

Hotel recommendation system based on cross-lingual feedbacks

Md. Fardin Rahman Ami
Dept. of CSE
Brac University
Dhaka
ID:20101549

Raufar Mostafa
Dept. of CSE
Brac University
Dhaka
ID:20101312

Maliha Jahan Maisha
Dept. of CSE
Brac University
Dhaka
ID:20101527

Taskia Siddika
Dept. of CSE
Brac University
Dhaka
ID:20301417

Abstract—The development of technology has revolutionized the decision-making process, including the selection of lodging. User-generated feedback and reviews play a crucial role in influencing the decisions of prospective visitors in the hospitality industry. These feedbacks are utilized by our system to generate appropriate recommendations for future customers. The foundation of the system is sentiment analysis, which involves categorizing reviews into positive and negative sentiments. A comprehensive preprocessing phase including data cleaning and feature extraction is implemented prior to training. The proposed system leverages a diverse set of models including k-Nearest Neighbors (KNN), Multinomial Naive Bayes (MNB), Logistic Regression, Decision Tree, Random Forest, Support Vector Classifier (SVC), AdaBoost, Multi-Layer Perceptron (MLP), Attention-based Long Short-Term Memory (ATTENTION-LSTM), Recurrent Neural Network (RNN) with LSTM and Convolutional Neural Network (CNN) with (LSTM). Each model was individually trained on a scrupulously preprocessed dataset, which included extensive data modification and feature extraction. The RNN-LSTM model emerges as the leader, achieving the maximum level of accuracy among the models. This demonstrates the capability of deep learning techniques to capture complex dependencies in textual data. The results of this initiative will have a substantial impact on the hospitality and tourism industries.

Index Terms—Sentiment Analysis, Tourism, CNN-LSTM, Hotel reviews, Machine-Learning, RNN-LSTM, Deep-Learning

I. INTRODUCTION

In the current digital planet, online evaluations exert an unprecedented amount of impact over consumer decisions, thereby rewriting the hospitality industry. Individuals readily share their experiences on platforms such as hotel websites, Facebook, and YouTube due to ubiquitous internet access. The effect of these feelings on online reservations is substantial. Sentiment Analysis (SA) with Machine Learning (ML) is used to generate a pattern of the deluge of written criticism. Our custom-tailored Natural Language Processing (NLP) model for SA decodes positive and negative sentiments from reviews. Our method utilizes a pipeline that transforms unprocessed text via data cleansing, tokenization, and vectorization. Then, Machine Learning and Deep Learning fuel classifications based on particular requirements. By automating sentiment extraction, our method bridges language nuances and fosters a robust framework for sentiment analysis by analysing with different models in total of 11 models including both neural network models like, CNN-LSTM, RNN-LSTM also

non-neural models likewise, decision-tree, KNN, MNB etc. This inspires the hotel industry to strategically respond to consumer perceptions in the dynamic digital era, improving growth.

II. RESEARCH METHODOLOGY

A. Dataset Collection

The dataset comprises diverse reviews encompassing a broad spectrum of guest perspectives. The data was scraped from Kaggle which contains 515,000 customer reviews and scoring of 1493 luxury hotels across Europe taken from Booking.com [2]. Written in a unique blend of formal and informal tone, it mirrors the contemporary linguistic practices of native speakers. Our dataset comprises reviews both positive and negative, where positive review holds 70% of the total and 30% being negative. We have split our dataset in 80-20 ratio for train-test purpose. This eclectic collection of text-based data forms the foundation for training and evaluating our sentiment analysis models, enabling accurate classification of sentiments as positive or negative, ultimately driving deep insights for effective decision-making in the hotel industry.

B. Data Preprocessing

Preprocessing means bringing the text into a more useable form and making it ready to be fed into the model by splitting a sentence into a list of words. When it comes to data preprocessing, there are many useful techniques. Specifically in this paper, tokenization is the first step in preprocessing. Subsequently, the removal of stop words, commonly used across languages, is implemented as they hold minimal significance in natural language processing. For English, words like "is", "the", and "and" fall under the category of stop words as well as punctuation marks, emoticons, pictorial icons, random words, single alphabets, and HTML tags which are removed from the review text which help in noise reduction of the data. Next comes the process of transforming a word into its root or lemma called lemmatization. This process enhances semantic consistency, converting words like *swimming* to *swim*, *was* to *be* and *mice* to *mouse*. To facilitate these operations, the system leverages *SpaCy*, a versatile open-source Python toolkit designed for advanced natural language processing

tasks. *SpaCy* efficiently executes tokenization, lemmatization, and stop word removal, streamlining preprocessing activities, particularly within the domain of English language processing.

