3rd International Conference in Technology, Humanities and Management
ICTHM 2025
28-30 January 2025
Maldives

TEXT MINING & NLP-DRIVEN EMOTION PREDICTION USING MACHINE LEARNING & DEEP LEARNING APPROACHES

Pyae Phyo Hein, Min Khant Ko, Kaung Set Kyaw
Department of Information and Communication Technology, Rangsit University, Thailand.

Pyae Phyo Hein: pyiphyohein7@gmail.com

Abstract

Text-based communication has become a common way for people to communicate their emotions in the current digital era, especially on social media. However, effectively understanding human emotions from textual data is still difficult. By combining text mining, machine learning (ML), deep learning (DL), and natural language processing (NLP) approaches, this work seeks to create a reliable emotion prediction system. The study made use of a publicly available Kaggle dataset that contained over 600,000 Twitter samples that were divided into six emotional categories: anger, joy, sadness, fear, surprise, and love. To guarantee the best possible data quality, extensive preprocessing techniques were used, such as class balancing, stemming, stop-word removal, handling chat words, tokenization, and feature extraction (TF-IDF and embeddings) etc. In this paper four models were investigated: **Support Vector Machine (SVM)**, **Extreme Gradient Boosting (XGBoost), Bidirectional Long Short-Term Memory (BiLSTM)** and **Convolutional Neural Network (CNN)**. Clear performance evaluation standards such as accuracy, precision, recall and F1-score are tested. BiLSTM achieved the highest model accuracy of **95.2%** followed by CNN 92.6%, SVM 91.3% and XGBoost 90.6% on Testing Data. The set results indicate how accurately deep learning models work in emotional expressions with BiLSTM demonstrating the high potential in handling sequential data. The results are also discussed with comparative tables, accuracy-loss graphs and confusion matrixs. This study provides the importance of integrating advanced NLP and modeling techniques for emotion prediction and sets a foundation for future research in improving text-based emotion analysis systems.

Keywords: *Emotion prediction, Text mining, Natural Language Processing (NLP), Machine Learning (ML), Deep Learning (DL)*

1. INTRODUCTION

Emotion is a dynamic cognitive and physical state that reacts to inputs like experiences, ideas, or social interactions. It includes subjective experience, cognitive processes, behavioral impacts, communication, and physiological responses. Given that machines are made to imitate humans for a variety of reasons, research on human emotion is especially important for understanding human-robot interaction and brain-computer evaluation. Therefore, a comprehensive study of human emotions and automated human emotion recognition is necessary (Khare et al., 2024). Emotion detection in literature is the process of determining the author's intended tone or emotional state through the use of language cues. To describe emotion in text, natural language processing (NLP) models look at word and phrase patterns that are often associated with specific emotions. While "sad" or "disappointed" denote negative emotions, words like "excited" or "thrilled," for example, typically signify positive ones. Through the use of linguistic analysis, readers or NLP models can identify the sentiment (whether positive, negative, or neutral) and deeper emotional states of a writer. The tone, word choice, sentence structure, and context all reveal the underlying feelings and emotions that are expressed in the text (Poria et al., 2019).

The advancements in text mining, machine learning, and natural language processing have made it easier to predict emotions in text automatically. Technology for emotion identification involves several steps: Text preprocessing: Tokenization, normalization, and stop word removal prepare the data for analysis. Feature Extraction: Techniques such as Bag of Words, Term Frequency-Inverse Document Frequency (TF-IDF), and word embeddings (such as Word2Vec, GloVe, or BERT) are used to convert textual data into numerical representations. Not only that, Machine learning algorithms (such neural networks, Naive

Bayes, and Support Vector Machines) are used to train models to classify emotions in text (Kalage & Bhuskade, 2018).

ML algorithms are trained on labeled emotional datasets in order to recognize and classify emotions. Although unsupervised methods can detect clusters of emotions in unstructured data, supervised learning methods are commonly used to classify emotions. Text mining is frequently employed as a first step in pipelines for emotion detection since it can identify patterns, extract keywords, and categorize text. Machines can understand human language and decipher emotional signs thanks to natural language processing (NLP), which includes techniques like named entity identification, sentiment analysis, and syntactic parsing. When combined, these technologies enable the prediction of emotions across many levels of text analysis, from individual words to the overall context (*Stanford CS 224N | Natural Language Processing with Deep Learning*, n.d.).

Emotion prediction is useful in business for evaluating reviews, social media posts, and customer feedback to determine client mood and enhance goods and services. Also in healthcare it can reveal information about the emotional and mental health of patients. Systems can, for instance, look for indications of anxiety or sadness in social media posts or therapy notes, supporting mental health and early intervention. At the nexus of machine intelligence, language processing, and human psychology, emotion prediction from textual data has potent applications in a variety of fields (Ramirez-Atencia et al., 2020).

The rest of the paper is structured as follows: **Section 2**: Literature Review; where covering background information and previous research that had focused on text-based emotions. **Section 3:** Methodology; the process flow, strategy, materials, and step-by-step techniques for the experiment, **Section 4**: Results and Outcomes; present and discuss each of the experiment's results and **Section 5:** Conclusions; the results and summary of the study and recommendations for the future research.

2. LITERATURE REVIEW

Emotion Prediction from textual data is an interesting subject due to its usage in fields such as in marketing, political science, psychology and human-computer interaction (Seyeditabari et al., 2018). The aim of emotion detection is to classify text according to distinct emotional states, such as joy, sadness, anger, or fear. Analysis of such information is valuable in the fields of studies where understanding the underlying emotions of textual data can inform decisions and improve user experiences Natural Language Processing (NLP) and machine learning have made it possible to develop more complex models for emotion detection, yet there are still challenges in achieving the accuracy, interpretability and applicability across varied types of text data.

