Medical Intent Classification Using Ensemble and Deep Learning Models

Javeria Nawal[1], Ghazia Arshad[1], Muzamil Ahmed[*,1, 2,], Malik Muhammad Ali Shahid[1], Hikmat Ullah Khan[3]

[1]Department of Computer Science, Namal University, Mianwali 42210, Pakistan.

[2]Department of Computer Science, COMSATS University Islamabad, Wah Campus, Wah Cantt 47010, Pakistan.

[3]Department of Information Technology, University of Sargodha, Sargodha 40100, Pakistan.

[*]**Correspondence:** Muzamil Ahmed (email: muzamil.ahmed@namal.edu.pk)

Citation | Nawal. J, Arshad. G, Ahmed. M, Shahid. M. M. A, Khan. H. U, "Medical Intent Classification Using Ensemble and Deep Learning Models", Special Issue. pp 207-219, Oct 2024

Received | Oct 12, 2024 **Revised |** Oct 17, 2024 **Accepted |** Oct 22, 2024 **Published |** Oct 27, 2024.

Introduction: Medical chatbots are innovative solutions that leverage Natural Language Processing (NLP) and Artificial Intelligence (AI) to enhance communication efficiency between healthcare providers and patients. In the realm of conversational AI, intent classification—the task of understanding a user's intent from natural language input—is both a complex and crucial aspect of the technology. This process is vital for ensuring that chatbots can accurately interpret and respond to patient queries in a meaningful and contextually appropriate manner.

Novelty Statement: This research proposes a hybrid approach that combines transformer-based embeddings with traditional deep learning models to reduce both complexity and computational cost in medical intent classification. By integrating the strengths of advanced transformer techniques with more established models, this approach aims to improve efficiency without sacrificing performance, making it more suitable for real-world healthcare applications.

Material and Method: This study investigates the use of context-aware word embeddings, including word2vec and sentence transformers, to capture rich semantic information from medical text. To refine the unstructured data, we apply various NLP preprocessing techniques, such as text cleaning, stop word removal, and lemmatization. For classification, we utilize a combination of ensemble-based and deep learning methods, including XGBoost, Random Forest, LSTM, and Bi-LSTM. These methods are tested on real-world data from 6,662 patients, with the dataset containing 25 distinct classes.

Result and Discussion: Empirical analysis demonstrates that the Bi-LSTM model, when combined with sentence transformers, achieves an accuracy of 95.23%, outperforming state-of-the-art models reported in the relevant literature.

Concluding Remarks: This research is expected to be highly beneficial to healthcare professionals by enhancing information extraction and enabling more effective handling of patient queries.

Keywords: NLP; Intent Classification; Word Embedding; Sentence Transformers; Health Informatics; Transformer Models.

Introduction:

Intent classification combines language analysis with AI techniques to predict users' intent from natural language content [1]. This process is a critical component of natural language understanding (NLU) in dialogue management and conversational AI systems. Due to its wide-ranging applications across various domains, intent classification has become a prominent research area within the field of Natural Language Processing (NLP), which has gained significant attention with the rise of technologies like ChatGPT. NLP integrates machine learning techniques with linguistics to address real-world challenges such as sentiment analysis, email filtering, text summarization, and the development of conversational agents or chatbots [2], [3]. Most online natural language content is available in unstructured text format, making it essential to explore methods for classifying electronic medical records and extracting meaningful data through text preprocessing techniques. Medical symptom text classification, in particular, facilitates the analysis of symptoms, streamlining patient care by freeing up time for healthcare professionals. By describing their symptoms in natural language, users can receive automated responses from a trained model that helps diagnose diseases based on the symptoms provided [4].

According to a survey [5], a significant number of adults prefer using online platforms to search for healthcare solutions before consulting with a clinician. Medical symptom text classification has a wide range of valuable applications in healthcare. One key application is enhancing diagnostic support systems, where AI models interpret patient-reported symptoms and suggest potential medical issues. This assists healthcare professionals in making more accurate and error-free decisions [6]. Additionally, this technology can support telemedicine by enabling automated symptom evaluation and providing feedback to patients remotely, helping ensure timely medical intervention. Furthermore, text classification can be used to monitor public health trends by analyzing symptoms shared on social media and health forums, potentially enabling the early detection of disease outbreaks [7].

Medical symptom text classification encompasses various approaches tailored to meet different healthcare needs [8]. One common approach is binary classification, which categorizes symptoms into two groups, typically indicating whether a specific condition is present or absent. Multi-class classification, on the other hand, sorts symptoms into multiple predefined categories, such as distinguishing between the flu, the common cold, or allergies. To achieve multi-class text classification for medical symptoms [9], [10], [11], [12], our proposed method combines both machine learning and deep learning (DL) models. We employ diverse encoding techniques such as TF-IDF, word2vec, and sentence transformers to capture the semantic meaning of medical texts effectively [13], [14].

Novelty Statement:

This study presents a hybrid approach that combines transformer-based models with traditional classification techniques to learn rich semantic representations while reducing computational costs. By leveraging high-level semantic information, our approach aims to enhance model performance. The results demonstrate that, through this integration, classification tasks can be both more efficient and more effective, achieving improved accuracy with less computational expense.

Objective:

The primary objectives of this study are as follows:

- To apply various NLP techniques—such as text cleaning, punctuation removal, lemmatization, and stop word elimination—to preprocess raw text data for analysis.
- To utilize feature-engineering methods for extracting context-aware representations that improve the model's ability to understand medical text.

- To implement ensemble-based and deep learning techniques, specifically XGBoost, Random Forest, LSTM, and BiLSTM, for the task of medical intent classification.
- To perform empirical analysis to assess the performance of the proposed models on a benchmark dataset, and to compare the results with state-of-the-art models.

