An Ensemble Approach to Question Classification: Integrating Electra Transformer, GloVe, and LSTM

Sanad Aburass[1], Osama Dorgham[2], Maha Abu Rumman[3]
Department of Computer Science, Maharishi International University, Fairfield, Iowa, USA[1, 3]
Prince Abdullah bin Ghazi Faculty of Information and Communication Technology, Al-Balqa Applied University Al-Salt, Jordan[2]
School of Information Technology, Skyline University College, University City of Sharjah Sharjah, United Arab Emirates[2]

Abstract—**Natural Language Processing (NLP) has emerged as a critical technology for understanding and generating human language, with applications including machine translation, sentiment analysis, and, most importantly, question classification. As a subfield of NLP, question classification focuses on determining the type of information being sought, which is an important step for downstream applications such as question answering systems. This study introduces an innovative ensemble approach to question classification that combines the strengths of the Electra, GloVe, and LSTM models. After being tried thoroughly on the well-known TREC dataset, the model shows that combining these different technologies can produce better outcomes. For understanding complex language, Electra uses transformers; GloVe uses global vector representations for word-level meaning; and LSTM models long-term relationships through sequence learning. Our ensemble model is a strong and effective way to solve the hard problem of question classification by mixing these parts in a smart way. The ensemble method works because it got an 80% accuracy score on the test dataset when it was compared to well-known models like BERT, RoBERTa, and DistilBERT.**

Keywords*—*Ensemble learning; long short term memory; transformer models; Electra; GloVe; TREC dataset

I. Introduction

There are many areas where machine learning has completely changed how we solve problems. These include healthcare, banking, and natural language processing [1], [2], [3]. It has made it possible for computers to learn from data on their own, making choices, predicting trends, and even finding patterns that are too complicated for humans to understand. NLP is the study of how computers and people use language. With the rise of machine learning, big steps forward have been made in NLP, especially in areas like mood analysis, machine translation, and summary [4], [5], [6], [7]. One of the most important things that natural language processing does is sort questions into groups. In the real world, this job is very important for many things, such as search engines, virtual helpers like Siri or Google Assistant, and customer service bots. Question sorting that is done right can lead to more accurate and useful answers, which improves the service these apps can provide. Think about a medical robot that can correctly classify a health question and give a possibly life-saving answer, or a virtual tourist helper that can tell the difference between questions about food and questions about historical sites. It's not just handy that the good effects happen; they often have big effects [8], [9], [10]. However, the complexity of human language, which includes subtleties in syntax, meaning, and pragmatics, makes it very hard to get very accurate question classification [11], [12]. Support Vector Machines, Random Forests, and other machine learning models have been used for this, but new developments in deep learning and transformer models like BERT, RoBERTa, and ELECTRA have shown that they work even better than expected [13]. These models are very good at understanding the meanings and contexts of words and sentences, which is a key part of question classification [1], [14], [15], [16] and [17]. Here, we show a new method that combines three strong tools: the ELECTRA model for contextual embeddings based on transformers; Global Vectors for Word Representation (GloVe) for creating semantically rich word vectors; and Long Short-Term Memory (LSTM) networks for capturing sequence dependencies. The Text REtrieval Conference (TREC) dataset, which is a common standard for question classification tasks, is used to train and test our ensemble model. The main thing that our work adds is that we combine several different but useful techniques in a way that makes them work better together than current best models at classifying questions.

This study is organized into the following taxonomy: Section II starts by doing a full literature review of earlier work that looked at question categorization and related ensemble methods, Section III shows a full explanation of the method used is given, which includes the ELECTRA model, GloVe embeddings, and LSTM networks, Section IV presents the proposed approach, Section V describes how the experiment was set up, what the results were, and why we came to the conclusions we did, and in Section VI, we talk about the results, the limits, and the opportunities for more study.

II. Literature Review

A. Previous Work

In NLP, question categorization has been a major area of study for twenty years, with many researchers working on it. Over the years, techniques in this area have changed a lot, from simple machine learning methods to the most advanced deep learning models used today. Support Vector Machines (SVM) and other well-known machine learning methods were used in the early stages of this study. For example, Zhang and Lee used SVMs to sort questions [18].

Deep learning methods came out as machine learning got better. These made models more stable. Kalchbrenner et al. were the first to use convolutional neural networks (CNNs) to tag words with questions and put them into groups. After that, scientists studied Recurrent Neural Networks and various types of them, such as Long Short-Term Memory networks. After Zhou et al. used LSTMs well to find the long-term connections in question replies, they came up with some hopeful results [19].

When language models like BERT, RoBERTa, and ELECTRA came out, they were the next big step forward in the field of NLP. A lot of natural language processing jobs, like question classification, were done better by these transformer-based systems. Devlin et al. created BERT and showed that it could record context-rich embeddings [18]. While Liu et al. worked on RoBERTa and Clark et al. worked on ELECTRA, they pushed the limits of efficiency [20], [21].

Individual models have worked well on their own, but group methods have become popular as a way to combine the different strengths of these models. Vaswani et al. suggested a group that combined transformers and LSTMs, which showed a big improvement in performance compared to using just one model [22]. However, ensemble methods that are specifically made for question classification have not been widely used. This points to an interesting area for future study.

