Journal of Ubiquitous Computing and Communication Technologies (ISSN: 2582-337X)
www.irojournals.com/jucct/

Integration of Sentiment Analysis and Speech Text Processing in Phonetic Flow System

Vandana[1], Ritik Rana[2], Akhilendra Khare[3], Subash Harizan[4]

[1,2]Department of CSE, Chitkara University Institute of Engineering and Technology, Punjab, India.
[3]Department of CSE, Galgotias University, Noida, India.
[4]Department of CSE, SRMIST, NCR Delhi, Gaziabad, India.

E-mail: [1]vandana.bajaj@chitkara.edu.in, [2]ritik7094.ca23@chitkara.edu.in, [3]a.khare0007@gmail.com, [4]subashharizan@gmail.com

Abstract

Human-computer interaction (HCI) applications increasingly rely on reading speech, understanding emotional context, and generating natural language. The heterogeneous set of approaches the existing solutions use for sentiment analysis, speech synthesis, and speech recognition results in an unbalanced user experience. Designing an integrated system that can perform speech-to-text (STT) and text-to-speech (TTS) processing, sentiment analysis of input text, and neighboring aware speech generation is the problem this paper attempts to solve. To identify complexity like sarcasm and negation, Bi – LSTM is selected because it can learn context from context words (previous and next words in a sentence). In spite of data sparsity conditions, GloVe embeddings improve model generalisation by offering deep semantic understanding from large corpora. Following experimental verification, our Bi-LSTM with GloVe embeddings achieves 90% sentiment classification accuracy that is 7-10% higher relative to standard baselines like SVM (82%) and Naïve Bayes (75%). With true positive values above 88%, the model achieves well-balanced performance on the positive, neutral, and negative classes. Due to its low latency and about 87% accuracy during live testing, the system is an excellent option for interactive systems. All these features are amalgamated in our Phonetic Flow System, which enhances them to develop an extensible system that supports quicker, more natural, and emotionally intelligent human-machine interaction.

Keywords: HCI, STT, TTS, Bi-LSTM.

DOI: https://doi.org/10.36548/jucct.2025.4.001
Received: 12.09.2025, received in revised form: 11.10.2025, accepted: 25.10.2025, published: 04.11.2025

1. Introduction

Nowadays, human computer interaction increasingly relies on speech and language technologies to enable systems to understand, respond, and interact with users in a more context-aware, intelligent and human-like way. Essentially, any field that demands precise communication accompanied by understanding of the context may benefit from this approach. These include medicine, education, assistive technology, customer service and many other areas. The Phonetic Flow System embodies and addresses these areas, combining sentiment analysis and text-to-speech synthesis with speech-to-text transcription in a single, modular platform. This system effectively operates by combining emotional intelligence with natural language comprehension: it accurately understands spoken language, determines the polarity of the mood and answers with a voice that mimics its human counterpart. The platform makes use of state-of-the-art deep learning methods, particularly pre-trained GloVe embeddings and Bi-Directional Long Short-Term Memory networks; it demonstrates high performance in processing complex linguistic and emotional input. It allows for modular deployments, maintains real-time interactions, and connects effortlessly to other conversational AI systems. A pipeline operating in real-time utilizing three stages, voice recognition, sentiment analysis, and speech synthesis, so-called Phonetic Flow System, sees precedence to coincide with. The most relevant argument to establish our emotionally intelligent system type as inviable compared to previous methods would be the unified nature. The design, implementation, and evaluation of the system are disseminated in this paper. The results indicate that our method enhances multimodal engagement and suggests its appropriateness for further extension and broad usage, including transformer-based sentiment modelling and even multilingual usage. Through a single or unified framework, sentiment classification, STT, and TTS were accomplished, resulting in an achievement of almost 90% accuracy on traditional baselines using Bi-LSTM + GloVe. Bi-LSTM is beneficial over CNN for capturing sequential text dependencies and is measurably applicable in applications that require accuracy combined with usability.

