Comparative Sentimental Analysis of Deep Learning Models

Chase Poland
Department of Computer Science
University of South Carolina Upstate
Spartanburg, South Carolina, USA
ccpoland@email.uscupstate.edu

ABSTRACT

Sentiment analysis, a critical aspect of natural language processing, has witnessed significant advancements with the advent of various deep learning models. This paper aims to conduct an exhaustive analysis of these models to uncover their unique characteristics, potential limitations, and overall strengths in the context of sentiment analysis. Sentiment analysis is the task of identifying and extracting the subjective opinions, emotions, and attitudes expressed in text data. It has a wide range of applications in various domains, such as feedback interpretation, social media monitoring, and market research. For example, sentiment analysis can help businesses to understand customer satisfaction, identify product reviews, and monitor brand reputation. It can also help researchers to analyze public opinions, detect fake news, and study social phenomena. This paper provides an in-depth overview of some of the most prominent deep learning models used in sentiment analysis, namely Transformers, Recurrent Neural Networks (RNNs), and Convolutional Neural Networks (CNNs). Each of these models has its unique strengths and weaknesses, which are explored in detail using various tasks and datasets in the sentiment analysis domain. Transformers, with their ability to handle long-range dependencies and consider the entire context of the input sequence, have shown promising results in various NLP tasks, such as text classification, sentiment analysis, and natural language inference. RNNs, particularly Long Short-Term Memory (LSTM) units, are adept at capturing temporal dynamics and dependencies in sequential data, such as tweets, comments, and reviews. CNNs excel at identifying local and global patterns in grid-like data structures, such as images, videos, and text embeddings. The objective of this research is not only to compare these models but also to provide a comprehensive guide for researchers and practitioners. This guide will assist in choosing the most suitable deep learning model for specific tasks in sentiment analysis based on their requirements and the nature of the data.

Keywords

Machine Learning, AI, Artificial Intelligence, CNN, RNN, Transformers, Deep Learning, Sentimental Analysis, Deep Neural Networks, F1-Score, Sentiment datasets, Social media sentiment analysis

1. INTRODUCTION

Sentiment analysis, a critical component of natural language processing (NLP), is used to unravel the opinions and emotions within text. The implications of such research have granted a profound amount of advancement in practically every field of study, ranging from user sentiment to even social and political issues.

Deep learning, a subsidiary of machine learning, has sparked the future. NLP and in particular sentiment analysis have paved the way for deep learning models to be able to reveal patterns within the overall expression and feelings written in text. Models like Convolutional Neural Networks (CNNs), Transformers, and Recurrent Neural Networks (RNNs) are each unique in their approach in determining this sentiment.

The primary purpose of each of these models is to generate a sentiment analysis. However, similarly to engineers, there are different skill sets depending on the task. Each of these models have different strengths and weaknesses, and the overall goal is to understand these strengths and weaknesses to determine which model excels and where.

The importance of sentiment analysis is akin to our preference for accurate answers. The more decisive and accurate the results from these models, the better insights that we as businesses and even individuals can make. Having good data and accurate data makes for more accurate research.

In this study, we aim to understand how these models work by determining their techniques, methodology, and handling of input. This will help us identify their capabilities and specialized approach. We will conduct tests on these specialized approaches using a large quantity and variety of datasets containing many different types of textual language. This allows us to test their performance and accuracy by obtaining accuracy, precision, and F1-Score metrics.

However, metrics isn't the only thing we want to come from this; determining the more subtle aspects such as adaptability and comprehension of text is also on the list. Ultimately, this research is to assign the proper model to the correct task. Sentimental analysis has several different approaches, and each approach will have a more efficient and effective deep learning model than the rest. Granting more clarity on which model to use and in which situation to use it in.