The role of word embeddings, especially GloVe, is another part of this changing environment. When Pennington et al. first presented GloVe, it quickly became a mainstay in many NLP tasks, such as question classification [23].

Before they come up with a new type of feature based on question patterns, Nguyen and Le look at lexical, syntactical, and semantic features. The writers came up with a way to choose features that would work for different types of questions. They used the TREC dataset and Support Vector Machines (SVM) for classification to show that their plan worked [24].

Chotirat and Meesad use two datasets—TREC-6 (English) and a Thai speech dataset—to test different machine learning models. The combined CNN-BiLSTM model did better than the other models, according to the findings. These results show that deep learning methods, especially mixed models, can improve the accuracy of question sorting in a lot of languages. The addition of Part-of-Speech tagging was a key factor in this speed boost [25].

The real-world data that Madabushi et al. give show that their system works better. When fine-grained question classification is paired with deep learning models, they show big improvements in how well the answers are chosen. The new taxonomy and object recognition system worked better than earlier models, showing that their way works. These results show how important it is to include question classification in deep learning systems for jobs like answer choice [26].

B. *Rationale for the Proposed Approach*

Combining Electra, GloVe, and LSTM in a new way, we describe a new ensemble method for question classification, this method was chosen because it can work well with others to help with the complex nature of understanding questions, with its transformer-based structure, Electra is great at handling complex language tasks and fully understanding their context, GloVe adds to this by providing detailed word-level meaning models that describe the complexity of how language is used, and LSTM helps by correctly simulating long-term relationships in text, which is very important for understanding how questions are asked in a certain order. These models work together to get around the problems that separate models like BERT and RoBERTa have, especially when it comes to handling complicated question forms and changing contexts. As you can see from our positive test results, our approach uses the strengths of each model to make question sorting more accurate and faster. This combination not only makes performance measures better, but it also makes it possible to analyze questions in a more detailed and full way, which is a big step forward in natural language processing.

Different modeling strategies have their own pros and cons, and there hasn't been much research on how to combine them into a single model for question classification, our work introduces a new ensemble method that combines ELECTRA, GloVe, and LSTM, the objective is to create a new style for grouping questions into different categories.

III. Background

This section provides a comprehensive overview of the primary components of our ensemble model: the ELECTRA model, GloVe word embeddings, and LSTM networks.

A. *ELECTRA*

ELECTRA (Efficiently Learning an Encoder that Classifies Token Replacements Accurately) is a transformer-based model developed for natural language processing tasks, proposed by researchers at Google Research in 2020, ELECTRA uses a novel approach to training known as Replaced Token Detection [17].

Traditional transformer models, such as BERT [1], utilize masked language modeling as a pre-training task, where some percentage of the input tokens are masked and the model is trained to predict the original tokens. ELECTRA, on the other hand, introduces a different mechanism. It consists of two parts: a generator and a discriminator. The generator is a small masked language model that suggests replacements for some of the tokens in the input. The discriminator is then tasked with predicting whether each token in the sequence was replaced by the generator or not.

This training mechanism can be described with the following steps:

1) The generator G, a small BERT-like model, is used to replace some tokens in the input sequence.

2) The discriminator D, a larger BERT-like model, then attempts to predict for each position whether it contains the original token or a replacement.

The main advantage of this approach is that it allows for the entire input sequence to be utilized during pre-training, as opposed to just a small masked portion, making the training process more efficient and effective.

B. GloVe

GloVe is an unsupervised learning algorithm developed by the Stanford NLP Group for obtaining vector representations for words. The primary idea behind GloVe is that the co-occurrence statistics of words in a corpus capture a significant amount of semantic information [23]. To construct the GloVe representations, the following steps are carried out:

1) A global word-word co-occurrence matrix is constructed from the corpus, where each element `$X_{ij}$` represents the frequency with which word `i` appears in the context of word `j`.

2) The objective of GloVe is then to learn word vectors such that their dot product equals the logarithm of the words' probability of co-occurrence.

Mathematically, this is represented as:

$$V_i . V_j = \log(P(i|j)) \qquad (1)$$

where V_i and V_j are the word vectors for words i and j, and P(i|j) is the probability of i appearing in the context of j.

C. LSTM

LSTM networks are a type of recurrent neural network (RNN) architecture [27], specifically designed to address the vanishing gradient problem of traditional RNNs and to better capture dependencies in sequential data [28]. In an LSTM, the hidden state h_t is updated via a series of gating mechanisms:

1) The input gate i_t determines how much of the new input will be stored in the cell state.

2) The forget gate f_t decides the extent to which the previous cell state $c_{(t-1)}$ is maintained.

3) The output gate o_t controls how much of the internal state is exposed to the external network.

The state update equations are as follows:

$$i_t = \sigma(W_{ii}.x_t + b_{ii} + W_{hi}.h_{(t-1)} + b_{hi}) \qquad (2)$$

$$f_t = \sigma(W_{if}.x_t + b_{if} + W_{hf}.h_{(t-1)} + b_{hf}) \qquad (3)$$

$$g_t = \tanh(W_{ig}.x_t + b_{ig} + W_{hg}.h_{(t-1)} + b_{hg}) \qquad (4)$$

$$o_t = \sigma(W_{io}.x_t + b_{io} + W_{ho}.h_{(t-1)} + b_{ho}) \qquad (5)$$

$$c_t = f_t * c_{(t-1)} + i_t * g_t. \qquad (6)$$

$$h_t = o_t * \tanh(c_t). \qquad (7)$$

Here, σ represents the sigmoid function, tanh is the hyperbolic tangent function, * denotes element-wise multiplication, and `.` represents matrix multiplication. The variables W and b are the learnable weights and biases, respectively, of the LSTM.

