Multi-label Emotion Classification using Machine Learning and Deep Learning Methods

Drashti Kher and Kalpdrum Passi[a]
School of Engineering and Computer Science, Laurentian University, Sudbury, Ontario, Canada

Keywords: Multi-label Emotion Classification, Twitter, Python, Deep Learning, Machine Learning, Naïve Bayes, SVM, Random Forest, KNN, GRU based RNN, Ensemble Methods, One-way ANOVA.

Abstract: Emotion detection in online social networks benefits many applications like personalized advertisement services, suggestion systems etc. Emotion can be identified from various sources like text, facial expressions, images, speeches, paintings, songs, etc. Emotion detection can be done by various techniques in machine learning. Traditional emotion detection techniques mainly focus on multi-class classification while ignoring the co-existence of multiple emotion labels in one instance. This research work is focussed on classifying multiple emotions from data to handle complex data with the help of different machine learning and deep learning methods. Before modeling, first data analysis is done and then the data is cleaned. Data pre-processing is performed in steps such as stop-words removal, tokenization, stemming and lemmatization, etc., which are performed using a Natural Language Processing toolkit (NLTK). All the input variables are converted into vectors by naive text encoding techniques like word2vec, Bag-of-words, and term frequency-inverse document frequency (TF-IDF). This research is implemented using python programming language. To solve multi-label emotion classification problem, machine learning, and deep learning methods were used. The evaluation parameters such as accuracy, precision, recall, and F1-score were used to evaluate the performance of the classifiers Naïve Bayes, support vector machine (SVM), Random Forest, K-nearest neighbour (KNN), GRU (Gated Recurrent Unit) based RNN (Recurrent Neural Network) with Adam optimizer and Rmsprop optimizer. GRU based RNN with Rmsprop optimizer achieves an accuracy of 82.3%, Naïve Bayes achieves highest precision of 0.80, Random Forest achieves highest recall score of 0.823, SVM achieves highest F1 score of 0.798 on the challenging SemEval2018 Task 1: E-c multi-label emotion classification dataset. Also, One-way Analysis of Variance (ANOVA) test was performed on the mean values of performance metrics (accuracy, precision, recall, and F1-score) on all the methods.

1 INTRODUCTION

With the increasing popularity of online social media, people like expressing their emotions or sharing meaningful events with other people on the social network platforms such as twitter, Facebook, personal notes, blogs, novels, emails, chat messages, and news headlines (Xiao Zhang, Wenzhong Li1 and Sanglu Lu, 2017).

Emotion is a strong feeling that deriving from person's mood or interactions with each other. Many ways are available for detecting emotions from the textual data, for example social media has made our life easier and by pressing just one button everyone can share personal opinion with the whole world. Emotion can be detected from the data with the help of data mining techniques, machine learning techniques and with the help of neural networks (Avetisyan, H and Bruna, Ondej and Holub, Jan, 2016). From the examination it was expressed that emotion detection approaches can be classified into three following types: keyword based or lexical based, learning based and hybrid. The most commonly used classifiers, such as SVM, naive bayes and hybrid algorithms (Avetisyan, H and Bruna, Ondej and Holub, Jan, 2016). Emotion mining is very interesting topic in many studies such as cognitive science, neuroscience, and psychology (Yadollahi, Ali and Shahraki, Ameneh Gholipour and Zaiane, Osmar R, 2017). Whereas emotion mining from text is still in its early stages and still has

[a] https://orcid.org/0000-0002-7155-7901

Kher, D. and Passi, K.
Multi-label Emotion Classification using Machine Learning and Deep Learning Methods.
DOI: 10.5220/0011532400003318
In *Proceedings of the 18th International Conference on Web Information Systems and Technologies (WEBIST 2022)*, pages 128-135
ISBN: 978-989-758-613-2; ISSN: 2184-3252

a long way to proceed, developing systems that can detect emotions from text has many applications.