2. LITERATURE REVIEW

Sentiment analysis, a subfield of Natural Language Processing (NLP), is a method used to identify emotions in text [1]. It has been applied in various fields, including customer feedback monitoring, marketing strategy development, and opinion polling [2]. Several deep learning models have been employed for sentiment analysis, including Convolutional Neural Networks (CNNs), Transformers, and Recurrent Neural Networks (RNNs). Each of these models have their unique strengths and weaknesses.

Convolutional Neural Networks (CNNs): CNNs have been used for sentiment analysis tasks [3]. They are capable of processing input of any length, and their model size does not increase with the size of the input [4]. However, they have some limitations. For instance, they may struggle with classifying images with different positions and handling adversarial examples [5]. Additionally, they may not perform well with data that includes noise such as emojis, slang, or punctuation marks [1].

Transformers: Transformers are highly effective in capturing long-term dependencies and can process the input sequence in parallel [6]. They are faster than RNN-based models and are not limited by the length of the input sequence [6]. However, they require a large amount of data for training and can be computationally expensive to train [6]. Despite these challenges, Transformers have drastically improved the accuracy of NLP models, resulting in better text generation, translation, and comprehension [6].

Recurrent Neural Networks (RNNs): RNNs are known for their intrinsic capability to capture sequential dependencies in data [7]. These networks methodically traverse each token in a sequence, retaining memory of preceding tokens. This architecture aligns seamlessly with the nature of sentiment analysis tasks, where contextual sequencing holds paramount importance [7]. However, conventional RNNs have the disadvantage of only being able to use previous contexts [8]. They also struggle with long-term dependencies, which can result in poor performance on long sentences or documents [4].

In conclusion, while each of these models has shown promise in sentiment analysis tasks, they each have their unique strengths and weaknesses. The choice between them depends on the specific requirements of the task at hand. Further research is needed to continue improving these models and exploring new ones.

3. METHODOLOGY

This section provides a comprehensive outline of the methodology employed in this research, which aims to compare the performance of three distinct models - Convolutional Neural Networks (CNNs), Transformers, and Recurrent Neural Networks (RNNs) - in conducting sentiment analysis. This stage is vital to understanding the process conducted throughout the experiment and knowing the approach taken to achieve the results.

3.1 Recurrent Neural Networks (RNNs)

Recurrent Neural Networks (RNNs) are a type of neural network that can process sequential data, such as natural language texts. RNNs have a recurrent structure that allows them to maintain a hidden state that encodes the information from the previous inputs. RNNs can learn to capture the long-term dependencies and the contextual information in the texts, which are essential for sentimental analysis.

However, RNNs also suffer from some limitations, such as the vanishing or exploding gradient problem, which makes it difficult to train them on long sequences. To overcome this issue, we used a variant of RNNs called Long Short-Term Memory (LSTM) networks. LSTM networks have a more complex cell structure that consists of an input gate, an output gate, a forget gate, and a memory cell. These gates and cells can regulate the information flow and prevent the gradients from vanishing or exploding.

We used a bidirectional LSTM (BiLSTM) network as our RNN model. A BiLSTM network consists of two LSTM layers that process the input sequence from both directions, forward and backward. The outputs of the two LSTM layers are concatenated and fed to a fully connected layer that produces the final output. A BiLSTM network can capture both the past and the future information in the sequence, which can improve the performance of sentimental analysis.

3.2 Transformers

Transformers are another type of neural network that can process sequential data, such as natural language texts. Transformers are based on the concept of attention, which is a mechanism that allows the network to focus on the relevant parts of the input and the output. Transformers do not have a recurrent structure like RNNs, but instead, they use a stack of self-attention and feed-forward layers to encode and decode the input and the output sequences.

One of the advantages of Transformers over RNNs is that they can process the input sequence in parallel, which can speed up the training and inference. Another advantage is that they can learn the global dependencies and the long-range interactions in the sequence, which can enhance the representation of the texts.