2. Related Work

To overcome the difficulties of automated speech recognition (ASR), numerous studies have used a variety of language modelling and acoustic processing techniques. Conventional methods used Gaussian Mixture Models (GMMs) and Hidden Markov Models

(HMMs) [5], which needed a lot of feature engineering and didn't work well in noisy or changing situations. By directly learning hierarchical and temporal patterns from raw or minimally processed input, deep learning models like Convolutional Neural Networks (CNNs), Recurrent Neural Network (RNNs), and especially Long Short – Term Memory (LSTM) and Bidirectional LSTM (Bi – LSTM) networks have greatly increased the accuracy of voice recognition [6]. Text – to – Speech (TTS) systems have been transformed concurrently by developments in natural voice synthesis. Through the modelling of prosody, rhythm, and intonation, deep learning – based techniques like Tacotron [7], WaveNet [8], and FastSpeech [9] produce speech outputs that are incredibly expressive and lifelike. These advancements have improved human – computer interaction in a variety of applications, from accessibility tools for those with vision or speech impairments to digital assistants. Similarly, sentiment analysis has progressed from utilising traditional machine learning methods like Logistic Regression, Naïve Bayes, and Support Vector Machines (SVMs) to more advanced deep learning architectures. Even when there is sarcasm, ambiguity, or informal language present, these more recent methods – such as Bi – LSTM, transformer models (like BERT [10] and RoBERTa [11]), and attention mechanisms – are able to classify sentiment more accurately because they have a deeper understanding of the syntactic structures and semantic relationships in the text. By combining sentiment analysis, text- to-speech, and speech-to-text into a single software framework, this work expands on existing technological developments. The suggested platform provides a smooth transition from audio input to textual emotional interpretation and back to audio output, in contrast to many systems that handle these elements separately. An end – to – end pipeline like this improves usability and engagement, making appropriate for real-time applications such as assistive technologies, emotion – aware systems, and conversational agents. The integration showcases the possibilities of deep learning – powered multimodal human – computer interaction in addition to the synergy between separate components.

2.1 Problem Statement

There is currently no integrated method that effectively manages all three tasks; instead, existing systems frequently focus on either voice recognition, speech synthesis, or sentiment analysis. Using a Bi – LSTM model, this study aims to increase sentiment analysis accuracy while creating a single system that combines these features. This clear differentiation

from currently available disjointed solutions highlights the originality and usefulness of the suggested method.

3. Proposed Work

3.1 System Architecture

- The system was developed using the Flask framework for Python [12], comprising online interfaces and APIs for speech-to-text, text-to-speech, and sentiment analysis modules.
- Speech to text: The Python library SpeechRecognition [13] uses the Google Speech Recognition API [14].
- Text-to-speech makes use of Pyttsx3 [15] for offline speech synthesis. Sentiment analysis is performed using a deep learning model Bi-LSTM that is pre-trained using GloVe word embeddings [2] and trained on text input.