Related Work:

This section reviews recent studies relevant to intent classification using machine learning and deep learning methods. In a study [15], the author proposed a biomedical text classification model that incorporates augmented word representation and distribution, as well as relational aspects. The approach suggests combining semantic relationships extracted from a large corpus with co-occurrence and pointwise mutual information techniques to learn enriched embeddings for biomedical text classification. The GloVe embeddings generated were then employed in deep learning models for improved classification performance.

In a subsequent study [16], the authors addressed challenges in classifying clinical text documents into specific medical specialties. They presented a machine learning approach consisting of text preprocessing, Word2Vec embeddings, and several classifiers. However, they noted limitations due to the dataset's small size and the limited number of categories, despite achieving an accuracy of 82% with a method that combined k-NN with Word2Vec.

Furthermore, a study by the authors of [17] applied capsule networks for classifying text from 44 medical subfields. The combination of capsule networks with LSTM (Long Short-Term Memory) networks resulted in an F1 score of 73.51%. While this was a notable achievement, the authors highlighted that the dataset may have been insufficient, which could impact the study’s generalizability.

In another study [18], conventional classification approaches were found to be ineffective for categorizing medical articles into broad categories like diabetes or cancer. To address this, the authors proposed a document-level medical article classification method using two types of features: CBFs (Content-Based Features), which focus on the writer’s stylistic choices and text difficulty, and DSBs (Domain-Specific Blocks), which rely on topic modeling (LDA) to filter keywords and assign articles to medical categories.

The challenge of obesity status extraction from unstructured clinical text data in Electronic Health Records (EHR) was explored by Hosseini et al. in [20]. They proposed an integrated model combining rule-based features and a knowledge-assisted deep learning (DL) model for classifying clinical texts related to obesity. The rule-based features involved predefined phrases indicative of obesity (e.g., "BMI 35"), while the DL model used a CNN (Convolutional Neural Network) trained on the text data with the aid of medical codes (UMLS CUIs). The results demonstrated high diagnostic accuracy, with an improved F1 score for obesity classification compared to existing methods. Additionally, the rule-based identification of trigger phrases proved effective in many cases.

In the study [21], the authors addressed the issue in China where patients often make incorrect initial medical specialty selections due to a lack of medical knowledge. To tackle this, they developed a Hybrid Model (HyM) that takes Chinese text descriptions of patient symptoms and assigns the most appropriate medical specialty from a list of eight specialties. The HyM combined features from four techniques: LSTM, Text-CNN, BERT, and TF-IDF, and was trained on over 40,000 offline hospital patient symptom descriptions. The model achieved an accuracy of 93.5% and an F-score of 90%.

The study in [22] discussed the challenges of trend analysis and identifying potential safety risks related to clinical trials, particularly because adverse events (AEs) are often documented in unstructured textual form. The authors proposed a method for classifying AEs into predefined subcategories using SVM (Support Vector Machines), Word2Vec, and TF-IDF. This method was applied to a dataset of 687 protocols, achieving an accuracy of 84%.

Lastly, a study [23] applied CNN and LSTM with TF-IDF on two datasets: THUCNews (long text) and Taobao reviews (short text). This approach aimed to improve text classification performance by leveraging both deep learning models and traditional text feature extraction techniques.

Material and Methods:

Figure 1 illustrates the proposed research framework for classifying medical users' intents from textual content. The first step in the framework is text preprocessing, which includes cleaning the text, removing punctuation, and eliminating stop words to prepare the data for model training. The second step involves generating word embeddings, such as Word2Vec, TF-IDF, and sentence transformers, to capture context-aware encodings of the text. Finally, ensemble-based and deep learning models, including XGBoost, Random Forest, LSTM, and BiLSTM, are applied in a recursive manner for classification. A detailed description of each step is provided in the following sections.

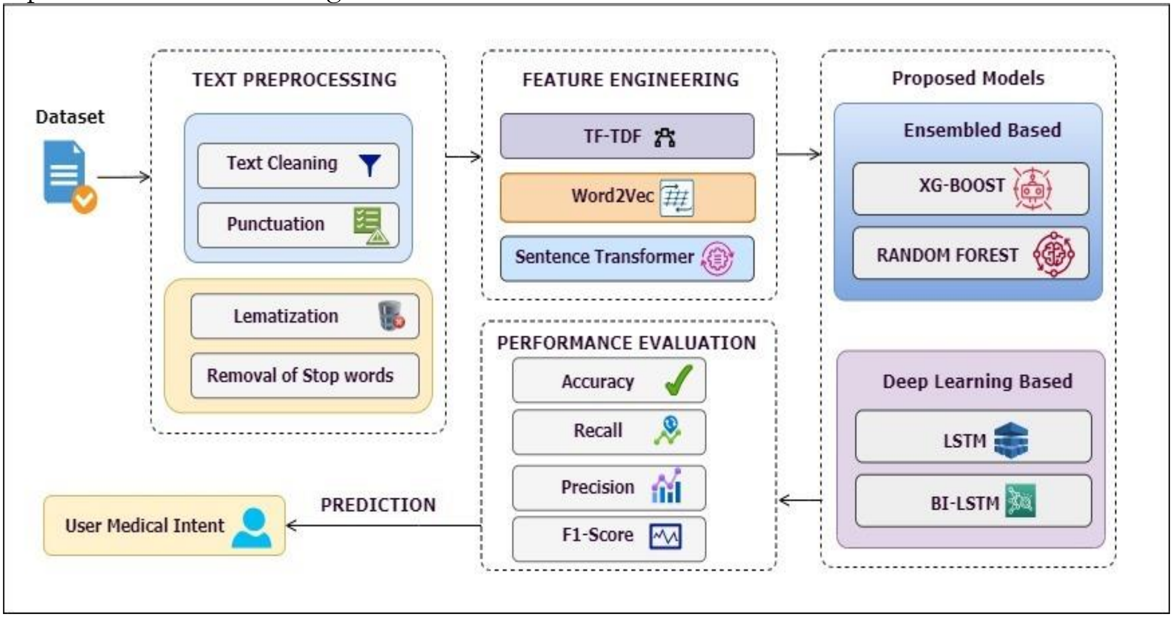

Figure 1. Flow diagram of the proposed study for medical intent classification

Text Preprocessing:

The text preprocessing steps involve several stages, including tokenization, text cleaning, stop word removal, and lemmatization. A detailed description of each step is provided below:

Text Cleaning:

Text cleaning is a crucial step in preparing written data, aiming to enhance its quality and consistency [24]. This process involves several tasks designed to eliminate irrelevant or extraneous elements. HTML tags, special characters, and unnecessary punctuation are removed to streamline the text and focus on meaningful content. Additionally, spelling corrections may be applied to address typographical errors, ensuring the accuracy of subsequent analyses. Proper handling of punctuation is also emphasized, as it plays a vital role in providing structure and clarity to the text during processing.

Stop Words Removal:

Stop words removal is the process of eliminating common, non-informative words from the text [24]. Words such as "the," "is," and "and" are considered stop words because they do not contribute significant meaning for pattern recognition in machine learning models. By removing these words, the focus shifts to more relevant terms, such as "fever" or "headache," which carry important information for classification tasks. This helps improve the efficiency and accuracy of the model by reducing noise and emphasizing key features.

Lemmatization:

This step involves reducing words to their base or dictionary form through lemmatization, ensuring consistency in the representation of medical symptoms. By applying lemmatization, we standardize the language used to describe symptoms, which is crucial for building an accurate classification model. This preprocessing technique enhances the model's performance and helps eliminate redundancy in the data [25], allowing it to focus on meaningful features for better classification accuracy.

Feature Engineering:

Feature engineering is a critical step in preparing data for machine learning models, as it converts raw textual data into fixed numerical representations. In this study, we employ three distinct encoding methods, as outlined below:

TF-IDF:

Term Frequency-Inverse Document Frequency (TF-IDF) is a fundamental technique in Natural Language Processing (NLP) that helps identify the most informative terms within a document corpus. It consists of two components: Term Frequency (TF), which measures how often a specific term (t) appears within a given document (d). The Term Frequency is calculated as shown in equation (1).

$$TF\ (t,d)\ =\ \frac{f\ (t,d)}{total\_words\_in\_document(d)} \qquad (1)$$

In this formula, f (t,d) represents the frequency of term t in document \(d \), while $total\_words\_in\_document(d)$ refers to the total number of words in document d. The second factor, Inverse Document Frequency (IDF), measures the importance of a term across the entire document collection (corpus). Terms that appear frequently in many documents are considered less informative for a specific document. IDF is calculated using the following equation (2).

$$IDF(t)\ =\ log\ (\frac{N}{doc\_freq(t)}) \qquad (2)$$

Here, N represents the total number of documents in the corpus, and $doc\_freq(t)$ denotes the number of documents that contain the term t. The logarithm (log) is applied to emphasize terms that appear in fewer documents, thus giving greater importance to rare or distinctive terms.

$$TF - IDF(t) = TF(t)\ +\ IDF(t) \qquad (3)$$

Subsequently, by multiplying TF and IDF, we get a final $TF - IDF$ score for each term within a document.

Word2Vec:

Word2Vec, short for "word to vector," bridges the gap between human and machine language by converting words into numerical representations known as word embeddings [26]. Word2Vec leverages the semantic relationships between words, placing them in a high-dimensional vector space where words with similar meanings are positioned closer together. The Skip-gram model, a key component of Word2Vec, takes a center word as input and predicts the surrounding contextual words within a specific window. The Word2Vec encoding of a textual document using the Skip-gram model is computed using the following equation (4).

$$\mathrm{p}(\mathrm{w_O}|\mathrm{w_I}) = \frac{\exp(\mathrm{v}'_{\mathrm{w_O}}\ \tau\ \mathrm{v}_{\mathrm{w_I}})}{\sum_{\mathrm{w}=1}^{\mathrm{W}} \exp(\mathrm{v}'_{\mathrm{w_O}}\ \tau\ \mathrm{v}_{\mathrm{w_I}})} \qquad (4)$$

Where w_o denotes word occurrence and Exp (w) is the exponent of the vector representation of the center word (w), Σ that represents summation over all possible context words within the window size around the center word, w=1 indicates that the summation starts from the first context word, Wc represents the weight matrix for the context words, T is the symbol which denotes the transpose operation and c represents a context word vector. The factor $p(w|c)$ represents the probability of the center word (w) given a context word (c).

Sentence Transformer:

Sentence transformers convert input sentences into high-dimensional vectors that capture their semantic meaning, enabling a deeper understanding of patient queries [24]. The similarity between two sentence embeddings, $e1$ and e2, is typically measured using a similarity metric such as cosine similarity.

$$Cosine\ Similarity(e1, e2) = \frac{e1 \cdot e2}{\|e1\|\|e2\|} \qquad (5)$$

Where $e1$ and e2 represent the vector embeddings of the input sentences, and ‖e1‖ and ‖e2‖ denote their respective magnitudes. This method plays a critical role in NLP tasks such as semantic textual similarity, information retrieval, and duplicate detection, where understanding the semantic relationships between text snippets or sentences is vital for precise analysis and informed decision-making.

Machine Learning Models

Machine Learning, a subfield of AI, enables machines to learn from existing data related to a specific problem. Ensemble-based methods in machine learning combine multiple models to improve predictive performance. Rather than relying on a single model for predictions, ensemble methods leverage several models and aggregate their outputs, enhancing the accuracy and robustness of the predictions.

