Research Square

Preprints are preliminary reports that have not undergone peer review.
They should not be considered conclusive, used to inform clinical practice,
or referenced by the media as validated information.

Enhancing Patient Experience Using Machine Learning and Deep Learning Based Sentiment Analysis

Jitendra Tembhurne
Indian Institute of Information Technology
Aishwarya Ukey
Indian Institute of Information Technology
Umakant Mandawkar
umakantmandawkar.sun@gmail.com

Symbiosis International University
Harsh Goud
Indian Institute of Information Technology
Sharvari Balapurkar
SVKM's Institute of Technology

Research Article

Keywords: Topic Modeling, LDA, Deep Learning, Patient Experience, Sentiment Analysis, Ensemble Techniques, Embedding Techniques, Healthcare Quality Evaluation

Posted Date: September 3rd, 2026

DOI: https://doi.org/10.21203/rs.3.rs-10524614/v1

License: This work is licensed under a Creative Commons Attribution 4.0 International License.
Read Full License

Additional Declarations: No competing interests reported.

Enhancing Patient Experience Using Machine Learning and Deep Learning Based Sentiment Analysis

Jitendra Tembhurne[1], Aishwarya Ukey[1], Umakant Mandawkar[2], Harsh Goud[3], Sharvari Balapurkar[4]

[1]Department of Computer Science and Engineering, Indian Institute of Information Technology, Nagpur,India
[2] Department of CSE, Symbiosis Institute of Technology, Nagpur Campus, Symbiosis International (Deemed University), Pune, India
[3]Department of Electronics and Communication Engineering, Indian Institute of Information Technology, Nagpur
[4]Department of Computer Engineering, SVKM's Institute of Technology, Dhule, Maharashtra, India

Corresponding Author:umakantmandawkar.sun@gmail.com

Abstract: An improvement in patient satisfaction and health care services depends on the full understanding of the sentiments of patients. From the analysis of the sentiments of patients, healthcare professionals will be able to find out how services can be improved, what are the recurring problems, and how services can be tailored to fit the needs of patients. This paper will examine how sentiment analysis has been utilized for patient experience and propose a new approach that makes use of modern techniques and models to derive meaningful insights about sentiment from textual data obtained from 21 hospitals. The new approach will employ such advanced algorithms as RNN, LSTM, GRU, BiLSTM, Random Forest, Decision Tree, XGBoost, and BERT. They will be used in order to enhance the overall performance of the model at various levels of sentiment and understand semantic dependencies in textual data. Different types of embeddings will be employed, namely word2vec, GloVe, and TF-IDF. In addition, a topic modeling by using LDA will be implemented in the process of obtaining meaningful information about sentiment categories in the analyzed texts. As part of the process, five topics will be selected for every sentiment category: positive, negative, and neutral.

Keywords: *Topic Modeling, LDA, Deep Learning, Patient Experience, Sentiment Analysis, Ensemble Techniques, Embedding Techniques, Healthcare Quality Evaluation.*

1. Introduction

The healthcare domain continues to evolve owing to developments in medicine, technology, and a growing emphasis on patient care. There are various measures employed to gauge the quality of healthcare services provided, but patient experience emerges as one of the most important aspects of care quality because it determines how satisfied patients are with their healthcare experience. This also affects the level of their compliance and eventually influences the outcome of the treatment process. Therefore, patient experience analysis has become an essential component of modern healthcare. Nowadays, patients have access to platforms to report their experiences via wikis, blogs, social networks, and other online forums [1, 2]. Nevertheless, such patient reports are not captured systematically due to their unstructured form.

In the recent past, there have been increasing significance accorded to the use of sentiment analysis in measuring the experience of patients. Sentiment analysis is among the natural language processing (NLP) methods, uses algorithms for identifying, extracting, and measuring subjective content in the form of text-based material. Most sentiment analysis studies conducted so far have focused on

collecting data from online sources such as websites and social networking sites like Facebook and Twitter. The information thus collected could be either positive, negative, or neutral sentiments related to specific areas of interest. Using sentiment analysis on patient data can help health care providers gain valuable insight about their patients' sentiments.

The current research will look at the implementation of machine learning and deep learning techniques for analyzing sentiments expressed by patients. In light of the fact that the amount, dispersion, and diversity of the data may prove quite large, extracting useful information out of them would become a difficult task [7]. This research aims at compiling a complete data set comprising patient reviews collected from different hospitals in several Indian cities. These reviews have been sorted into three sentiment categories, namely positive, negative, and neutral. Some of the models being tested include Recurrent Neural Network (RNN), LSTM, GRU, Bidirectional LSTM, Random Forest, Decision Tree, XGBoost, and BERT. The embedding techniques used in this study include word2vec, GloVe, and TF-IDF. The purpose of the current research is to achieve high precision in the sentiment categorization process while at the same time uncovering hidden topics that lie beneath each individual sentiment class through topic modeling. LDA has enabled the extraction of important topics from positive, negative, and neutral reviews.

The structure of this paper is outlined below: Section 2 will present an extensive literature review of the methods for machine learning and deep learning that have been used in sentiment analysis in patient reviews. The methodology adopted in this paper will be presented in Section 3, where the experimental setup will be explained. Section 4 will describe the findings, while Section 5 will outline the conclusions drawn from the analysis.

2. Related Work

Also known as sentiment analysis, opinion mining is the technique of finding out and classifying the emotions expressed in texts. Sentiment analysis can be helpful in making patients happy by providing useful insights in the healthcare industry. Techniques for assessing the sentiment polarity of texts were developed during preliminary work by Liu et al. (2012) and Pang and Lee (2008). Such techniques have currently been applied in the healthcare sector. Opinion mining and sentiment analysis techniques have been reviewed in detail by Pang and Lee (2008) [9], and the sentiment analysis techniques have been extensively surveyed by Liu et al. (2012) [8]. Various traditional ML models like Support Vector Machine, Naive Bayes, and Decision Tree algorithms have been commonly used for sentiment analysis. Hegde and Bhat (2018) were successful in achieving an accuracy rate of 89.17% by using SVM for sentiment classification [Pang and Taboada corpus dataset with 10 cross-validations]. The above experiment proved that SVM was effective in handling binary classification. Likewise, Rahman et al. (2019) analyzed patient feedback by using the Random Forest algorithm that could effectively handle different types of datasets while obtaining an accuracy rate of 85%.

In the field of healthcare, the use of naive Bayes classifiers for sentiment analysis has also been observed, which is famous for its efficiency and simplicity. Sentiment analysis of patients' reviews in health forum was conducted by Piryani et al. (2017) using Naive Bayes classifier to achieve an F1-score of 0.78 [12]. Although these traditional classifiers provide a base for sentiment analysis, they may face some difficulties in recognizing complex language patterns. Besides, sentiment analysis through an LSTM recurrent neural network has been applied to identify trends in public sentiment. In relation to the COVID-19 outbreak, Yin et al. (2020) proposed a new approach to detect the trending topics and sentiment trends in social media messages. Aspect based sentiment analysis coupled with topic

modeling was used by Jang et al. (2020) on the COVID-19 Twitter dataset. The change in subject trends during the pandemic period has also been explored by the researchers.

A major transformation has occurred in sentiment analysis through deep learning, with improved accuracy and the ability to cope with complex language structures. The use of RNNs and their variants, LSTMs, and GRUs has proven quite effective when applied to the task of sentiment categorization. According to Yadav et al. (2020), LSTM networks performed exceptionally well in sentiment categorization with the help of patient evaluations, yielding an F1-score of 0.86 [13]. Likewise, Zhang et al. (2018) reached an accuracy rate of 88% by applying GRU in sentiment categorization in the healthcare sector [14]. There is a possibility that CNN can learn from local features in texts that makes them useful in sentiment analysis applications. According to Kim (2014), CNNs are capable of producing state-of-the-art performance in several datasets used for sentiment analysis, including patient reviews[15]. Their model, which yielded an accuracy rate of 89%, is just one example of this application.

Researchers have tried to improve on sentiment classification by incorporating the use of hybrid and ensemble techniques, whereby the strengths of more than one method are combined. In an attempt to exploit features both locally and globally, Li and Wu (2021) have proposed a hybrid approach of CNN and LSTM, which yielded an accuracy of 91%. This clearly shows the benefits of using a hybrid method [16]. It has further been shown that ensemble methods such as the combination of Random Forests and deep learning methods can help to improve classification performance. An ensemble of LSTM along with Random Forest yielded 92% accuracy [17].