TABLE I
DATA PREPROCESSING

<i>Before Processing</i>	<i>After Processing</i>
Well located mind taking train. The bathroom crystal walls comfortable. The room size also pretty small.	well located mind taking train bathroom crystal walls comfortable room size also pretty small
All staffs are friendly, helpful. The food is excellent great choice. The rooms are lovely well.	staffs friendly helpful food excellent great choice rooms lovely well

C. Feature Extraction

In the initial phase, all features except the text and label columns were retained. The text data, being natural language, often lacked proper formatting. It contained elements like HTML tags, punctuation marks, diverse symbols, and stop-words. These elements hold negligible semantic significance within sentences and have limited impact on stress and non-stress classification. To address this, a preprocessing step was undertaken to normalize the text and eliminate unnecessary symbols, resulting in a simplified form containing only words. Lemmatization was employed to obtain word root forms, enriching the text's consistency. To integrate this processed data into the model, the words needed to be transformed into a format that the model could comprehend. Various word embedding techniques exist, including word2vec, TF-IDF, countvectorizer, and rapid text. Among these, we explored the use of word2vec and countvectorizer methods. However, word2vec's negative value vector conversion posed challenges, making it less adaptable for certain models. Consequently, word2vec embedding yielded less satisfactory outcomes. As an alternative, countvectorizer was selected for word embedding assignments, facilitating subsequent comparisons.

D. Word representation

Since computers do not understand words or their context, it is needed to convert text into the appropriate, machine-interpretable form. The aforementioned representation refers to vectors that are spatially arranged in a manner that vectors in close proximity exhibit greater semantic similarity by mathematical representation.

1) *Count Vectorizer*: Count Vectorizer is a method that generates a matrix in which each distinct word is denoted by a column, and each text sample from the document is represented as a row. The purpose of this method is to convert a given text into a vector representation based on the frequency of each word in the entirety of the text. The present study employs the vector representation as a characteristic. Count vectors were retrieved from the dataset and subsequently utilized as input for many models.

TABLE II
WORD PREPROCESSED

<i>Reviewed text</i>	<i>Preprocessed text</i>
Rude people. Do not stay, and despite the fact cool hotel, the place sucks, rudest people, am disappointed	"rude", "people", "stay", "despite", "cool", "hotel", "place", "suck", "rude", "people", "disappointed"
Enjoyed the hotel. Location and service costs are excellent, good room. Recommend	"enjoy", "hotel", "location", "service", "cost", "excellent", "good", "room", "recommend"
Great location, clean comfortable, close pike. Market in walking distance, breakfast nice and fresh	"great", "location", "clean", "comfortable", "close", "pike", "market", "walking", "distance", "breakfast", "nice", "fresh"

2) *Global Vector Model (GloVe)*: The Global Vector model is a computational framework that represents words in a distributed manner. The algorithm under consideration is an unsupervised learning technique that integrates the characteristics of two distinct model families, specifically the global matrix factorization and local context window approaches. The GloVe model is a log-bilinear model that utilizes a weighted least-squares goal function during training. This objective function is applied to a global word-word co-occurrence matrix. The matrix shown illustrates the probability of co-occurrence between terms within a certain corpus.

E. Word Embedding

Converting raw text into a numerical form is crucial for training learning models. Word embedding (Mandelbaum and Shalev, 2016) is the contemporary choice for this purpose, transforming text into vectors while capturing word relationships [1]. In our sequential model, the first layer is an embedding layer that converts user reviews into trainable vector format. The embedding dimension is 100, and the maximum sequence length is set according to the longest review.

III. MODEL IMPLEMENTATION

A. Sentiment analysis using machine learning

Classifiers:

For our proposed system we have used several Machine Learning models and some deep learning models to classify the data. As for the machine learning models we have opted to use k-Nearest Neighbors (KNN), Multinomial Naive Bayes (MNB), Logistic Regression, Decision Tree, Random Forest, Support Vector Classifier (SVC), AdaBoost, Multi-Layer Perceptron (MLP) and for the models for deep learning we have used neural network models like Attention-based Long Short-Term Memory (ATTENTION-LSTM), Recurrent Neural Network (RNN) with LSTM and Convolutional Neural Network (CNN) with (LSTM).