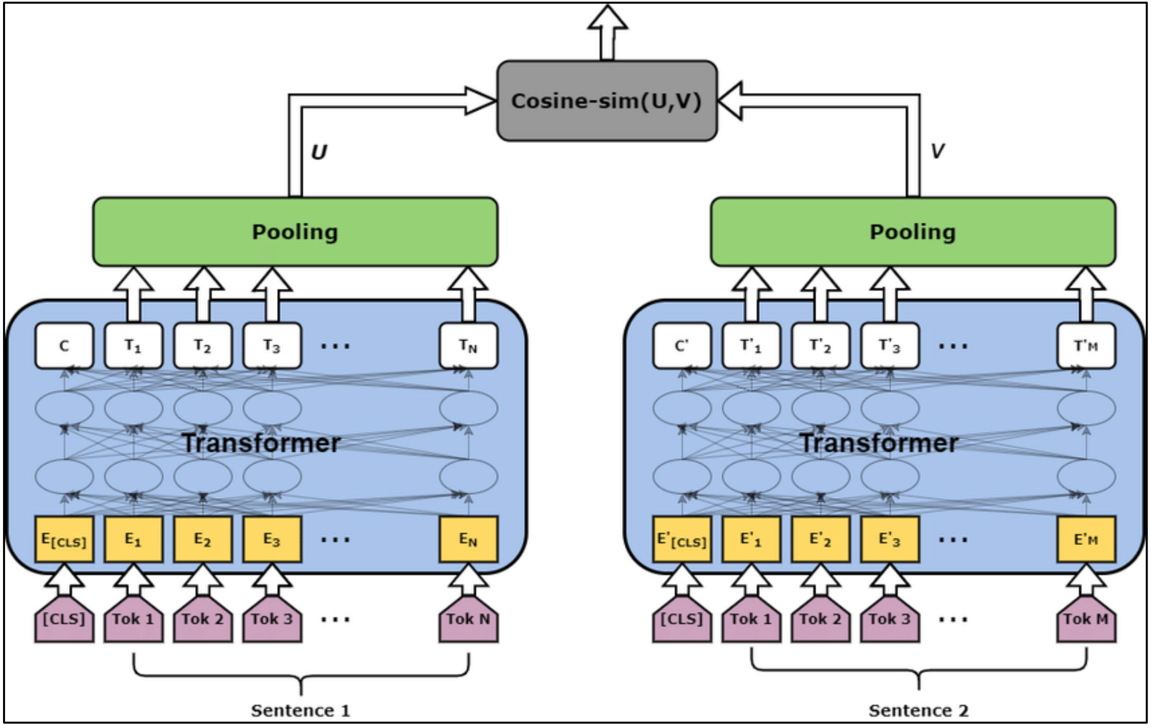

Figure 2. Architecture of Sentence Transformer for Medical Intent Classification

In this study, we employed two distinct ensemble-based machine learning models to predict patient intent from natural language text: 1) Random Forest and 2) XG-Boost. The detailed architecture and workings of each model are outlined below:

Random Forest:

Random Forest is a powerful ensemble learning algorithm that has gained significant popularity in recent years. By combining multiple decision trees, Random Forest delivers robust and accurate predictions. Each tree is trained on a random subset of the data and features, which helps reduce overfitting and enhances the model's ability to generalize to unseen data. The process of constructing a forest of decision trees involves two key steps.

The first step is Bootstrap Aggregation (Bagging), where Random Forest generates a collection of decision trees by randomly sampling n instances (with replacement) from the original dataset of size N. The second step is subset feature selection, where, at each node of a tree, a random subset of m features (from the total M features) is chosen as candidates for splitting. For a new data point (x), each tree in the forest votes for a specific class label, and the final prediction is determined by aggregating the votes across all trees. The final prediction is given by the following equation (6):

$$\hat{y} = argmax\_c \sum (T\_i(x) == \mathrm{c}) \qquad (6)$$

Where ŷ denotes the Predicted class label, c denotes the Class label, $T\_i(x)$ denotes the Prediction of the i-th tree in the forest for data point x and $argmax\_c$ denotes operator that finds the argument (class label) with the maximum value.

Extreme Gradient Boosting (X-G-Boost):

XGBoost has become a prominent ensemble learning technique due to its ability to build robust and highly accurate machine learning models. The XGBoost classification process occurs in two key steps:

Sequential Learning:

In this phase, XGBoost gradually builds its knowledge base, much like a student preparing for an exam. Instead of trying to learn everything at once, it adds new decision trees sequentially. Each new tree is designed to correct the errors made by the previous ones, thus improving the model with every iteration.

Gradient Loss Minimization:

This step focuses on minimizing the loss function, which quantifies prediction errors. Mathematically, let F(x) represent the current model's prediction for a given data point $x, y_$i be the true label for the data point i, and $L(y\_i, F(x\_i))$ be the loss function. The goal of XGBoost is to minimize the following objective, represented by equation (7).

$$Obj = \sum\_i L(y\_i\ , F(x\_i)) + \Omega(F) \qquad (7)$$

Where, $\Omega(F)$ represents a regularization term that penalizes model complexity, preventing overfitting.

Deep Learning (DL) Models:

The advent of deep learning (DL) models has brought about a significant transformation in the field of artificial intelligence, particularly in understanding the semantics and comprehension of natural language text. Inspired by the human brain's remarkable ability to learn from vast amounts of data, DL models excel at uncovering hidden patterns and relationships within complex datasets. In this study, we employed two powerful DL models: Long Short-Term Memory (LSTM) networks and Bidirectional LSTM (Bi-LSTM) networks, to effectively capture the intricate nuances of language and enhance model performance.