Sentiment analysis provides a broad categorization of text, by using opposite tones, like positive, negative or neutral. And this has been proven useful and has been utilized in customer feedback analysis, marketing and social media monitoring (Cambria et al., 2013). Emotion detection, on the other hand, can provide a better approach by recognizing specific emotions, For example, (Lerner & Keltner, 2000) demonstrated that fear and anger represent negative feelings, however it provides different behavioral outcomes: people with fear show caution, while people who are angry are more aggressive. This difference shows the value of emotion detection, understanding the emotions in various analyses and providing better targeted responses.

Emotion detection techniques have been developed and improved, moving from simple lexicon based methods to complex deep learning models. These methods vary in complexity, types of emotions and resource requirements that they can accurately detect.
Lexicon Based Approaches: Lexicon-based methods use existing lists of words associated with specific emotions like NRC Emotion Lexicon (Mohammad & Turney, 2013) and Wordnet-Affect (Strapparava & Mihalcea, 2008). These resources are effective for identifying clear emotional expressions but struggle with unclear or fanciful language.
Supervised Machine learning techniques: Supervised machine learning models, such as Support Vector machines(SVM) and naive bayes, require labeled datasets and rely on features such as bag of words or n

grams. These models are most effective to use in studies that use large social media dataset. For instance, (Dzikovska et al., 2012) used SVM to classify emotions on Twitter using hashtags and emoticons as stand-ins for labels, achieving an accuracy of 82% for detecting emotions like happiness. (Wang, n.d.) used a similar method with Naive Bayes and SVM on Twitter data, reporting an F-measure as high as 0.72 for emotions like joy but lower performance for other emotions, revealing the difficulty in achieving high accuracy for multiclass emotion classification.
Deep learning approaches: such as Convolutional Neural Networks(CNNs) and Recurrent Neural Networks(RNNs), including Long Short Term Memory(LSTM) networks, have shown to be effective for emotion detection due to their ability to capture complex patterns and informations in text. (Hasan et al., 2019) applied a CNN-LSTM hybrid model for Twitter data, achieving significant improvements over traditional machine learning models, especially in complex emotions.

The introduction of deep learning and transformer based models has marked a significant improvement in emotion detection accuracy. Deep learning models like CNN and LSTM can handle a series of texts and capture complex data, which are important for understanding emotions that are expressed indirectly. (Hasan et al., 2019) and (Wang, n.d.) have shown that deep learning models perform better than the traditional methods particularly on social media data, which often contain informal and expressive language Transformer models, particularly BERT, demonstrate high accuracy and the ability to handle nuanced expressions. By providing context-sensitive descriptions , BERT based models have addressed many of the limitations faced by earlier methods. However, there are still errors, where these models require extensive computational resources and are vulnerable to data imbalances, which can have negative impact on accuracy for less frequently expressed emotions.

Emotion detection systems continue to face several challenges.The central concern is the lack of sufficiently high quality, balanced datasets which are important for the training of models. The current datasets are also lacking proportion, with more focus on some emotions than others, thus creating biases in classifiers (Seyeditabari et al., 2018). Moreover, some problems, such as vague expressions and metaphoric languages, still remain, for example, 'Lost his cool' which could imply anger, although anger was not mentioned, will require the models to understand the emotion from the context in which it was used. These weaknesses can be fixed by adding common-sense knowledge into emotion detection with models like, SenticNet (Cambria et al., 2014) might provide models with better interpretation of the vague expressions. We believe that researchers need to focus more on developing larger, balanced datasets, improving model prediction and refining techniques for handling context-dependent emotions. By addressing these difficulties, emotion detection models can be more reliable and applicable to a wide range of real world applications.

3. METHODOLOGY

This section of the paper provides approaches used to create the methodology explaining step by step text mining and natural language processing (NLP) techniques for preprocessing, machine learning (ML) and deep learning (DL) models to predict emotions in text. The methodology involves a systematic process starting from collecting data, preprocessing the data to extract features, comparing and selecting a model that has the ability to produce strong and more accurate results as seen in Fig 3.1. Our experiment will use a sizable corpus of textual data, guaranteeing a rich dataset for testing and training in order to get reliable and robust performance results.

