of a CNN and found that the proposed model achieved better results than some CNNs but not all of them. In this application, global vectors, the word2vec model, TF-IDF, and bag-of-words were used for feature extraction [13].

Given the surveyed literature above, it is quite evident that many of the studies conducted are comparative in nature. In the same way, our research will be comparison of different feature extraction methods; similar to Zheng et al, TF-IDF will be used for extraction. However, as opposed to Zheng et al, the TF-IDF will be compared against the ELM word embeddings. Moreover, for feature selection, Zainuddin et al and data science competitions have proven the utility of PCA as a feature selection method. Therefore, the PCA method will be compared to IG. Furthermore, most of the papers surveyed suggested implementing deep learning approaches in their future work. Thus, in this paper we propose a ResNet as our deep learning approach which has not been used in the surveyed literature. Finally, in addition to the ResNet, we propose using the XGBoost algorithm. Both of these techniques will be compared against the standard machine learning techniques used in sentiment detection such as SVM and naive bayes.

III. Implementation

A. Data Set

For this paper we will be using the will be the Stanford Twitter Sentiment Corpus (Sentiment140) *Stanford Twitter Corpus* (STS) [14]. The data was created by Alec Go, Richa Bhayani, and Lei Huang, who were Computer Science graduate students at Stanford University. On their website as referenced above, the use cases stated include: brand management, polling and planning a purchase. The file consists of six fields:

- the polarity of the tweet (e.g. 0 = negative, 2 = neutral, 4 = positive)
- the id of the tweet (e.g. 2087)
- the user that tweeted (e.g. robotickilldozr)
- the text of the tweet (e.g. Lyx is cool)

In this research, the data will be divided into training and testing portions as illustrated in Fig. 1.

B. Pre-processing

For any machine learning algorithm to work for any NLP task the text needs to be represent as numerical data for statistical analysis. However, it is unlikely for most text to immediately be ready ready for feature extraction. Because, if this is the case, the feature extraction will generate a lot of features that are not important to the task at hand and thereby generate noise which will affect the final model's accuracy and its ability to generalize. Therefore, to alleviate the noise in text data, the data must first go through the stages of pre-processing.

1) Tokenization: Tokenization is the process of breaking down continuous text into words, phrases, and removing spaces from a given text. The resulting words, phrases, and or symbols are referred to as tokens [15].

2) Stop words removal: In the context of social media, text and document classification is often fill with common words like 'an', 'this', 'that', 'is' that often carry a small weight relative to other words. These words are known as stop words. Most *natural language toolkits* (NLTK) have a list of stop words to be used during the pre-processing phase. This list is then used to reliminate the words from text or documents this consequently decrease the resulting features.

3) Stemming: Stemming is transforms a word to obtain its variants using different linguistic processes like affixation (adding affixes). The most frequently used stemmer is the Porter Stemmer, this paper will also use this stemmer.

4) Spelling correction: With social media posts, there will often be words that are deliberately misspelled to achieve some sort of emphasis. For instance, someone would write 'Amazzzziiiiing', instead of just amazing. These are some of the spelling mistakes that one must deal with during pre-processing. Also, these are not the only spelling mistakes that are there but these are the most prevalent in our context. To mitigate this, we will use the python *pattern* library to correct the spelling mistakes contained in our text sample.

5) Noise removal: In the case of social media or blog posts, there can be a lot of symbols and characters that add no value to a machine learning model, but are rather

