

AI Assignement

by Hammada Ballali Ebbi

Submission date: 19-Apr-2024 02:52PM (UTC+0100)

Submission ID: 230012829

File name: 49540_Hammada_Ballali_Ebbi_AI_Assignement_571450_1783268548.docx (218.44K)

Word count: 4571

Character count: 26759

**Meta ensemble deep learning techniques
for sentiment analysis on the IMDB
(Internet Movie Database) dataset.**



Prifysgol Cymru
Y Drindod Dewi Sant
University of Wales
Trinity Saint David

School of Applied Computing

Wales Institute of Science and Art (WISA)

L7 - Artificial Intelligence (ACCA7026)

Dr. Seena Joseph

Hammada Ballali Ebbi 2308208

19/04/2024

Contents

Abstract:.....	3
1.Introduction	1
2. Literature Review	3
3. Methodology	8
3.1 Dataset	8
3.2 Implementation.....	10
3.2.1 CNN Model	10
3.2.2 LSTM Model	12
3.2.3 GRU Model.....	13
3.2.4 BERT Model	14
3.2.5 ML Model using ensemble techniques.	15
4. Discussion.....	16
5. Conclusion	17
6. References.....	18

Table of figures

Figure 1 Sentiment analysis diagram.....	3
Figure 2. Accuracy metric of the architecture model using precision recall f-score and accuracy derived from f-score retrieved by M. R. Haque, S. Akter Lima, and S. Z. Mishu [4]	5
Figure 3. IMDB dataset sample	9
Figure 4. Sample of words used for training purposes.	10

Abstract:

With the emergence of Internet forums, web services and social networks, a very wide range of possibilities was opened in the field of communication, giving a complete turn to the study areas focused on information-gathering behaviours. Although it is of great interest to be able to correctly collect the general appreciation of a sample of users on a specific topic, the enormous amount of information becomes very difficult to process and analyse. Machine Learning models are presented as a very useful tool when addressing this type of tasks, giving impulse to the field of study in Artificial Intelligence for sentiment analysis applying Machine Learning techniques. Different models have been proposed in recent years in order to achieve acceptable precision levels in this type of analysis. In this research, some of these models will be evaluated and compared with the objective of establishing the advantages and disadvantages of the application of each approach. Afterwards, with the objective of carrying out an effective sentiment analysis, different models will be applied to the IMDB data set, with the main aim of designing a model that applies meta ensemble learning techniques that allows achieving high levels of precision. Finally, this model will be evaluated and analysed by studying the achieved results.

1.Introduction

In today's interconnected world, gathering opinions and sentiments from a diverse range of sources has become integral to decision-making processes. With the proliferation of internet platforms and the explosion of user-generated content, individuals can easily access a wealth of opinions on various subjects, providing valuable insights into public sentiment. Sentiment analysis, a key area of study in machine learning, plays a crucial role in extracting and analysing sentiments from textual data, such as movie reviews, blogs, and social media posts.

Sentiment analysis, closely related to natural language processing (NLP) and text mining, aims to identify, and quantify the attitudes and opinions expressed in text. By analysing the sentiment conveyed in reviews, sentiment analysis can provide valuable insights into the favourability or negativity surrounding a particular subject, such as a movie. In the context of movie reviews, sentiment analysis allows us to gauge the overall sentiment towards a film, helping viewers decide whether it's worth their time [1].

Machine learning has emerged as a crucial component of artificial intelligence, finding applications across diverse fields owing to its ability to process and extract insights from vast datasets. However, the landscape of machine learning is vast, encompassing various algorithms and methodologies, each with its own set of strengths and limitations.

In recent years, different approaches have been applied in the field of sentiment analysis, highlighting the studies that have applied LSTM, CNN, GRU and BERT techniques, achieving different levels of precision. This encompasses different deep learning techniques, characterised by their ability to handle complex problems and automatically extract features from unstructured data to more conservative models. These techniques offer powerful tools for tasks ranging from image recognition to natural language processing. However, training deep learning models can be resource intensive and require careful hyperparameter tuning to avoid overfitting.

