Emotion Detection: Exploring Innovative Approaches and Novel Methods

Ali Alzurufi
Department of Mathematics and Computer Science
Lawrence Technological University
Southfield, Michigan, USA
aalzurufi@ltu.edu

***Abstract*—Text based emotion detection has become an important field of research in the domain of natural language processing. Digital communication, whether it's through social media platforms or online businesses, provide text through which a person's mental state, thoughts, and opinions can be inspected. Emotion detection is a subset of sentiment analysis, predicting a person's unique emotions given a context. An issue regarding text based emotion detection is the absence of tone, facial expressions, pitch which can more easily classify a person's emotions. Throughout recent years, researchers have implemented and tested various natural language processing tasks to effectively derive emotional state from text. Methods such as keyword and lexicon approach have shown limitations due to their focus on semantic relations. In this study, a deeper dive into determining whether methods such as text generation and preprocessing hold any significance when it comes to classifying emotions is taken. A machine learning approach was also implemented, consisting of Stochastic Gradient Descent, Gradient Boosting, and Logistic Regression classification methods. This was done to evaluate the performance of the model, which resulted in an accuracy of 88.4 percent.**

***Index Terms*—Emotion Detection, Text Generation, Feature Extraction, Data Preprocessing**

I. Introduction

With the emergence of Artificial Intelligence (AI) in the 1950's, and the drastic change it brought with it, researchers have utilized it through various fields such a Natural Language Processing (NLP), machine learning, and deep learning. A challenge posed through NLP is the ambiguity of human interaction through text. The fields of NLP remain unclear due to its computational and linguistic techniques, which help computers understand and generate human-computer interactions in the form of text and speech [5]. A key area within the domain of NLP is sentiment analysis, which identifies a person's opinions expressed in a piece of text to determine their attitude towards a particular topic or product. The issue with sentiment analysis is that it broadly classifies text as either positive, negative, or neutral. A semantic understanding refers to the meaning or understanding of language and its associated concepts. It involves the interpretation and comprehension of words, phrases, sentences, and the relationships between them [1]. A subset of sentiment analysis, emotion detection is a means of identifying distinct human emotion types such as furious, cheerful, or depressed [2]. In emotion detection, the problem is identifying the human emotions from a piece of text that best represents the author's mental state [6]. A semantic understanding plays a crucial role in emotion detection because it allows for a deeper understanding of the meaning and the context of the language, capturing richness and complexity of human language, and leading to more accurate assessment of text based emotion.

II. Literature Review

Below are a few relevant studies that used different techniques to detect emotion from text including [5], [4], and [7]

[5] proposed a hybrid model consisting of deep learning and machine learning algorithms to predict emotions. The deep learning consists of Convolutional Neural Network (CNN) and Bidirectional Gated Recurrent unit (Bi-GRU). For machine learning, The Support Vector Machine (SVM) classifier was used. The data was taken from three different datasets including ISEAR, WASSA, and Emotion-stimulus which have different types of text such as normal sentences, tweets, and dialogs. Due to the data being in raw form, they preprocessed it by applying Natural Language Toolkit (NLTK) tools such as tokenization, lemmatization, and stop word removal. To reduce the dimensionality of large data, TF-IDF was applied. Word2Vec was applied after the dataset had been preprocessed and converted to vector form. A pipeline was created to combine both the CNN and the BI-GRU models. The data is put through embedding layers, and then through each of the models respectively, it was then combined and fed to the SVM classifier. The evaluation of these models resulted in an accuracy of 80.11 percent.

[4] proposed a hybrid approach combining rule-construction and machine learning approaches into a single model. Rule-based approach focuses on the grammatical errors and abbreviations. preprocessing was done through stop word removal, tokenization, and parts of speech tagging. The intensity of the word is calculated next. After this, the negation in the sentence is checked and the emotion is extracted from the sentence. The hybrid approach uses the basic rule-based algorithm to retrieve semantics and syntactic analysis using dictionaries to extract features from the input data. When the search keywords match the words in the

emotion list, a vector is created from each token and the model then fits the sentences into the respective target classes. The evaluation of this approach resulted in an accuracy of 65.23 percent.