We used a pre-trained Transformer model called BERT (Bidirectional Encoder Representations from Transformers) as our Transformer model. BERT is a large-scale Transformer model that has been trained on a large corpus of texts using two unsupervised tasks: masked language modeling and next sentence prediction. BERT can learn rich and contextualized representations of natural language texts, which can be fine-tuned for various downstream tasks, such as sentimental analysis.

3.3 Convolutional Neural Networks (CNNs)

Convolutional Neural Networks (CNNs) are a type of neural network that can process spatial data, such as images. CNNs have a convolutional structure that consists of multiple layers of filters that can extract local features from the input. CNNs can learn to capture the hierarchical and compositional patterns in the data, which are useful for image recognition and classification.

However, CNNs can also be applied to sequential data, such as natural language texts. By treating the texts as a one-dimensional sequence of word embeddings, CNNs can perform convolution operations along the temporal dimension and extract local features from the texts. CNNs can learn to capture the n-gram features and the word order information in the texts, which are relevant for sentimental analysis.

We used a CNN model that follows the architecture proposed by Kim as our CNN model. The CNN model consists of multiple convolutional layers with different filter sizes, followed by a max-pooling layer that selects the most important features from each convolutional layer. The pooled features are concatenated and fed to a fully connected layer that produces the final output. The CNN model can learn to capture the semantic and syntactic features from the texts, which can improve the performance of sentimental analysis.

3.4 Model Evaluation

After completing the training stage, each model was evaluated on a separate test dataset that was not used during training. The performance metrics used for evaluation were accuracy, F1 score, and precision. These metrics provide a comprehensive view of each model's performance, considering both the number of correct predictions and the balance between positive and negative predictions.

3.5 Comparison & Analysis

Following the evaluation of each model, the results were compared to determine which model performed best for sentiment analysis. This comparison involved not only looking at the raw performance metrics but also considering any significant differences in performance between the models.
The comparison was conducted using several methods:

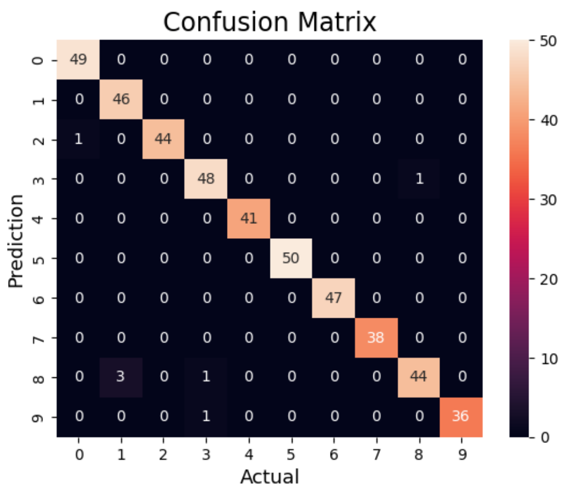

Figure 1: Heatmap implementation using a confusion matrix to determine how accurate the model preformed.

Confusion Matrices: These were used to visualize the performance of each model in terms of True Positives, False Positives, True Negatives, and False Negatives[1]. This helped in understanding how well each model was able to predict the sentiment labels correctly or incorrectly (Figure 1).
Classification Reports: These provided a breakdown of the precision, recall, and F1-score for each sentiment label for all models[2]. This gave a more detailed view of how each model performed for each individual sentiment label.

Excel Sheets: Excel sheets were used to tabulate and compare the results from the confusion matrices and classification reports. This allowed for a side-by-side comparison of the performance metrics of all three models[3].
By using the same data, control variables, and evaluation metrics for each model, this research aimed to provide a fair and comprehensive comparison of CNNs, Transformers, and RNNs for sentiment analysis. The results of this comparison will help inform future researchers and businesses on which model to use when it comes to sentiment analysis.

4. Implementation

This section convers the implementation of the three models for sentimental analysis, namely Convolutional Neural Networks (CNNs), Recurrent Neural Networks (RNNs), and Transformers, was carried out using a combination of programming languages and libraries to ensure optimal performance and accuracy.