In natural language processing, another important development is transfer learning which involves utilizing already trained models such as BERT. With the exceptional skills of BERT in understanding semantics and context, sentiment analysis has seen significant progress. For instance, in their work, Zhou et al. (2022) reported state-of-the-art results with 94% accuracy when they employed BERT in sentiment classification of patients' reviews [18]. More complex natural language processing methods can be adopted in the healthcare sector as there is no need for large datasets due to pre-trained models.

Other than sentiment classification, topic modeling approaches such as LDA have also proven useful in identifying hidden topics in patient reviews. Health practitioners will be able to learn about specific topics in which there could be some concerns through identification of hidden topics related to different sentiments. Through use of LDA on patients' reviews, Wang and Liu (2019) were able to identify key topics that were associated with patient happiness and displeasure [20]. This has shown that knowledge on topics covered in patient reviews is essential to tackle problems in a targeted manner. Table 1 below demonstrates a comparative study of different approaches used for sentiment classification of patients.

Table 1: Comparison of existing models for patient sentiment classification based on accuracy

Model	**Dataset**	**Accuracy (%)**
Ref. [10]	Pang Corpus [21] and Taboada Corpus [22]	89.17
Ref. [11]	Movie Review Dataset V1.0 [23]	87.85
Ref. [24]	Movie Review Dataset V1.0 [23]	86.00
Ref. [25]	Movie Review Dataset V1.0 [23]	82.70
Ref. [26]	Movie Review Dataset V1.0 [23]	87.52

Ref. [27]	Movie Review Dataset V2.0 [23]	85.90
Ref. [28]	Movie Review Dataset V2.0 [23]	87.60
Ref. [29]	Movie Review Dataset V2.0 [23]	91.00
Ref. [18]	Patient reviews	94.00
Ref. [31]	US major airlines [30]	86.00
Ref. [32]	E-commerce sites [33]	88.48
Ref. [34]	Airy rooms Dataset	97.65
Ref. [12]	Health forum reviews	78.00

3. Methodology

The structure of the proposed system is presented in Figure 1, which comprises the following phases: data acquisition, preprocessing, split, model selection, training, evaluation, and topic modeling. Every one of them is essential for the correct and accurate sentiment analysis of patient reviews. In order to develop models capable of comprehensively understanding and classifying sentiments, the current study aims at systematically dealing with every phase. In the proposed architecture, several machine learning methods will be applied to the prepared data set through the training phase, specifically the Random Forest Classification, Decision Tree, and XGBoost Classification. Other models like RNN, LSTM, GRU, and Bidirectional-LSTM will be tested in addition to two kinds of embedding, GloVe and Word2Vec. The last model, Transformer called BERT, will be applied for comparison purposes. Lastly, topics will be extracted from all the three sentiment categories by means of the topic modeling algorithm referred to as LDA.

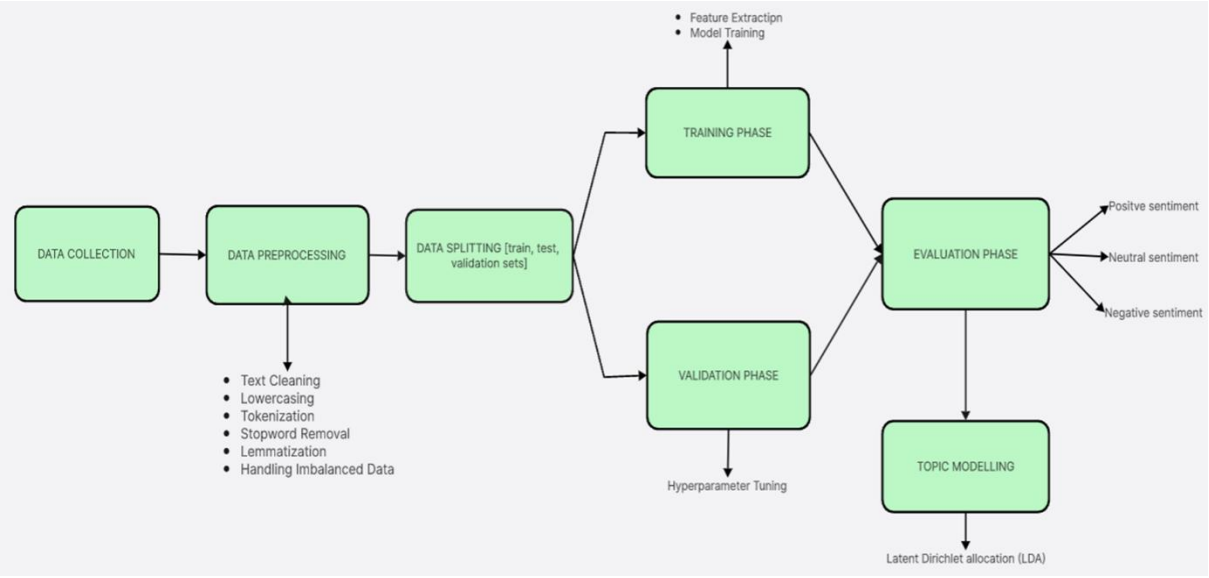


Fig. 1: Proposed Architecture of Sentiment Analysis of Patient Reviews

3.1. Dataset

The first requirement includes acquiring a large number of patient reviews. In our case, we will collect reviews for 21 hospitals in various cities of India. These reviews are available on websites related to

healthcare where patients can freely comment on their experiences and opinions regarding the services provided by the hospitals. We will acquire around 10,000 patient reviews from these websites The data obtained in the review consists of varied sentiments categorized as "positive," "negative," or "neutral".

The powerful features provided by libraries such as BeautifulSoup, Requests, and Pandas assist in making the data collection process effective and thorough. By effectively parsing, cleaning, and formatting the collected data in a DataFrame, a well-defined dataset is formed, suitable for use in the analysis and building models. Through this process, we will improve the quality of services in the healthcare sector.

3.2. Data Pre-processing

Data preprocessing is an important step involved in preparing data from an unstructured state to a well-organized form suitable for sentiment analysis. The accuracy and reliability of the results are enhanced through this step since it ensures that the data are formatted in such a manner as to allow easy processing and analysis using machines. The process of preprocessing involves several key steps. Lower casing of all texts is one such key step that ensures uniformity in the data being processed. Case insensitivity becomes a problem in this process. For example, the words "Good" and "good" would be treated as different tokens if not lower-cased.

Tokenization involves the division of the text into tokens or words that act as the basic units for analysis. In other words, each token represents a word or a part of text that the model understands. This step forms the basis of further text extraction and text processing. In order to reduce dimensionality, stop words are stripped from the text, as they refer to commonly used terms that add little meaning to the text. Common terms like 'and,' 'the,' and 'is' may be frequently used, but they add nothing substantial to the sentiments portrayed by the text. The process of stripping these words shifts focus to the words that actually express the sentiment of the review. The NLTK library is used to strip off these stop words. Moreover, the stem library helps to strip off all special characters including ['!"#$%&'()*+,-./:;<=>?@[\]^_`{|}~'] that do not carry any semantic value. Whitespace stripping is done using Python.

First, Lemmatization is carried out to standardize the text by bringing words into their base form. For instance, the words "running," "ran," and "runs" can all be considered variants of the same word "run." This step is useful in standardizing the text so that various word forms become one token, thus enabling the model to understand better. The WordNetLemmatizer class in the nltk.stem module in Python can be used to perform this task. Lastly, addressing the issue of contractions is important during text preprocessing. Contractions like "don't" should be replaced with "do not," while "can't" should become "cannot."

Figure 2 represents the text pre-processing process that is vital for natural language processing activities. These processes cover contraction handling, removal of special characters and stop words, as well as converting all texts to lower case. Other processes covered in the next stages include lemmatization, orthographic correction, and removing of extra white spaces.