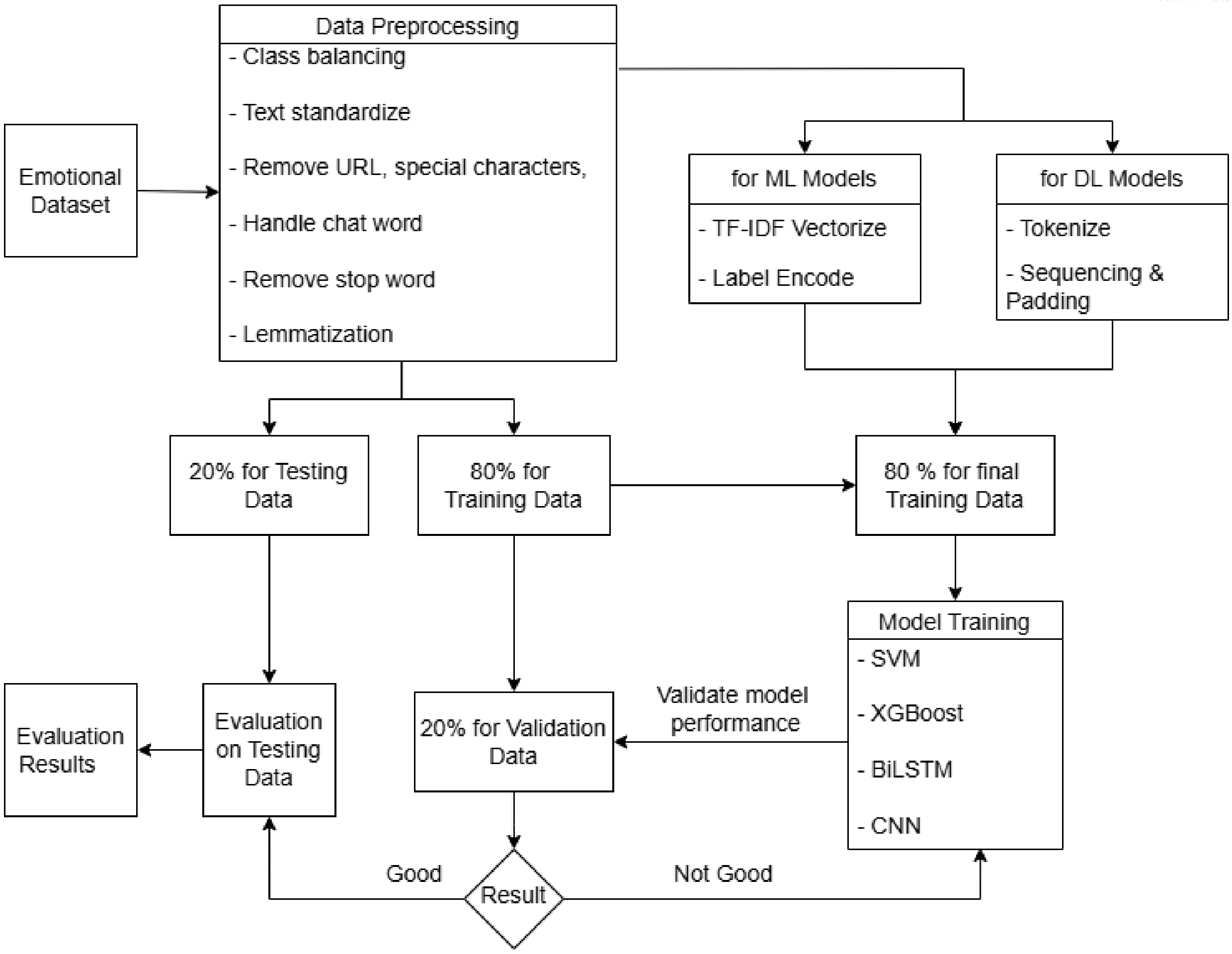


Fig 3.1: Process Flow of Methodology

3.1 Data Source

In this study, we use the **Emotion Dataset** (*Emotions*, n.d.) from Kaggle, a publicly available dataset created nine months ago by scraping tweets from Twitter. This dataset was selected because of the fact that it offers a vast and varied collection of more than 600,000 text samples that are each classified with one of six different emotions: **sadness, joy, love, anger, fear and surprise.** We believe this dataset is relatively relevant for training emotion prediction models, because of recent collections and language usage that is typical on social media, the dataset covers a broad range of language, slangs and expression styles. The model, because of its wide representation, can better detect emotions in real world text data and is therefore a better fit for fields such as emotion classifying, social media insights, and customer feedback analysis.

3.2 Programming language and Tools

The main language that is being used in the project is Python **3.11** due to its richness of Deep Learning (DL), Machine Learning (ML), and Natural Language Processing (NLP) packages. For text mining NLP preprocessing we use libraries such as **NLTK** and **spaCy**. To build machine learning models, we use a popular library named **scikit-learn** which provides tools for various types of Machine learning algorithms. Deep learning models are using **TensorFlow** and **Keras**, which provides awesome support for neural network training and design. Finally, we show the model performance on the **Seaborn** and **Matplotlib**. Because of unavailable powerful hardware resources, we will use **Google Colab** as an IDE where we can use the Cloud environment which provides the NVIDIA T4 GPU.

3.3 Data Preprocessing

In the data preprocessing stage, we apply a series of Text Mining & NLP techniques to clean and prepare the text data for model training.Here are the step by step process we did in the stage of Data

Preprocessing:

Data Balancing

Although the initial dataset was nearly 400K text data, it was unbalanced across emotion classes, which can hinder model accuracy. To handle this, we decided to downsample to the smallest output class frequency to ensure that all classes are balanced. Setting the sample count at **14,972** text samples per class we used the least frequent class as the baseline. This was an improvement over previous techniques because the current method ensured that each category of emotions was well represented and the model was made fair for all emotions.

Basic Data Cleaning

1. **Column Selection:** Removed unnecessary columns, retaining only text and label columns needed for analysis.
2. **Text Standardization**: Converted all text to lowercase to avoid case sensitivity issues.
3. **URL and Special Character Removal:** Stripped URLs, special characters, and extra whitespace to ensure clean, readable text. The reason is that these characters do not contribute meaningful information for emotion prediction and add more noise to data. By removing,it will help to streamline the text data, making it more standardized and easier for the model to interpret accurately.
4. **Handling Chat words:** As the data was scraped from Twitter, we removed emojis, replaced slang words, and any "chat speak" (e.g., "u" for "you"), as these can introduce noise into our emotion prediction models.
5. **Stopword Removal:** Removed common stopwords to focus the models on the most meaningful words.
6. **Stemming**: Applied stemming to reduce words to their root forms (e.g., "playing" becomes "play"), which helps the model focus on the core meaning of each word which can focus and contribute a lot in the model's interpretability to detect emotions.