The intelligent tutoring system can decide on teaching materials, based on users mental state and feelings in E-learning applications. The computer can monitor users emotions to suggest appropriate music or movies in human computer interaction (Yadollahi, Ali and Shahraki, Ameneh Gholipour and Zaiane, Osmar R, 2017). Moreover, output of an emotion-mining system can serve as input to the other systems. For instance, Rangel and Rosso (Yadollahi, Ali and Shahraki, Ameneh Gholipour and Zaiane, Osmar R, 2017)(Rangel and Paolo Rosso,2016) use the emotions identified within the text for author identification, particularly identifying the writers age and gender. Lastly, however not the least, psychologists can understand patients emotions and predict their state of mind consequently. On a longer period of time, they are able to detect if a patient is facing depression, stress that is extremely helpful since he/she can be referred to counselling services (Yadollahi, Ali and Shahraki, Ameneh Gholipour and Zaiane, Osmar R, 2017). There is analysis on detecting emotions from text, facial expressions, images, speeches, paintings, songs, etc. Among all, voice recorded speeches and facial expressions contain the most dominant clues and have largely been studied (Carlos Busso, Zhigang Deng, Serdar Yildirim, Murtaza Bulut, Chul Min Lee, Abe Kazemzadeh, Sungbok Lee, Ulrich Neumann, and Shrikanth Narayanan, 2004)(Alicja Wieczorkowska, Piotr Synak, and Zbigniew W. Ra´s., 2006). Some types of text can convey emotions such as personal notes, emails, blogs, novels, news headlines, and chat messages. Specifically, popular social networking websites such as Facebook, Twitter, Myspace are appropriate places to share one’s feelings easily and largely.

1.1 Multi-label Classification for Emotion Classification

Emotion mining is a multi-label classification problem that requires predicting several emotion scores from a given sequence data. Any given sequence data can possess more than one emotion, so the problem can be posed as a multi-label classification problem rather than a multi-class classification problem. Both machine learning and deep learning were used in this research to solve the problem.

1.1.1 Machine Learning based Approach

For the machine learning models, data cleaning, text preprocessing, stemming, and lemmatization on the raw data were performed. The text data was transformed to vectors by using the TF-IDF method, then multiple methods were used-to predict each emotion. SVM, Naive Bayes, Random Forest, and KNN classifiers were used extensively to build the machine learning solution. After all the training, various performance metrics measures were plotted for each model concerning every emotion label as a bar plot.

1.1.2 Deep Learning based Approach

For the deep learning, dataset is loaded, then preprocessed, and encoded to perform deep learning techniques on it. From this research shows that RNN based model performs well on text data, GRU model was built with an attention mechanism to solve the problem by training for multiple epochs to obtain the best accuracy.

2 DATA AND PREPROCESSING

In this research, 10,983 English tweets were used for multi-label emotion classification from (“SemEval-2018”, 2018), (Mohammed, S., M.; Bravo-Marquez, F.; Salameh, M.; Kiritchenko, S, 2018). The dataset of emotions classification includes the eight basic emotions (joy, sadness, anger, fear, trust, disgust, surprise, and anticipation) as per Plutchik (1980) (Jabreel M., Moreno A, 2019) emotion model, as well as a few other emotions that are common in tweets which are love, optimism, and pessimism. Moreover, python 3.7.4 version was used for data preprocessing, multi-label emotion classification, and data visualization.

Data preprocessing is the most crucial data mining technique that transforms the raw data into a useful and efficient format. Real-world information is frequently inconsistent, incomplete, or missing in specific behaviours and is likely to contain lots of errors. It is a demonstrated technique of resolving such issues. It prepares raw data for further processing. Different tools are available for data preprocessing. Data preprocessing is divided into a few stages which is show in Figure 1.

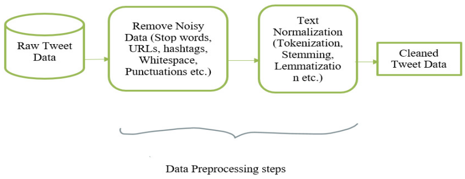
Raw Tweet Data
Remove Noisy Data (Stop words, URLs, hashtags, Whitespace, Punctuations etc.)
Text Normalization (Tokenization, Stemming, Lemmatization etc.)
Cleaned Tweet Data
Data Preprocessing steps

Figure 1: Structure of Data Preprocessing.