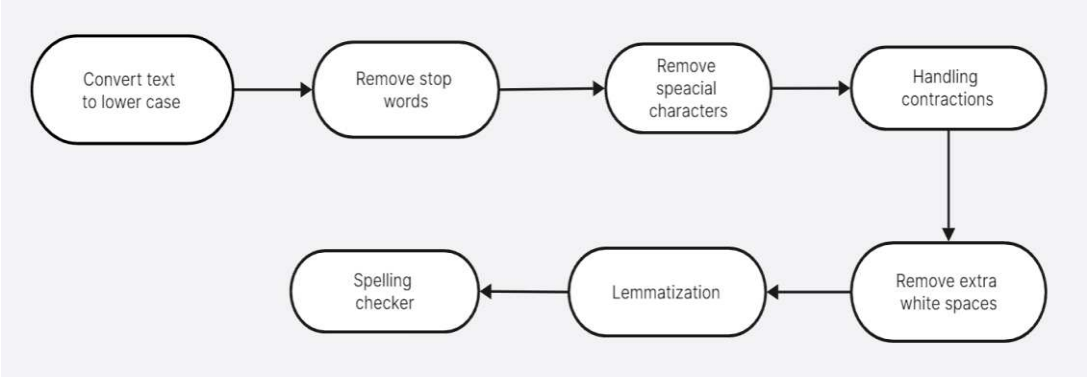


Fig. 2: Data preprocessing pipeline

3.3. Machine Learning Approach

The current research uses machine learning techniques to classify the text data using the Term Frequency-Inverse Document Frequency (TF-IDF). The use of the TF-IDF ensures that a correct measurement of the importance of a word within a document is achieved, hence making it possible to distinguish between relevant and irrelevant keywords. Such an operation makes it possible to use several machine learning techniques such as random forest, decision tree, and XGBoost for classifying the data. While positive reviews are classified with the number zero, negative reviews are denoted with the number one and neutral reviews have the number two.

$$accuracy\ (y,\ \hat{y}) = \frac{1}{n}\sum_{i=0}^{n-1} 1(\hat{y}\ = y_i) \tag{1}$$

$$precision = \frac{True\ positive}{True\ positive + False\ positive} \tag{2}$$

$$recall = \frac{True\ positive}{True\ positive + False\ negative} \tag{3}$$

$$f1\ score = 2 \times \frac{precision \times recall}{precision + recall} \tag{4}$$

Random Forest model is preferred owing to its robustness and capability to deal with high dimensional data. An accuracy of 0.8817 is recorded by the model, suggesting good model performance. In relation to class 0, F1 score, recall and precision measures are respectively computed to be 0.92, 0.96 and 0.89. For class 1, F1 score, recall and precision measures are computed to be 0.72, 0.57 and 0.96 respectively. For class 2, measures of F1 score, recall and precision are respectively calculated as 0.84, 0.86 and 0.83. Despite varying performance levels across different classes, the performance of this model can be rated excellent for class 0.

Many evaluation metrics were adopted in determining the effectiveness of the Decision Tree model when used for text classification purposes. Accuracy of this machine learning model is 0.8435 and is slightly less compared to that of the Random Forest classifier. Regarding class 0, precision, recall, and F1 scores are 0.90, 0.89, and 0.89 respectively. As for class 1, recall and precision scores are 0.63 and 0.68 respectively, while F1 score is 0.65. For class 2, recall and precision scores are 0.85 and 0.78 respectively, and F1 score is 0.81. Despite being a successful one, recall and precision scores of the decision tree model are different from those of the random forest classifier.

The maximum accuracy provided by the usage of XGBoost, which is the fastest model for machine learning, among all other models is 0.8986. The precision and recall rate of the class zero are 0.93 and 0.94 respectively, while its F1-score is 0.93 according to the classification report. The precision, recall, and F1-score of class one are 0.90, 0.72, and 0.80 respectively. Likewise, the precision and recall rates

of the class two are 0.80 and 0.90 respectively, while its F1-score is 0.85. It is safe to conclude here that XGBoost is the best model, providing the highest accuracy with precision and recall rate.

The accuracy, precision, recall, and F1 score for XGBoost classifier were better than that of Random Forest and Decision Tree classifiers. From the analysis above, it is clear that the performance of Random Forest classifier was outstanding;but compared to the XGBoost, its performance was not as efficient in case of minority class classification. The Decision Tree classifier was the one which showed variation in its performance depending upon the type of data class.

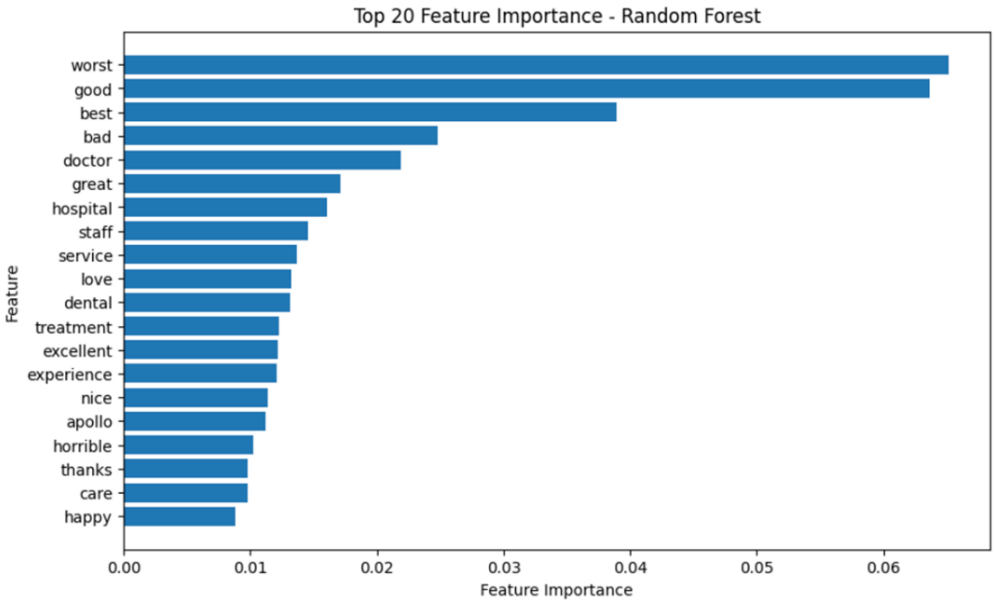
Decision tree techniques use entropy in Eq. 5 to quantify the uncertainty or impurity in a dataset. It measures the dataset's degree of disorder or randomness.

Information gain in Eq. (6) (where D_i is the subset of D after splitting by an attribute) is a metric used to assess the decrease in impurity or entropy following the splitting of a dataset according to a specific attribute.

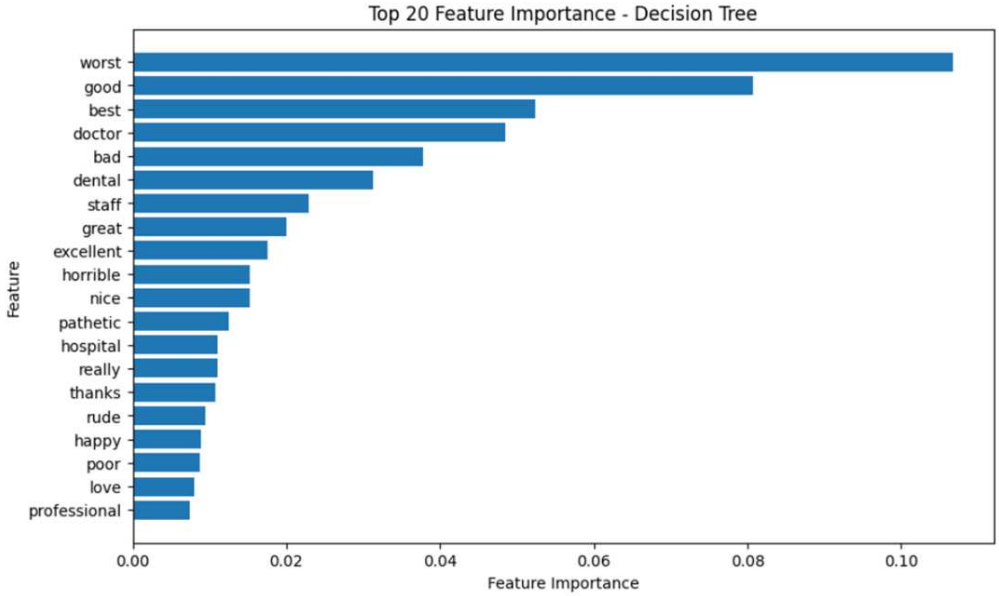
$$H(D) = -\sum_{i=1}^{c} p_i log_2(p_i) \quad (5)$$