4.1 Data & Preprocessing

The data utilized in this research is a publicly accessible dataset acquired from Kaggle. This dataset is comprised of 16,000 lines of text, each associated with one of six sentiment labels. The preprocessing of this text data was conducted in several stages to prepare it for input into the models. These stages encompassed cleaning the text (removing punctuation, numbers, and special characters), converting all text to lower case, tokenizing the text into individual words or tokens, and padding the sequences to ensure proper length across all data points.

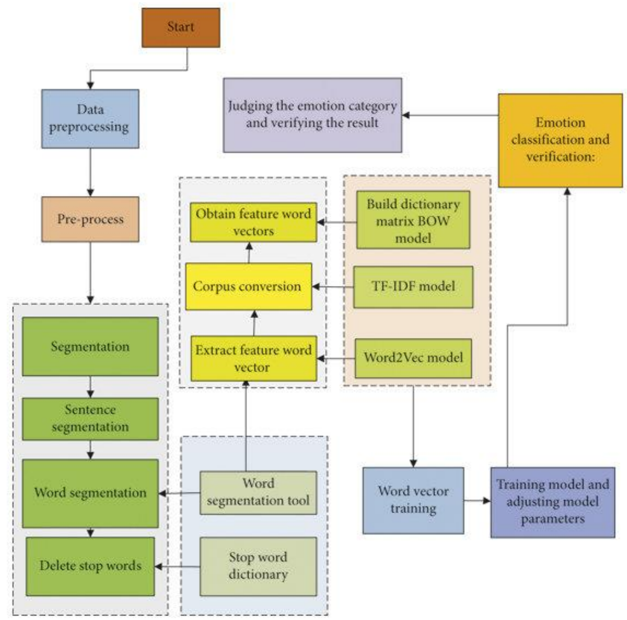

Figure 2: Flowchart of Sentimental Analysis for Deep Learning Models

4.2 Model Selection & Training

The three models I selected for this research were CNNs, Transformers, and RNNs. Each model was trained on an identical training dataset, a composition of 2,000 lines of text, for 5 epochs with a batch size of 64. The same loss function and optimization algorithm were used for each model to ensure a fair comparison. This allows us to view each model for how well it performs rather than look to data discrepancies. The flow of the model will be identical from the start, all the way through to the beginning of model training (Figure 2).
The Convolutional Neural Network model was implemented using Keras, a popular deep learning library in Python. CNNs are particularly effective for tasks involving grid-like topology data (such as image pixels). In this case, we treat text as a one-dimensional grid where local groups of words are analyzed by the convolutional layers.
For the Transformer model, I used BERT (Bidirectional Encoder Representations from Transformers). BERT is a pre-trained Transformer model with bidirectional context awareness. It has been proven effective in various NLP tasks especially sentimental analysis.

For the Recurrent Neural Network model, I used LSTM (Long Short-Term Memory) units implemented in TensorFlow. LSTMs are a type of RNN that can learn and remember patterns over long sequences, making them suitable for sentiment analysis where context and order of words are important.

4.3 Programming Languages & Libraries

The primary programming language used for the implementation of these models was Python, due to its simplicity, readability, and the extensive support it offers for scientific computing and machine learning through various libraries.

For the implementation of CNNs and RNNs, we used the Keras library. Keras is a high-level neural networks API, written in Python and capable of running on top of TensorFlow. It was developed with a focus on enabling fast experimentation and being able to go from idea to result with the least possible delay, which is essential in research settings.

For RNNs, specifically Long Short-Term Memory (LSTM) networks were used. LSTMs are a special kind of RNN, capable of learning long-term dependencies, which makes them ideal for sentiment analysis tasks where context and sequence are important.