Ensemble learning, on the other hand, involves combining multiple baseline models to create a more robust and accurate predictive model. This concept began to be introduced forcefully in the 90s (Leo Breiman, "Bagging Predictors", 1996) [2] when greater emphasis began to be placed on studies that applied the combination of different ML techniques to specific problems. In his paper titled By leveraging the diversity of these constituent models, ensemble learning can mitigate the risk of overfitting and improve performance across various

domains. Ensemble techniques such as bagging, boosting or stacking have been widely successful and outperformed single models in numerous applications.

While traditional ensemble learning has primarily focused on integrating traditional machine learning models, recent efforts have sought to extend ensemble techniques to deep learning architectures. However, these attempts often rely on simplistic averaging methods that may not fully exploit the potential of deep learning models. Despite various strategies proposed for combining baseline learners in ensemble deep learning, challenges remain in terms of generalisation, training difficulties, and other issues.

The applications of sentiment analysis extend across various domains, including finance, healthcare, politics, and social events. Businesses can use sentiment analysis to gauge customer satisfaction, predict sales performance, and analyse market trends. Additionally, sentiment analysis can be applied in political contexts to analyse public opinion and predict election results.

Applying machine learning (ML) techniques for sentiment analysis on movie reviews involves utilising algorithms to automatically classify the sentiments expressed within textual data. Leveraging datasets such as the IMDB dataset, which contains labelled movie reviews with corresponding sentiment polarities, ML models are trained to recognize patterns and extract features indicative of positive or negative sentiments. Through the iterative training process, ML models learn to generalise from the training data, enabling them to accurately predict sentiment in unseen movie reviews. The application of ML for sentiment analysis on movie reviews facilitates valuable insights for viewers, filmmakers, and industry professionals, aiding in decision-making processes, gauging audience reactions, and enhancing the overall movie-watching experience [1].

Thus, sentiment analysis plays a crucial role in extracting and analysing sentiments from textual data, providing valuable insights into public opinion. By leveraging machine learning techniques, researchers can automate sentiment analysis tasks and gain actionable insights from large volumes of textual data. As the volume of user-generated content continues to grow, sentiment analysis will remain an essential tool for understanding public sentiment and making informed decisions.

In this sense, this study proposes and applies four ML models CNN, LSTM, GRU and BERT, with the objective of being able to extract and combine the strengths of each technique in an ensemble model applied to the IMDB dataset, in order to perform classification on user reviews (Fig 1).

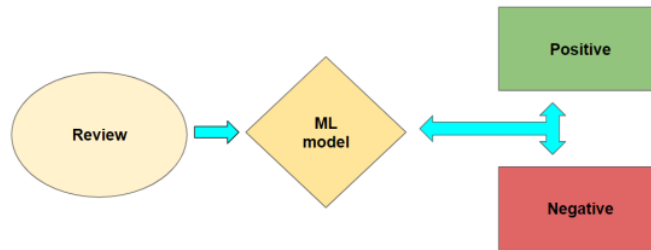


Figure 1 Sentiment analysis diagram.

2. Literature Review

Multiple studies have been carried out on this subject in recent years applying different approaches to the problem of sentiment analysis based on language processing, with very

different results depending on the type of technique used, the degree of optimisation and the type of dataset.

A study published by A. Yener and A. Verma in 2017, *Deep CNN-LSTM with combined kernels from multiple branches for IMDb review sentiment analysis* in the Journal of IEEE [3], applies a Deep Learning model which combines CNN techniques with LSTM, so that the results obtained from applying the convolution, activation (using ReLU) and pooling layers serve as input for a new layer applying LSTM, which intuitively fits the needs of the study due to the type of sequential data to which it is applied. This process is performed through multiple branches to finally apply concatenation in the last layer to merge them all together.. The authors observed that the main challenge in applying this approach is to reduce the number of epochs in order to avoid overfitting. Even so, for a reduced number of epochs the model achieves acceptable accuracy rates of over 80%.