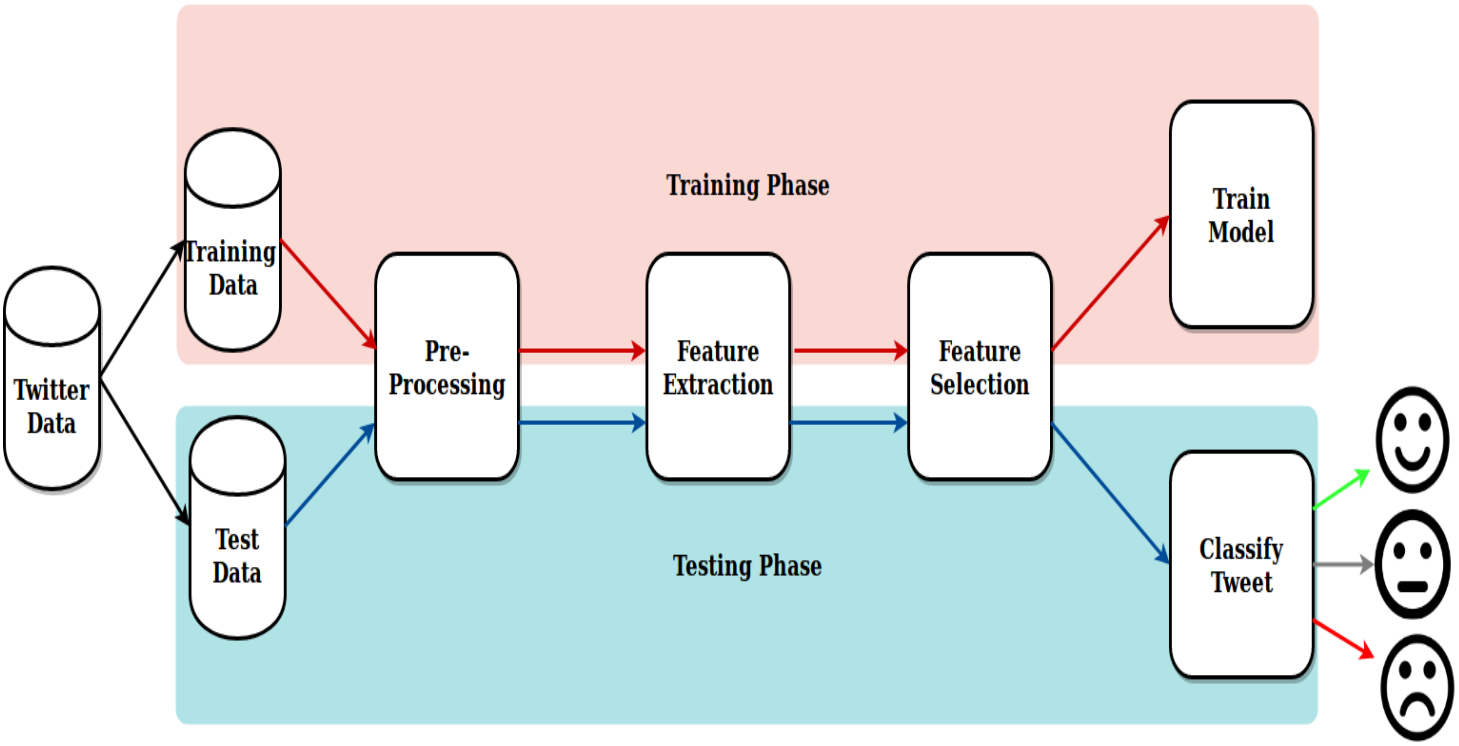

Fig. 1: Process flow chart

relevant in the context of whatever social media platform the post is from. From example the hashtag '#' and the tag '@' characters on Twitter. Noise removal is performed as an attempt to remove some of these characters. However, because this technique requires the use of regular expressions, it is done at the potential expense of losing punctuation marks which are critical in understanding sentences.

C. Feature Extraction

This section will give a brief overview on the feature extraction techniques that will be compared in this paper for the experimentation section. Namely, the *term frequency-inverse document frequency*, ELM-based word embeddings, and word2vec embedding. This is not an exhaustive review of the techniques, but rather a high-level description of the techniques and how they work.

TF-IDF is one of the simplest techniques of text feature extraction. A reductionist definition would be that TF-IDF is a method based on counting the number of words in a given document and assigning a weight according to how frequently the word appears across the documents.

As mentioned in the literature review section of this paper, our approach will implement an ELM-based word embedding as implemented by Lauren et al [11]. ELM use a standard feed-forward neural network structure with a single hidden layer, as illustrated in Fig. 2.

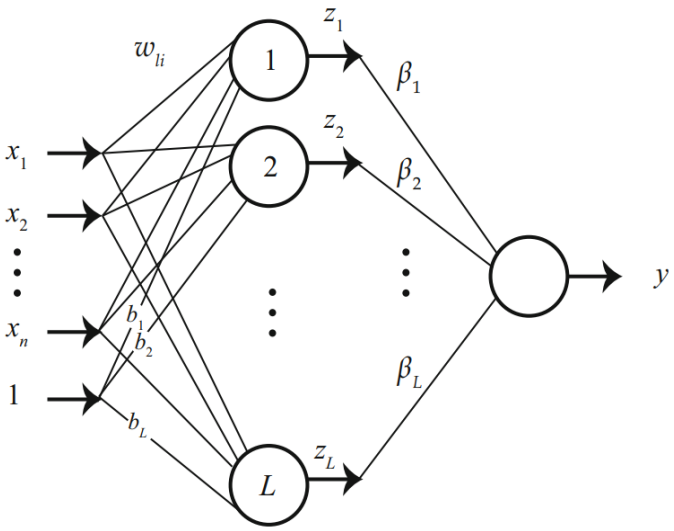

Fig. 2: ELM architecture

D. Feature Selection

For any machine learning algorithm to learn and gain meaningful insight and at times generalize quickly, one must perform feature selection. Feature selection is a process by which an algorithm is used to determine which attributes in a vector are most useful in determining a label or class value. For this paper, we will use *information gain* (IG) and *principal component analysis* (PCA) for feature selection.