B. Model Analysis

After preprocessing the data by tokenizing and lemmatization, we used the count vector feature to modify the data used in the machine learning models and gloVe for the data

to be fed in the deep learning models. We have found quite comparatively well accuracy and results from the Logical Regression and Random Forest model from the machine learning side but overall, the best results were from the neural network side which used the RNN-LSTM model. We have trained the model 15 epochs with Adam (learning rate 0.001) as a optimizers and loss function as binary crossentropy. Then we have achieved 96.02% accuracy.

The table 1 presents performance metrics for various machine learning and deep learning models used in the study. These metrics evaluate the effectiveness of each model in categorizing reviews into positive and negative sentiments.

TABLE III
MODEL PERFORMANCE METRICS

Model Names	F1 Score Positive (%)	F1 Score Negative (%)	Accuracy (%)
KNN	94	74	89.77
MNB	96	85	93.69
Logistic Regression	96	87	94.3
Decision Tree	94	77	89.94
Random Forest	96	86	93.92
SVC	96	86	94.11
AdaBoost	93	69	88.14
MLP	96	84	93.16
Attention-LSTM	97	89	94.89
RNN-LSTM	97	91	96.02
CNN-LSTM	97	90	95.5

C. Model Training

To start our study, we carefully chose several traditional machine learning models known for their distinct abilities. Our aim in using these models was two-fold. Initially, we expected these models to provide useful baseline results that could act as reference points for assessing more advanced models. Secondly, we aimed to identify any inherent patterns or tendencies in the data that could guide us toward the most effective modeling approach. The unique feature of attention mechanisms, allowing the model to focus on specific parts of the input data, could be crucial in identifying subtle cues related to human stress levels. Additionally, the use of pre-trained GloVe embeddings to align the dataset meticulously within the framework of deep learning models. Both the RNN and CNN used the Long Short-Term Memory (LSTM) architecture, which included components of the Convolutional Neural Network (CNN) into the LSTM model. The training methodology used for each of these deep learning models included 15 epochs, with a batch size of 32.

IV. RESULT ANALYSIS AND DISCUSSIONS

Non-neural models were first selected due to their adaptability, since they include a broad spectrum of machine learning methods, which include both linear and non-linear approaches. This enabled us to examine several approaches for predicting reviews and get a holistic understanding of the issue. Moreover, these models have gained significant recognition in the respective discipline, providing a robust basis for our

preliminary inquiry. However, we recognized the potential for improvement by incorporating neural models, particularly those with attention mechanisms. Neural models, known for their ability to capture intricate patterns in complex datasets, are a valuable asset. The unique feature of attention mechanisms, enabling models to focus on specific data elements, holds promise for identifying subtle cues in sentiment analysis. Neural models' adaptability, ability to prioritize relevant information, and capacity to uncover hidden features make them an essential addition to enhance our performance in predicting positive and negative reviews. From Fig1, MNB, Random Forest, Logistic Regression, SVC and MLP shows drastic high score for the F1 scores of both positive and negative reviews among the basic 8 ML models that we have been implemented. In addition to that, in Fig2 the comparison between deep learning models has been plotted with F1 scores of both positive and negative reviews.

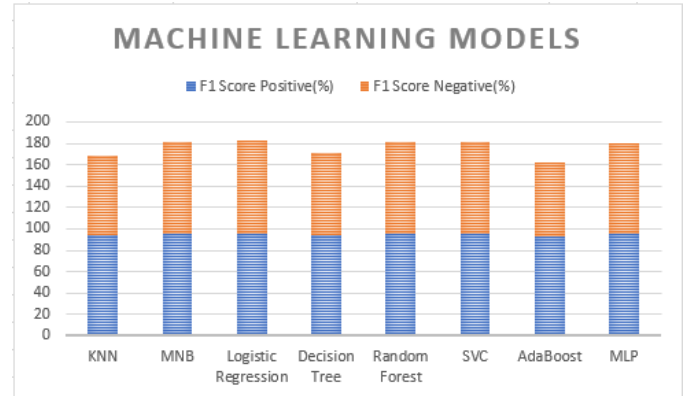


Fig. 1. Comparisons of outcomes among Machine Learning Models

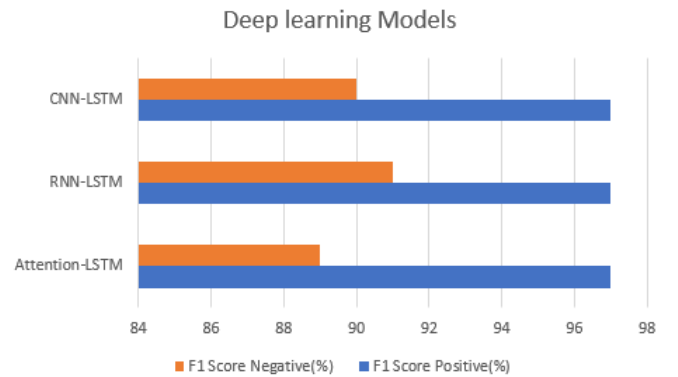


Fig. 2. Comparisons of outcomes among Deep Learning Models

A. Comprehensive Model Comparison and Analysis

Given is comparative comparison of performance indicators for several machine learning models in three categories: F1 Score Positive (%), F1 Score Negative (%), and Accuracy (%). Initially, it is evident that the models demonstrate a notable