Another study published by M. R. Haque, S. Akter Lima, and S. Z. Mishu in 2019, *Performance Analysis of Different Neural Networks for Sentiment Analysis on IMDb Movie Reviews* [4], utilises word embedding to establish a vector space for sentence representation in reviews. Subsequently, three distinct neural network architectures, namely CNN, LSTM, and LSTM-CNN, are employed to extract features from the sentences. These extracted features are then inputted into a multilayer perceptron network for sentiment classification (positive or negative). The primary aim is to identify optimal deep neural network architectures for enhanced sentiment classification performance on the IMDb movie reviews dataset. The architectures are implemented using Keras, a high-level API operating atop TensorFlow, a deep learning framework developed by Google to facilitate efficient deep learning operations. The study concludes that the CCN model performs better achieving higher rates of accuracy than the other two models (Fig 2).

Evaluation Measure	CNN	LSTM	LSTM-CNN
Accuracy	0.90	0.88	0.89
Recall	0.95	0.82	0.90
Specificity	0.84	0.90	0.87
Precision	0.87	0.90	0.87
F-Score	0.91	0.86	0.88

 2. Accuracy metric of the architecture model using precision recall f-score and accuracy derived from f-score retrieved by M. R. Haque, S. Akter Lima, and S. Z. Mishu [4]

Another study published by S. M. Qaisar,, and S. Z. Mishu, in 2019, *Sentiment Analysis of IMDb Movie Reviews Using Long Short-Term Memory [5]*, proposes an experiment that discusses the common practice of dividing datasets into training and testing vectors for classification tasks, typically using an 80-20 split, and employing 10-fold cross-validation to mitigate bias in results. Preprocessing steps for IMDb movie review data include removing symbols, converting text to lowercase, eliminating hybrid links, removing stop words, and applying stemming techniques. Vectorization, performed using Python libraries and Doc2Vec model, encodes reviews into numeric values. The Long Short-Term Memory (LSTM) classifier, based on Recurrent Neural Network (RNN), addresses issues with long-term dependencies. It utilises the Adam optimizer for adaptive learning rates and consists of three layers: the input layer with 50 nodes representing words, the LSTM layer with 101 memory units, and the output layer with two nodes for classifying positive and negative sentiments. The model achieves an accuracy of 89.9% that can be improved by further optimisation and data cleaning.

A research published in 2020 by R. Ni and H. Cao, *Sentiment Analysis based on GloVe and LSTM-GRU* [6], presents a different approach. It applies a GRU model to two datasets, the IMDB dataset used in this research and the Review_polarity dataset which is somewhat more complex to manipulate because the reviews are generally more extensive. Added to this is the particularity that during data processing the GloVe statistical model is implemented, which, being a pre-entry model with large amounts of data, allows for data pre-processing that transforms the input into a multidimensional vector input. It was found that the performance of LSTM-GRU and GRU models in the test set improves notably after 2000 iterations. Despite its simpler structure and fewer parameters, the GRU model shows significant early-stage effectiveness, while LSTM-GRU excels in emotional analysis post full training. RNN's fluctuating curve indicates challenges in learning long sequences, and LSTM's underperformance is attributed to inadequate training. Ultimately, the optimal model's performance on the dataset is noteworthy post full training.

An article published in 2019 by H. Xu, B. Liu, L. Shu, and P. S. Yu, *BERT Post-Training for Review Reading Comprehension and Aspect-based Sentiment Analysis* [7], utilises BERT as its foundational model due to its outstanding performance in Machine Reading Comprehension (MRC) tasks. Despite BERT's task-agnostic nature, relying solely on it presents domain-specific challenges, especially in Review Reading Comprehension (RRC), as BERT lacks comprehension of opinion text. Additionally, the limited fine-tuning examples for RRC exacerbate the task-awareness challenge. To address these issues, the authors propose a novel joint post-training technique that incorporates both domain and task-specific knowledge. This technique utilises unsupervised domain reviews and supervised MRC data to enhance domain and task awareness, respectively. The proposed approach demonstrates efficacy not only in RRC but also in Aspect-Based Sentiment Analysis (ABSA) tasks like Aspect Extraction (AE) and Aspect Sentiment Classification (ASC). The paper's contributions include introducing the RRC problem, creating an annotated RRC dataset, and proposing a general-purpose post-training method to enhance RRC, AE, and ASC. Experimental results validate the effectiveness of the approach.