The data preprocessing steps that are performed before starting machine learning and deep learning methods are as follows.

Data Cleaning: For data cleaning, sometimes tweets possess certain usernames, URLs, hashtags, whitespace, Punctuations, etc., which is not helpful in machine learning algorithms to get better accuracy. Then, remove all noisy data from every tweet. All special characters are replaced with spaces. This step is performed as special characters do not help much in machine learning modeling. Every tweet is transformed into lower case. Also, duplicate tweets are identified and removed.

Remove Stop Words: Stop words are words that are finalized in the Natural Language Processing (NLP) step. "Stop words" or "Stop word lists" consists of those words which are very commonly used in a Language, not just English. Stop word removal is important because it helps the machine learning models to focus on more important words which result in more accurate prediction. Stop word removal also helps to avoid problems like the curse of dimensionality as it reduces the dimensionality of the data. It is important to note that there is a total of 179 stop words available in the English language using NLTK library (Manmohan singh, 2020).

Tokenization: In simple terms, tokenization is a process of turning sequence data into tokens. It is the most important natural language processing pipeline. It turns a meaningful piece of text into a string char named tokens.

Stemming: Stemming is a process of turning inflected words into their stemmed form. Stemming also helps to produce morphological variants of a base word. Stemming is the part of the word which adds inflected word with suffixes or prefixes such as (-ed, -ize, -s, -de, mis). So, stemming results in words that are not actual words. Stemming is created by removing the suffixes or prefixes used with a word.

Lemmatization: The key to this process is linguistics and it depends on the morphological analysis of each word. Lemmatization removes the inflectional endings of words and returns the dictionary form of the word, which is also known as "Lemma". Lemmatization also uses wordnet, which is a lexical knowledge base. Lemmatization is performed after stemming, and it is performed on the tokenized words.

3 METHODS

Machine learning and a deep learning-based approaches were used to solve the multi-label emotion recognition problem on emotion classification from twitter data. Both machine learning and deep learning algorithms were applied after applying domain knowledge-based data cleaning, NLP based data preprocessing, and feature engineering techniques. Different feature engineering and preprocessing techniques were applied for both the solutions.

3.1 Machine Learning Methods for Emotion Classification

The most popular machine learning methods such as Naïve bayes, SVM, Random Forest, and KNN have been discussed in this section. For the Machine learning models, data cleaning, text preprocessing, stemming, and lemmatization on the raw data were performed. Feature engineering converts the text/string data to a format that machine learning algorithms would interpret. It is an important step before applying any of the mentioned machine learning algorithms. The overview of applying machine learning techniques to the emotion classification labeled data and analysis is shown in Figure 2.

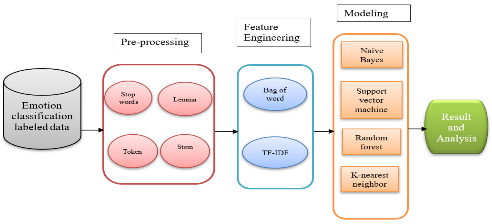

Figure 2: Overview of applying machine learning techniques.

Feature Engineering: The cleaned and preprocessed tokens of tweets are obtained after all the preprocessing where each token is a "string". Machine learning models cannot work with strings, they only work with numbers. The tokens are transformed into numbers by using the methods given below.

Bag of Words (BOW)

Term frequency and Inverse document frequency (TF-IDF)

It is always a better idea to use TF-IDF rather than BOW as the TF-IDF feature engineering technique also preserves some semantic nature of the sequence. For this research, the TF-IDF feature engineering technique was used to encode tokens as numbers.

Naïve Bayes: Naive Bayes is a machine learning classifier and it used to solve classification problems. It uses Bayes theorem extensively for training. It can solve diagnostic and predictive problems. Bayesian Classification provides a useful point of view for evaluating and understanding many learning algorithms. It calculates explicit probabilities for hypothesis, and it is robust to noise in input information (Hemalatha, Dr. G. P Saradhi Varma, Dr. A. Govardhan, 2013). In this multilabel classification, single Naive Bayes model is trained for predicting each output variable.