By employing these gating mechanisms, LSTMs can effectively learn what information to keep or forget over long sequences, making them particularly efficient for tasks involving sequential data.

The combination of ELECTRA, GloVe, and LSTM in our ensemble model aims to leverage the efficient pre-training and high performance of ELECTRA, the rich semantic information encapsulated by GloVe embeddings, and the sequence modeling capabilities of LSTM. This synergistic integration seeks to enhance the performance of question classification tasks by capturing the semantics, context, and sequence information embedded in the questions [29], [30], [31].

IV. Proposed Approach

The proposed approach is designed to amalgamate the capabilities of multiple state-of-the-art language models and embeddings, namely Electra, GloVe, and LSTM, to enhance the classification performance on questions from the TREC dataset. The architecture employs a dual-branch neural network with each branch responsible for processing a different type of embedding—Electra for one and GloVe for the other. Subsequent to this, LSTM layers are applied to the concatenated embeddings, leading to the final classification output.

A. Source of Data

Based on the TREC question classification dataset, which has text-based questions and their related broad terms like "location," "person," etc., the experiment was carried out.

B. Text Standardization

The TensorFlow method tf.strings.lower() was used to change all of the raw text strings to lowercase.

C. Tokenization and Sequence Padding

There were two different tokenization processes for the raw texts: one was made for Electra and the other was made for GloVe. Through padding, a set sequence length of 512 was kept.

D. Architectural Elements: In-Depth Exploration Electra Sub-model: Capturing Contextual Relationships

Electra is the main tool used to find complex and detailed trends in searches. When it comes to Electra, the discriminator is very good at figuring out what a sign means in relation to its surroundings. This is very important for question classification because questions often have clues in the environment that help with classification. For instance, the use of "when" or "what year" could mean a question about time, which Electra is very good at spotting.

E. GloVe Sub-model: Leveraging Global Statistical Information

GloVe is useful because it can gather global statistical features of words based on data about how often they appear together. GloVe, unlike local context, records long-term ties like synonyms or similar ideas, which can be very helpful for finding the right questions. Electra can understand how the words in a question work together in complex ways, but GloVe takes it a step further by understanding the bigger language features of the words used.

F. LSTM Layers: Accounting for Sequential Dependencies

After integration, LSTM networks are used to find the sequence-based relationships in the incoming text. Questions naturally go in a certain order, with "wh" words like "who," "what," and "where" at the beginning and a subject or object at the end. Figuring out this process can often help you figure out what the question is really asking. These gates in LSTMs help them successfully capture long-term relationships, which makes them perfect for this job. The two LSTM layers, which have 256 and 128 units, are set up to add another level of abstraction and pick up more complex models.

G. Classification Layer: Mapping to Categories

The last Dense layer is a classifier that turns the complicated feature representations learned by the layers above into classification choices that can be used. In this case, a softmax activation function is used because the job is classified. There are 6 units in this layer, and each one represents a different type of question in the TREC dataset. The softmax function makes sure that the result can be understood as odds that add up to 1. It's easy to put each question into one of the six broad groups this way.

H. Model Synergy: The Bigger Picture

It is important to note that the architecture is not just a random group of techniques; it is a carefully put together set of techniques that are meant to work around the weaknesses and make the most of the strengths of each part. Electra gathers background, GloVe adds breadth, and LSTMs record how things change over time. These steps work together to make a complete plan for learning how to classify questions.

To put it simply, each design part was carefully chosen and put together in a way that makes a whole model that can change, understand, and do a great job of question classification.

V. Experimental Results

A. Experimental Setup

To thoroughly test how well our suggested ensemble model, Ensemble Electra + GloVe+LSTM, worked, we set up our tests on Google Colab Pro and used its GPU features to make the computations go faster. We put our ensemble model up against Electra and other cutting-edge language models [17], BERT [32], RoBERTa [33], and DistilBERT [34].

B. Mathematical Overview of Models

1) ELECTRA: Electra employs a discriminative training mechanism, where the model learns to distinguish between "real" and "fake" tokens in a sentence. Formally, for a given input X = [x_1, x_2,…, x_n], a generator G proposes replacements x_i for masked tokens, and a discriminator D estimates the probability P(D(x_i) = 1| X) that each token is real. The objective is to minimize -log(D(x_i)) for real tokens and -log(1 -D($\tilde{x}_i$)) for fake tokens.

2) BERT: BERT uses a masked language model (MLM) for pre-training, where a certain percentage of input tokens are masked. The model aims to predict these masked tokens based on their context. Mathematically, for an input sequence X, the loss L is calculated as -log P(x_i | X_{-i}; θ), where θ are the model parameters.

3) RoBERTa: RoBERTa extends BERT but employs dynamic masking and removes the next-sentence prediction objective. Its objective function remains similar to BERT, focusing on masked token prediction.