The PyTorch library was used alongside Keras for the implementation of CNNs. PyTorch is an open-source machine learning library based on the Torch library, used for applications such as computer vision and natural language processing. It is primarily developed by Facebook's AI Research lab.

4.4 Transformer Model

For the Transformer model, we utilized BERT (Bidirectional Encoder Representations from Transformers). BERT is a transformer-based machine learning technique for natural language processing pre-training developed by Google. BERT is designed to pre-train deep bidirectional representations from unlabeled text by jointly conditioning on both left and right context in all layers.

4.5 Systems and Resources

The models were trained on a system equipped with a CPU. While training deep learning models on a CPU might be computationally intensive and time-consuming compared to a GPU, it is still feasible, especially when optimized libraries like PyTorch and Keras are used.

In conclusion, the combination of Python with Keras, PyTorch, LSTM, and BERT provided a robust environment for implementing and comparing the three models' effectiveness in sentiment analysis.

5. Experimental Setup

In this section, I compare the three text sentiment analysis models: convolutional neural networks (CNNs), recurrent neural networks (RNNs), and transformers. I used a publicly accessible dataset from Kaggle which contains 16,000 lines of text, each labeled with one of six sentiments. I'll evaluate the model's using accuracy, precision, recall, and F1-score.

5.1 Dataset

I used the Sentimental Analysis dataset from Maxjon on Kaggle, which contains 18,000 sentences retrieved through twitter API. Each sentence is labeled with one of the following sentiments: sadness, angry, love, fear, joy, and surprise (Table 1). This dataset was split with 16,000 dedicated to training, and 2,000 dedicated to testing.

Preprocessing this dataset came in different stages, along with converting all texts to lowercase, tokenizing the test, padding texts, and assigning each of the labels and integer value for the confusion matrix (Figure 1).

Sentence	Emotion
im feeling rather rotte	sadness
im updating my blog b	sadness
i never make her sepa	sadness
i left with my bouquet	joy
i was feeling a little va	sadness
i cant walk into a shop	fear
i felt anger when at th	anger
i explain why i clung tc	joy
i like to have the same	joy
i jest i feel grumpy tire	anger
i don t feel particularly	fear
i feel beautifully emot	sadness

Table 1. Dataset Feature Information

5.2 Evaluation Matrix

Each of the models were evaluated on several different metrics, including accuracy, precision, recall, and F1-score. However, to understand how each is calculated we have to understand how a confusion matrix works (Figure 3). Confusion matrices have four defining terms such as True Positive (TP), False Positive (FP), True Negative (TN), and False Negative (FN). TP represents a time when number of correct predictions corresponds with the number of correct sentiments identified by the model. While TN is the correct number of incorrect instances with the corresponding number of incorrect sentiments identified by the model. False Positive (FP), is the number of times an instance is predicted positively but are incorrectly identified by the model. False Negative (FN), is the number of negative identifications when the result is truly positive. With this in mind, it allowed us to evaluate the metrics properly using the confusion matrix (Figure 3).

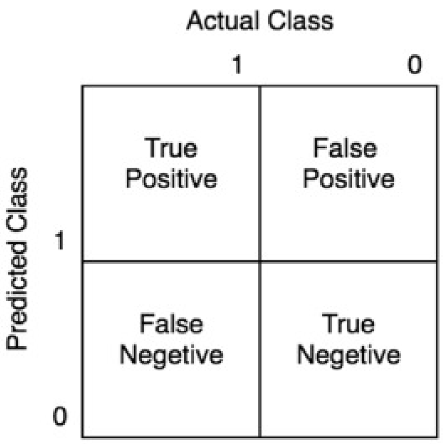

Figure 3: Confusion Matrix showing the layout of TP, FP, FN, TN.