Support Vector Machine: The support vector machine is a supervised learning distance-based model. It is extensively used for classification and regression. The main aim of SVM is to find an optimal separating hyperplane that correctly classifies data points and separates the points of two classes as far as possible, by minimizing the risk of misclassifying the unseen test samples and training samples (García-Gonzalo, E., Fernández-Muñiz, Z., García Nieto, P.J., Bernardo Sánchez, A., Menéndez Fernández, M, 2016). It means that two classes have maximum distance from the separating hyperplane.

Random forest: It is an ensemble learning method for classification and regression. Each tree is grown with a random parameter and the final output is achieved by aggregating over the ensemble (R. Gajjar and T. Zaveri, 2017). As the name suggests, It is a classifier that contains a number of decision trees on different subsets of the given dataset and takes the average to improve the predictive accuracy of that dataset. Rather than depending on one decision tree, the random forest takes the prediction from each tree and based on the majority votes of predictions, and it predicts the final output.

K-Nearesr Neighbor: K-Nearest Neighbor is one of the simplest Machine Learning algorithms based on Supervised Learning technique. It assumes the similarity between the new data and available data and put the new data into the category that is the most similar to the available categories. It reserves all the available data and classifies a new data point based on the similarity. This means when new data comes out then it can be easily classified into a well suite category by using K- NN algorithm. It can be used for Classification as well as for Regression but mostly it is used for the Classification problems. KNN algorithm at the training phase just stores the dataset and when it gets new data, then it classifies that data into a different category that is much similar to the new data.

3.2 Deep Learning based Emotion Classification

Deep learning adjusts a multilayer approach to the hidden layers of the neural network. In machine learning approaches, features are defined and extracted either manually or by making use of feature selection methods. In any case, features are learned and extricated automatically in deep learning, achieving better accuracy and performance. Figure 3 shows the overview of deep learning technique. deep learning currently provides the best solutions to many problems in the fields of image and speech recognition, as well as in NLP.

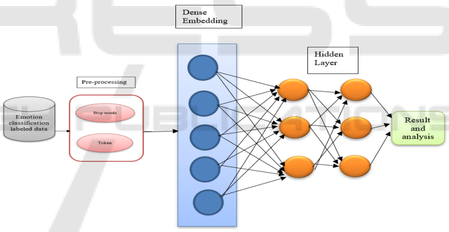

Figure 3: Overview of applying deep learning techniques.

Feature Exrtraction: Feature extraction is the name for methods that combine and/or select variables into features, effectively reducing the amount of data that must be processed, while still accurately and completely describing the original data set.

Word Embedding: Word embeddings are the texts changed into numbers and there may be different numerical representations of the same content. As it turns out, most of the machine learning algorithms and deep learning architectures are unable to process strings or plain text in their raw form (NSS, 2017). They require numbers as inputs to perform any sort of work, which is classification, regression etc. Moreover, with the huge amount of data that is present within the text format, it is basic to extract knowledge out of it and build applications (NSS, 2017). So, word embeddings are used for converting all text documents into a numeric format.

Word2vec: It could be a two-layer neural net that processes text (Pathmind Inc., 2022) . The text corpus takes as an input, and its output may be a set of vectors. Whereas it is not a deep neural network it turns text into a numerical form that deep neural network can process. The main purpose and usefulness of Word2vec is to group the vectors of similar words together in vector space (Pathmind Inc., 2022).

Gated Recurrent Unit based Recurrent Neural Network: In this research, Simple recurrent neural networks are not used because they do not have long term dependencies. The way to solve gated recurrent units used. For solving, the vanishing gradient problem of a standard RNN, GRU uses two gates: update gate and reset gate. GRUs can be trained on data stored for a long time without removing irrelevant data or cleaning the data.

4 RESULTS

The following evaluation parameters were used to evaluate the performance of the classifiers.