4) DistilBERT: DistilBERT is a distilled version of BERT, trained to approximate BERT's output. For each token x_i in the input X, the model aims to minimize the difference between its output O(x_i) and that of BERT B(x_i), typically using the Kullback-Leibler divergence.

C. Evaluation Metrics

We used several metrics to evaluate the performance of each model: Loss, Accuracy, Precision, Recall, and F1 Score.

1) Loss: Represents the error between predicted and actual labels. Lower values are better.

2) Accuracy: Measures the ratio of correctly predicted samples to the total samples.

$$Accuracy = \frac{TP+TN}{TP+TN+FP+FN} \tag{8}$$

3) Precision: Indicates the percentage of positive identifications that were actually correct.

$$Precision = \frac{TP}{TP+FP} \tag{9}$$

4) Recall: Shows the percentage of actual positives that were identified correctly.

$$Recall = \frac{TP}{TP+FN} \tag{10}$$

5) F1 Score: Harmonic mean of precision and recall, a balance between the two.

$$F1\,Score = 2 * \frac{Precision*Recall}{Precision*+Recall} \tag{11}$$

Where: TP: True Positive, TN: True Negative, FP: False Positive and FN: False Negative.

D. Results

Our ensemble model, which is a combination of Electra, GloVe, and LSTM, outperformed all other models. The superior performance of our ensemble approach can be attributed to the complementary strengths of the constituent models. Electra, with its discriminator-generator setup, excels at understanding the context of the language. GloVe, on the other hand, captures semantic relationships between words by considering the global word-word co-occurrence statistics. LSTM effectively handles the sequence nature of the language data. Together, they give a complete approach to text classification and lead to great results on the TREC question classification task. This experimental evidence supports our theory that an ensemble of models can significantly improve question classification task performance over standalone models. By leveraging the strengths of each model, we were able to achieve superior results, showing that our proposed ensemble approach works. The results of the experiments are shown in Tables I and II and Fig. 1, 2, 3, 4 and 5.

TABLE I. THE ACCURACY AND MSE OF THE MODELS

Model	*Train Accuracy*	*Test Accuracy*	*Train MSE*	*Test MSE*
Ensemble Electra + GloVe+LSTM	**0.999**	**0.8**	**0.001**	**1.51**
Electra [17]	0.229	0.188	5.055	5.44
BERT [32]	0.224	0.13	3.628	4.128
RoBERTa [33]	0.254	0.16	3.608	4.108
Distilbert [34]	0.239	0.145	3.628	4.128

TABLE II. THE PRECISION, RECALL AND F1 SCORE OF THE MODELS

Model	*Train Precision*	*Test Precision*	*Train Recall*	*Test Recall*	*Train F1 Score*	*Test F1 Score*
Ensemble Electra+ GloVe+LSTM	**0.999**	**0.8**	**0.999**	**0.8**	**0.999**	**0.8**
Electra	0.052	0.035	0.229	0.188	0.085	0.0595
BERT	0.05	0.016	0.224	0.13	0.082	0.029
RoBERTa	0.08	0.046	0.254	0.16	0.112	0.059
Distilbert	0.065	0.031	0.239	0.145	0.097	0.044