$$IG(D,A) = H(D) - \sum_{i=1}^{n} \frac{|D_i|}{|D|} \times H(D_i) \quad (6)$$

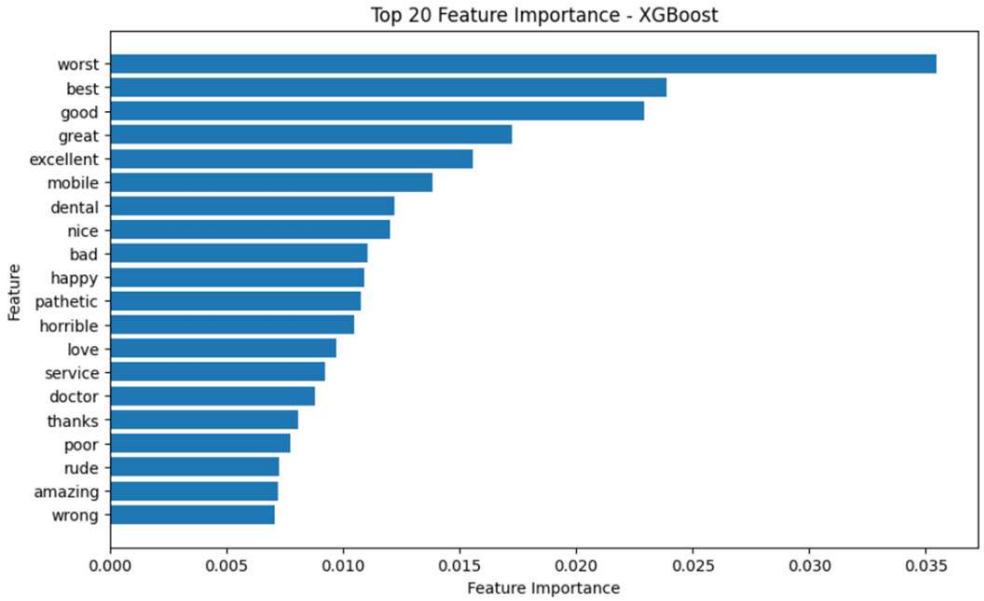
Feature importance scores are calculated and visualized to elucidate the impact of different features in the Random Forest classifier. The top 20 features (shown in Figure 3) are extracted from the TF-IDF vectorizer based on their relevance scores.



(i) Feature Importance w.r.t. Random Forest Classifier



(ii) Feature Importance w.r.t. Decision Tree Classifier



(iii) Feature Importance w.r.t. XGBoost Classifier

Fig. 3: Top 20 Feature Importance (i) Random Forest Classifier (ii) Decision Tree Classifier and (iii) XGBoost Classifier

The TF-IDF score for a term t in a document d from the document set D is the product of its term frequency and inverse document frequency in Eq. 7

$$tf\ idf(t,d,D) = tf(i,d)\ .\ idf(t,D) \tag{7}$$

Where:

$$tf(t,d) = \log(1 + frq(t,d)) \tag{8}$$

$$idf(t,D) = \log(\frac{N}{count(d \in D : t \in D)}) \tag{9}$$

The most significant features are displayed at the top of a horizontal bar chart, effectively visualizing their significance. This analysis focuses on the critical factors that affect the process of classification and gives important information about the working mechanism of the model. The above table shows that specific phrases are highly effective for the decisions made by the model.

3.4. Deep Learning Approach

This paper is an effort to implement and evaluate different types of deep learning models for POS tagging. The procedure includes a number of vital steps: data pre-processing, use of GloVe and Word2Vec independently to incorporate the text data, and development of different RNN structures, namely Simple RNN, LSTM, GRU, and Bidirectional LSTM. Initially, the analysis begins with a data set of 8,626 labeled sentences which consists of a lexicon of 37 different POS tags and 15,394 tokens. There are three data pre-processing steps: padding of the sequence, vectorizing the data, and tokenizing it. Each sentence is tokenized into numerical vectors suitable for deep learning techniques. To ensure uniformity of all input variables, the sequence is padded up to 100 tokens per sentence.

GloVe embeddings, which have 300 dimensions, are used for token representation in the vector space. This layer is applied at the beginning of the model as an embedding layer to get effective semantic representation. GloVe embeddings pretrained embeddings can be used to initialize the weights for better semantic representations.

Moreover, Word2Vec word embeddings are investigated where word embeddings are represented using 300 dimensional vectors. The size of the vocabulary depends on the word tokenizer, and therefore, an embedding matrix is created that represents the semantic vectors for those words in the corpus. Words that are not in Word2Vec have their corresponding embedding vector set to zero.

3.4.1. Simple RNN

As mentioned before, the initial model for POS tagging was a Simple RNN, which had the objective of predicting the POS tag for each individual word. The model consists of an embedding layer, a SimpleRNN layer containing 64 units, and a TimeDistributed Dense layer employing Softmax activation.This model, utilizing the Adam optimizer and categorical cross-entropy loss, contains 4,644,030 trainable parameters. The training utilized a batch size of 128 across 10 epochs. Figure 4 illustrates the model's accuracy with word2vec and GloVe.

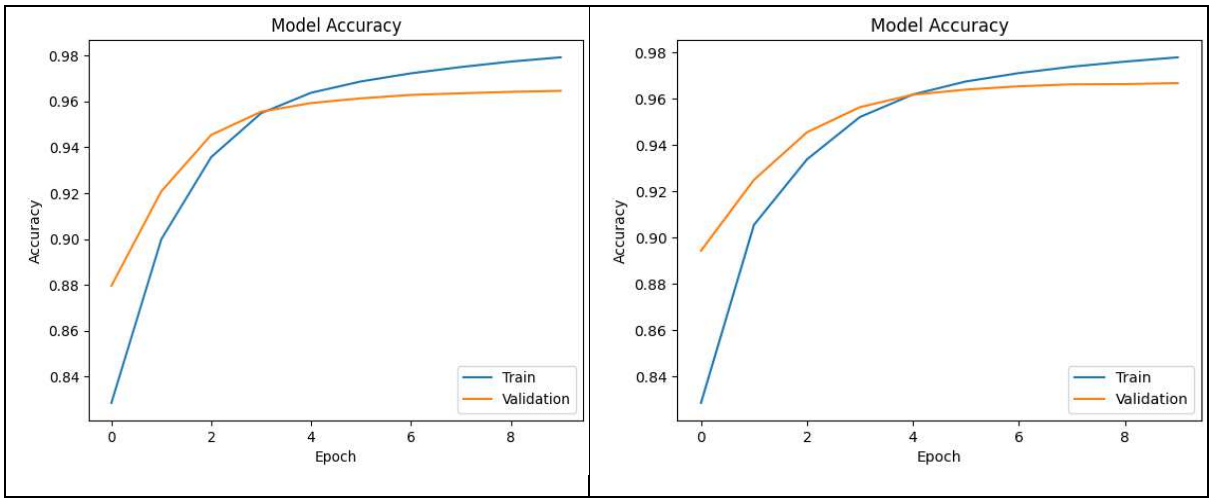


Fig. 4: Training history (i) with word2vec and (ii) with GloVe

3.4.2. LSTM

The second method uses LSTM units, which are capable of modeling long-range dependencies. The LSTM network resembles that of the RNN network, except for the fact that it uses an LSTM layer instead of a SimpleRNN layer. Trained using the Adam optimizer with categorical cross-entropy loss, the number of parameters to be trained in this method is 4,714,110, with a batch size of 128 and for 10 epochs. Figure 5 illustrates the model's accuracy with word2vec and GloVe.