Accuracy: It is a ratio of correctly predicted emotion class to the total number of observation emotion class.

Precision: It is a ratio of correctly predicted emotion class to the total number of positive predicted class.

Recall: It is a ratio of correctly predicted positive emotion class to all observation in true actual class.

F1 score: F1 score is the degree of calculating the weighted average of precision and recall. It ranges between 0 to 1 and it is considered perfect when it is 1 which means that the model has low false positives and low false negatives.

Confusion Matrix: A confusion matrix is used for summarizing the performance of a classification algorithm.

The most commonly used performance evaluation metrics for classification problems are accuracy, Precision, recall and F1 score. Evaluation parameters are measured with the help of confusion matrix.

Figure 4 shows that the Naïve Bayes classifier achieved the best performance with respect to precision (0.80) on average of all emotions. Moreover, KNN method has high precision for Pessimism (0.951) emotion compared to the other methods but did not perform well overall compared to Naïve Bayes. For precision, machine learning methods achieved better result compared to deep learning methods. For deep learning models, GRU based RNN with RmsProp optimizer (0.59) performed well compare to Adam optimizer (0.52).

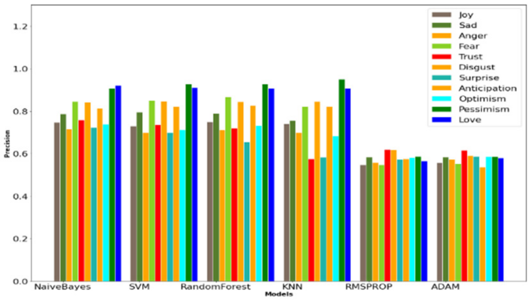

Figure 4: Precision of various algorithms at emotion category.

Figure 5 shows that the Random Forest classifier achieved the best performance with respect to recall (0.819) for average of all emotions. Also, SVM and Naïve Bayes perform well with a recall of 0.81 and 0.815, respectively. Moreover, K-nearest Neighbor (KNN) classifier has low recall value for trust (0.465) and surprise (0.384) emotion but overall KNN performed well with an average recall of 0.749. For deep learning methods, GRU based RNN with RmsProp optimizer (0.632) performed well compare to Adam optimizer (0.452). Figure 5 shows the recall of the classifiers for each emotion category.

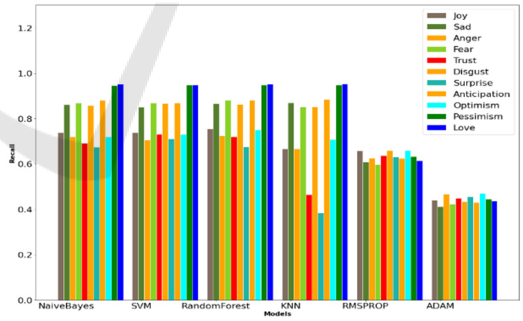

Figure 5: Recall of algorithms at emotion category.

Figure 6 shows that the support vector machine (SVM) classifier achieved the best performance with respect to F1 score (0.798) for average of all emotions. Moreover, K-nearest Neighbor (KNN) classifier has quite low result (0.671) compared to Random Forest (0.794), Naïve Bayes (0.762), and SVM. For deep learning models, both the models performed similar in all emotions. But GRU based RNN with RmsProp optimizer (0.595) performed well compare to Adam optimizer (0.486). Figure 6 shows F1 score of the classifiers for each emotion category.

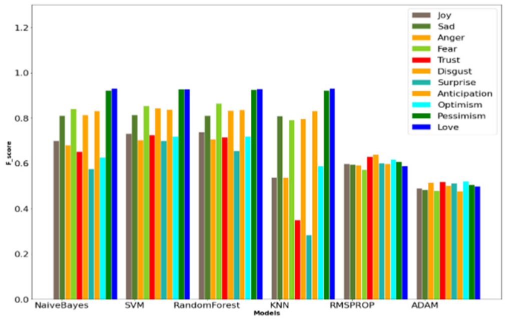

Figure 6: F1 score of algorithms at emotion category.