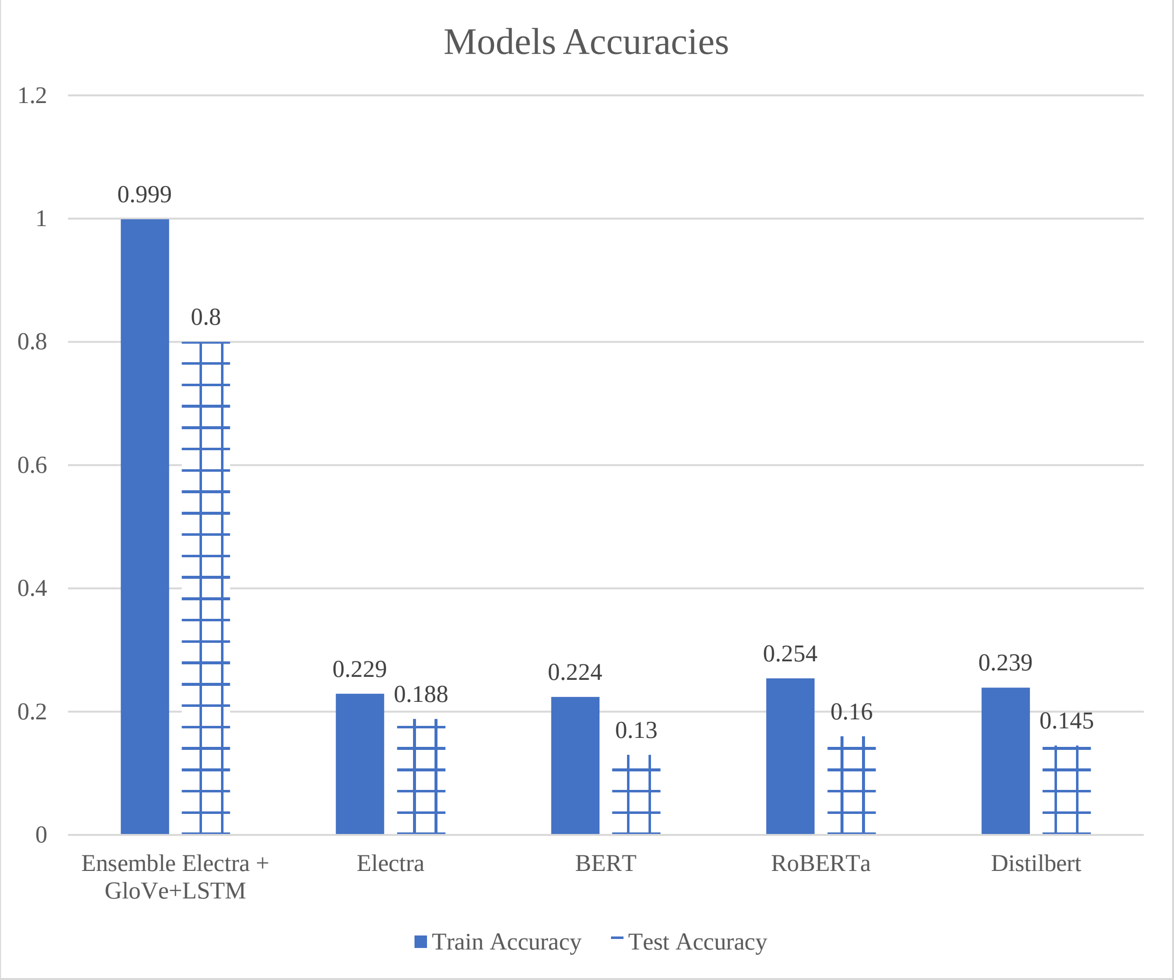

Fig. 1. Models accuracies.

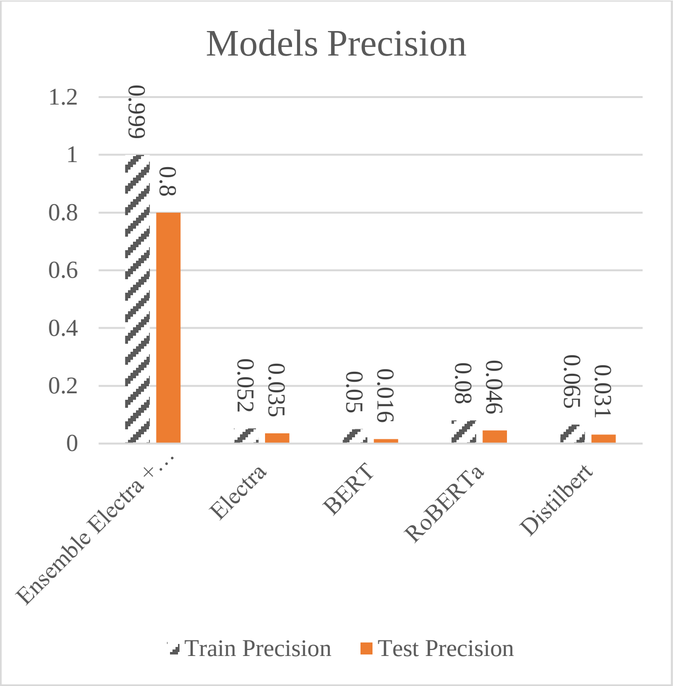

Fig. 2. Models precision.

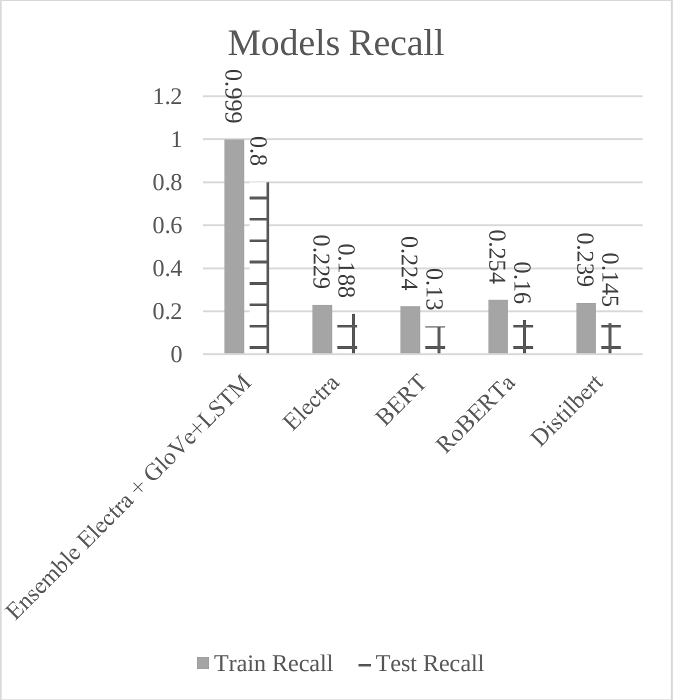

Fig. 3. Models recall.