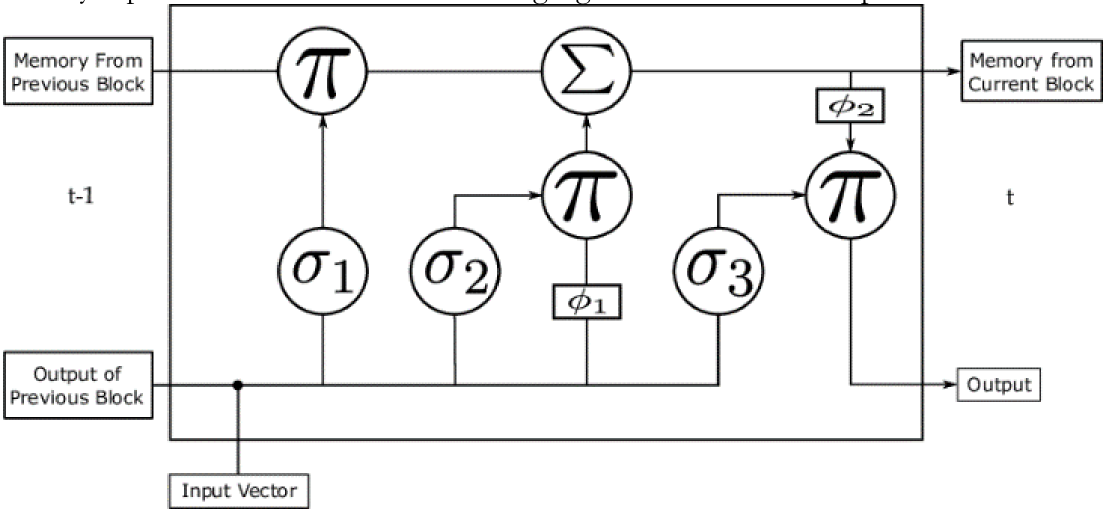

Figure 3. Model Architecture of Deep Learning based Long Short-Term Memory

Long Short-Term Memory (LSTM):

LSTM (Long Short-Term Memory), a specialized type of Recurrent Neural Network (RNN), is designed to learn long-term dependencies in sequential data. A typical LSTM unit consists of a repeating module with four neural network layers interacting in a specific manner. The module is controlled by three gate activation functions (ф1, ф2, and ф3) along with two output activation functions (ф1 and ф2). The symbol π represents element-wise multiplication, while Σ denotes element-wise addition. A key component of LSTMs is the cell state, which acts as a "memory line" running from the memory of the previous block (S_(t-1)) to the current block (S_t). This cell state allows information to flow directly through the network. The LSTM network can decide how much of the previous information should be passed along, and this is controlled by the first layer (σ1). The operation of this layer is illustrated in Figure 3.

$$c\,ft \;=\; \sigma1(Wc\,f \;\cdot\; [Ot-1, xt] +\; bc\,f) \qquad (8)$$

$$It \;=\; \sigma2(WI \;\cdot\; [Ot-1, xt] +\; bI) \qquad (9)$$

$$\tilde{S}t \;= tanh\,(WS \;\cdot\; [Ot-1, xt] +\; bS\,) \qquad (10)$$

$$S\,t \;=\; c\,ft \,\times\, S\,t1 \;+\; It \,\times\, \tilde{S}t-1 \qquad (11)$$

A sigmoid layer (σ2) determines the values to be updated, while a tanh layer (ф1) generates a vector of new candidate values (S˜t), as shown in equation (4). The combination of these values is then added to the cell state. Finally, the cell state is updated using the following equation (5).

Bidirectional Long Short-Term Memory (Bi-LSTM):

The Bi-LSTM network is capable of capturing both forward and backward information from the text, making it highly effective for extracting contextual features from sentences.

$$hi^{\rightarrow} = LSTMa^{i-1}, xi \qquad (12)$$

$$hi^{\leftarrow} = LSTMa^{i+1}, xi \qquad (13)$$

$$hi = hi^{\rightarrow}; hi^{\leftarrow} \qquad (14)$$

$$h_c = h_1, h_2, h_3, \cdots, h_m \qquad (15)$$

Where $a < i >$represents the hidden layer state of the current memory cell. $hi^{\rightarrow}$and $hi^{\leftarrow}$ represent the hidden state of the forward and back memory network at the $i-th$ character position, respectively. hi represents a combination of hidden states in both directions.

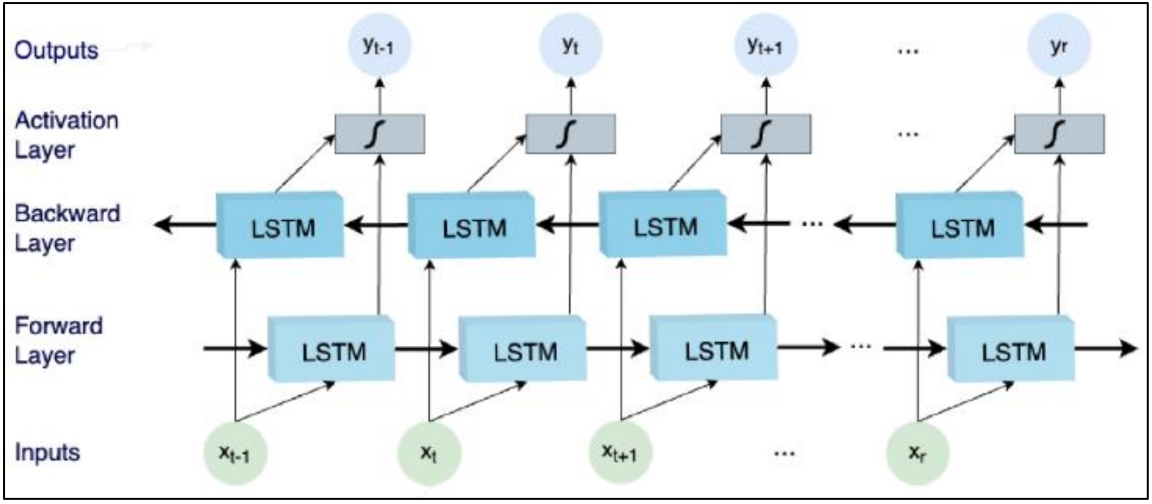

Figure 4. Model Architecture of Deep Learning based Long Short-Term Memory

Result and Discussion:

This section presents the results of the proposed framework for the intent classification task. As shown in Table 1, the XG-Boost model outperforms the Random Forest model across various encoding methods. When using Sentence Transformers, XG-Boost achieves the highest accuracy of 89.00%, with a precision of 87.22%, recall of 88.93%, and an F1-Score of 88.07%.