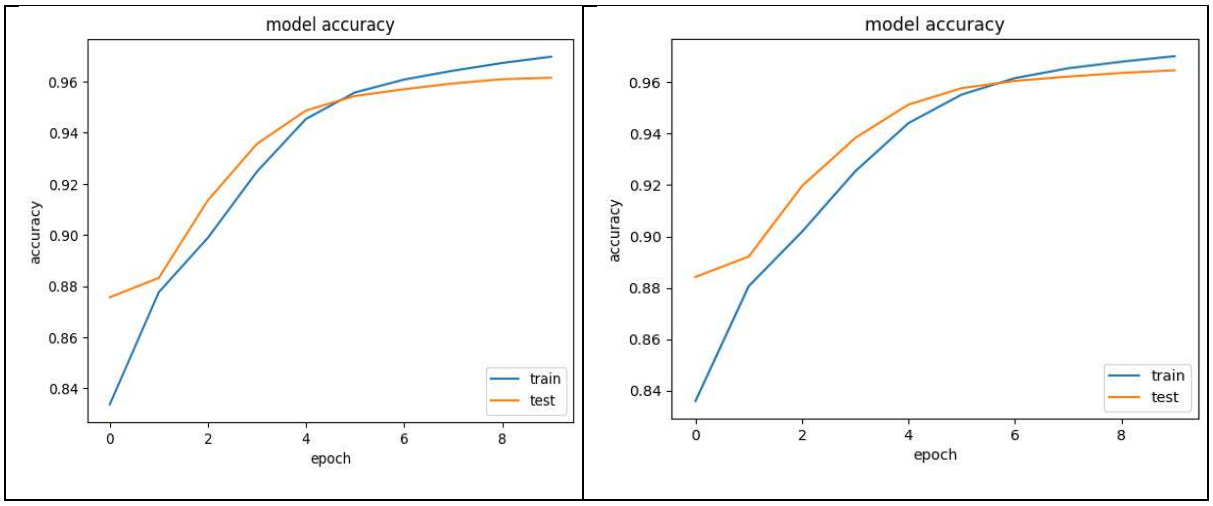


Fig. 5: Training history (i) with word2vec and (ii) with GloVe

3.4.3. GRU

This architecture employs a GRU model for sequence classification tasks, starting with an Embedding layer that converts input word indices into fixed-size dense vectors, specifically 300-dimensional vectors. Then, a 64-GRU cell GRU layer is added. This layer processes the input sequence while preserving an internal state to identify sequential dependencies. The GRU layer generates an output at every time step by returning sequences. The Time Distributed layer does the same, adding a Dense layer for every time step. Due to the equal number of units in the Dense layer and output classes (38 here), the structure enables the model to output multiple output classes at every time step. Finally, the Adam optimizer and categorical cross-entropy are chosen to build the model. The above neural network architecture is well suited for problems that require the contextual understanding of words, such as named entity recognition, sentiment analysis, and sequence labeling. The above model built using Adam optimizer and categorical cross-entropy has 4,690,942 parameters and is trained on a batch size of 128 for 10 epochs. Figure 6 shows accuracy of word2vec and GloVe models.

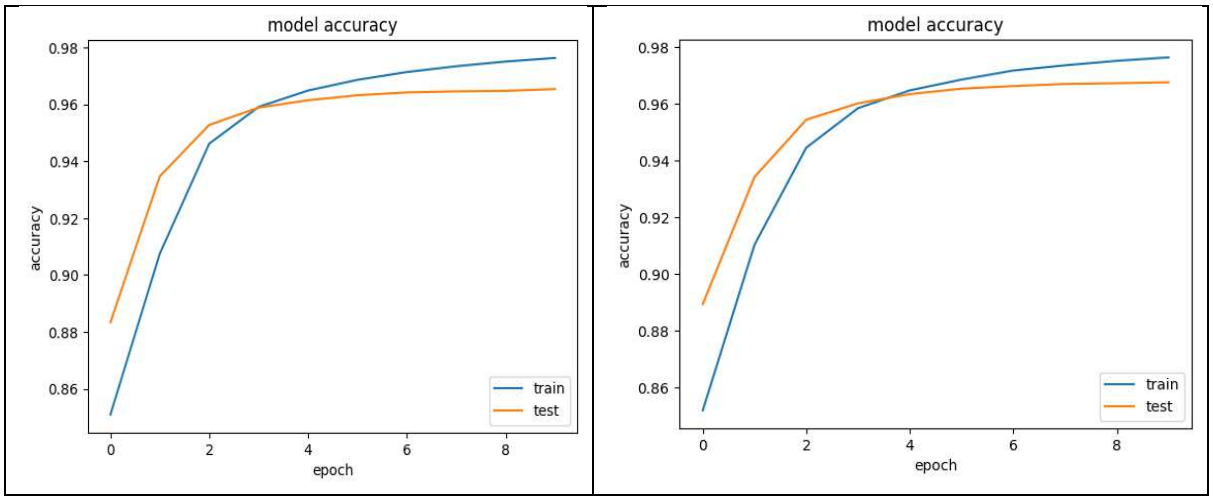


Fig. 6: Training history (i) with word2vec and (ii) with GloVe

3.4.4. Bidirectional LSTM

For the final model, bidirectional LSTM neural networks have been used to capture context from both ends of the sentence. This is because the model analyzes the input sequence not only from the start but also from the end, working on the sequences in both directions. This particular design, which is ideal for classifying sequences, has an initial layer called "Embedding," which takes in the input indices of words and produces dense vectors (of 300 dimensions in this case). Next, two layers of LSTMs are combined to create a Bidirectional LSTM, and each layer contains 64 LSTM cells which produce outputs by returning sequences. The model is capable of predicting several different classes of outputs independently per time step, through the use of TimeDistributed layer along with Dense layer for every time step, where the number of units in the Dense layer will be equal to the number of classes (in this case, 38 classes). The final construction of the model utilizes Adam optimization along with categorical cross-entropy as the loss function. The architecture of this model is best suited for situations where the understanding of context is important to the words, in order to make the predictions accurately, such as named entity recognition or part-of-speech tagging.

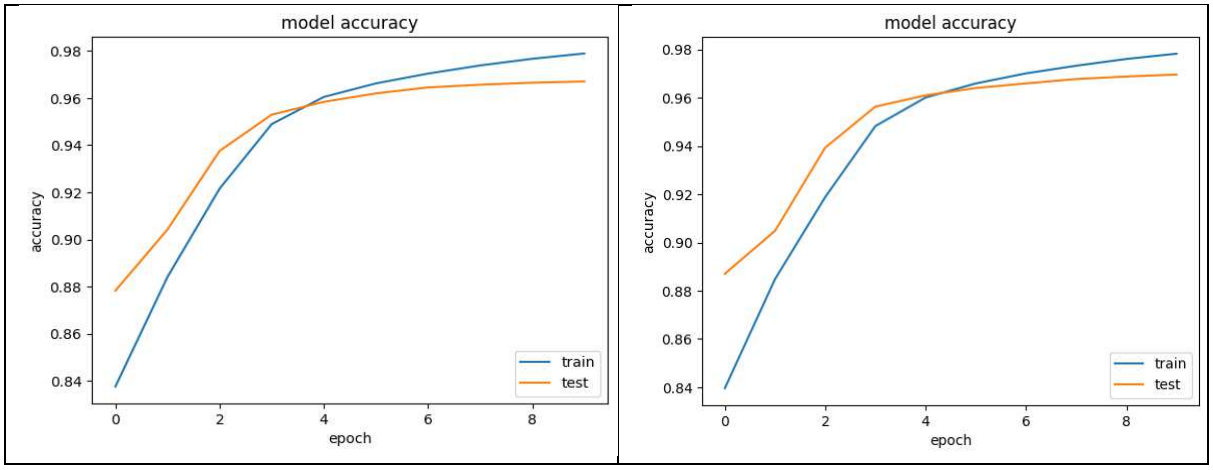


Fig. 7: Visualization of training process (i) using word2vec and (ii) using GloVe

The diagram in Figure 8 illustrates the architecture starting with the input layer receiving the input. The input data is then passed to the embedding layer where the data will be represented in numbers. Thereafter, the TimeDistributed layer uses any of the following recurrent layers—SimpleRNN, LSTM, GRU, or Bidirectional LSTM—to the elements of the sequences. Recurrence layers perform better when it comes to handling dependencies that can extend further in time into the past. This layer's output is then passed on to the dense layer, which is a fully-connected layer that will serve for predictions. Lastly, the output layer generates a probability distribution of the possible outcomes using the softmax activation function.

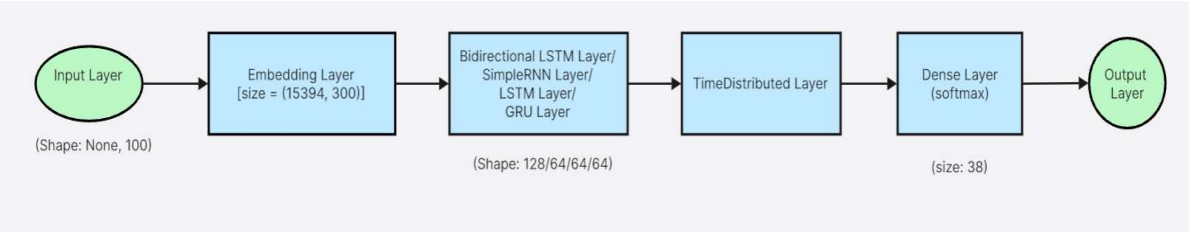


Fig. 8: Architecture of neural network models

3.5. Transformer Approach

This method utilizes a model based on BERT for categorizing customer reviews into three sentiment classes. First, the dataset containing customer reviews along with their corresponding sentiments is

collected and then preprocessed. Textual data are processed through a BERT tokenizer in the Hugging Face library to ensure that each review is either truncated or padded to 512 tokens. In the preprocessing stage, the reviews are converted into an appropriate format that can be used in BERT model training. Data are then split into training and validation datasets based on an 80-20 ratio in a stratified manner.