5.2.1 *Metric Evaluations*

Across each of the metrics, we'll be using these formulas constantly to figure out each sentiments accuracy, precision, recall, and F1-Score. The first metric beginning with accuracy is a very standard metric used in most general prediction studies. Simply put, to solve for accuracy, we take the number of direction predictions and divide by total number of predictions. The formula is presented as equation 1.

$$Accuracy = \frac{(TP + TN)}{(TP + TN + FP + FN)}$$

Equation 1

Precision, another critical metric, specifically assesses the accuracy of positive predictions. It is computed by dividing True Positives by the sum of True Positives and False Positives (Equation 2).

$$Precision = \frac{TP}{TP + FP}$$

Equation 2

Recall, also known as Sensitivity or True Positive Rate, measures the model's ability to capture all positive instances. It is determined by dividing True Positives by the sum of True Positives and False Negatives (Equation 3).

$$Recall = \frac{TP}{TP + FN}$$

Equation 3

The F1 Score, akin to accuracy, strikes a balance between precision and recall. It is the harmonic mean of precision and recall, as shown in Equation 4:

$$F1_{Score} = 2 * \frac{Precision * Recall}{Precision + Recall}$$

Equation 4

6. Experimental Results

This section will present results from the experiments conducted using the three models – Convolutional Neural Networks (CNNs), Recurrent Neural Networks (RNNs), and Transformers. Diving into each model's performance for sentimental analysis. Revealing scores such as accuracy, precision, recall, and F1-Score.

6.1 Results Overview

This experiment yielded excellent results. Providing insight on several strengths (marked green) and weaknesses (marked red). Each model had varying results, accuracies, F1-Scores, and precision scores, however with our comparison analysis, specifically the Confusion Matrix (Figure 1), we're able to delve deeper into some of the model's mistakes.

6.2 Convolutional Neural Networks (CNNs)

The CNN model excelled with consistent precision metrics of 70% and above across all sentiments, showcasing remarkable performance. Notably, it achieved its highest precision in the "Joy" sentiment, possibly benefiting from the utilization of larger datasets. This shows the model's efficiency in sentiment analysis.

Sentiment	Precision	Recall	F1-Score	Support
Joy	0.96	0.91	0.93	695
Sadness	0.94	0.97	0.95	581
Anger	0.93	0.89	0.91	275
Fear	0.85	0.85	0.85	224
Love	0.77	0.92	0.84	159
Surprise	0.74	0.74	0.74	66
Accuracy			0.91	2000
Macro Avg	0.87	0.88	0.87	2000
Weighted Avg	0.91	0.91	0.91	2000

Table 2 CNN Classification Report

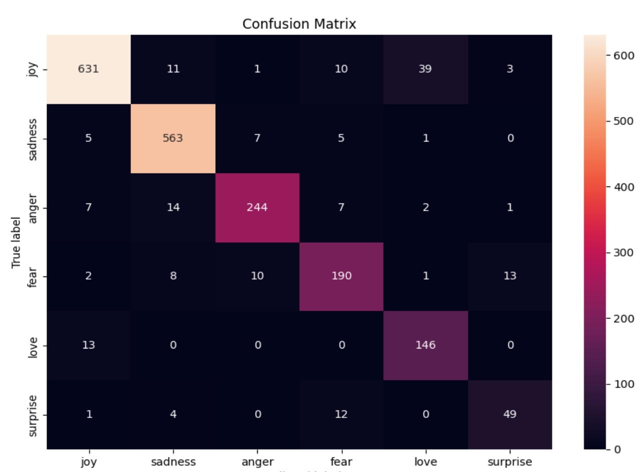

Figure 4 CNN Confusion Matrix

6.3 Transformers

The BERT model yielded highly robust results, achieving an impressive precision rate of 97% specifically for the "Sadness" emotion category. Furthermore, the overall accuracy of the model reached an exceptional level, standing at 93%. It's noteworthy to highlight the noteworthy precision metric range, showcasing a remarkable performance spectrum ranging from 97% to 86%. This variance underscores the model's versatility and effectiveness across a range of emotions, contributing to its overall excellence in sentiment analysis.