Exploratory Analysis

Not only that**,** To gain insights into each emotion type, we generated word clouds for each class. This visualization helped identify commonly used words in each emotion, enhancing our understanding of the data and informing the model setup.

Data Splitting

After preprocessing, we split the data:

- **80% for training** and **20% for testing** to evaluate model performance.
- Within the training set, we further split out **20% for validation** to fine-tune the model during training.

Preprocessing for Machine Learning (ML) Models

For traditional ML models, we performed:

- **TF-IDF Vectorization**: Text data was converted into numerical characteristics using **TF-IDF** (Term Frequency-Inverse Document Frequency), which is a popular technique in NLP. TF-IDF allows the model to know which words are the most important by first capturing the significance of each word from a document relative to the whole dataset.
- **Label Encoding**: Encoded each emotion label as a numerical value, preparing it for use in classification algorithms.

Preprocessing for Deep Learning (DL) Models

For DL models, we applied:

1. **Tokenization**: Tokenized the text with a vocabulary limit of 60,000 words, using a special token

for out-of-vocabulary(**OOV**) words.

2. **Embedding Preparation**: Set an embedding dimension of 64 to represent each word as a vector in a lower-dimensional space.
3. **Sequencing and Padding**: Converted text to fixed-length sequences and applied padding to ensure uniform input shapes for DL models.

This comprehensive preprocessing pipeline prepared our dataset for both ML and DL approaches, providing a strong foundation for training and improving model's emotion prediction accuracy.

3.4 Model & Algorithms

In this work, we investigated and created several models both deep learning and traditional machine learning methods to come up with reliable emotion prediction. We selected **Support Vector Machine (SVM)** and **Extreme Gradient Boosting (XGBoost)** for machine learning , moreover **Bidirectional Long Short-Term Memory (BiLSTM)** and **Convolutional Neural Network (CNN)** for deep learning.

Support Vector Machine (SVM)

Support Vector Machines(SVM) is a classification framework that operates by finding the optimal hyperplane that separates data points belonging to different classes in a feature space The hyperplane is determined by maximizing the margin , which is the distance between the hyperplane and the nearest data points of each class (called support vectors). Key Components of the SVM are support vectors which are the critical data points that determine the hyperplane, kernel trick which allows the SVM to work efficiently in nonlinear settings and optimization used to find the optimal hyperplane (Yue et al., 2003). In our experiment the SVM model with parameters of(C=0.1 and kernel ='linear') is applied as the best SVM model.

Extreme Gradient Boosting(XGBoost)

Extreme Gradient Boosting(XGBoost) uses the gradient boosting framework to optimize loss function by adding decision trees to minimize residual errors. XGBoost can handle sparse datasets with missing or zero values. It assigns a default direction for missing values, enabling efficient split computation and robustness. XGBoost also uses a depth-first tree building approach by pruning trees by removing splits with a negative gain, ensuring compact and effective trees (Chen & Guestrin, 2016). The Model parameters of (learning_rate=0.1, max_depth=10, n_estimators=100) are set up for our experiment.

Bidirectional Long Short-Term Memory (BiLSTM)

BiLSTM is a type of Recurrent Neural Network (RNN) that processes sequential data in both forward and backward directions. This allows it to capture information from both past and future contexts. For applications like natural language processing, it is essential to be able to capture long-range dependencies in the data, which is made possible by this bidirectional technique. BiLSTM outperforms conventional RNNs by processing the input in both directions, which helps it comprehend the context of each sequence piece (Sun et al., 2023). For our Model, we developed BiLSTM model with total 9 layers: Embedding as a input layer, 2 bidirectional layers with 256, 128 neurons respectively, 3 pairs of dropout and Dense layer with 0.5 dropout and 'relu' activation, and among them last dense layer served as output layer with 6 classes and 'softmax' activation function. Then the model is compiled with 'categorical_crossentropy' loss function and 'adam' optimizer. In figure 3.2 (*The Architecture of the BiLSTM Model.*, n.d.), an example architecture showing bidirectional layer for BiLSTM Model.