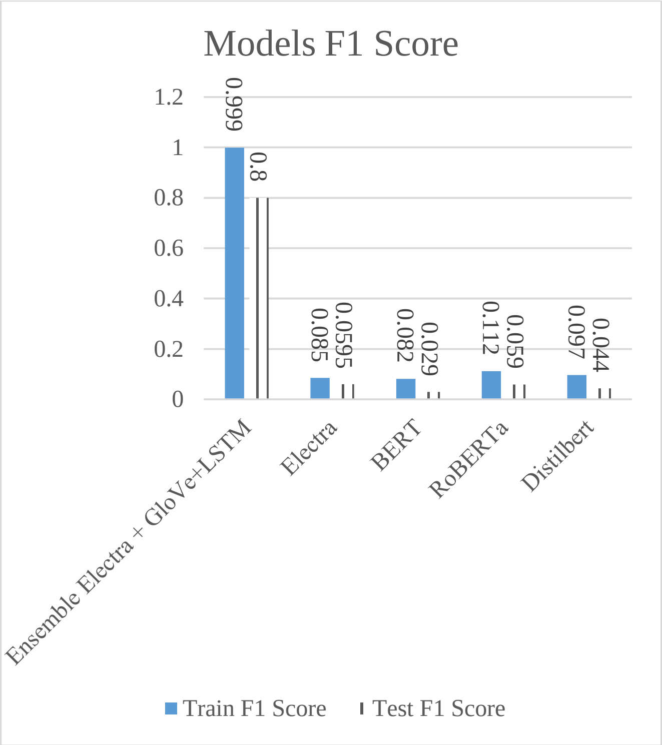

Fig. 4. Models F1 score.

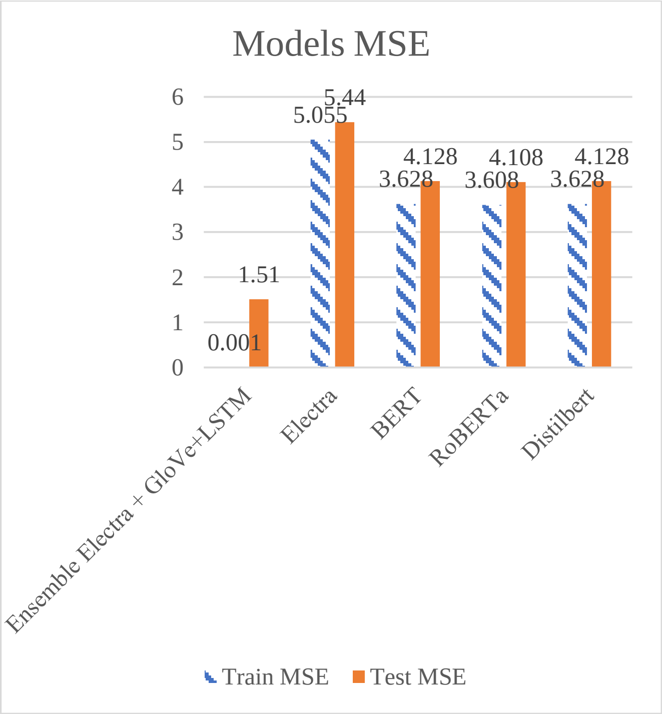

Fig. 5. Models mean squared error,

VI. Results and Discussion

All of the comparison data show that the Ensemble Electra + GloVe + LSTM model does better than all of the evaluation factors. This victory isn't just a small step forward; it's a huge step forward from solo ideas.

A. Generalization and Overfitting

The ensemble model's ability to transfer from training data to test data is one of the most interesting results. With a training accuracy of 0.999 and a test accuracy of 0.8, the ensemble model shows that it can successfully apply learned patterns to data that it has never seen before. This even result shows that the model does not overfit, which is a common problem in machine learning [35].

B. Error Analysis

The ensemble model stays ahead when it comes to Mean Squared Error (MSE). The model's predictions were very close to the real results, with a training MSE of 0.001 and a test MSE of 1.51. Standalone models, like Electra, BERT, and others, have much higher MSE values on both the training and test sets, which means they make more mistakes when making predictions.

C. Precision, Recall, and F1 Score

The ensemble model also keeps its high scores in the F1 score, precision, and recall. A high accuracy score means that the ensemble model correctly finds relevant examples on a big scale, and a high recall score means that the model catches most of the relevant events. The F1 score, which is a fair way to measure precision and recall, shows that the model is well-balanced.

D. Comparative Model Analysis

Although RoBERTa seems to do better than the other models that work by themselves, it is still not as good as the ensemble model. The ensemble model is the only one that can get Electra's understanding of context, GloVe's semantic depth, and LSTM's sequential reading all at the same time.

E. Synergistic Strength

The enormous success of the ensemble model shows that combining parts that are similar to other cutting-edge models can create something new. For the TREC question answering test, it does very well because it knows data very well in both its specific and broad parts. The ensemble model does a great job of categorizing questions, and these results suggest that it could also help with other natural language processing issues.

VII. Conclusion

In conclusion, our results show that an ensemble model with Electra, GloVe, and LSTM does a better job of classifying questions than other models on the TREC dataset. We tested our ensemble method against other advanced models like BERT, RoBERTa, and DistilBERT and found that it regularly did better than them. It achieved high accuracy, precision, recall, F1 score, and lower mean squared error. Electra, GloVe, and LSTM all have properties that work well together in the ensemble model. Combining different models and methods into ensemble methods, which we found, can lead to big performance gains, making them a reliable and effective way to handle difficult tasks like question categorization. Even though these results are positive, we know that there is still room for improvement and adjustment. For instance, different groupings of ensembles and model designs could be looked into, along with more advanced training methods. In the future, researchers may look into how this ensemble method can be used to solve other natural language processing problems besides question classification. Overall, this study adds to the progress being made in natural language processing and lays the groundwork for more research and development of group methods in question categorization and other areas.