In comparison, the Random Forest model with Sentence Transformers yields an accuracy of 85.92%, precision of 86.34%, recall of 87.11%, and an F1-Score of 86.72%. Furthermore, the XG-Boost model with TF-IDF encoding also demonstrates superior performance, attaining an accuracy of 87.23% and an F1-Score of 85.85%.

Table 1. Results of Ensemble based Methods with diverse Word Embedding over Medical Speech Transcription and Intent Dataset

Model	**Encoding Method**	**Accuracy %**	**Precision %**	**Recall %**	**F1-Score %**
Random Forest	TF-IDF	83.45	81.24	84.76	83.00
	Word2Vec	81.67	80.13	82.25	81.18
	Sentence Transformers	85.92	86.34	87.11	86.72
X-G Boost	TF-IDF	87.23	83.67	88.12	85.85
	Word2Vec	78.89	77.15	79.44	78.28
	Sentence Transformers	**89.00**	**87.22**	**88.93**	**88.07**

According to Table 2, with TF-IDF encoding, the highest accuracy of 93.23% is achieved after 25 epochs. For Word2Vec embeddings, the LSTM model performs optimally at 20 epochs, achieving an accuracy of 95.23%, precision of 93.67%, recall of 95.12%, and an F1-Score of 94.85%. Similarly, with Sentence Transformers, the model reaches its highest performance at 20 epochs, with an accuracy of 95.92%, precision of 91.34%, recall of 96.11%, and an F1-Score of 93.72%.

Table 2. Results of Deep Learning based LSTM Model with diverse embedding over Medical Speech Transcription and Intent Dataset

Model	**Encoding Method**	**Epochs #**	**Accuracy**	**Precision**	**Recall**	**F1-Score**
LSTM	TF-IDF	5	88.45	86.24	90.76	88
		10	89.45	87.24	91.76	89
		15	88.67	86.13	90.25	88.18
		20	89.92	91.34	92.11	91.72
		25	93.23	88.67	93.12	90.85
		30	84.89	83.15	85.44	84.28
	Word2Vec	5	87.45	88.24	91.76	89
		10	88.67	87.13	91.25	89.18
		15	90.92	91.34	92.11	91.72
		20	95.23	93.67	95.12	94.85
		25	84.89	83.15	86.44	85.28
		30	95	93.22	94.93	94.07
	Sentence Transformers	5	93.45	88.24	92.76	90
		10	89.45	90.24	92.76	91
		15	94.67	91.13	94.25	92.18
		20	95.92	91.34	96.11	93.72
		25	93.23	93.67	94.12	94.85
		30	84.89	86.15	88.44	87.28

Table 3 presents the results of the Bi-LSTM model for medical intent classification using different embedding methods and epoch counts. For TF-IDF encoding, the highest accuracy of 95.23% is achieved at 25 epochs, along with precision of 91.67%, recall of 96.12%, and an F1-Score of 93.85%. However, as training progresses to 30 epochs, the performance significantly drops to 87.89% accuracy, indicating potential overfitting. For Word2Vec encoding, the model performs best at 20 epochs, achieving an accuracy of 97.23%, precision of 95.67%, recall of 97.12%, and an F1-Score of 96.85%.

Table 3. Results of Deep Learning based Bi-LSTM Model with diverse embedding over Medical Speech Transcription and Intent Dataset

Model	Encoding Method	Epochs #	Accuracy	Precision	Recall	F1-Score
Bi-LSTM	TF-IDF	5	90.45	88.24	93.76	90
		10	91.45	89.24	94.76	91
		15	91.67	88.13	93.25	90.18
		20	92.92	94.34	95.11	94.72
		25	95.23	91.67	96.12	93.85
		30	87.89	86.15	88.44	87.28
	Word2Vec	5	89.45	90.24	93.76	91
		10	90.67	89.13	93.25	91.18
		15	92.92	93.34	94.11	93.72
		20	94.23	95.67	94.12	96.85
		25	86.89	85.15	88.44	87.28
		30	94.00	95.22	96.93	96.07
	Sentence Transformers	5	95.45	90.24	94.76	92
		10	91.45	92.24	94.76	93
		15	96.67	93.13	96.25	94.18
		20	97.92	93.34	98.11	95.72
		25	95.23	95.67	96.12	96.85
		30	86.89	88.15	90.44	89.28