[7] Investigated the effectiveness of Support Vector Classifier (SVC), LinearSVC, Random Forest Classifier, and Decision Tree Classifier for the identification of textual emotions. The study was carried out on the Emotion Classification dataset with six emotional groups. Data preprocessing was applied to the dataset as well as feature extraction. The feature extraction method consisted of a DictVectorizer which is used to convert feature arrays represented as lists of standard python dictionary objects to NumPy/SciPy representations using scikit-learn-estimators. DictVectorizer implements nomial discrete features with "attribute-value" pairs where the value is restricted to a list of discrete possibilities without ordering. The dataset was split with a 3 to 1 ratio of 70 percent training and 30 percent testing. After implementing the models, it was found that the Random Forest Classifier had the best accuracy but the Decision Tree Classifier had the best average performance in terms of efficiency, sensitivity and f1 score at 84.8 percent, 74.2 percent, and 94.1 percent respectively.

III. Methods and Design

The methods proposed in this study consist of the following: Text generation, Data preprocessing, Feature Extraction, hyperparameter tuning, and model building. These methods proved to be beneficial through evaluation and resulting in a high accuracy.

A. Dataset description/fields

For this research, the Emotion detection dataset made available from kaggle machine learning has been utilized. [1]

B. Text Generation

The GPT-2 model, an unsupervised text generation model was used in this study to generate text for the imbalanced data classes of "love" and "surprise". The typical usage of GPT-2 consists of generating synthetic text samples in response to a textual input [3]. By feeding the GPT-2 model a set of prompts, the goal was to produce a set of text for the emotion classes of love and surprise that would help with handling the imbalances in the data and the distribution of samples per each emotion. The generated text was then applied to the orignal dataset before the preprocessing stage.

C. Data Preprocessing

Data that is unstructured can make it difficult to perform sentiment analysis and emotion detection. Preprocessing is a critical stage in data cleaning since the quality of the data significantly impacts many approaches that follow preprocessing [2]. The preprocessing methods used in this study consist of converting the text into lowercase, removal of symbols and numbers through regular expression, the removal of stop words, tokenization, and lemmatization.

[1] http://www.kaggle.com

Stop-word is an unwanted frequently occurring word which holds needless space in a dataset and expands the processing period. A sample of text is "you're", "you've", "you'll", "couldn't", "didn't", "doesn't", etc. [8]. These words hold no sentimental value and should be removed to imrove performance.

Tokenization is the process of breaking down either the whole document or paragraph or just one sentence into chunks of words called tokens [2]. For example, a sentence such as "I love programming", would be broken down into three tokens: "I", "love", "programming".

Lemmatization is the process of breaking down a word into it's root meaning in order to better identify similarities.

The NLTK library was utilized in effectively implementing stop word removal, tokenization, and lemmatization to the dataset.

"Fig 1." represents the imbalance of the emotion classes. The majority classes of 'joy' and 'sadness' dominate the atmosphere while the minority classes of 'love' and 'surprise' lack in comparison. The goal of text generation and data preprocessing is to address these imbalances and to boost the number of samples of the minority class.

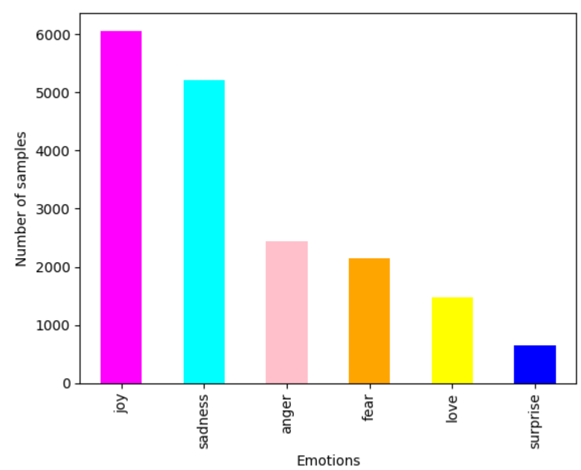

Fig. 1. Distribution of samples among the emotion classes

D. Feature Extraction

Feature extraction helps to reduce the amount of redundant data from the dataset. Through this reduction of data, the speed of learning and generalization are increased. The method that was conducted in this study is Term Frequency Inverse Document Frequency (TF-IDF). This is a weighting method that when applied, reduces the dimensionality of the feature space and the noise contained within the dataset, therefore drastically improving the outcome of the classifier [9]. A 10 to 1 ratio was applied in determining the number of features to use with TF-IDF. The dataset used in this study contained 18,000 instances, and so the number of features applied was 1800.