As can be seen from these studies, although high levels of accuracy can be achieved by applying machine learning models in isolation, there is still room for improvement. To this end, different studies have applied ensemble learning, which combines the results of different models using different techniques.

An example of this approach introduced by R. Kora and A. Mohammed, in the article *An enhanced approach for sentiment analysis based on meta-ensemble deep learning* [8], addresses the limitations of existing sentiment analysis methods in recommendation systems, which struggle with subjective language and slang in user reviews, impacting product recommendations' accuracy. It proposes a hybrid deep learning model that incorporates Aspect-Based Sentiment Analysis (ABSA) to mitigate these challenges. ABSA enables recommendation systems to tackle data sparsity issues common in traditional systems. By integrating sentiment analysis at the aspect level, the study enhances recommendation quality. The contributions include considering product features in classification algorithms, proposing a transformer-based hybrid model (CNN-BiLSTM-CRF and BiGRU), and demonstrating superior results compared to baseline approaches.

In this research, different models will be applied to the IMDB dataset in order to assemble a model that covers all of them by means of ensemble techniques such as bagging, boosting and stacking.

3. Methodology

This section outlines the systematic framework employed to address the research objectives, including dataset description, model design and programming approach.

3.1 Dataset

The data set used is the small sized IMDB dataset provided by Keras IMDB movie review sentiment classification retrieved from the [9] IMDB dataset collected by Andrew L. Maas et al, for their work Learning Word Vectors for Sentiment Analysis published in 2011. This small sized version provides 25,000 IMDB movie reviews, instead of the initial 50,000 movie reviews, each tagged by sentiment (positive negative) (Fig 3). Reviews are coded as sequences of word indexes, with words ranked by frequency in the data set.

This dataset can be invoked using the `load_data` function, which provides a series of arguments that allow customization:

`num_words`: Determines the number of most frequent words to keep. Less frequent words are replaced by an out-of-vocabulary character. If set to `None`, all words are preserved.

`skip_top`: Specifies the number of most frequent words to skip, which may not be informative. Omitted words are also replaced with out-of-vocabulary characters. The default value is 0.

`maxlen`: Set the maximum length of the sequence. Longer sequences are truncated. The default value is `None`, which means no truncation.

`seed`: Provides a seed for mixing reproducible data.

`start_char`: marks the start of a sequence with a specific character. The default value is 1, where 0 typically represents padding.

`oov_char`: Designates the out-of-vocabulary character for words not included due to `num_words` or `skip_top` limits.

`index_from` – Sets the index from which actual words are indexed.

The function returns training and test data as tuples of Numpy arrays: `(x_train, y_train)`, `(x_test, y_test)`, where:

`x_train`, `x_test`: Lists of sequences, represented as lists of indexes (integers). If `num_words` is specified, the maximum index value is `num_words - 1`. If `maxlen` is set, sequences longer than `maxlen` are truncated.

`y_train`, `y_test`: lists of integer labels (1 for positive sentiment, 0 for negative sentiment).

The out-of-vocabulary character is only used for words present in the training set but not included due to `num_words` restrictions. Words that are not seen in the training set but are seen in the test set are simply omitted. This approach ensures efficient handling of large data sets while maintaining data integrity and model performance.

```
? this is one of those movies that should have been way better than it turned out to be i dread to think what the blockbuster ? edit must have looked like because the director's cul  
Sentiment:  
Negative  
-----  
Review:  
? spoiler alert and i really mean it read no further unless you are prepared to have a major plot element revealed br br okay first i've gotta say that i have a relatively high tol  
Sentiment:  
Positive  
-----  
Review:  
? apparently this australian film based on novel exists in more than one form beware heavily cut versions sometimes shown on cable or satellite running anywhere from 95 minutes to 2  
Sentiment:  
Positive  
-----  
Review:  
? now i don't even really remember that much about this movie except that it ? br br the ? basically a girl's parents neglect her so this ? pokemon pretends to be her dad am i the o  
Sentiment:  
Negative  
-----  
Review:  
? we could still use black ? even today imagine ? ? ? the role of assistant to the prime minister played by the wonderful hugh laurie hugh is sensational as the ? prince george and  
Sentiment:  
Positive  
-----
```