Notice that GRU based RNN with RmsProp, Random Forest and SVM perform relatively better over other methods. The efficacy of the task is achieved through the ensemble modelling. In ensemble modelling, the predictions of different models are combined to produce improved performance over any individual model in classifying the emotions. This approach helps in reducing the variance and improves the generalization. The following two popular ensemble techniques have been used in this study: (i) majority voting, and (ii) weighted average.

In majority voting approach, predictions of different algorithms have been combined and the majority vote is predicted. In weighted average approach, predictions of algorithms have been combined with certain weightage. The weightage of each algorithm is generally assigned based on the individual performance of that algorithm on the data. In this research, F1 score of the algorithm is considered to be its weight.

The ensemble methods combine the predictions of all the other methods to produce an improved prediction. These ensemble methods considered in this research are parallel in nature which means all the models are independent of each other. Figure 7 shows that both ensemble techniques achieved the best result with respect to precision (0.818, 0.813), recall (0.829, 0.83) and F1 score (0.789, 0.799) for average of all emotions respectively. Moreover, both the ensemble techniques perform better than any individual method. Figure 7 compares performance metrics of ensemble methods against other individual algorithms.

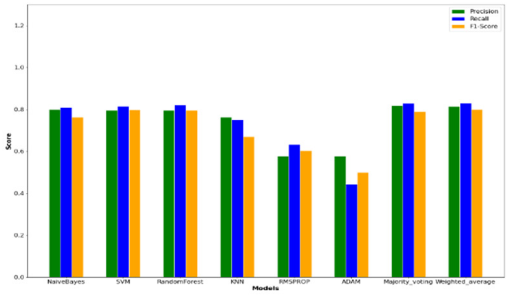

Figure 7: Comparison of performance metrics of algorithms against ensemble methods.

Mohammed et al. (Jabreel M., Moreno A, 2019) achieved 0.59 accuracy, 0.57 precision, 0.61 recall, and 0.56 F1 score using GRU based RNN classifier, which was used in this research as a reference. In comparison, different classifiers were used for measuring all evaluation parameters for emotion classification labeled data set. GRU based RNN with RmsProp optimizer classifier gave high accuracy for multi-label emotion classification from emotion classification dataset (SemEval-2018), even though, other methods give better performance. Table 1 shows that the comparison of all methods for emotion classification dataset.

Table 1: Comparison of all methods.

Number	Parameters	Naïve Bayes	SVM	Random Forest	KNN	GRU based RNN with Adam Optimizer	GRU based RNN with RmsProp Optimizer
1	Accuracy	0.809	0.815	0.819	0.757	0.79	0.823
2	Precision	0.80	0.794	0.794	0.762	0.526	0.596
3	Recall	0.812	0.815	0.82	0.75	0.452	0.632
4	F1-score	0.762	0.798	0.794	0.67	0.486	0.595
5	AUC	0.79	0.81	0.79	0.59	0.81	0.84

Table 2: ANOVA test results on performance metrics.

Metric	Naïve Bayes	SVM	Random Forest	KNN	RmsProp	Adam	Majority voting Method mean	Weighted average method mean	P-value
Precision	0.80	0.798	0.80	0.736	0.607	0.539	0.819	0.814	$6.85*10^{-9}$
Recall	0.812	0.819	0.824	0.763	0.588	0.463	0.829	0.832	$1.72 *10^{-8}$
F1 Score	0.766	0.80	0.801	0.70	0.581	0.497	0.789	0.802	$1.36 * 10^{-14}$
Accuracy	0.812	0.819	0.824	0.763	0.827	0.795	0.817	0.805	$1.4 * 10^{-5}$

Overall, the better performance is achieved by using machine learning methods for all evaluation parameters. But GRU based RNN with Rmsprop optimizer performed the best in terms of accuracy, with the highest accuracy (0.823) compared to other classifiers. The results also show a huge improvement compared to the results of Mohammed et al. (Jabreel M., Moreno A, 2019) for the same dataset. Figure 8 shows that comparison of all evaluation parameters using different classifiers.