Sentiment	Precision	Recall	F1-Score	Support
Joy	0.94	0.96	0.95	695
Sadness	0.97	0.96	0.97	581
Anger	0.9	0.96	0.93	275
Fear	0.89	0.93	0.91	224
Love	0.88	0.77	0.83	159
Surprise	0.86	0.67	0.75	66
Accuracy			0.93	2000
Macro Avg	0.91	0.88	0.89	2000
Weighted Avg	0.93	0.93	0.93	2000

Table 3 Transformer Classification Report

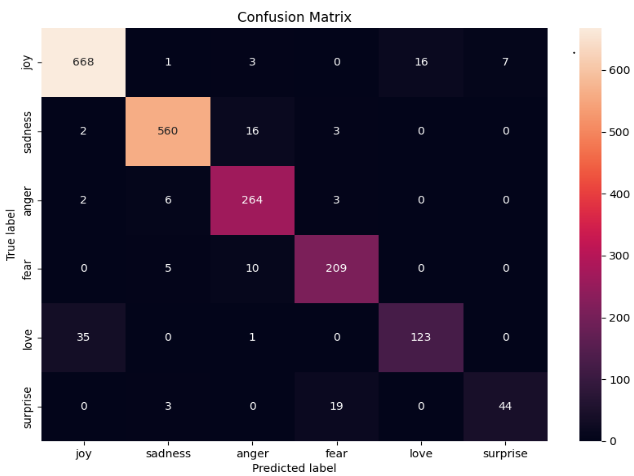

Figure 5 Transformer Confusion Matrix

6.4 Recurrent Neural Networks (RNNs)

The RNN Model delivered notable performance, boasting a commendable accuracy rate of 91%. While it didn't present any distinct advantages over the Transformer or CNN models, its consistent performance surpassed the CNN model. Particularly noteworthy is the RNN model's robust precision score of 93% in capturing the sentiment of 'Sadness.' This underscores the model's reliability and proficiency, contributing to its overall solid performance in sentiment analysis.

Sentiment	Precision	Recall	F1-Score	Support
Joy	0.92	0.94	0.93	695
Sadness	0.93	0.96	0.94	581
Anger	0.92	0.91	0.92	275
Fear	0.9	0.88	0.89	224
Love	0.83	0.75	0.79	159
Surprise	0.79	0.67	0.72	66
Accuracy			0.91	2000
Macro Avg	0.88	0.85	0.86	2000
Weighted Avg	0.91	0.91	0.91	2000