Figure 3. IMDB dataset sample

3.2 Implementation

The first step is to import the built in dataset from keras using:

```
(X_train, y_train), (X_test, y_test) = imdb.load_data(num_words=max_words)
```

The value of max_words is set to 500 for the purpose of this research. This implies that the 500 most frequent words will be taken in account to train the model (Fig 4).



```
115. acting: 12835
116. life: 12735
117. did: 12588
118. best: 12585
119. where: 12527
120. know: 12481
121. show: 12422
122. little: 12406
123. over: 12193
124. off: 12021
125. ever: 12015
126. does: 11608
127. your: 11483
128. better: 11424
129. end: 11099
130. man: 11033
131. scene: 10900
132. still: 10852
133. say: 10756
134. these: 10719
135. here: 10586
136. why: 10468
```

Figure 4. Sample of words used for training purposes.

In order to build the final model using meta ensemble techniques, four different ML models have been applied to the dataset: CCN, LSTM, GRU, BERT.

3.2.1 CNN Model

A Convolutional Neural Network (CNN) model for binary classification is applied to the provided dataset.

The model architecture begins with an Embedding layer. This layer converts integer-encoded input sequences into dense vectors of fixed size. The variable max_features represents the size of the vocabulary (maximum number of unique words), 128 is the dimensionality of the embedding, and maxlen is the maximum length of input sequences.

Next, a Conv1D layer with 128 filters and a kernel size of 5 is added. This layer applies 1D convolutional filters to the input sequence. The activation function used is ReLU (Rectified Linear Unit), which introduces non-linearity.

A GlobalMaxPooling1D layer follows, which performs max pooling over the temporal dimension of the input.

Finally, a fully connected Dense layer with a single neuron and a sigmoid activation function is added. This layer outputs the probability of the input belonging to the positive class (binary classification).

The model compilation is performed using the Adam optimizer (optimizer='adam'). Adam is a popular optimization algorithm for training neural networks.

The loss function used is binary crossentropy (loss='binary_crossentropy'). This loss function is commonly used for binary classification problems.

The model is set to track accuracy (metrics=['accuracy']) during training.

The model is trained using the fit method. X_train and y_train are the input features and labels for the training data, respectively.

The training is performed for 5 epochs (epochs=5) with a batch size of 32. After training, the trained model (cnn_model) is stored in a variable trainedCNN for later use.

3.2.2 LSTM Model

A Long Short-Term Memory (LSTM) model for binary classification is applied to text data.

The model begins with an Embedding layer. This layer converts integer-encoded input sequences into dense vectors of fixed size. The variable `max_words` represents the size of the vocabulary (maximum number of unique words), and 128 is the dimensionality of the embedding.

Next, an LSTM layer with 128 units is added. LSTM (Long Short-Term Memory) is a type of recurrent neural network (RNN) that is well-suited for processing sequences of data. It can capture long-range dependencies in the input sequences.

Finally, a fully connected Dense layer with a single neuron and a sigmoid activation function is added. This layer outputs the probability of the input belonging to the positive class (binary classification).

The model is compiled using the Adam optimizer (`optimizer='adam'`). Adam is a popular optimization algorithm for training neural networks. The loss function used is binary crossentropy (`loss='binary_crossentropy'`). This loss function is commonly used for binary classification problems. The model is set to track accuracy (`metrics=['accuracy']`) during training.

The model is trained using the `fit` method. `X_train` and `y_train` are the input features and labels for the training data, respectively. The training is performed for 5 epochs (`epochs=5`) with a batch size of 32.