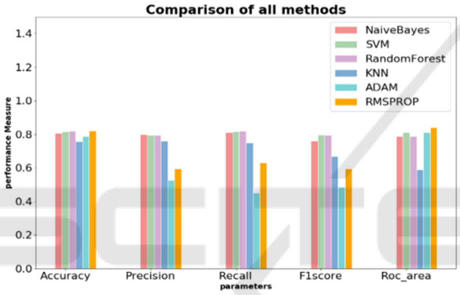

Figure 8: Comparison of all methods.

Statistical Analysis: To conclude, or to choose the best method from these ensemble methods as well as all classifiers, statistical one-way ANOVA test was performed. Test for statistical significance helps to measure whether the difference between the performance metrics observed via all methods is significant or not.

In this research, One-way Analysis of Variance (ANOVA) test is performed on the mean values of performance metrics on all the methods (shown in Table 2). The null hypothesis (H0) states that all models demonstrate similar performance. H0 is accepted if no statistically significant difference ($P > 0.05$) is observed in the mean value of the performance metrics for the different models under study. The alternate hypothesis (H1) is accepted and H0 is rejected if a statistically significant performance difference ($P < 0.05$) is found to exist (S. Rajaraman, Sameer K. Antani, 2020). One-way ANOVA is an omnibus test and needs a post-hoc study to identify all the methods demonstrating this statistically significant performance differences (S. Rajaraman, Sameer K. Antani, 2020).

Table 2 summarizes the ANOVA test results for performance metrics. It is observed that the P -values are lower than 0.05 for the performance metrics. This means that the methods are statistically significant (null hypothesis H_0 is rejected) when evaluated on the basis of these performance metrics. F1 score is the consonant mean of both precision and recall. It is a better measure of incorrectly classified cases and used when it needs to maintain higher precision and recall instead of just focussing on one. In this study, the mean value of F1 score is higher for weighted average ensemble method (0.802) compared to that of majority voting ensemble method (0.789). This shows, that weighted average method has proved to be the best model in view of achieving higher F1 score and model built using weighted average method would result in higher F1 score over other methods.

5 CONCLUSIONS

In this research, twitter data was analysed for emotion classification. Since each tweet is associated with multiple emotions not just limited to one, this problem has been formulated as multi-label emotion classification. The popular machine learning classifiers and GRU based Recurrent Neural Network with Adam and RmsProp optimizer were used to solve multi-label emotion classification problem.

The popular ensemble techniques such as Majority voting and Weighted average methods were used for reducing the variance and improve the generalization. These methods have been proved to be more accurate in terms of all the performance metrics (accuracy, precision, recall, and F1 score). Also, One-way Analysis of Variance (ANOVA) test is performed on the mean values of performance metrics on all the methods.

From the results, it is concluded that accuracy increased from 0.59 to 0.823 using GRU based RNN with RmsProp optimizer classifier which is 23.3% (0.233) higher, precision increased from 0.57 to 0.80 using Naive Bayes classifier which is 23% (0.23) higher, recall increased from 0.56 to 0.82 using Random Forest classifier which is 26% (0.26) more and F1 score increased from 0.56 to 0.798 using SVM which is 23.8% (0.238) higher than Mohammed et al. (Alicja Wieczorkowska, Piotr Synak, and Zbigniew W. Ra´s., 2006) research paper results on emotion classification dataset (SemEval-2018). Highest value of AUC (0.84) was achieved for GRU based RNN with RmsProp optimizer. For visualization, Matplotlib library was used in Jupyter Notebook to compare all the results using machine learning and deep learning methods.

Future Work: In the future, the present analysis can be extended by adding more feature extraction parameters and different models can be applied and tested on different datasets. The present research focusses on establishing the relations between the tweet and emotion labels. More research can be done in the direction of exploring relations between the phrases of tweet and emotion label. Transfer learning with some existing pre-trained models for classification and data fusion from different data sources can be a good direction to explore to improve the robustness and accuracy. In this study, dataset comes from only twitter source, but other social networks can be used for creating this type of dataset. For this research, emotion classification dataset was used from the research paper of Mohammed et al., but new dataset can be created to explore the same problem.