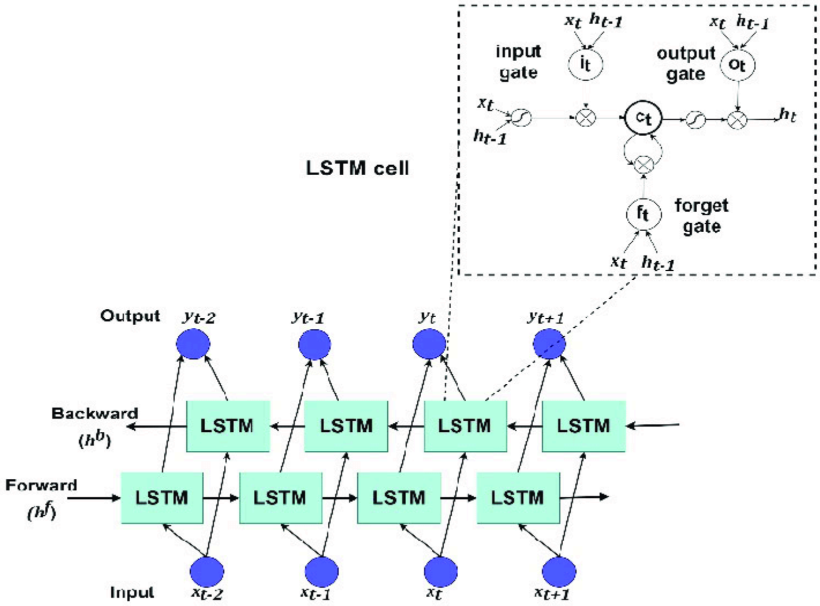


Fig 3.2: Example Architecture of BiLSTM Model

Convolutional Neural Network (CNN)
Convolutional neural networks, or CNNs, are popular deep learning technologies. Although they are specifically suited for photos as inputs, they can also be utilized for other continuous answers, text, and signals(*Learn About Convolutional Neural Networks*, n.d.). We created a CNN architecture for our model that has 10 layers in total. The first layer is an embedding layer that creates numerical representations from text sequences. After that, two 1D convolutional layers - one with 256 filters and the other with 128 filters are used to extract local characteristics from the text. In order to minimize dimensionality and avoid overfitting, two max pooling layers are used. Next, the output is reshaped into a 1D vector via a flattening layer. Finally, three dense layers with ReLU activation and 0.5 dropout are used, and the output layer with 6 neurons and softmax activation classifies the text into 6 different emotion categories. The CNN model is compiled with 'categorical_crossentropy' loss function and 'adam' optimizer. Example architecture of CNN model is as shown in fig 3.3(*The Overall Architecture of the Convolutional Neural Network...*, n.d.).

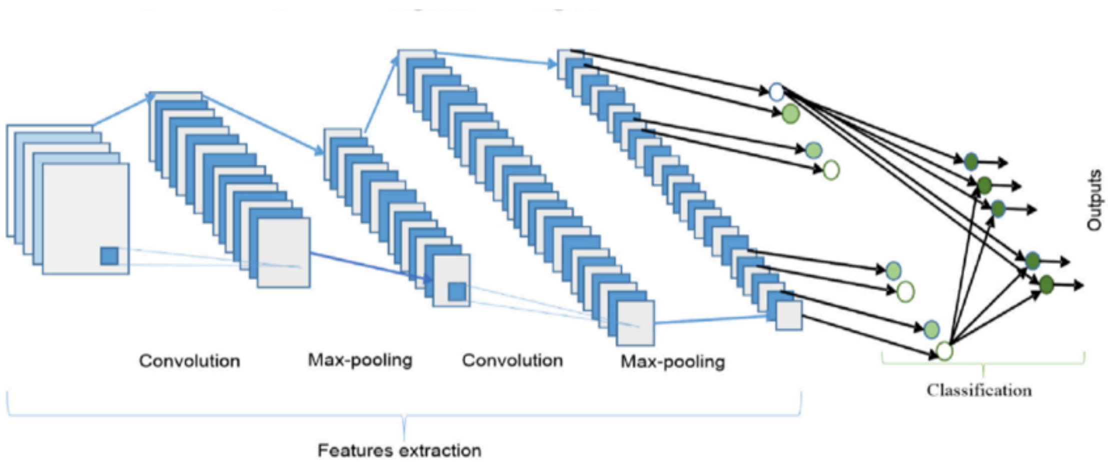


Fig 3.3: Example Architecture of CNN Model

3.5 Experiment Setup
When training the model, we split it into 2 sections: one for Traditional Machine Learning Algorithms and other for Deep Learning Neural Networks. We implement Grid Search for ML algorithms but for DL, we manually tested by changing the parameters because of the lack of powerful hardware resources.

Machine Learning Models
As we mentioned in the data preprocessing step, we convert the data to TF-IDF and apply a **grid search** method. For **SVM**, the regularization parameter C (0.1, 1), kernel types (linear, RBF, polynomial) and gamma (auto, scale) are tested. For the **XGBoost** model learning rate (0.01, 0.1), max depth (3, 6), and number of estimators (100, 200) are applied respectively. Moreover, XGBoost optimizes performance while managing model Logloss. Both models are trained with a cross validation value of 3 folds. This combination maximized the model's classification accuracy on text data and Grid search will find the best parameters and best model. After that, models were validated on Validation data, and if the result is satisfied it will be evaluated on final testing Data.

Deep Learning Models
For Both **BiLSTM** and **CNN** neural networks, we developed an early stopping mechanism that will track the model's loss and restore its optimal weight after three iterations of patience. When the accuracy of the model no longer increases during training, even if it does not reach the goal epochs, it will automatically end. To verify the model's accuracy during the process, we put up a batch size of 32, 10 epochs, an early terminating callback mechanism, and validation data. The models will next proceed through the evaluation procedure using test data.

3.6 Evaluation Metrics
To evaluate the model's performance comprehensively we set up 5 Key evaluation metrics:Accuracy, Recall, Precision, F-1 Score, and Confusion matrix. The terms that we will use in equations are: true positive (TP), true negative (TN), false positive (FP) and false negative (FN) (*Model Evaluation Techniques in Machine Learning | by Fatmanurkutlu | Medium*, n.d.).

Accuracy: The ratio of accurate predictions to total predictions, where the formula is
$Accuracy = (TP+TN) / (TP+TN+FP+FN)$

Precision: The proportion of correctly predicted positive instances out of all predicted positives.
$Precision = TP / (TP+FP)$

Recall: the proportion of actual positive instances correctly identified by the model.
$Recall = TP / (TP+FN)$

F-1 Score: Represent how many times a model made a correct prediction across the entire dataset.
$F\text{-}1 = (2 \times precision \times recall) / (precision+recall)$

Confusion Matrix: A table, as you see in fig 3.4, summarizing true positives (TP), true negatives (TN), false positives (FP), and false negatives (FN).

		actual	
		YES	NO
predicted	YES	TP	FP
	NO	FN	TN

Fig 3.4 : Confusion Matrix

4. RESULT AND DISCUSSION
In the experiments, we tested four Models as mentioned above: SVM, XGBoost, BiLSTM and CNN. Result scores were collected by splitting the dataset to 80% for the training set, 20% of training data for validation and 20% for testing set. Detailed metrics: accuracy, precision, recall, and F1-score, were calculated for each model to provide a comprehensive performance analysis. These metrics demonstrated that deep learning models (BiLSTM and CNN) outperformed machine learning models (SVM and XGBoost) in handling nuanced emotional expressions as shown in table 4.1.