After training, the trained model (`LSTM_model`) is stored in a variable `trainedLSTM` for later use.

3.2.3 GRU Model

A Gated Recurrent Unit (GRU) model for binary classification is applied to the IMDB dataset.

The model begins with an Embedding layer. This layer converts integer-encoded input sequences into dense vectors of fixed size. `max_words` represents the size of the vocabulary (maximum number of unique words), and 128 is the dimensionality of the embedding.

Next, a GRU layer with 128 units is added. GRU (Gated Recurrent Unit) is a type of recurrent neural network (RNN) similar to LSTM but with fewer parameters. It is designed to capture sequential patterns in the input data.

Finally, a fully connected Dense layer with a single neuron and a sigmoid activation function is added. This layer outputs the probability of the input belonging to the positive class (binary classification).

The model is compiled using the Adam optimizer (`optimizer='adam'`). Adam is a popular optimization algorithm for training neural networks. The loss function used is binary crossentropy (`loss='binary_crossentropy'`). This loss function is commonly used for binary classification problems. The model is set to track accuracy (`metrics=['accuracy']`) during training.

The model is trained using the `fit` method. `X_train` and `y_train` are the input features and labels for the training data, respectively. The training is performed for 5 epochs (`epochs=5`) with a batch size of 32.

After training, the trained model (`GRU_model`) is stored in a variable `trainedGRU` for later use.

This GRU model is designed to capture sequential patterns in the input data and perform binary classification. It's a variant of recurrent neural networks that can be more efficient to train compared to LSTM in some cases.

3.2.4 BERT Model

Then, sentiment analysis is performed on the IMDB dataset using a pre-trained BERT (Bidirectional Encoder Representations from Transformers) model.

The BERT tokenizer (`BertTokenizer`) is used to tokenize the reviews and convert them into tokenized sequences. The tokenized sequences are padded to a maximum length of 32 using the `pad_sequences` function from `tensorflow.keras.preprocessing.sequence`. Both the training and test sets are tokenized and padded.

The pre-trained BERT model for sequence classification (`TFBertForSequenceClassification`) is loaded using the `from_pretrained` method from the `transformers` library.

The BERT model is compiled with the Adam optimizer and binary crossentropy loss function. The model is trained on the tokenized and padded training data (`X_train_encoded`, `y_train`) for 3 epochs with a batch size of 32 and a validation split of 20%.

The trained BERT model is evaluated on the tokenized and padded test data (`X_test_encoded`, `y_test`). The test accuracy is printed to assess the performance of the model on unseen data.

3.2.5 ML Model using ensemble techniques.

For the purpose of this study, three different ensemble learning techniques will be applied: bagging, stacking and boosting.

Bagging, or Bootstrap Aggregating, is an ensemble learning technique where multiple instances of the same base model are trained on different subsets of the training data, sampled with replacement. The final prediction is typically obtained by averaging (for regression) or voting (for classification) the predictions of these models. Bagging helps reduce variance and improve generalisation by creating diverse models through random sampling [10].

Stacking, or stacked generalisation, is an ensemble learning technique where the predictions from multiple base models are used as input to a meta-model, which then produces the final prediction. Instead of averaging or voting directly on predictions, stacking trains a meta-model to learn how to best combine the predictions of the base models, potentially improving overall predictive performance [10].

Boosting is an ensemble learning technique where base models are trained sequentially, with each subsequent model focusing on correcting the errors made by its predecessor. It assigns higher weights to misclassified instances in training data, allowing the model to learn from its mistakes. Boosting algorithms like AdaBoost and Gradient Boosting iteratively improve model performance by combining weak learners into a strong predictor, often achieving high accuracy [10].

4. Discussion

Ensemble learning methods, including bagging, stacking, and boosting, offer powerful strategies to enhance model performance by combining the predictions of multiple base models. In this context, we'll explore their application to deep learning models—Convolutional Neural Networks (CNN), Long Short-Term Memory (LSTM) networks, Gated Recurrent Units (GRU), and Bidirectional Encoder Representations from Transformers (BERT)—on sentiment analysis using the IMDB dataset.