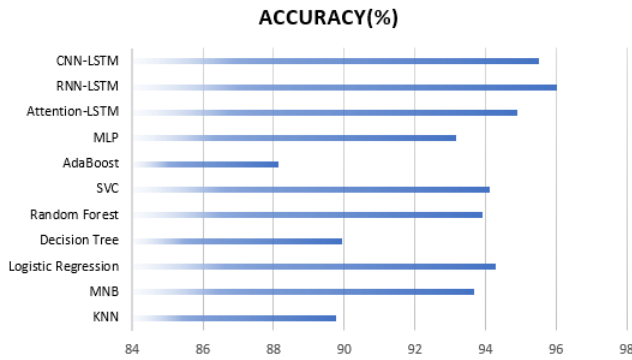


Fig. 3. Comparisons of outcomes among Deep Learning Models

level of proficiency, as shown by F1 scores over 90% continuously for the positive class. It is worth mentioning that both Attention-LSTM and RNN-LSTM exhibit exceptional F1 scores in the negative class, namely 97% and 91% respectively. This indicates their notable proficiency in detecting negative outcomes. Regarding the overall accuracy, the models demonstrate commendable performance, as Logistic Regression, SVC, RNN-LSTM, and CNN-LSTM achieve accuracy scores beyond 94%. Once again, the RNN-LSTM model demonstrates its superiority by achieving an amazing accuracy rate of 96.02%, while the Attention-LSTM model closely trails behind with a commendable accuracy rate of 94.89%. Upon comparing the F1 scores of the positive and negative classes, it becomes apparent that the majority of models exhibit a somewhat equitable performance.

However, a subset of models, including AdaBoost and KNN, have a propensity towards predicting the positive class, resulting in much lower F1 scores for the negative class. When examining the trade-off between accuracy and recall, as measured by F1 scores, both Logistic Regression and Random Forest demonstrate a satisfactory equilibrium in accurately recognizing positive and negative examples. This is seen in their balanced F1 scores across both classes. Additionally, it is worth mentioning that the Multi-Layer Perceptron (MLP) has a notable F1 score of 96% for the positive class, while exhibiting a comparatively lesser score of 84% for the negative class. This observation implies that the MLP excels in accurately recognizing positive situations. When considering model complexity and interpretability, it is seen that simpler models such as Multinomial Naive Bayes and Decision Tree provide satisfactory performance, but not as remarkable as more sophisticated models like LSTMs. One example of a classification algorithm, Naive Bayes, obtains an accuracy of 80.48%, whilst another algorithm, Decision Tree, has a higher accuracy rate of 89.94%.

Models such as RNN-LSTM and CNN-LSTM demonstrate superior performance in terms of generating high F1 scores and accuracy, simpler models such as Logistic Regression and Random Forest provide a well-balanced level of performance.

B. Model Selection

The CNN-LSTM and RNN-LSTM models have shown superior performance in the context of sentiment analysis tasks using textual input. These models have repeatedly shown outstanding performance across several measures, making them very suitable for this particular application. Both the CNN-LSTM and RNN-LSTM models demonstrated exceptional performance in terms of F1 scores for both positive and negative classes. Notably, the RNN-LSTM model outperformed its counterparts by achieving an amazing F1 score of 91% specifically for the negative class.