Models	Training	Validation	Testing			
	Accuracy	Accuracy	Accuracy	Precision	Recall	F-1 Score
SVM	93.5%	91.3%	91.3%	91.5%	91.4%	91.3%
XGBoost	92.9%	90.5%	90.6%	90.8%	90.7%	90.6%
BiLSTM	96.3%	94.5%	95.2%	95.4%	95.2%	95.2%
CNN	94.2%	93.2%	92.6%	92.8%	92.6%	92.5%

Table 4.1. Results summary for all models

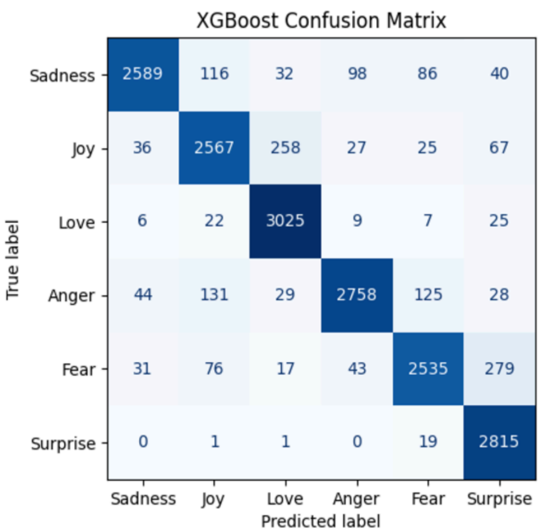
BiLSTM: Showed remarkable ability in capturing sequential and contextual information, with the greatest accuracy of 95%. All emotion classes showed consistently strong precision, recall, and F1-scores.

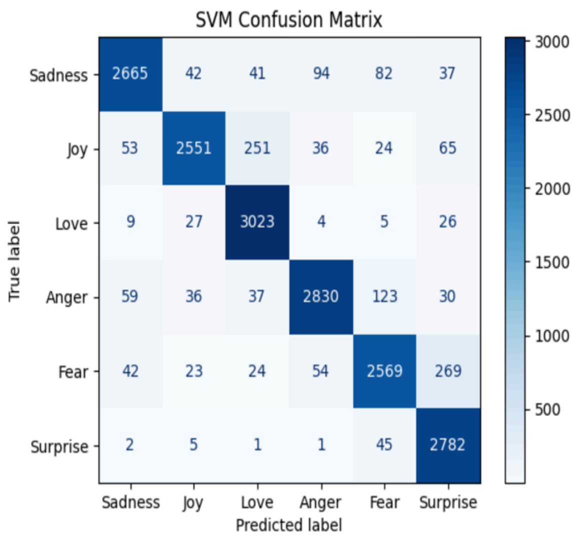
CNN: Identified patterns with a 93% accuracy rate, demonstrating strong performance. Because of its limited ability to capture long-term dependencies, it performed not as good as BiLSTM.

SVM: Handled TF-IDF characteristics for emotion prediction with an accuracy of 91%. It had a good mix between memory and precision, but it had trouble with subtle linguistic patterns in minority emotion classes.

XGBoost: Had the lowest accuracy of all the models, at around 90%. Compared to deep learning models, it was less successful at capturing the intricacy of text semantics, but being competent at modeling interactions between features.

We also generated a confusion matrix for each of the models to identify and more understanding of the prediction of each of the classes.