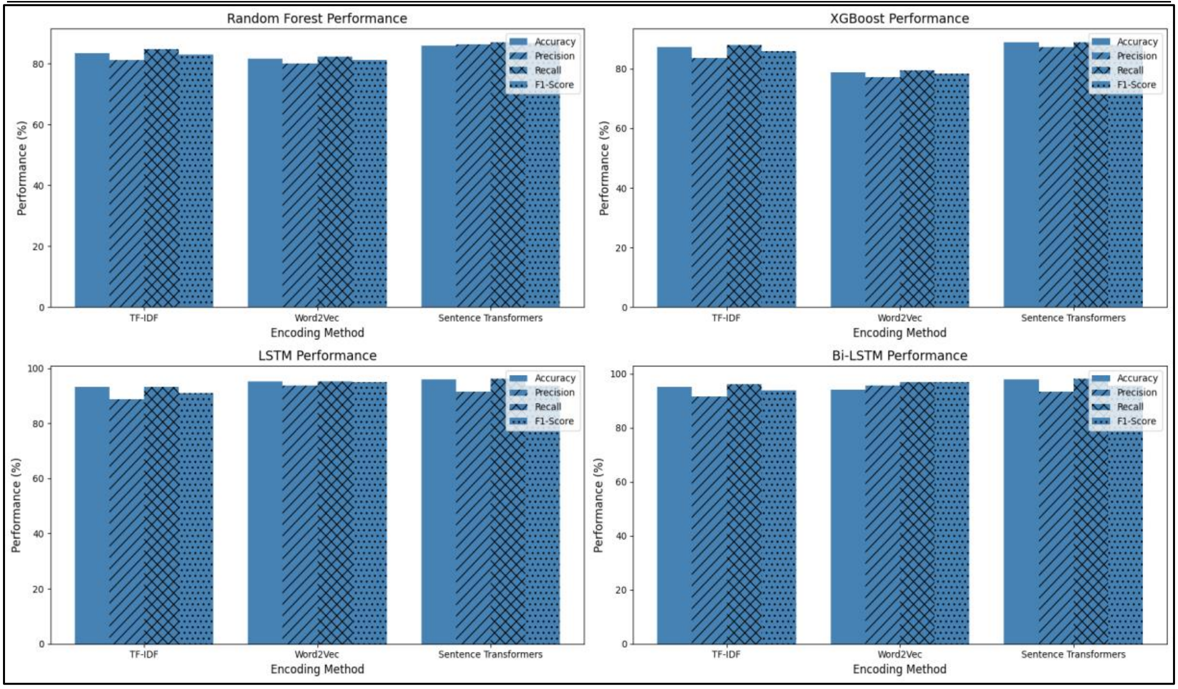

Figure 5. Performance of the proposed model with diverse embedding over different epochs

According to Figure 5, the Sentence Transformer combined with the Bi-LSTM model demonstrates the best performance at 20 epochs, achieving an accuracy of 97.92%, precision of 93.34%, recall of 98.11%, and an F1-Score of 95.72%. Table 4 provides a comparison of the proposed models with state-of-the-art methods.

Table 4. Comparison of Proposed Model with State-of-art Methods

Ref	Year	Model/Method	Performance
[17]	2022	Capsule network + TF-IDF	73.25 % Acc
[16]	2023	K-Nearest Neighbor + Word2Vec	82.51 % Acc
[22]	2021	Support Vector Machine + TF-IDF	79.58 % Acc
[15]	2023	Recurrent Neural Network + GloVe	83.20 % Acc
[19]	2019	CBFs and topic modeling (LDA)	84.15 % F1
[20]	2019	CNN + Rule-based Features	80.58 % Acc
[21]	2023	Hybrid Model (HyM) CNN+ TF-IDF	89.50 % Acc
Proposed		Bi-LSTM + Sentence Transformers	95.23 % Acc

Conclusion:

In this study, we conducted an empirical analysis of medical intent classification using both ensemble-based and deep learning (DL) approaches. To assess the performance and effectiveness of the proposed models, we utilized a publicly available dataset that includes medical speech, transcription, and intent data, encompassing 25 to 30 different classes and approximately 6,662 patient records. The analysis incorporated two machine learning models, XG-Boost and Random Forest, as well as two deep learning models, LSTM and Bi-LSTM. The results indicate that the XG-Boost model outperforms Random Forest, especially when paired with Sentence Transformers. However, among the deep learning models, Bi-LSTM achieved the highest performance with Sentence Transformers at 20 epochs, reaching an accuracy of 97.92% and an F1-Score of 95.72%. These models have a wide range of applications, including medical conversational bots, clinical decision support systems, and telehealth services. In the future, we plan to develop our own dataset, enhance our feature engineering techniques, and explore the use of advanced transformer-based architectures for more semantic-aware medical intent classification.

References:

[1] C. Zeng, S. Li, Q. Li, J. Hu, and J. Hu, “A Survey on Machine Reading Comprehension—Tasks, Evaluation Metrics and Benchmark Datasets,” Applied Sciences, vol. 10, no. 21, 2020, doi: 10.3390/app10217640.

[2] J. He, L. Peng, Y. Zhang, B. Sun, R. Xiao, and Y. Xiao, “Machine Reading Comprehension with Rich Knowledge,” Intern J Pattern Recognit Artif Intell, vol. 36, no. 05, p. 2251004, Apr. 2022, doi: 10.1142/S0218001422510041.

[3] X. Xu, T. Tohti, and A. Hamdulla, “A Survey of Machine Reading Comprehension Methods,” in 2022 International Conference on Asian Language Processing (IALP), 2022, pp. 312–317. doi: 10.1109/IALP57159.2022.9961260.

[4] R. G. Reddy, M. A. Sultan, E. S. Kayi, R. Zhang, V. Castelli, and A. Sil, “Answer Span Correction in Machine Reading Comprehension,” 2020, arXiv. doi: 10.48550/ARXIV.2011.03435.

[5] R. Baradaran, R. Ghiasi, and H. Amirkhani, “A Survey on Machine Reading Comprehension Systems,” Nat Lang Eng, vol. 28, no. 6, pp. 683–732, 2022, doi: DOI: 10.1017/S1351324921000395.

[6] J. Liu, Y. Chen, and J. Xu, “Document-level event argument linking as machine reading comprehension,” Neurocomputing, vol. 488, pp. 414–423, 2022, doi: https://doi.org/10.1016/j.neucom.2022.03.016.

[7] F. Li, Y. Shan, X. Mao, X. Ren, X. Liu, and S. Zhang, “Multi-task joint training model for machine reading comprehension,” Neurocomputing, vol. 488, pp. 66–77, 2022, doi: https://doi.org/10.1016/j.neucom.2022.02.082.