REFERENCES

Xiao Zhang, Wenzhong Li1, Sanglu Lu. (2017). *Emotion detection in online social network based on multi-label learning,*

Database Systems for Advanced Applications- 22nd International Conference, pp. 659-674

Avetisyan, H and Bruna, Ondej and Holub, Jan. (2016). *Overview of existing algorithms for emotion classification*

Uncertainties in evaluations of accuracies, Journal of Physics: Conference Series, vol:772

Yadollahi, Ali and Shahraki, Ameneh Gholipour and Zaiane, Osmar R. (2017). *Current State of Text Sentiment Analysis from Opinion to Emotion Mining,* ACM. Survey , pp.1-25

Rangel and Paolo Rosso. (2016). *On the impact of emotions on author profiling*, Information Processing & Management 52, pp.73–92

Carlos Busso, Zhigang Deng, Serdar Yildirim, Murtaza Bulut, Chul Min Lee, Abe Kazemzadeh, Sungbok Lee, Ulrich Neumann, and Shrikanth Narayanan. (2004). *Analysis of emotion recognition using facial expressions, speech, and multimodal information*, In Proceedings of the 6th International Conference on Multimodal Interfaces. ACM, pp. 205–211

Alicja Wieczorkowska, Piotr Synak, and Zbigniew W. Ra´s. (2006). *Multi-label classification of emotions in music*, In

Intelligent Information Processing and Web Mining. Springer, pp. 307–315

SemEval-2018 Task 1: Affect in Tweets (Emotion Classification Dataset):

https://competitions.codalab.org/competitions/17751#learn_the_details-datasets

Mohammed, S., M.; Bravo-Marquez, F.; Salameh, M.; Kiritchenko, S. (2018). *Semeval-2018 task 1: Affect in Tweets*, In

Proceedings of the 12th InternationalWorkshop on Semantic Evaluation, New Orleans, LA, USA, pp. 1–17

Jabreel M., Moreno A. (2019). *A Deep Learning-Based Approach for Multi-Label Emotion Classification in Tweets*, Appl. Sci. 9:1123. doi: 10.3390/app9061123

Manmohan singh. (2020). *Stop the stopwords using different python libraries*, https://medium.com/towards-artificial-intelligence/stop-the-stopwords-using-different-python-libraries-ffa6df941653

Hemalatha, Dr. G. P Saradhi Varma, Dr. A. Govardhan. (2013). *Sentiment Analysis Tool using Machine Learning Algorithms*,

IJETTCS, Vol 2, Issue 2

García-Gonzalo, E., Fernández-Muñiz, Z., García Nieto, P.J., Bernardo Sánchez, A., Menéndez Fernández, M. (2016). *Hard- Rock Stability Analysis for Span Design in Entry-Type Excavations with Learning Classifiers*, *9*, 531, DOI: https://doi.org/10.3390/ma9070531

R. Gajjar and T. Zaveri. (2017). *Defocus blur radius classification using random forest classifier*, 2017 International Conference on Innovations in Electronics, Signal Processing and Communication (IESC), pp.219-223, DOI: https://doi.org/10.1109/IESPC.2017.8071896

NSS (2017). “*An intuitive understanding of Word Embedding: From Count vectors to word2vec*”,

https://www.analyticsvidhya.com/blog/2017/06/word-embeddings-count-word2veec

“*A Beginner’s guide to word2vec and neural word embeddings*”, https://wiki.pathmind.com/word2vec

S. Konstadinov. (2017). *Understanding GRU networks*, https://towardsdatascience.com/understanding-gru-networks- 2ef37df6c9be

S. Rajaraman, Sameer K. Antani. (2020). *Modality-specific deep learning model ensembles toward improving TB detection in chest radiographs*, IEEE access: practical innovations, open solutions vol.8 :27318-27326, DOI: 10.1109/access.2020.2971257.