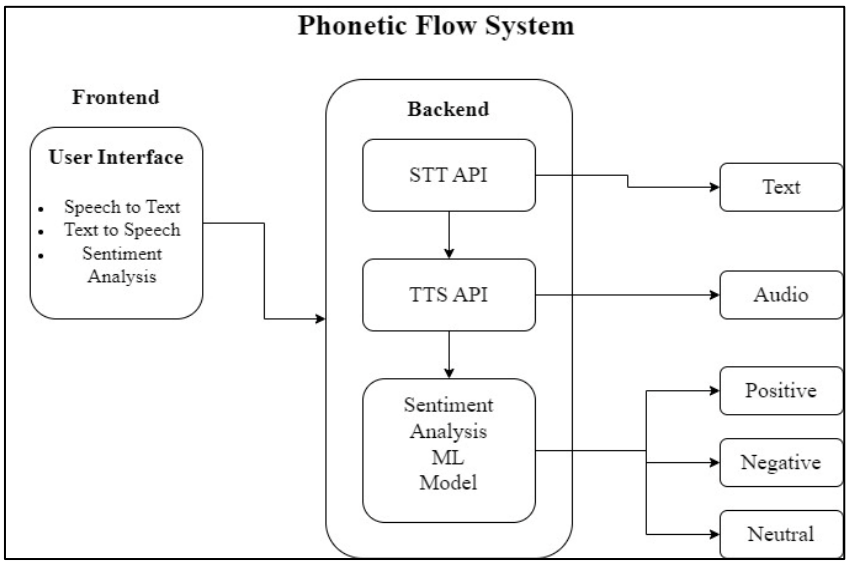

Figure 1. End-to-End Workflow from User Input to Text, Audio, and Sentiment Output

3.2 Dataset Description

For training and testing, we used the Twitter Entity Sentiment Analysis Kaggle Dataset. The dataset contains over 60,000 labeled samples with sentiment labels such as Positive, Negative, Neutral, and Irrelevant across various themes. We were interested in the

positive, negative, and neutral classes for the classification of our system. Duplicate entries and entries with null values were removed, and preprocessing techniques such as removing URLs, lowercasing, expanding contractions, tokenizing, stemming, and handling stopwords were performed. The dataset was split into an 80/20 train-test split. Padding at 167 characters guaranteed fixed-length inputs, and label encoding changed categories into numerical representations.

3.3 Sentiment Analysis Model

The Bi-LSTM model classifies sentiment into three categories, namely positive, negative, and neutral, by processing input text sequences using embedding, bidirectional LSTM layers, and fully connected layers. The dimensions of the pre-trained GloVe vectors used for embedding are 300.

Justification: Bi-LSTM was chosen over CNN because it captures the bidirectional context and sequential dependencies in words, which are crucial in emotion tasks such as sarcasm or negation recognition. However, this choice has its own set of disadvantages. First, GloVe embeddings are static and cannot change the meaning of words based on the surrounding context, making them less powerful than contextual embeddings like BERT. Bi-LSTM models also require longer training cycles and more computation than simpler models, which may restrict scalability. Moreover, the pre-trained vectors of GloVe are mainly based on English, limiting the potential of the system in adapting to multilingual and dialectal data. These points highlight the various directions in which future development should be carried out in transformer-based systems.

- Model parameters include a maximum input length of 167 tokens, an output size of three classes, and a hidden dimension of 64.
- Training: The model was trained using cross-entropy loss and later fine-tuned for classification accuracy.

3.4 Data Preprocessing

The dataset was preprocessed systematically for its suitability for sentiment analysis. The feature text, which contains the visual text, was reformatted. Its column names were changed to feature underscores and converted to lowercase for ease of reference. Text with underlying meanings, as well as null-valued rows in the review text, were purged; additional

missing values were checked and removed as per the context of the data. The process to regularize the text data, composed of HTML tags, emoticons, URLs, and various symbols, was performed while fulfilling the regular expression. Remaining English contractions and words were expanded and converted to lowercase. The text was divided into tokens, a process known as tokenization. Each token was stemmed to its root, and in many cases, words and phrases were removed when necessary, while stop words that carry content were retained. In this dataset, all other data was reshaped and processed, tokenized into individual units, and then padded as required; sentiment labels were converted to numeric values to facilitate model training and make processing clear and helpful.

4. Implementation

It is implemented as a Flask web application [18] that provides a set of RESTful API endpoints for speech and sentiment analysis tasks. This is employed because it is lightweight, flexible, and easy to work with regarding machine learning models. Some of the functionalities that the user-friendly interface of this application allows interaction with are text-to-speech synthesis, sentiment analysis, and speech transcription. Various models and utility functions operating on audio or textual data in real time are integrated into the backend. The basic functionality of the application is briefly represented by three important routes listed in Table 1.