E. Train and Test

Splitting the data into training and testing sets is essential to avoid overfitting. This is an instance where a machine learning model fits its training data too well and fails to reliably fit additional data. In this case of this study, the standard training split of 80/20 was used where 80 percent of the data was used as training data. The preprocessing was applied to this training data and then the test set, which was the other 20 percent of the data, was evaluated using the machine learning models.

F. Hyperparameter Tuning

The hyperparameters on the GPT-2 model were tuned to best produce the results of the generated samples. The parameters include number of generated sentences, sentence length, temperature, top k and top p.

The temperature parameter was set to 0.49 because a lower temperature value allows the model to become more deterministic, leading to a more focused output.

The top k parameter was set to 100 because it allows for a broader exploration of the vocabulary and will potentially produce a more diverse output.

The top p parameter was set to 0.83, which includes a larger portion of the probability mass, leading to more diverse and creative outputs.

G. Model building

Choosing a proper model is an important task to evaluate the performance and it is dependent on the nature of the dataset and its characteristics. The model used in this study is machine learning approach, through which a voting classifier was implemented consisting of Stochastic Gradient Descent (SGD), Gradient Boosting (GB), and Logistic Regression (LR) classifiers.

A voting classifier is an ensemble machine learning model that combines the predictions of multiple individual classifiers to make a final prediction. By leveraging the strengths of these different classifiers, the overall performance is improved. A hard voting classifier was implemented in this case where the ensemble votes for a class, and the class that receives the majority of votes is the final predicted class. A hard voting approach is beneficial in terms of reducing variance, improving accuracy, it's robustness to outliers, and versatility.

IV. Results and Evaluation

The results of this study are fruitful due to the increase improvements in the classification rate of the imbalanced emotion classes. The classification report along with the confusion matrix are represented below, proving text generation and preprocessing had an effect on these classes. There is a slight trade of between precision and recall of the imbalanced classes, but the overall f1 score is higher.

When looking at the two imbalanced classes targeted in this study, "love" had a precision of 0.72, while "surprise" had a precision of 0.73. A higher precision means that the model had correct positive predictions. Precision is calculated by taking the true positives and dividing them by the true positives added with the true negatives.

$$\text{Precision} = \frac{\text{True Positives}}{\text{True Positives + False Positives}} \tag{1}$$

According the recall values, which were fairly high, it can be seen that the model was effective in capturing or identifying a large proportion of the positive instances. "love" emotion had a recall of 0.89, and "surprise" had a recall of 0.94. Recall is calculated by taking the true positives and dividing them by the true positives added with the true negatives.

$$\text{Recall} = \frac{\text{True Positives}}{\text{True Positives + False Negatives}} \tag{2}$$

The f1 scores of the imbalanced classes were improved a good amount. "love" resulted in a 0.79 f1 score and "surprise' resulted in a 0.83 f1 score. This shows that there was a good balance between the precision and the recall. The model used was effective in identifying positive instances while minimizing both false positives and false negatives. The f1 score is calculated by taking the precision multiplied by the recall and diving it by the precision added with the recall. The result is then multiplied by 2, giving us the f1 score.

$$\text{F1 Score} = 2 \times \frac{\text{Precision} \times \text{Recall}}{\text{Precision + Recall}} \tag{3}$$

"Fig 2." represents the confusion matrix for the ensemble model after evaluation. It can be seen that there is a low amount of times in which the imbalanced classes were falsely classified as another emotion.