In the context of text classification, IG measures the bits of information gained, while deciding the class which a text belongs to by checking how frequently a word occurs in the text [16].

On the other hand, PCA computes new variables called *principal components* [17] to achieve its main goals, which are:

- extract the most important information from the data table
- compress the size of the data set by keeping only this important information
- explain and simplify the description of data
- analyze the structure of observations and variables

The resulting values are in the form of linear combinations of the original variables.

REFERENCES

[1] R. Xia, C. Zong, and S. Li, "Ensemble of feature sets and classification algorithms for sentiment classification," *Information Sciences*, vol. 181, no. 6, pp. 1138–1152, 2011.

[2] B. Liu, "Sentiment analysis and opinion mining, Morgan & Claypool publishers, May 2012," *Author Profiles Mr. Sohom Ghosh is a student at Heritage Institute of Technology, Kolkata. He is pursuing B. Tech in Computer Science and Engineering. His research interests include Data Mining, Social Network Analysis and Machine Learning*.

[3] S. C. Guntuku, D. B. Yaden, M. L. Kern, L. H. Ungar, and J. C. Eichstaedt, "Detecting depression and mental illness on social media: an integrative review," *Current Opinion in Behavioral Sciences*, vol. 18, pp. 43–49, 2017.

[4] A. C. Pandey, S. R. Seth, and M. Varshney, "Sarcasm detection of amazon alexa sample set," in *Advances in Signal Processing and Communication*. Springer, 2019, pp. 559–564.

[5] M. Oussalah, F. Bhat, K. Challis, and T. Schnier, "A software architecture for twitter collection, search and geolocation services," *Knowledge-Based Systems*, vol. 37, pp. 105–120, 2013.

[6] F. G. Contratres, S. N. Alves-Souza, L. V. L. Filgueiras, and L. S. DeSouza, "Sentiment analysis of social network data for cold-start relief in recommender systems," in *World Conference on Information Systems and Technologies*. Springer, 2018, pp. 122–132.

[7] L. Zheng, H. Wang, and S. Gao, "Sentimental feature selection for sentiment analysis of chinese online reviews," *International journal of machine learning and cybernetics*, vol. 9, no. 1, pp. 75–84, 2018.

[8] Z. Jianqiang and G. Xiaolin, "Comparison research on text pre-processing methods on twitter sentiment analysis," *IEEE Access*, vol. 5, pp. 2870–2879, 2017.

[9] M. M. Fouad, T. F. Gharib, and A. S. Mashat, "Efficient twitter sentiment analysis system with feature selection and lassifier ensemble," in *International Conference on Advanced Machine Learning Technologies and Applications*. Springer, 2018, pp. 516–527.

[10] N. Zainuddin, A. Selamat, and R. Ibrahim, "Hybrid sentiment classification on twitter aspect-based sentiment analysis," *Applied Intelligence*, pp. 1–15, 2018.

[11] P. Lauren, G. Qu, J. Yang, P. Watta, G.-B. Huang, and A. Lendasse, "Generating word embeddings from an extreme learning machine for sentiment analysis and sequence labeling tasks," *Cognitive Computation*, vol. 10, no. 4, pp. 625–638, 2018.

[12] M. Al-Smadi, O. Qawasmeh, M. Al-Ayyoub, Y. Jararweh, and B. Gupta, "Deep recurrent neural network vs. support vector machine for aspect-based sentiment analysis of arabic hotels reviews," *Journal of Computational Science*, vol. 27, pp. 386–393, 2018.

[13] M. Hanafy, M. I. Khalil, and H. M. Abbas, "Combining classical and deep learning methods for twitter sentiment analysis," in *IAPR Workshop on Artificial Neural Networks in Pattern Recognition*. Springer, 2018, pp. 281–292.

[14] T. Inc. (2018) Quarterly results. [Online]. Available: https://investor.twitterinc.com/financial-information/quarterly-results/default.aspx

[15] L. M. Abualigah and A. T. Khader, "Unsupervised text feature selection technique based on hybrid particle swarm optimization algorithm with genetic operators for the text clustering," *The Journal of Supercomputing*, vol. 73, no. 11, pp. 4773–4795, 2017.

[16] C. A. Gonçalves, E. L. Iglesias, L. Borrajo, R. Camacho, A. S. Vieira, and C. T. Gonçalves, "Comparative study of feature selection methods for medical full text classification," in *International Work-Conference on Bioinformatics and Biomedical Engineering*. Springer, 2019, pp. 550–560.

[17] M. Morchid, R. Dufour, P.-M. Bousquet, G. Linares, and J.-M. Torres-Moreno, "Feature selection using principal component analysis for massive retweet detection," *Pattern Recognition Letters*, vol. 49, pp. 33–39, 2014.