- **Transcription:** It expects a POST request with an attached mp3 or wav file. This will run that file through the speech recognition model-which at this point is the Google Speech API-that converts spoken language into plain text. This endpoint lets the user convert speech inputs into text data, which can then be processed or analyzed as desired.

- **TTS:** The TTS engine, like pyttsx3 or gTTS, synthesizes the provided text into an audio file upon receiving a raw text input via a POST request. This approach can be used to convert written text to spoken speech in accessibility applications, voice assistants, and language learning programs.

- **Predict:** It classifies the sentiment of the text provided by a user for any POST request into neutral, negative, or positive using a pre-trained BiLSTM model. Preprocessing, tokenization, and padding are performed on the input text before

vectorization for use by the model for prediction. The BiLSTM model captures the context in both directions to improve the accuracy of the sentiment analysis.

Table 1. Overview of System API Endpoints and their Functions

Endpoint	Method	Functionality
transcribe	POST	Converts uploaded audio file into transcribed text
text-to-speech	POST	Synthesizes audio output from input text
predict	POST	Performs sentiment analysis on user-provided text

5. Simulation Results

The simulation was executed using the Python software [19] The speech-to-text module does well when it comes to turning spoken words into text. For precise and instantaneous transcription, it uses the Google Speech Recognition API and supports a variety of audio formats, including .wav and mp3. After using approaches for ambient noise control, the system is able to successfully record and convert a variety of voice inputs, including accented or noisy audio. Because of this, users can effectively communicate with the device using voice commands or recorded inputs. Alternatively, the text-to-speech (TTS) module uses user-supplied text to generate natural – sounding audio output. With consistent synthesis speed and clear articulation, the module enables English pronunciation using a TTS engine such as pyttsx3 or gTTS. Applications like voice assistants, reading assistance for the blind, and language learning resources can all make use of this essential component for improving accessibility. Both modules exhibit low latency and are appropriate for interactive or real – time use cases.

A Bidirectional Long Short-Term Memory (Bi-LSTM) neural network, which is trained using pretrained word embedding GloVe and preprocessed textual input, powers the sentiment analysis capability. The Bi-LSTM design greatly enhances the model's capacity to identify negations, sarcasm, and sentiment subtleties by capturing contextual information from both the words that come before and after a sentence.

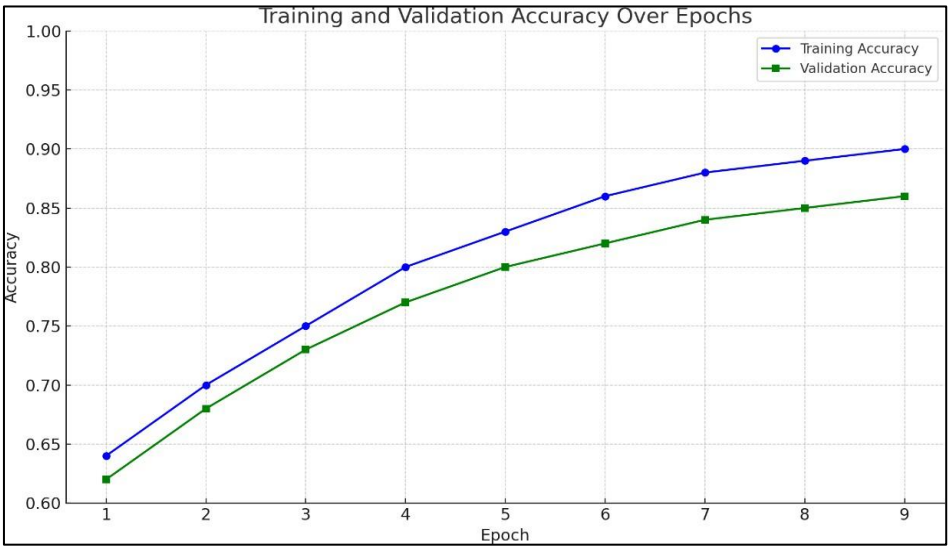

Figure 2. Training and Validity Accuracy by Varying Epochs

A labeled dataset that was split into training and test sets was used to assess the model. It successfully distinguished between positive, neutral, and negative sentiments with high classification accuracy. Figure 1 depicts the training and validation accuracy over several epochs, provides an illustration of the performance measures, and shows a consistent learning curve with little overfitting. The confusion matrix, as shown in Table 2, indicates low misclassification across all classes and high true positive rates, further supporting the model's predictive power.