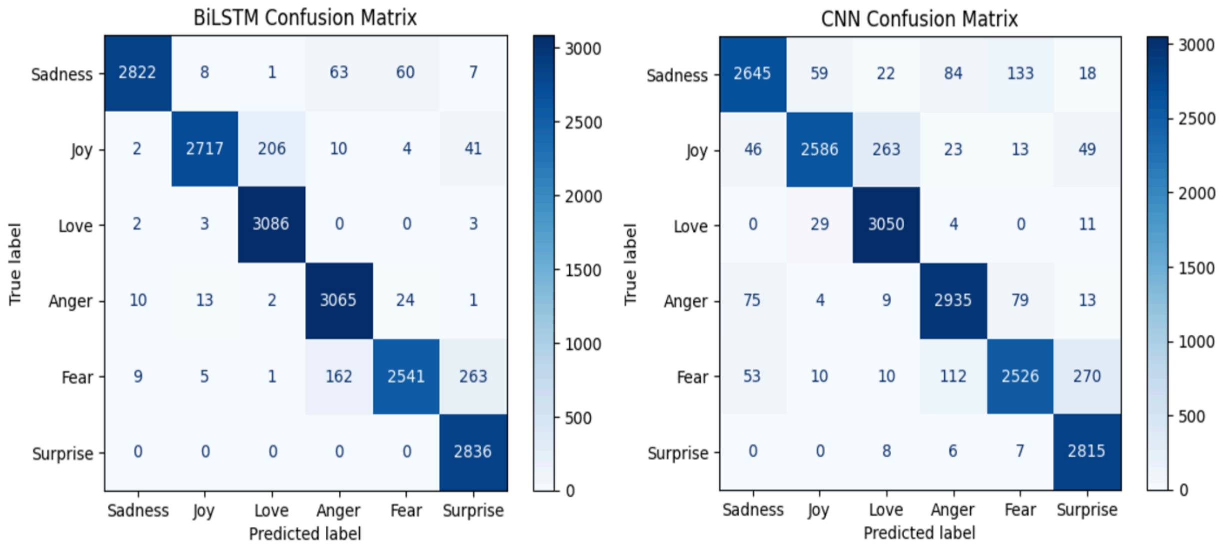


Fig 4.2 : Confusion Matrixs of all Models

Looking at the performance of each model, we can see that the BiLSTM model performs better than the others, which emphasizes how crucial it is to capture sequential information in text data. It had a major advantage over previous models since it could simulate word dependencies. So we visualized the BiLSTM model Accuracy and Loss graph to monitor and analyze the model during the training process according to each epoch as shown in fig 4.3.

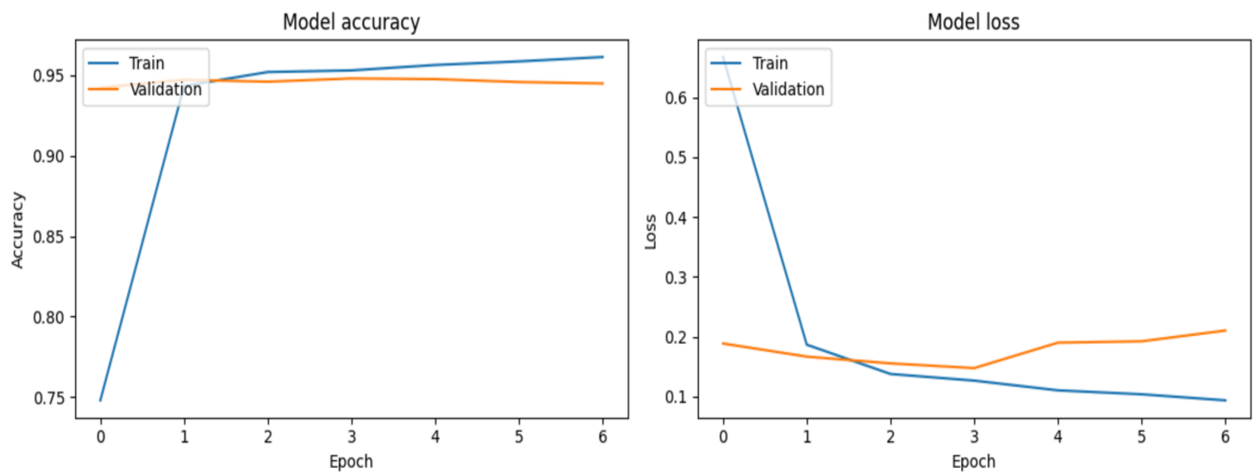


Fig 4.3 The accuracy and Loss Graph of BiLSTM Model

5. CONCLUSION

This experiment gives a clear comparison of machine learning (SVM, XGBoost) and deep learning models (BiLSTM, CNN) by using text mining and natural language processing (NLP) preprocessing techniques to predict emotions from text. A Kaggle dataset with over 600,000 samples, but was balanced to each class of around 15,000 samples, was used for the experiments, and the outcomes demonstrated how successful each strategy was to predict six basic emotions: sadness, joy, love, anger, fear and surprise . With the highest testing accuracy of 95.2% among the models, BiLSTM demonstrated its capacity to identify sequential and contextual patterns in text.

The study also highlighted how crucial preprocessing methods like tokenization, stemming, vectorization, and class balancing are to improving model performance. A thorough comparison of model efficacy was made by extensive evaluation metrics, such as accuracy, precision, recall, F1-score, and confusion matrices. The accuracy-loss graph representations provided more evidence of the deep learning models' dependability in this task.

In summary, this study shows that using deep learning and sophisticated preprocessing methods greatly enhances textual emotion prediction. While the findings support the capabilities of BiLSTM, more diverse emotions, larger datasets, more comprehensive preprocessing techniques and transformer-based designs might all be investigated in future studies to enhance the field's progress.

Acknowledgement
I would like to express my gratitude to the following organizations and individuals for their invaluable support in completing this research project.

First of all, I would like to express my sincere gratitude to my **Asst.Prof.Dr. Kritsada Sriphaew** (Aj.Kong), Dean of Rangsit International College, for his support, guidance, and lectures not only in class but also the project starting from dataset collection till the end of the project experiment. After presenting my project's plan, your insights and feedback have been helpful in shaping the scope and directions of this study.

I am grateful to all my professors **Dr. Prashaya Fusiripong, Assoc.Prof.Dr. Todsanai Chumwatana**, **Asst. Prof. Dr. Suttisak Jantavongso**, **Dr. Herison Surbakti,** and **Dr. Billel Arbaoui** for their instructions and lessons throughout my college years.

Additionally, Thank you to **Ms. Aye Khin Khin Hphone (TA),** and my colleagues **Mr. Min Khant Ko**, **Mr**. **Kaung Set Kyaw** for their assistance with dataset, interpretation, feedback and other necessary help which made this paper possible.

I would also like to express my gratitude to **Mr**. **Phyo Thu Htet**, Founder of Simbolo Myanmar, who laid down the introduction and foundation of AI and NLP to me which helped me a lot throughout this project, experiments and paper.

Finally, I thank my institution, **Rangsit University International College**, for providing the necessary facilities and resources to conduct my research.