TABLE I
Classification Report

	precision	recall	f1-score	support
anger	0.86	0.86	0.86	510
fear	0.84	0.81	0.83	378
joy	0.92	0.89	0.91	1182
love	0.72	0.89	0.79	300
sadness	0.95	0.88	0.92	1089
surprise	0.73	0.94	0.83	145
accuracy			0.88	3604
macro avg	0.84	0.88	0.86	3604
weighted avg	0.88	0.88	0.88	3604

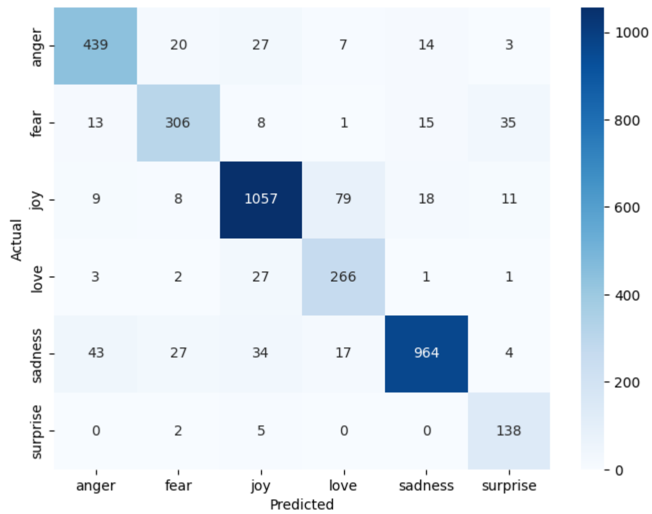

Fig. 2. Confusion Matrix for the Ensemble Model

V. Conclusion

Throughout this study, many different methods were implemented to try and the improve the classification rate of the imbalanced emotion classes 'love' and 'surprise'. A text generation approach using GPT-2, with tuned hyperparameters was done in order to improve the imbalances by generating synthetic text through the use of textual based prompts.

A preprocessing approach was applied to remove less meaningful information from the dataset and to further improve classification rate by using methods such as lowercasing, symbol and number removal, stop word removal, tokenization, and lemmatization.

Furthermore, classification was done using a hard voting classifier consisting of Stochastic Gradient Descent, Gradient Boosting, and Logistic Regression, resulting in an overall accuracy of 88 percent. For the imbalanced classes, "love" had a precision of 0.72, recall of 0.89, and f1 score of 0.79. The emotion "surprise had a precision of 0.73, recall of 0.94, and f1 score of 0.83. It can be concluded that methods such as text generation and preprocessing do in fact have an impact on the classification rate of emotion classes

References

[1] Chang, Y., Wang, X., Wang, J., Wu, Y., Zhu, K., Chen, H., Yang, L., Yi, X., Wang, C., Wang, Y., Ye, W., Zhang, Y., Chang, Y., Yu, P. S., Yang, Q., Xie, X. (2023, July 18). A Survey on Evaluation of Large Language Models. ArXiv.org. https://doi.org/10.48550/arXiv.2307.03109

[2] P. Nandwani, R. Verma. A review on sentiment analysis and emotion detection from text, 2022.

[3] A. Bartoli, E. Medvet, Exploring the potential of GPT-2 for Generating Fake Reviews of Research Papers, 2020

[4] S. Gaythari, G. Maragatham, A Survey on Text Based Emotion Detection, 2021.

[5] Bharti, S. K., Varadhaganapathy, S., Gupta, R. K., Shukla, P. K., Bouye, M., Hingaa, S. K., Mahmoud, A. (2022). Text-Based Emotion Recognition Using Deep Learning Approach. Computational Intelligence and Neuroscience, 2022, 2645381. https://doi.org/10.1155/2022/2645381

[6] M. N. Alrasheedy, R. C. Muniyandi, F. Fauzi. Text-Based Emotion Detection and Applications: A Literature Review, 2022

[7] A. Shameem, G. Rameshbabu, L. Vigneshwaran, K. Sundar, K. Veena, (2022). Text Emotion Detection Using Machine Learning And NLP, International Journal of Scientific Research in Science, Engineering and Technology. 30 May 2022, https://doi.org/10.32628/ijsrset

[8] Ab. Nasir, A. F., Nee, E. S., Choong, C. S., Abdul Ghani, A. S., Abdul Majeed, A. P. P., Adam, A., Furqan, M. (2022). Text-based emotion prediction system using machine learning, 2022.

[9] Al-Habib, Hasanuddin, et al. Text Processing Using Support Vector Machine for Scientific Research Paper Content Classification. Text Processing Using Support Vector Machine for Scientific Research Paper Content Classification — Atlantis Press, Atlantis Press, 22 May 2023, www.atlantis-press.com/proceedings/iconnsmal-22/125987144.