Table 2. Performance of the Sentiment Classification Model Across Positive, Neutral, And Negative Classes

	Predicted Positive	**Predicted Neutral**	**Predicted Negative**
Actual Positive	88	6	6
Actual Neutral	4	91	5
Actual Negative	3	5	92

5.1 Comparative Evaluation

Table 3. Comparative Evaluation of Sentiment Analysis Models, Highlighting Accuracy, Strengths, and Weaknesses

Model	Accuracy	Strengths	Weaknesses
Naïve Bayes	75%	Simple, Fast	Poor with negations
SVM	82%	Strong Baseline	Lacks deep context
Bi-LSTM + GloVe	90%	Contextual, robust, low latency	Slightly higher training cost

5.2 Error Analysis

- Sarcasm sentences (e.g., “Oh great, another delay”) are mistakenly categorised as positive.
- Neutral or slightly positive texts (e.g., “It’s okay”) are not consistently categorised.
- Accuracy is decreased by code-switching and dialectal variances.

5.3 Real-Time vs Dataset Performance

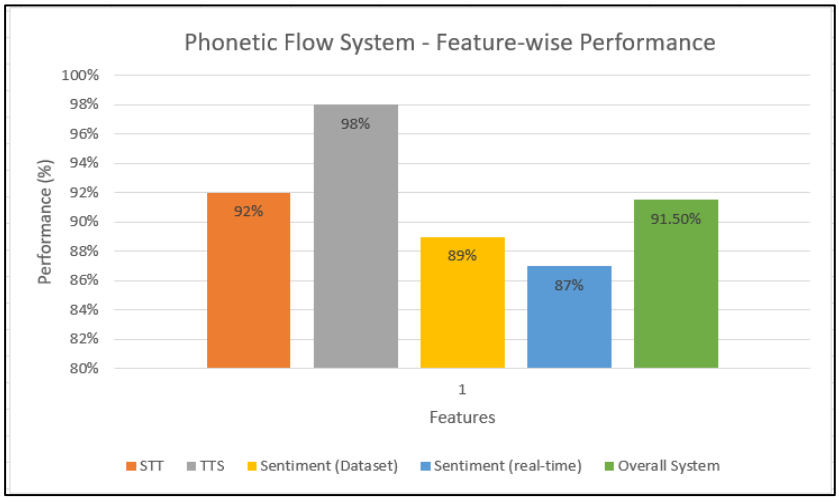

Figure 3. Performance based on Features

The real-time model's accuracy was about 87%, whereas the dataset's accuracy was 90%. This discrepancy can be explained by the fact that the dataset provides entire sentences for evaluation, enabling the model to fully capture contextual information and attain higher

accuracy, whereas the real-time system evaluates sentiment word-by-word and aggregates the results, sometimes resulting in neutral or intermediate classifications.

As demonstrated in Figure 3, the results validate that Bi-LSTM and pretrained word embedding complement each other well, allowing the model to generalize successfully even on complex sentences containing a variety of emotions. The remarkable efficacy of the model in every sentiment category further highlights its usefulness in real-world applications like conversational AI systems, social media monitoring, and customer feedback analysis.

6. Privacy and Ethical Considerations

The technology preserves user privacy by processing inputs instantly and not storing unprocessed audio or transcripts. Every dataset is anonymized and accessible to the general public. Preprocessing and balanced class representation help to reduce potential biases. Instead of being used for monitoring, the architecture is meant for constructive uses like assistive technologies and accessibility. System flaws like decreased accuracy in noisy or dialectal environments are highlighted to preserve transparency.