[8] A. Mohammadi, R. Ramezani, and A. Baraani, “A Comprehensive Survey on Multi-hop Machine Reading Comprehension Approaches,” 2022, arXiv. doi: 10.48550/ARXIV.2212.04072.

[9] J. Liu, L. Cui, H. Liu, D. Huang, Y. Wang, and Y. Zhang, “LogiQA: A Challenge Dataset for Machine Reading Comprehension with Logical Reasoning,” 2020, arXiv. doi: 10.48550/ARXIV.2007.08124.

[10] H. T.-T. Le, V.-D. Ho, D.-V. Nguyen, and N. L.-T. Nguyen, “Integrating Semantic Information into Sketchy Reading Module of Retro-Reader for Vietnamese Machine Reading Comprehension,” in 2022 9th NAFOSTED Conference on Information and Computer Science (NICS), 2022, pp. 53–58. doi: 10.1109/NICS56915.2022.10013390.

[11] S. Back, S. C. Chinthakindi, A. Kedia, H. Lee, and J. Choo, “NeurQuRI: Neural Question Requirement Inspector for Answerability Prediction in Machine Reading Comprehension,” in International Conference on Learning Representations, 2020. [Online]. Available: https://openreview.net/forum?id=ryxgsCVYPr

[12] S. Yuan et al., “Large-Scale Multi-granular Concept Extraction Based on Machine Reading Comprehension,” in The Semantic Web ISWC 2021, Springer International Publishing, 2021, pp. 93–110. doi: 10.1007/978-3-030-88361-4_6.

[13] F. K. Alarfaj, I. Malik, H. U. Khan, N. Almusallam, M. Ramzan, and M. Ahmed, “Credit Card Fraud Detection Using State-of-the-Art Machine Learning and Deep Learning Algorithms,” IEEE Access, vol. 10, pp. 39700–39715, 2022, doi: 10.1109/ACCESS.2022.3166891.

[14] M. Ahmed, H. Khan, T. Iqbal, F. Khaled Alarfaj, A. Alomair, and N. Almusallam, “On solving textual ambiguities and semantic vagueness in MRC based question answering using generative pre-trained transformers,” PeerJ Comput Sci, vol. 9, p. e1422, Jul. 2023, doi: 10.7717/peerj-cs.1422.

[15] Md. A. Parwez, Mohd. Fazil, M. Arif, M. T. Nafis, and Md. R. Auwul, “Biomedical Text Classification Using Augmented Word Representation Based on Distributional and Relational Contexts,” Comput Intell Neurosci, vol. 2023, no. 1, p. 2989791, 2023, doi: https://doi.org/10.1155/2023/2989791.

[16] L. Almazaydeh, M. Abuhelaleh, A. Al Tawil, and K. Elleithy, “Clinical Text Classification with Word Representation Features and Machine Learning Algorithms.,” International Journal of Online & Biomedical Engineering, vol. 19, no. 4, 2023.

[17] Q. Zhang, Q. Yuan, P. Lv, M. Zhang, and L. Lv, “Research on Medical Text Classification Based on Improved Capsule Network,” Electronics (Basel), vol. 11, no. 14, 2022, doi: 10.3390/electronics11142229.

[18] R. López, J. Tejada, and M. Alexandrov, “MEDICAL TEXTS CLASSIFICATION BASED ON KEYWORDS USING SEMANTIC INFORMATION,” Transactions on Business and Engineering Intelligent Applications, p. 64, 2014.

[19] A. Al-Doulat, I. Obaidat, and M. Lee, “Unstructured Medical Text Classification using Linguistic Analysis: A Supervised Deep Learning Approach,” in 2019 IEEE/ACS 16th International Conference on Computer Systems and Applications (AICCSA), 2019, pp. 1–7. doi: 10.1109/AICCSA47632.2019.9035282.

[20] L. Yao, C. Mao, and Y. Luo, “Clinical text classification with rule-based features and knowledge-guided convolutional neural networks,” BMC Med Inform Decis Mak, vol. 19, no. 3, p. 71, 2019, doi: 10.1186/s12911-019-0781-4.

[21] C. Mao, Q. Zhu, R. Chen, and W. Su, “Automatic medical specialty classification based on patients’ description of their symptoms,” BMC Med Inform Decis Mak, vol. 23, no. 1, p. 15, 2023, doi: 10.1186/s12911-023-02105-7.

[22] E. Richard and B. Reddy, "Text Classification for Clinical Trial Operations: Evaluation and Comparison of Natural Language Processing Techniques," Ther Innov Regul Sci, vol. 55, no. 2, pp. 447–453, 2021, doi: 10.1007/s43441-020-00236-x.

[23] H. Zhou, "Research of Text Classification Based on TF-IDF and CNN-LSTM," J Phys Conf Ser, vol. 2171, no. 1, p. 12021, Jan. 2022, doi: 10.1088/1742-6596/2171/1/012021.

[24] M. Ahmed, H. U. Khan, M. A. Khan, U. Tariq, and S. Kadry, "Context-Aware Answer Selection in Community Question Answering Exploiting Spatial Temporal Bidirectional Long Short-Term Memory," ACM Trans. Asian Low-Resour. Lang. Inf. Process., Jun. 2023, doi: 10.1145/3603398.

[25] M. Ahmed, H. U. Khan, S. Iqbal, and Q. Althebyan, "Automated Question Answering based on Improved TF-IDF and Cosine Similarity," in 2022 Ninth International Conference on Social Networks Analysis, Management and Security (SNAMS), 2022, pp. 1–6. doi: 10.1109/SNAMS58071.2022.10062839.

[26] S. Iqbal, R. Khan, H. U. Khan, F. K. Alarfaj, A. M. Alomair, and M. Ahmed, "Association Rule Analysis-Based Identification of Influential Users in the Social Media," 2022.