In this study, the 'bert-base-uncased' model from the Hugging Face repository will be used along with an extra classification layer that generates logit scores for three different sentiment classes. The Hugging Face Trainer is configured with various training parameters such as number of epochs, batch size, and evaluation strategy, which apply to both training and evaluation stages. The training process of the model depends on these parameters, while the model's performance depends on metrics such as accuracy, precision, recall, and F1 score. In order to overcome imbalanced classes, the metric averaging approach uses the weighted average technique.

The approach that the BERT model adopts to perform sentiment analysis is shown in Figure 9. The first step entails the preprocessing of the input data, which is then split into the training and validation data sets. This is followed by the tokenization of the data using the BERT tokenizer and then initialization of the BERT model for sequence classification. Data set classes for training and validation are then created, after which a trainer is defined with specific training parameters.

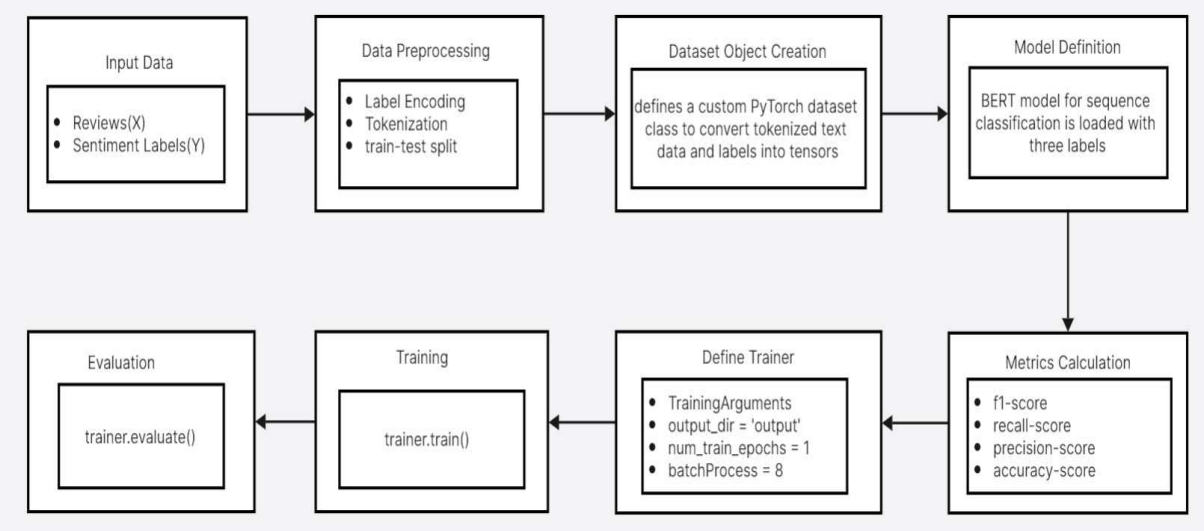


Fig. 9: General architecture of BERT model

3.6. Topic Modeling

It was intended to reveal the topics hidden in customer reviews through Latent Dirichlet Allocation (LDA). Sentiment classification helped conduct topic modeling independently for both negative, positive, and neutral reviews, thus enabling an insight into hidden themes in each particular type of sentiment.

Initially, we have selected only those reviews that were classified as "negative". TF-IDF vectorization is subsequently employed to transform this subset of reviews into a Document-Term Matrix (DTM). The vectorizer preprocessed the text by eliminating accents, converting it to lowercase, and removing common stop words. To improve DTM quality, terms present in fewer than ten reviews or in over 90% of the reviews are eliminated. This transformation generated a sparse matrix, emphasizing term importance throughout the corpus by modifying term frequencies with their inverse document frequencies.

After running the Gibbs sampling,

1). Estimate Topic Distributions (θ) for each Document in Eq. 10 ($n_{d,k}$: Number of words in document d assigned to topic k, n_d: Total number of words in document d.)

$$\theta_{d,k} = \frac{n_{d,k} + \alpha}{n_d + K\alpha} \tag{10}$$

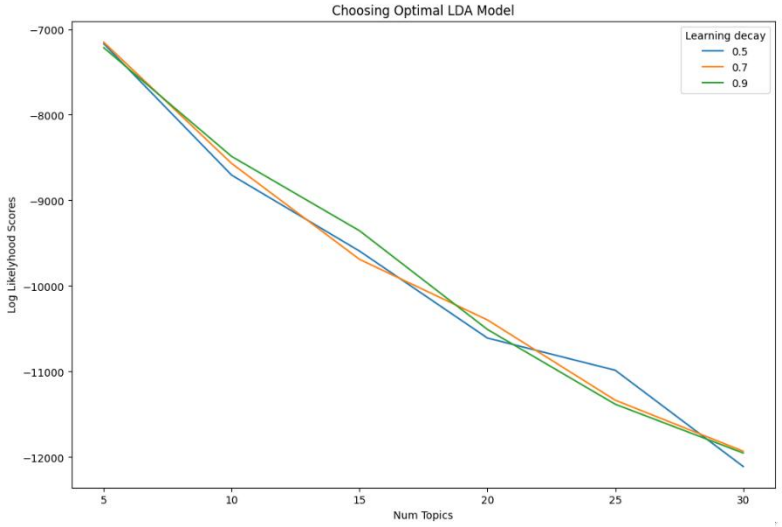
2). Estimate Word Distributions (β) for each Topic in Eq. 11 ($n_{k,w}$: Number of times word w is assigned to topic k, n_k: Total number of words assigned to topic K.)

$$\beta_{k,w} = \frac{n_{k,w} + \mu}{n_k + V\mu} \tag{11}$$

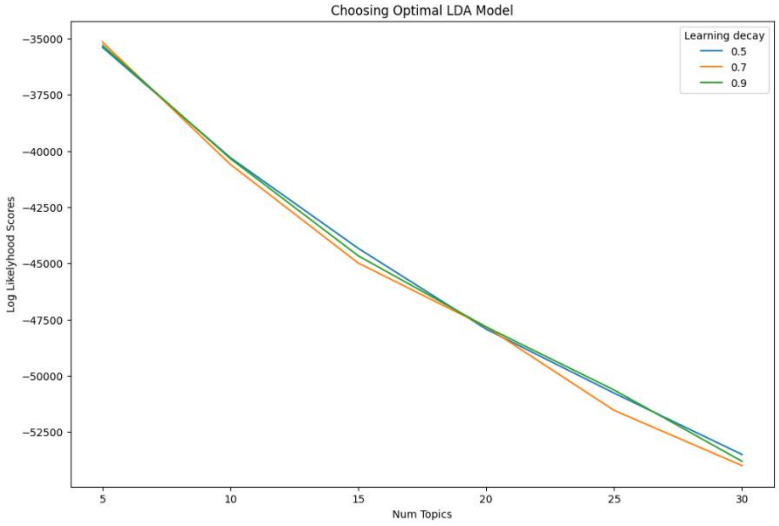
Here *k* is the number of topics and V is the vocabulary size.

To identify the best LDA model, a grid search is conducted across various hyperparameters, specifically learning decay and the number of topics Decay values of 0.5, 0.7, and 0.9 for learning will be considered during the search, and the range of number of topics will be from 5 to 30 topics. All the combinations will be tested with the help of GridSearchCV using log-likelihood scoring. The model with maximum log-likelihood scoring is deemed the most suitable one, as it provides the best topic modeling with respect to negative reviews.