7. Conclusion

Therefore, the Phonetic Flow System is a single package that integrates sentiment analysis, speech-to-text, and text-to-speech to enable intelligent user-machine interaction. An on-the-fly review of the Google Speech Recognition API and TTS engines powers bidirectional performance. The sentiment analysis subsection consists of a Bidirectional Long Short-Term Memory neural network with pre-trained GloVe embeddings that achieve approximately 90% classification accuracy, outperforming other widespread techniques, including SVM and Naïve Bayes. In simple words, it incorporates all the top AI ethics behaviors into a single real-time process. Desirable future modifications include working with multiple languages and dialects, conducting sentiment analysis in real-time during conversation, and improving user performance with multimodal analysis, including visual factors. This is a modular architecture with numerous APIs that can be incorporated into software such as chatbots or learning programs.

References

[1] J. Patel, “Twitter Entity Sentiment Analysis,” Kaggle, 2020. [Online]. Available: https://www.kaggle.com/datasets/jp79 7498e/twitter-entity-sentiment- analysis [Accessed on 22-11-2024]

[2] Pennington, Jeffrey, Richard Socher, and Christopher D. Manning. "Glove: Global vectors for word representation." In Proceedings of the 2014 conference on empirical methods in natural language processing (EMNLP), 2014, 1532-1543.

[3] Graves, Alex, and Jürgen Schmidhuber. "Framewise phoneme classification with bidirectional LSTM and other neural network architectures." Neural networks 18, no. 5-6 (2005): 602-610

[4] https://pypi.org/project/SpeechRecogn%20ition/

[5] McKinney, Wes. "Data structures for statistical computing in Python." scipy 445, no. 1 (2010): 51-56.

[6] https://pypi.org/project/pyttsx3/

[7] https://pypi.org/project/gTTS

[8] PyTorch, “An open source machine learning framework.” [Online]. Available: https://pytorch.org

[9] Harris, Charles R., K. Jarrod Millman, Stéfan J. Van Der Walt, Ralf Gommers, Pauli Virtanen, David Cournapeau, Eric Wieser et al. "Array programming with NumPy." nature 585, no. 7825 (2020): 357-362.

[10] Hugging Face, “Transformers: State-of-the-art Natural Language Processing.” [Online]. Available: https://huggingface.co/transformers

[11] Wang, Yuxuan, R. J. Skerry-Ryan, Daisy Stanton, Yonghui Wu, Ron J. Weiss, Navdeep Jaitly, Zongheng Yang et al. "Tacotron: Towards end-to-end speech synthesis." arXiv preprint arXiv:1703.10135 (2017).

[12] Oord, Aaron van den, Sander Dieleman, Heiga Zen, Karen Simonyan, Oriol Vinyals, Alex Graves, Nal Kalchbrenner, Andrew Senior, and Koray Kavukcuoglu. "Wavenet: A generative model for raw audio." arXiv preprint arXiv:1609.03499 (2016).

[13] Ren, Yi, Yangjun Ruan, Xu Tan, Tao Qin, Sheng Zhao, Zhou Zhao, and Tie-Yan Liu. "Fastspeech: Fast, robust and controllable text to speech." Advances in neural information processing systems 32 (2019).

[14] Devlin, Jacob, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. "Bert: Pre-training of deep bidirectional transformers for language understanding." In 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), 2019, 4171-4186.

[15] Bird, Steven, Ewan Klein, and Edward Loper. Natural language processing with Python: analyzing text with the natural language toolkit. " O'Reilly Media, Inc.", 2009.

[16] https://code.visualstudio.com

[17] Juhttps://jupyter.org

[18] https://flask.palletsprojects.com

[19] https://docs.python.org/3.8/