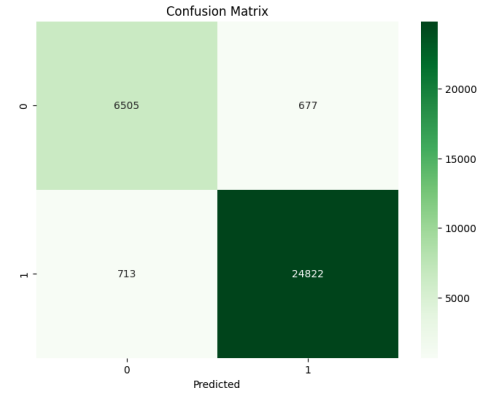


Fig. 4. RNN-LSTM Confusion Matrix

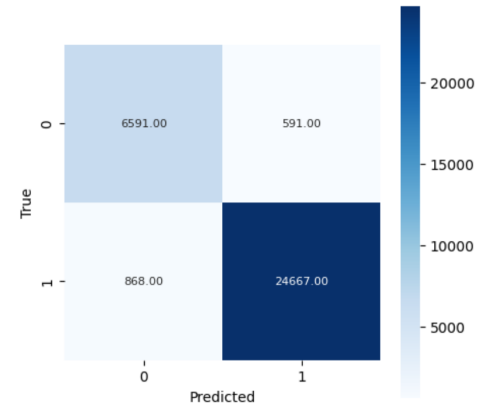


Fig. 5. CNN-LSTM Confusion Matrix

This demonstrates their aptitude in discerning positive and negative attitudes in textual data, underscoring their appropriateness for sentiment analysis tasks that need an equitable categorization of sentiments. Moreover, in relation to the overall accuracy, the RNN-LSTM model exhibited superior performance compared to all other models, with an accuracy rate of 96.02%. The CNN-LSTM model closely followed with an accuracy rate of 95.5%. The aforementioned accuracy rates demonstrate a remarkable proficiency in accurately categorizing feelings within textual material, a critical factor for the efficacy of sentiment analysis applications. The delicate trade-off between accuracy and recall has significant importance in

sentiment research. These models demonstrate a commendable equilibrium, making them dependable options for collecting nuanced thoughts. The strong performance shown by the system in this aspect is especially advantageous for sentiment analysis tasks that need a significant level of precision in sentiment categorization.

The use of CNN-LSTM and RNN-LSTM models in sentiment analysis is justified due to their established efficacy, particularly in cases where precise sentiment categorization has significant significance. These models possess the capability to effectively manage the complexities inherent in text input and extract significant sentiment information, making them the favored options for constructing a sentiment analysis model.

V. FUTURE WORKS

In the realm of future endeavors, a significant avenue lies in the development of a specialized dataset that facilitates cross-lingual learning. This innovative dataset would encompass diverse languages, enabling the sentiment analysis model to transcend language barriers and extract sentiment nuances across different linguistic landscapes. By engaging in cross-lingual learning, the model could gain a more comprehensive understanding of the universal aspects of sentiment while accommodating the intricacies that arise from linguistic variations.

Furthermore, the integration of BERT (Bidirectional Encoder Representations from Transformers) stands as a promising augmentation strategy. BERT, renowned for its contextual embedding capabilities, could profoundly enhance the model's comprehension of context-dependent sentiment cues. By assimilating BERT into the existing architecture, the sentiment analysis model could capitalize on its proficiency in capturing intricate contextual relationships, potentially leading to a re- refined accuracy in sentiment classification.

The integration of cross-lingual learning and BERT has the potential to provide a novel phase of sentiment analysis, marked by a broader range of languages and an increased awareness of contextual factors. This all-encompassing strategy could enable a model to flourish in understanding and translating sentiment in a variety of languages, dialects, and socioeconomic settings, resulting in more precise and subtle sentiment analysis results on a global basis.

VI. CONCLUSION

The findings of our research resulted in beneficial results. Through the utilization of advanced deep learning architectures, the system effectively processed and understood feedback from several languages, leading to enhanced levels of precision and reliability in hotel suggestions. The RNN-LSTM and CNN-LSTM models exhibited their adeptness in capturing the subtleties of customer input, hence enabling the system to effectively identify emotions and thoughts across several languages. The sentiment analysis task exhibited

a notable level of accuracy, as evidenced by the constant attainment of F1 scores over 90% by both models. One notable advantage of the recommendation system is in its capacity to offer tailored hotel recommendations by leveraging sentiment analytics. Consequently, this feature aids passengers in making well-informed choices. The use of deep learning models, namely RNN-LSTM and CNN-LSTM, enhances the system's ability to comprehend the sentiment-laden characteristics of client feedback, thus enhancing the overall quality of hotel suggestions. This method essentially offers a helpful tool for tourists looking for lodgings by utilizing the strength of cross-lingual sentiment analysis.

REFERENCES

- [1] Mandelbaum, R., Shalev, N. (2016). Word embedding clustering and similarity. arXiv preprint arXiv:1612.02796.
- [2] <https://www.kaggle.com/code/abdallahmohamedamin/sentiment-analysis-using-cnn-lstm-and-cnn-lstm>