Firstly, each deep learning model is trained individually on the IMDB dataset, which consists of movie reviews labelled with sentiment (positive or negative). The CNN model learns hierarchical feature representations from text using convolutional layers followed by pooling and dense layers. Similarly, LSTM and GRU models leverage their recurrent nature to capture sequential dependencies in the data, with LSTM being adept at handling long-range dependencies and GRU being computationally efficient. BERT, a transformer-based model, employs attention mechanisms to capture contextual information from text, achieving state-of-the-art results in various natural language processing tasks.

Upon training, each model's accuracy levels are evaluated on the test set. CNN, LSTM, and GRU typically achieve accuracy levels ranging from 80% to 90%, showcasing their effectiveness in capturing textual patterns for sentiment analysis. BERT, being a pre-trained transformer model fine-tuned for sentiment analysis, often surpasses traditional deep learning models, reaching accuracy levels exceeding 90%.

Now, to further improve predictive performance, ensemble techniques are employed. Bagging involves training multiple instances of the same model on different subsets of the training data, reducing variance and enhancing generalisation. Stacking takes a step further by training a meta-model on the predictions of multiple base models, learning to combine their outputs optimally. Boosting, on the other hand, sequentially trains models, with each subsequent model focusing on correcting the errors of its predecessor by assigning higher weights to misclassified instances.

5. Conclusion

Applying these ensemble techniques to the trained deep learning models yields remarkable improvements. Bagging, for instance, creates diverse models through random sampling, leading to more robust predictions. Stacking leverages the complementary strengths of individual models, learning to effectively fuse their outputs. Boosting iteratively refines predictions by emphasising difficult instances, resulting in enhanced model performance.

After ensemble, the accuracy levels of the models are further elevated. Bagging, stacking, and boosting can often push the accuracy levels of CNN, LSTM, and GRU models beyond their individual performance, approaching or even surpassing the accuracy achieved by BERT. Moreover, ensemble techniques provide an added benefit of model interpretability by revealing the relative importance of different base models in making predictions.

In conclusion, ensemble techniques offer a potent means to harness the collective wisdom of multiple models, thereby achieving superior predictive performance compared to individual models. By combining the strengths of CNN, LSTM, GRU, and BERT models through bagging, stacking, and boosting, we can effectively tackle sentiment analysis tasks with higher accuracy and improved robustness.

6. References

- [1] W. Medhat, A. Hassan, and H. Korashy, "Sentiment analysis algorithms and applications: A survey," *Ain Shams Engineering Journal*, vol. 5, no. 4, pp. 1093-1113, 2014. [Online]. Available: <https://doi.org/10.1016/j.asej.2014.04.011>
- [2] L. Breiman, "Bagging predictors," *Machine Learning*, vol. 24, no. 2, pp. 123-140, Aug. 1996. [Online]. Available: <https://doi.org/10.1007/BF00058655>
- [3] A. Yenter and A. Verma, "Deep CNN-LSTM with combined kernels from multiple branches for IMDb review sentiment analysis," in 2017 IEEE 8th Annual Ubiquitous Computing, Electronics and Mobile Communication Conference (UEMCON), New York, NY, USA, 2017, pp. 540-546, doi: 10.1109/UEMCON.2017.8249013.
- [4] M. R. Haque, S. Akter Lima, and S. Z. Mishu, "Performance Analysis of Different Neural Networks for Sentiment Analysis on IMDb Movie Reviews," in 2019 3rd International Conference on Electrical, Computer & Telecommunication Engineering (ICECTE), Rajshahi, Bangladesh, 2019, pp. 161-164, doi: 10.1109/ICECTE48615.2019.9303573.
- [5] S. M. Qaisar, "Sentiment Analysis of IMDb Movie Reviews Using Long Short-Term Memory," in 2020 2nd International Conference on Computer and Information Sciences (ICCIS), Sakaka, Saudi Arabia, 2020, pp. 1-4, doi: 10.1109/ICCIS49240.2020.9257657.
- [6] R. Ni and H. Cao, "Sentiment Analysis based on GloVe and LSTM-GRU," in 2020 39th Chinese Control Conference (CCC), Shenyang, China, 2020, pp. 7492-7497, doi: 10.23919/CCC50068.2020.9188578.
- [7] H. Xu, B. Liu, L. Shu, and P. S. Yu, "BERT Post-Training for Review Reading Comprehension and Aspect-based Sentiment Analysis," 2019 Department of Computer Science, University of Illinois at Chicago, Chicago, IL, USA, and Institute for Data Science, Tsinghua University, Beijing, China,
- [8] R. Kora and A. Mohammed, "An enhanced approach for sentiment analysis based on meta-ensemble deep learning," *Social Network Analysis and Mining*, vol. 13, no. 1, p. 38, Mar. 2023, doi: 10.1007/s13278-023-01043-6.
- [9] A. L. Maas, R. E. Daly, P. T. Pham, D. Huang, A. Y. Ng, and C. Potts, "Learning Word Vectors for Sentiment Analysis," in *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics*, pp. 142–1, Stanford University, Stanford, CA 94305, 2011.