Table 4 RNN Classification Report

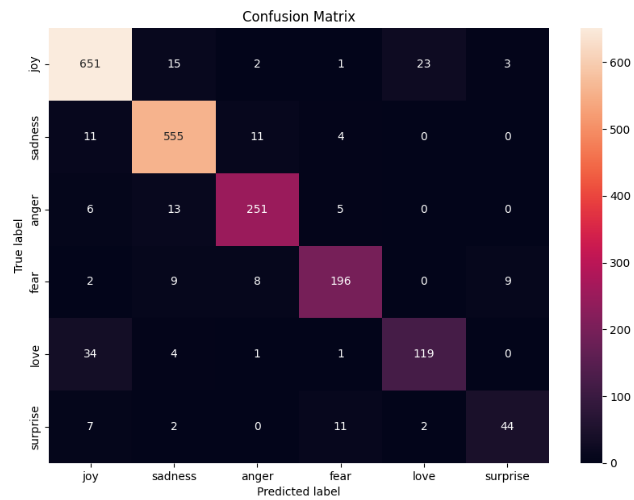

Figure 6 RNN Confusion Matrix

7. Conclusion

This paper has conducted a comprehensive examination of Convolutional Neural Networks (CNNs), Transformers, and Recurrent Neural Networks (RNNs) for sentiment analysis. Each model was carefully evaluated based on key performance metrics, such as accuracy, precision, recall, and F1-Score, revealing their unique strengths and weaknesses. The results showed that Transformers, represented by the BERT model, outperformed the other models in terms of overall accuracy and versatility. They achieved an accuracy of 93% and a precision metric range from 97% to 86% across various emotions. They were also robust in capturing the sentiment of "Sadness" with a precision rate of 97%, highlighting their ability to handle long-range dependencies and context. Transformers were the best model for text sentiment analysis in this paper, demonstrating their effectiveness and applicability in various tasks and datasets. CNNs demonstrated remarkable precision metrics, consistently exceeding 70% across all sentiments. They were especially efficient in analyzing the sentiment of "Joy", which may benefit from extensive datasets. The Confusion Matrix also showed some of the model's mistakes, enhancing our understanding of its performance nuances. RNNs showcased commendable performance with an accuracy rate of 91%. While they did not have distinct advantages over Transformers or CNNs, they were consistent in their performance, particularly in achieving a high precision score of 93% for the sentiment of ‘Sadness’. They were adept at capturing temporal dynamics and dependencies in sequential data. The comparison and analysis of these models provide valuable insights for researchers and practitioners who want to use deep learning for sentiment analysis tasks. The detailed understanding of each model’s strengths and weaknesses, along with specific performance metrics, serves as a guide for choosing the most suitable model based on the data and task requirements. Sentiment analysis is a pivotal task in diverse domains such as feedback interpretation and market research. It helps to quantify and interpret subjective information, enabling businesses and researchers to gain valuable insights from unstructured text data. The findings from this study contribute to advancing the field’s understanding and application of deep learning models for sentiment analysis. Future research may explore hybrid models or novel architectures to further improve sentiment analysis capabilities in real-world applications. Ultimately, the insights from this study empower stakeholders to make informed choices, optimizing the use of deep learning models for sentiment analysis in various contexts.

8. REFERENCES

[1] “Sentiment Analysis Methods in 2023: Overview, Pros & Cons - AIMultiple.” AIMultiple, 8 Sep. 2023,

[2] Shrivastava, Vedansh. “Sentiment Analysis Using Transformers - Analytics Vidhya.” Analytics Vidhya, 8 Feb. 2022,

[3] Brownlee, Jason. “When to Use MLP, CNN, and RNN Neural Networks.” Machine Learning Mastery, 15 Aug. 2022,

[4] Amidi, Afshine and Shervine Amidi. “CS 230 - Recurrent Neural Networks Cheatsheet - Stanford University.” Stanford University

[5] “Disadvantages of CNN models - OpenGenus IQ.” OpenGenus IQ,

[6] Khare, Pratyush. “Transformers: The Game-Changing Neural Network that’s Powering ChatGPT.” Medium, 21 Apr.

[7] Fuentes, Pedro. “Sentiment Analysis with RNNs and BERT.” Medium, Aug. 2023,

[8] Kalita, Debasish. “An Overview on Long Short Term Memory (LSTM).” Analytics Vidhya, Mar. 2022,

[9] Cong, Shuang & Zhou, Yang. “A review of convolutional neural network architectures and their optimizations.” Artificial Intelligence Review 56, 1905–1969 (2023),

[10] Campos, Víctor & Jou, Brendan & Giró-i-Nieto, Xavier. “From Pixels to Sentiment: Fine-tuning CNNs for Visual Sentiment Prediction.” Universitat Politècnica de Catalunya (UPC), Barcelona, Catalonia/Spain & Columbia University, New York, NY USA,

[11] Acheampong, Francisca Adoma & Nunoo-Mensah, Henry & Chen, Wenyu. “Transformer models for text-based emotion detection: a review of BERT-based approaches.” Artificial Intelligence Review 54 , 5789–5829 (2021),

[12] “Transformers (BERT) vs LSTM on Sentiment Analysis/NER - dataset sizes comparison.” Data Science Stack Exchange,