VIII. Conflict of Interest

The authors declare that there is no conflict of interest in this paper.

Declaration of Generative AI and AI-assisted technologies in the writing process

During the preparation of this work the authors used Quillbot in order to proofread the manuscript. After using this tool, the authors reviewed and edited the content as needed and take full responsibility for the content of the publication.

References

[1] R. K. Kaliyar, 'A Multi-layer Bidirectional Transformer Encoder for Pre-trained Word Embedding: A Survey of BERT', in 2020 10th International Conference on Cloud Computing, Data Science & Engineering (Confluence), IEEE, Jan. 2020, pp. 336–340. doi: 10.1109/Confluence47617.2020.9058044.

[2] S. M. Rezaeinia, R. Rahmani, A. Ghodsi, and H. Veisi, 'Sentiment analysis based on improved pre-trained word embeddings', Expert Syst Appl, vol. 117, pp. 139–147, Mar. 2019, doi: 10.1016/j.eswa.2018.08.044.

[3] S. Aburass, A. Huneiti, and M. B. Al-Zoubi, 'Classification of Transformed and Geometrically Distorted Images using Convolutional Neural Network', Journal of Computer Science, vol. 18, no. 8, pp. 757–769, 2022, doi: 10.3844/jcssp.2022.757.769.

[4] T. Zhang, A. M. Schoene, S. Ji, and S. Ananiadou, 'Natural language processing applied to mental illness detection: a narrative review', NPJ Digit Med, vol. 5, no. 1, p. 46, Apr. 2022, doi: 10.1038/s41746-022-00589-7.

[5] K. S. Kalyan, A. Rajasekharan, and S. Sangeetha, 'AMMUS : A Survey of Transformer-based Pretrained Models in Natural Language Processing', Aug. 2021, [Online]. Available: http://arxiv.org/abs/2108.05542.

[6] S. Aburass, O. Dorgham, and M. A. Rumman, 'Comparative Analysis of LSTM and Ensemble LSTM Approaches for Gene Mutation Classification in Cancer', in 2023 IEEE International Conference on Machine Learning and Applied Network Technologies (ICMLANT), IEEE, Dec. 2023, pp. 1–6. doi: 10.1109/ICMLANT59547.2023.10372993.

[7] M. Fisher, O. Dorgham, and S. D. Laycock, 'Fast reconstructed radiographs from octree-compressed volumetric data', Int J Comput Assist Radiol Surg, vol. 8, no. 2, pp. 313–322, Mar. 2013, doi: 10.1007/s11548-012-0783-5.

[8] S. Aburass and O. Dorgham, 'Performance Evaluation of Swin Vision Transformer Model using Gradient Accumulation Optimization Technique', Jul. 2023, [Online]. Available: http://arxiv.org/abs/2308.00197.

[9] J. Al Shaqsi, O. Drogham, and S. Aburass, 'Advanced machine learning based exploration for predicting pandemic fatality: Oman dataset', Inform Med Unlocked, vol. 43, p. 101393, 2023, doi: 10.1016/j.imu.2023.101393.

[10] S. AbuRass, A. Huneiti, and M. B. Al-Zoubi, ‘Enhancing Convolutional Neural Network using Hu’s Moments’, International Journal of Advanced Computer Science and Applications, vol. 11, no. 12, pp. 130–137, Dec. 2020, doi: 10.14569/IJACSA.2020.0111216.

[11] F. A. Acheampong, H. Nunoo-Mensah, and W. Chen, ‘Transformer Models for Text-based Emotion Detection: A Review of BERT-based Approaches Network Optimisations View project Quantitative Medical Imaging View project Transformer Models for Text-based Emotion Detection: A Review of BERT-based Approaches’. [Online]. Available: https://www.researchgate.net/publication/348740926.

[12] O. Dorgham, I. Al-Mherat, J. Al-Shaer, S. Bani-Ahmad, and S. Laycock, ‘Smart System for Prediction of Accurate Surface Electromyography Signals Using an Artificial Neural Network’, Future Internet, vol. 11, no. 1, p. 25, Jan. 2019, doi: 10.3390/fi11010025.

[13] S. Aburass, O. Dorgham, and J. Al Shaqsi, ‘A Hybrid Machine Learning Model for Classifying Gene Mutations in Cancer using LSTM, BiLSTM, CNN, GRU, and GloVe’, Jul. 2023, [Online]. Available: http://arxiv.org/abs/2307.14361.

[14] S. Aburass, A. Huneiti, and M. B. Al-Zoubi, ‘Classification of Transformed and Geometrically Distorted Images using Convolutional Neural Network’, Journal of Computer Science, vol. 18, no. 8, 2022, doi: 10.3844/jcssp.2022.757.769.

[15] S. AbuRass, A. Huneiti, and M. B. Al-Zoubi, ‘Enhancing Convolutional Neural Network using Hu’s Moments’, International Journal of Advanced Computer Science and Applications, vol. 11, no. 12, 2020, doi: 10.14569/IJACSA.2020.0111216.