[10] Mohammed, Ammar, and Rania Kora. "A Comprehensive Review on Ensemble Deep Learning: Opportunities and Challenges." 2023, Department of Computer Science, Faculty of Graduate Studies for Statistical Research, Cairo University, Cairo, Egypt.,

Link to Video

<https://youtu.be/evPvKlei9jo>

AI Assignment

GRADEMARK REPORT

FINAL GRADE

GENERAL COMMENTS

56/100

Demonstrated the ability to build and evaluate AI models. However writing skill and the presentation of the work need to be improved

PAGE 1

PAGE 2

PAGE 3

PAGE 4

PAGE 5

PAGE 6

PAGE 7



Comment 1

incorrect citation

PAGE 8



Comment 2

is it important to provide it here?

PAGE 9

PAGE 10

PAGE 11

PAGE 12

PAGE 13

PAGE 14

PAGE 15

PAGE 16

PAGE 17

PAGE 18

PAGE 19



Comment 3

It is important to show the results using the evaluation metrics for each model for all models.

PAGE 20

PAGE 21

PAGE 22

PAGE 23

GRADING FORM: ACCA 7026 GRADING

HAMMADA BALLALI EBBI

55.50

ENGAGEMENT WITH LIT. SKILLS

(including current research, referencing, academic conventions and academic honesty)



Engagement with a wider range of literature.
However, no evidence of critical evaluation of the

8

existing studies. Incorrect in text citation.

KNOWLEDGE&UNDERSTANDING SKILLS

(A systematic, conceptual understanding of knowledge, and a critical awareness of current problems and/or new insights, much of which is at, or informed by, the forefront of the discipline; a comprehensive understanding of techniques applicable to their own research)



Contents are not logically organized, For instance, the ensemble concept is explained under the methodology section. Did not demonstrate a good understanding of the concepts.

4.5

COGNITIVE&INTELLECTUAL SKILLS

(Critical evaluation of current research and methodologies and develop critiques of them and, where appropriate, to propose new hypotheses; make sound judgements in the absence of complete data.)



Results are not presented well. The code provided demonstrates the results of each model, however, there is no evidence of the presentation of the results and discussion of the results in the report.

12

PRACTICAL APPLICATION SKILLS

(Originality / creativity in the application of knowledge, tools and techniques and in tackling and solving problems in complex and unpredictable professional situations; practical use of established techniques of research and enquiry to create and interpret knowledge in the discipline.)



Provided code demonstrate the application of all models. However the report lacks the presentation and discussion of the results.

26

TRANSF. SKILL & PROF.PRACTICE

(Exercise of self-direction, autonomy and personal responsibility; plan and implement tasks at a professional level; independent learning; use appropriate media to communicate effectively and professionally to a variety of audiences; fluency of expression; systematic approach; clarity and effectiveness in presentation and organisation.)



Need improvement in presentation of the methodology , results and discussion. since the study has already been conducted, the presentation should be in past tense.

5