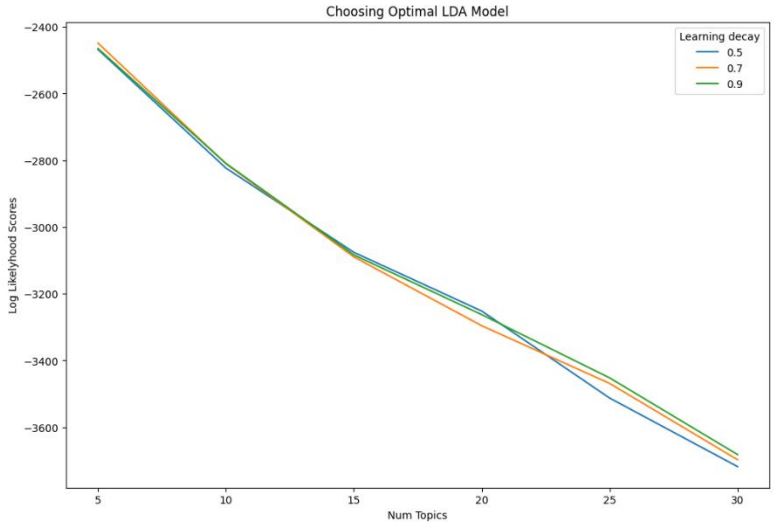
The log-likelihood scores and perplexities are computed to assess the LDA models. High log-likelihood scores indicate that the model fits well; they show how well the model fits the data observed. Perplexity, on the other hand, is yet another key measurement of the predictive power of the model, wherein high numbers indicate that the model does not generalize well to new data. The results from the grid search are given below to help identify the best model.



(a) LDA on negative comments



(b) LDA on positive comments



(c) LDA on neutral comments

Lastly, the task was interpreting the topics that the optimal LDA algorithm produced. By selecting the top contributing words for each topic, we were able to determine what the key themes that occur in the negative comments were. This process was repeated for both the positive and neutral sentiments so that the resulting topics could be compared between sentiments. The topic is described based on its top words, which are characteristic for each of the sentiments. Figure 10 shows the LDA applied to negative, positive, and neutral comments.

4. Result and Discussion

In this part, we will address the output obtained from our experimentation with respect to the accuracy performance and other parameters. Our experiments have all been done with the help of the Deep Learning package [35], where the hardware specifications are as follows: AMD Ryzen 7 5700U with Radeon Graphics, 1801 MHz, 8 Core(s), 16 Logical Processor(s), 16GB RAM. Also, Python 3.7 has been used for programming.

Using accuracy and comprehensive classification criteria, this study assesses the performance of three machine learning classifiers: Random Forest, Decision Tree, and XGBoost. According to the results,

XGBoost achieves the best accuracy (0.899), followed by Decision Tree (0.843) and Random Forest (0.882). For each model, comprehensive classification reports are given so that their performance in various classes can be further examined. Table 2 displays the evaluation results of independent model's based on machine learning approach.

Table 2: Independent Model's Accuracy, Precision, Recall, and F1-score based on Machine Learning approach.

Model	**Accuracy (%)**	**Weighted Average**		
		Precision (%)	**Recall (%)**	**F1-score (%)**
Random Forest	88.17	89.33	79.66	82.66
Decision Tree	84.34	78.66	80	78.33
XGBoost	89.85	87.66	85.33	86

The evaluation of four different neural network models with GloVe embeddings has been done in Table 3 based on the test data. The four models are a simple RNN, LSTM, GRU, and Bi-directional LSTM. For the LSTM model, the accuracy is 0.9578, while the loss is 0.1444. On the other hand, for the RNN model, the accuracy is 0.9602, while the loss is 0.1339. For the GRU model, the accuracy is 0.9609, and the loss is 0.1264. The Bi-directional LSTM model shows superior performance since the accuracy is 0.9628 with a loss of 0.1232. In summary, this model performed slightly better in terms of minimized loss and maximum accuracy for the given data set.

Table 3: Deep Learning Model's Accuracy, loss for Patient experience dataset with GloVe.

Model	**Accuracy (%)**	**Loss Score**
RNN	96.01	0.133
LSTM	95.77	0.144
GRU	96.09	0.126
Bidirectional LSTM	96.27	0.123

The following table is Table 4 showing the performance of four different neural network models trained with word2vec embeddings on a test set. The four models considered here are a simple RNN model, an LSTM model, a GRU model, and a bidirectional LSTM model. The LSTM model showed a performance of 0.9621 and 0.1289 for its accuracy and loss respectively, while the RNN model showed a performance of 0.9654 and 0.1174 for its accuracy and loss respectively. The GRU model showed a performance of 0.9661 and 0.1128 for its accuracy and loss respectively. The best-performing model among the four is the bidirectional LSTM model whose accuracy is 0.9680 and loss is 0.1070.

Table 4: Deep Learning Model's Accuracy, Loss for Patient experience dataset with word2vec.

Model	**Accuracy (%)**	**Loss Score**
RNN	96.53	0.117
LSTM	96.21	0.128
GRU	96.57	0.112

Bidirectional LSTM	96.76	0.107

As illustrated in Figure 11 below, this is the outcome of applying Latent Dirichlet Allocation (LDA) to a dataset consisting of negative reviews. The table consists of six columns; each represents a particular topic discovered using the algorithm. These words describe the topic under consideration, which could be either positive, negative, or neutral about health care.

	Healthcare Experience and Gratitude	Negative Clinic Experience and Dental Issues	Negative Surgical Experience and Patient Dissatisfaction	Medical Staff Issues and Treatment Concerns	Poor Hospital Service Experience and Carelessness
0	doctor	bad	waste	doctor	worst
1	hospital	experience	surgery	hospital	hospital
2	patient	clinic	money	patient	service
3	money	dental	false	staff	experience
4	day	pathetic	year	wrong	poor
5	worst	service	mother	treatment	horrible
6	treatment	expensive	rude	time	staff
7	time	love	painful	health	apollo
8	admitted	terrible	doctor	problem	doctor
9	thank	spine	dentist	appointment	careless

(i) Negative review

	Best Hospital and Patient Care	Exceptional Dental Care and Love	Advanced Surgery and Pristyncare Services	Thankful for Treatment and Healing Efforts	Positive Experience with Good Service
0	hospital	dental	hospital	treatment	good
1	doctor	love	surgery	thank	service
2	pet	clinic	doctor	thanks	hospital
3	best	prison	pristyncare	doctor	staff
4	patient	experience	care	healing	best
5	animal	care	pain	hand	doctor
6	good	treatment	apollo	good	nice
7	money	dentist	patient	sir	excellent
8	new	teeth	day	appreciate	treatment
9	staff	amazing	good	effort	experience

(ii) Positive review

	Timely Response and Mobile Health Check	Apollo Hospital Experience and Reviews	Quality Service in Surgery and Patient Care	Professional Doctor Care and Support Team	Effective Treatment and Emergency Response
0	response	hospital	surgery	doctor	treatment
1	mobile	apollo	patient	care	money
2	received	review	service	team	narayana
3	view	experience	animal	prison	heart
4	check	pain	dental	helpful	emergency
5	health	bangalore	staff	pristyncare	hrudayalaya
6	yashoda	sir	hospital	place	father
7	ran	time	clinic	avoid	food
8	medical	road	problem	management	went
9	beware	narasimhan	care	dear	got

(iii) Neutral review

Fig. 11: Topics associated with (i) negative review, (ii) positive review, (iii) neutral review

In Fig. 12, the wordcloud shows frequently used terms that are taken from hospital internet evaluations. The terms "patient," "doctor," "hospital," and "treatment" are the most frequently used ones, indicating that these are the elements of care that patients most frequently discuss in their evaluations. While negative terms like "bad," "experience," and "cost" are less common but still prevalent, positive words like "good," "staff," and "well" are also commonly cited. Interestingly, terms like "junk," "fake," and

"irrelevant" also surface, implying that some reviewers might have provided false information. Overall, the wordcloud highlights both positive and negative elements of care and offers a preliminary analysis of the emotion expressed in online reviews of this institution.

Fig. 12: Wordcloud for all sentiment

5. Conclusion and Future Research Direction

This study presented a novel approach to sentiment analysis in patient experience analysis. The presented approach divides patient input from 21 hospitals spread across different Indian cities into three categories: good, negative, and neutral attitudes. It does this by using sophisticated machine learning algorithms. Robust techniques and models are utilized to extract significant inferences from textual data. These insights can be used by healthcare professionals to identify problem areas, identify reoccurring concerns, and tailor their services to better meet the needs of their patients.

Numerous sophisticated algorithms, including RNN, LSTM, GRU, Bidirectional LSTM, Random Forest, Decision Tree, XgBoost, and BERT, are used in our methodology. Bidirectional LSTM slightly outperformed other recurrent neural networks with the accuracy of 96.76 [with word2vec] and 96.27 [with glove]. The embedding models, including the TF-IDF model, GloVe model, and the word2vec model, are used to capture the semantic relationships within the dataset. Additionally, themes are identified in sentiment-labeled data using the Latent Dirichlet Allocation (LDA) based topic modeling technique that brings up vital topics and locations where improvements in the provision of healthcare services are needed. This sentiment analysis and topic modeling technique gives a comprehensive understanding of the opinions expressed by patients.