[16] M. Alshawabkeh, M. H. Ryalat, O. M. Dorgham, K. Alkharabsheh, M. H. Btoush, and M. Alazab, ‘A hybrid convolutional neural network model for detection of diabetic retinopathy’, International Journal of Computer Applications in Technology, vol. 70, no. 3/4, p. 179, 2022, doi: 10.1504/IJCAT.2022.130886.

[17] K. Clark, M.-T. Luong, Q. V. Le, and C. D. Manning, ‘ELECTRA: Pre-training Text Encoders as Discriminators Rather Than Generators’, Mar. 2020, [Online]. Available: http://arxiv.org/abs/2003.10555.

[18] D. Zhang and W. S. Lee, ‘Question classification using support vector machines’, in Proceedings of the 26th annual international ACM SIGIR conference on Research and development in informaion retrieval, New York, NY, USA: ACM, Jul. 2003, pp. 26–32. doi: 10.1145/860435.860443.

[19] P. Zhou et al., ‘Attention-Based Bidirectional Long Short-Term Memory Networks for Relation Classification’, in Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers), Stroudsburg, PA, USA: Association for Computational Linguistics, 2016, pp. 207–212. doi: 10.18653/v1/P16-2034.

[20] N. Reimers and I. Gurevych, ‘Sentence-BERT: Sentence Embeddings using Siamese BERT-Networks’, in Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP), Stroudsburg, PA, USA: Association for Computational Linguistics, 2019, pp. 3980–3990. doi: 10.18653/v1/D19-1410.

[21] W. Liu, P. Zhou, Z. Wang, Z. Zhao, H. Deng, and Q. JU, ‘FastBERT: a Self-distilling BERT with Adaptive Inference Time’, in Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics, Stroudsburg, PA, USA: Association for Computational Linguistics, 2020, pp. 6035–6044. doi: 10.18653/v1/2020.acl-main.537.

[22] D. Britz, A. Goldie, M.-T. Luong, and Q. Le, ‘Massive Exploration of Neural Machine Translation Architectures’, in Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing, Stroudsburg, PA, USA: Association for Computational Linguistics, 2017, pp. 1442–1451. doi: 10.18653/v1/D17-1151.

[23] J. Pennington, R. Socher, and C. D. Manning, ‘GloVe: Global Vectors for Word Representation’. [Online]. Available: http://nlp.

[24] N. Van-Tu and L. Anh-Cuong, ‘Improving Question Classification by Feature Extraction and Selection’, Indian J Sci Technol, vol. 9, no. 17, May 2016, doi: 10.17485/ijst/2016/v9i17/93160.

[25] S. Chotirat and P. Meesad, ‘Part-of-Speech tagging enhancement to natural language processing for Thai wh-question classification with deep learning’, Heliyon, vol. 7, no. 10, p. e08216, Oct. 2021, doi: 10.1016/j.heliyon.2021.e08216.

[26] H. Tayyar Madabushi, M. Lee, and J. Barnden, ‘Integrating Question Classification and Deep Learning for improved Answer Selection’, in Proceedings of the 27th International Conference on Computational Linguistics, E. M. Bender, L. Derczynski, and P. Isabelle, Eds., Santa Fe, New Mexico, USA: Association for Computational Linguistics, Aug. 2018, pp. 3283–3294. [Online]. Available: https://aclanthology.org/C18-1278.

[27] J. Chung, C. Gulcehre, K. Cho, and Y. Bengio, ‘Empirical Evaluation of Gated Recurrent Neural Networks on Sequence Modeling’, Dec. 2014, [Online]. Available: http://arxiv.org/abs/1412.3555.

[28] A. Graves, ‘Long Short-Term Memory’, 2012, pp. 37–45. doi: 10.1007/978-3-642-24797-2_4.

[29] O. Sagi and L. Rokach, ‘Ensemble learning: A survey’, WIREs Data Mining and Knowledge Discovery, vol. 8, no. 4, Jul. 2018, doi: 10.1002/widm.1249.

[30] Z.-H. Zhou, Ensemble Methods Foundations and Algorithms. 2012.

[31] X. Dong, Z. Yu, W. Cao, Y. Shi, and Q. Ma, ‘A survey on ensemble learning’, Front Comput Sci, vol. 14, no. 2, pp. 241–258, Apr. 2020, doi: 10.1007/s11704-019-8208-z.

[32] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova, ‘BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding’, Oct. 2018, [Online]. Available: http://arxiv.org/abs/1810.04805.

[33] Y. Liu et al., ‘RoBERTa: A Robustly Optimized BERT Pretraining Approach’, Jul. 2019, [Online]. Available: http://arxiv.org/abs/1907.11692.

[34] V. Sanh, L. Debut, J. Chaumond, and T. Wolf, ‘DistilBERT, a distilled version of BERT: smaller, faster, cheaper and lighter’, Oct. 2019, [Online]. Available: http://arxiv.org/abs/1910.01108.

[35] S. Aburass, ‘Quantifying Overfitting: Introducing the Overfitting Index’, 2023. Accessed: Nov. 10, 2023. [Online]. Available: https://arxiv.org/abs/2308.08682.