References
Cambria, E., Olsher, D., & Rajagopal, D. (2014). SenticNet 3: A Common and Common-Sense Knowledge Base for Cognition-Driven Sentiment Analysis. *Proceedings of the AAAI Conference on Artificial Intelligence*, *28*(1), Article 1. https://doi.org/10.1609/aaai.v28i1.8928

Cambria, E., Schuller, B., Xia, Y., & Havasi, C. (2013). New Avenues in Opinion Mining and Sentiment Analysis. *IEEE Intelligent Systems*, *28*(2), 15–21. IEEE Intelligent Systems. https://doi.org/10.1109/MIS.2013.30

Chen, T., & Guestrin, C. (2016). XGBoost: A Scalable Tree Boosting System. *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, 785–794. https://doi.org/10.1145/2939672.2939785

Dzikovska, M. O., Bell, P., Isard, A., & Moore, J. D. (2012). Evaluating language understanding accuracy with respect to objective outcomes in a dialogue system. In W. Daelemans (Ed.), *Proceedings of the 13th Conference of the European Chapter of the Association for Computational Linguistics* (pp. 471–481). Association for Computational Linguistics. https://aclanthology.org/E12-1048

Emotions. (n.d.). Retrieved November 17, 2024, from https://www.kaggle.com/datasets/nelgiriyewithana/emotions

FIGURE 3. The architecture of the BiLSTM model. (n.d.). ResearchGate. Retrieved November 17, 2024, from https://www.researchgate.net/figure/The-architecture-of-the-BiLSTM-model_fig2_358201557

Figure 9. The overall architecture of the Convolutional Neural Network... (n.d.). ResearchGate. Retrieved November 17, 2024, from https://www.researchgate.net/figure/The-overall-architecture-of-the-Convolutional-Neural-Network-CNN-includes-an-input_fig4_331540139

Hasan, M., Rundensteiner, E., & Agu, E. (2019). Automatic emotion detection in text streams by analyzing Twitter data. *International Journal of Data Science and Analytics*, *7*(1), 35–51. https://doi.org/10.1007/s41060-018-0096-z

Kalage, A. A., & Bhuskade, A. (2018). Optimum Coordination of Directional Overcurrent Relays Using Advanced Teaching Learning Based Optimization Algorithm. *2018 IEEE Global Conference on Wireless Computing and Networking (GCWCN)*, 187–191. https://doi.org/10.1109/GCWCN.2018.8668611

Khare, S. K., Blanes-Vidal, V., Nadimi, E. S., & Acharya, U. R. (2024). Emotion recognition and artificial intelligence: A systematic review (2014–2023) and research recommendations. *Information Fusion*, *102*, 102019. https://doi.org/10.1016/j.inffus.2023.102019

Learn About Convolutional Neural Networks. (n.d.). Retrieved November 17, 2024, from https://www.mathworks.com/help/deeplearning/ug/introduction-to-convolutional-neural-networks.html

Lerner, J. S., & Keltner, D. (2000). Beyond valence: Toward a model of emotion-specific influences on judgement and choice. *Cognition and Emotion*, *14*(4), 473–493. https://doi.org/10.1080/026999300402763

Model Evaluation Techniques in Machine Learning | by Fatmanurkutlu | Medium. (n.d.). Retrieved November 17, 2024, from https://medium.com/@fatmanurkutlu1/model-evaluation-techniques-in-machine-learning-8cd88deb8655

Mohammad, S. M., & Turney, P. D. (2013). Crowdsourcing a Word–Emotion Association Lexicon. *Computational Intelligence*, *29*(3), 436–465. https://doi.org/10.1111/j.1467-8640.2012.00460.x

Poria, S., Majumder, N., Mihalcea, R., & Hovy, E. (2019). *Emotion Recognition in Conversation: Research Challenges, Datasets, and Recent Advances* (arXiv:1905.02947). arXiv. https://doi.org/10.48550/arXiv.1905.02947

Ramirez-Atencia, C., Rodriguez-Fernandez, V., & Camacho, D. (2020). A revision on multi-criteria decision making methods for multi-UAV mission planning support. *Expert Systems with Applications*, *160*, 113708. https://doi.org/10.1016/j.eswa.2020.113708

Seyeditabari, A., Tabari, N., & Zadrozny, W. (2018). *Emotion Detection in Text: A Review* (arXiv:1806.00674). arXiv. https://doi.org/10.48550/arXiv.1806.00674

Stanford CS 224N | Natural Language Processing with Deep Learning. (n.d.). Retrieved November 17, 2024, from https://web.stanford.edu/class/cs224n/

Strapparava, C., & Mihalcea, R. (2008). Learning to identify emotions in text. *Proceedings of the 2008 ACM Symposium on Applied Computing*, 1556–1560. https://doi.org/10.1145/1363686.1364052

Sun, Y., Zhang, J., Yu, Z., Zhang, Y., & Liu, Z. (2023). Bidirectional Long Short-term Neural Network Based on the Attention Mechanism of the Residual Neural Network (ResNet–BiLSTM–Attention) Predicts Porosity through Well Logging Parameters. *ACS Omega*, *8*(26), 24083–24092. https://doi.org/10.1021/acsomega.3c03247

Wang, W. (n.d.). (PDF) Harnessing Twitter “Big Data” for Automatic Emotion Identification. *ResearchGate*. https://doi.org/10.1109/SocialCom-PASSAT.2012.119

Yue, S., Li, P., & Hao, P. (2003). SVM classification:Its contents and challenges. *Applied Mathematics-A Journal of Chinese Universities*, *18*(3), 332–342. https://doi.org/10.1007/s11766-003-0059-5