Future scope involves making use of patient reviews not only from major multispecialty hospitals via different portals and forums but also from physical survey via phone, social media applications, etc. This would facilitate participation from patients regarding their views and enable the hospital to get an idea about the issues that exist in their operations or infrastructure. Different sophisticated variants of neural networks, ensemble models, and transfer learning can be utilized for this purpose.

Declaration Statements:

Ethics Approval : Not Applicable

Consent to Participate : Not Applicable

Consent for Publication : Not Applicable

Funding: This research did not receive any specific funding

Conflict of Interest: The authors declare no conflict of interest

Acknowledgements: I would like to express my very great appreciation to the co-authors of this manuscript for their valuable and constructive suggestions during the planning and development of this research work.

Informed consent: Not Applicable

Ethical approval: Not Applicable

Author Contribution: All authors have made substantial contributions to conception and design, revising the manuscript, and the final approval of the version to be published. Also, all authors agreed to be accountable for all aspects of the work in ensuring that questions related to the accuracy or integrity of any part of the work are appropriately investigated and resolved.

Data Availability Statement:

The datasets generated during and/or analysed during the current study are available from the corresponding author on reasonable request.

References

[1] Gao GG, McCullough JS, Agarwal R, Jha AK. A shifting landscape of doctor quality ratings: An analysis of patients' online ratings of their doctors over five years. J Med Internet Res 2012;14(1):e38 [FREE Full text] [doi:10.2196/jmir.2003] [Medline: 22366336]

[2] Greaves F, Millett C. Consistently increasing numbers of online ratings of healthcare in England. J Med Internet Res 2012;14(3):e94 [FREE Full text] [doi: 10.2196/jmir.2157] [Medline: 22742977]

[3] Cartouche, I.; Elmendili, S.; Elmendili, F. Sentiment Analysis of Health Care: Review. E3S Web Conf. 2021, 319, 01064. [CrossRef]

[4] Hu, G.; Han, X.; Zhou, H.; Liu, Y. Public Perception on Healthcare Services: Evidence from Social Media Platforms in China. Int. J.Environ. Res. Public Health 2019, 16, 1273. [CrossRef]

[5] Hossain, D.; Azam, S.; Ali, J.; Sabit, H. Drugs Rating Generation and Recommendation from Sentiment Analysis of Drug Reviews using Machine Learning. In Proceedings of the 2020 Emerging Technology in Computing, Communication and Electronics (ETCCE), Dhaka, Bangladesh, 21–22 December 2020. [CrossRef]

[6] Ramírez-Tinoco, F.J.; Alor-Hernández, G.; Sánchez-Cervantes, J.L.; Salas-Zárate, M.D.P.; Valencia-García, R. (Eds.) Use of Sentiment Analysis Techniques in Healthcare Domain. In Current Trends in Semantic Web Technologies: Theory and Practice. Studies in Computational Intelligence; Springer: Cham, Switzerland, 2019; Volume 815, pp. 189–212. [CrossRef]

[7] F. Greaves, D. Ramirez-Cano, C. Millett, A. Darzi and L. Donaldson, "'Use of sentiment analysis for capturing patient experience from free-text comments posted online," J.Med.Internet Res., vol. 15, no. 11, Nov 1, pp. e239., 2013

[8] Liu, B., Hu, M., & Cheng, J. (2012). Opinion observer: Analyzing and comparing opinions on the web. Proceedings of the 14th International Conference on World Wide Web, 342-351.

[9] Pang, B., & Lee, L. (2008). Opinion mining and sentiment analysis. Foundations and Trends in Information Retrieval, 2(1-2), 1-135.

[10] Hegde, R., & Bhat, S. (2018). Sentiment analysis of healthcare reviews using support vector machine. Journal of Data Mining in Genomics & Proteomics, 9(1), 2168-9458.
[11] Rahman, T., Hasan, K., & Afroz, F. (2019). An analysis of sentiment of patient reviews using Random Forest classifier. IEEE Access, 7, 124878-124888.
[12] Piryani, R., Madhavi, D., & Singh, V. K. (2017). Analytical mapping of opinion mining and sentiment analysis research during 2000-2015. Information Processing & Management, 53(1), 122-150.
[13] Yadav, A., Vishwakarma, D. K., & Yadav, S. K. (2020). Sentiment analysis using deep learning architectures: A review. Artificial Intelligence Review, 53(6), 4335-4385.
[14] Zhang, H., Zhang, S., & Pan, Y. (2018). Improving sentiment analysis by using a combination of convolutional and recurrent neural networks. Procedia Computer Science, 130, 167-174.
[15] Kim, Y. (2014). Convolutional neural networks for sentence classification. Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP), 1746-1751.
[16] Li, J., & Wu, X. (2021). Hybrid CNN-LSTM model for short text sentiment classification. Applied Sciences, 11(3), 1048.
[17] Liu, B., Zhou, Y., & Jiang, X. (2019). Combining ensemble learning and deep learning for sentiment analysis. International Journal of Machine Learning and Cybernetics, 10(2), 381-397.
[18] Zhou, W., Wu, Y., & Zhang, Z. (2022). Sentiment analysis of patient reviews using BERT model. IEEE Transactions on Computational Social Systems, 9(2), 500-509.
[19] Devlin, J., Chang, M.-W., Lee, K., & Toutanova, K. (2019). BERT: Pre-training of deep bidirectional transformers for language understanding. Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers), 4171-4186.
[20] Wang, Y., & Liu, X. (2019). Exploring latent themes in patient reviews using topic modeling. Health Informatics Journal, 25(1), 121-13
[21] B. Pang and L. Lee, "A sentimental education: Sentiment analysis usingsubjectivity summarization based on minimum cuts," in Proceedings ofthe ACL, 2004.
[22] M. Taboada, C. Anthony, and K. Voll, "Methods for creating semanticorientation dictionaries," in Conference on Language Resources andEvaluation (LREC), 2006, pp. 427–432
[23] Dataset : cs.cornell.edu/people/pabo/moviereview-data/.
[24] T. Mullen, N. Collier, Incorporating topic information into sentiment analysis models, In Proceedings of the ACL 2004on Interactive poster and demonstration sessions , Article 25
[25] E. Riloff, S. Patwardhan, J. Wiebe, Feature subsumption foropinion analysis , ACL, Proceedings of the 2006 Conferenceon Empirical Methods in Natural Language Processing, pp.440-448, 2006.20.
[26] X. Bai, Predicting consumer sentiments from online text,Decision Support Systems 50 (4), pp. 732-742, 2011
[27] A. Kennedy, D. Inkpen, Sentiment Classification of MovieReviews Using Contextual Valence Shifters, ComputationalIntelligence, Vol. 22, No. 2 . pp. 110-125. 2006.
[28] B. Agarwal, N. Mittal, Sentiment Classification using RoughSet based Hybrid Feature Selection, WASSA 2013: 4thWorkshop on Computational Approaches to Subjectivity,Sentiment and Social Media Analysis
[29] A. Konig, E. Brill, Reducing the human overhead in textcategorization, In Proceedings of the 12th ACM SIGKDDconference on knowledge discovery and data mining, pp.598-603, 2006
[30] www.kaggle.com/datasets/shaivyac/us-airline-dataset
[31] Kang, H. W., Chye, K. K., Ong, Z. Y., & Tan, C. W. (2021). The Science of Emotion: Malaysian Airlines Sentiment Analysisusing BERT Approach. Conference Proceedings: International

Conference on Digital Transformation and Applications (ICDXA2021). doi:10.56453/icdxa.2021.1013.

[32] Geetha, M. P., & Karthika Renuka, D. (2021). Improving the performance of aspect based sentiment analysis using fine-tunedBert Base Uncased model. International Journal of Intelligent Networks, 2, 64–69. doi:10.1016/j.ijin.2021.06.005

[33] T.U. Haque, N.N. Saber, F.M. Shah Sentiment analysis on large scale Amazon product reviews 2018 IEEE International Conf. On Innovative Research and Development, ICIRD 2018, Thailand (2018), pp. 1-6

[34] Azhar, A. N., & Khodra, M. L. (2020). Fine-tuning Pretrained Multilingual BERT Model for Indonesian Aspect-based SentimentAnalysis. 2020 7th International Conference on Advance Informatics: Concepts, Theory and Applications (ICAICTA).doi:10.1109/icaicta49861.2020.9428882.

[35] Chollet, F (2015) Keras. https://github.com/fchollet/keras.