VOICE-ENABLED CUSTOMER ASSISTANCE KIOSK WITH INTENT CLASSIFICATION AND SECURE OTP AUTHENTICATION FOR BANKING

Dr. E. Terence
Department of Electronics and Communication Engineering
Hindustan Institute of Technology and Science Chennai, India
eterence@hindustanuniv.ac.in

Sworn Prabha Pradhan
Department of Electronics and Communication Engineering
Hindustan Institute of Technology and Science Chennai, India
swornprabha@gmail.com

Sam Leo Sahayaraj I
Department of Electronics and Communication Engineering
Hindustan Institute of Technology and Science Chennai, India
Samleo300305@gmail.com

A Santhosh
Department of Electronics and Communication Engineering
Hindustan Institute of Technology and Science Chennai, India
asanthosh085@gmail.com

Dr. Arikesh A
Department of Electronics and Communication Engineering
Hindustan Institute of Technology and Science Chennai, India
aarikesh@hindustanuniv.ac.in

Dr. B. Rajesh Shyamala Devi
Department of Electronics and Communication Engineering
Hindustan Institute of Technology and Science Chennai, India
rsyamla@hindustanuniv.ac.in

Abstract: ***Presented here is a kiosk for bank-related support that responds to spoken or written questions, combining intent detection through NLP with OTP verification. Voice entries reach text form first - achieved via Google's Speech-to-Text interface - before any analysis occurs. Initial handling splits input into parts, strips irrelevant words, reduces terms to root forms using Porter's method, then captures weighted term significance via TF-IDF. Classification relies on a LinearSVC model, shaped by exposure to 8,000 categorized examples across ten banking intentions. System behavior emerges from structured translation, filtering, weighting, and prediction stages woven together. Evaluation uses a separate test set containing 5,000 instances, divided by an 80/20 ratio. Accuracy reaches 91.67 percent, while precision - on a macro level - stands at 0.910. Recall measures slightly lower, at 0.908, yet supports stable detection across classes. The F1 metric settles near both, recording 0.910, indicating balanced performance. Contrasting methods reveals LinearSVC surpasses Naive Bayes, which scored 83.2 percent. It also exceeds Logistic Regression, limited here to 88.4 percent. Outcomes suggest efficient natural language processing systems may enable timely understanding of user goals.***

Keywords

Natural Language Processing; intent classification; TF-IDF; LinearSVC; speech recognition; OTP authentication; banking kiosk; voice interface.

I. INTRODUCTION

Where automation now runs across finance, demand grows for chat tools handling sudden, unclear questions right away. Years ago, bank machines used rigid menus - choices came from fixed lists - something often confusing people new to screens, particularly when rushed at busy times. Speaking seems easier than pressing buttons for numerous persons; however, putting voice systems into banks happens rarely because understanding varied intentions stays hard. How client's express needs changes widely between areas - a challenge current solutions tackle weakly although progress occurred. Even where rules guard money matters closely, blocking speech-based checks alone, slight progress appears if accuracy walks with caution. Yet confidence rises at a crawl - every step measured twice before taken.

Though transformer-based models like BERT and GPT excel at understanding user intent, their heavy computing needs make them poorly suited for kiosk devices at the network edge [1]–[3]. In contrast, simpler machine learning methods offer fast responses and stable resource use - traits useful for immediate, local processing [8]. Despite this, combining such efficient classifiers with voice interfaces and secure login routines inside one cohesive framework for banking services remains underexplored in research.
One more gap shows up in today's rollout setups - protection. Most talking bank tools run on fixed logins like PINs or passphrases, open targets for onlookers or recorded breaches. Even if double-check steps were tested alone [15], fitting them smoothly into spoken NLP engines inside one self-service unit remains largely unexplored. Security slips through when voice machines meet outdated entry

rules. This paper addresses these gaps by proposing and evaluating a voice-enabled customer assistance kiosk that combines a TF-IDF/LinearSVC intent classification pipeline with OTP-based session authentication. The specific contributions of this work are as follows:

(1) A collection of 8,000 labelled questions, focused on banking actions, forms the core. Ten distinct purposes guide its structure. Varied ways customers express needs shape each entry. This design reflects how people naturally ask about financial matters.
(2) A structure built as one cohesive unit operates through distinct modules. From this setup, voice transforms into written form before moving forward. Following that step, meaning is identified using language analysis methods. One-time codes verify identity during progression. Outputs form only after these stages complete. Deployment occurs once all components link seamlessly.
(3) A detailed comparison examines LinearSVC alongside Naive Bayes and Logistic Regression, using a separate test set containing five thousand instances. Performance broken down by category appears together with measures assessing whether differences are likely due to chance. Each model's output undergoes scrutiny under consistent conditions, ensuring results reflect actual behaviour rather than random variation. Evaluation includes formal checks for reliability across all categories involved.
(4) A test shows how a fast TF-IDF paired with LinearSVC reaches solid results. Efficiency gains appear when comparing run times against heavy transformer models. Despite simplicity, performance stays close to advanced methods. Speed matters where resources are limited. Accuracy does not always require complexity. Lower computation needs open doors for wider use. Trade-offs exist, yet they favour lightweight systems here.

II. LITERATURE SURVEY

Progress in understanding human language by machines has mainly come through designs built on transformers. Models such as BERT, GPT, and PaLM - given substantial size - perform well on many tasks involving written expression [1]–[4]. Work by Liu et al. [5] indicates strong precision when applying these structures to detect goals in conversations without predefined topics. Despite such capability, their operation requires significant graphical processor resources; furthermore, delays during output generation often exceed what interactive station devices can tolerate under live conditions.

Study into intent recognition for practical conversations used multiple methods. Notably,[6] Zhang and colleagues compared network types like convolutional and recurrent systems, finding hybrids combining CNNs with LSTMs work reliably on common data collections. Work by Chen together with Liu [7] supported such outcomes - hybrid deep networks outperformed older techniques especially when tested on extensive, evenly distributed datasets. Later, research led by Gupta [8] revealed simpler setups such as support vector machines using TF-IDF measures can match precision levels within focused domains, yet consume much less computing power. Such efficiency matters greatly under conditions resembling those faced in kiosk installations analysed here.

One way to sort chatbots divides them by method [9],[10] some pull answers from set lists, others build replies anew each time. Though fixed-response designs limit surprise outcomes, their outputs stay consistent - important when rules tightly govern operations. Flexible reply generators allow fluid talk, yet sometimes invent details without warning, raising concerns about reliability [11],[12]. Where financial services apply strict standards, predefined answer paths often fit better than free-form ones. A study examining real-world bank chat tools found repeated gaps in how safety measures connect with core functions [13].

Chatbot technologies have evolved in design and functionality. Existing systems are categorized into retrieval-based and generative models [9], [10]. Retrieval-based approaches provide more consistent responses but are not flexible. Generative models offer context-aware interactions but are more complex [11], [12]. Although recent developments have improved response generation and conversational quality challenges related to scalability, cost and real-time deployment persist [13].

Progress in speech handling for machine communication stands notable, as industry tools like Google's transcription service report inaccuracies under ten percent when identifying clearly spoken English [14]. Despite this, limited research examines how misrecognized words influence purpose detection precision specifically in financial services software.

Within chat-based platforms, confirming user identity draws notable focus - especially through voice patterns and layered checks. Though OTP methods are common in online banking, their role in spoken-language kiosks lacks structured assessment. Emphasis on verifying access during active sessions was noted by Richardson and Heck [15], especially when queries involve private data. Most live setups treat sign-in steps as entry points instead of blending them throughout interaction flows. Because integration remains shallow, continuity between validation and task execution often breaks down unexpectedly.

Present findings show missing elements in systems that link low-intensity text review, speech processing, along with tightly integrated verification steps inside automated money-handling stations. Addressing these void forms the core focus here. Prior attempts often study linguistic methods in isolation or depend on general-purpose datasets; instead, this effort traces a complete chain of actions using specific transaction-based dialog logs, employing precise comparative evaluation strategies.

III. PROPOSED SYSTEM

A. System Overview and Problem Definition

One goal stands clear: handle everyday bank tasks using normal speech or typed words. Whether asking about money available, moving funds between accounts, placing deposits, taking cash out, or checking past actions - voice or text works. A key difficulty lies in turning free-form human questions into exact banking purposes without delay. Accuracy must hold even as speed matters during live use. Protection of private steps within each conversation remains necessary throughout. Interpretation happens fast, yet stays precise under time pressure.

Driven by limits in today's kiosk systems, rigid menus force people through fixed paths. Because choices stack in layers, understanding them takes effort - more so for those less familiar with digital tools. These setups fail when speech patterns differ, even slightly, from expected inputs. Instead of structured prompts, freeform phrases become possible once language processing adapts to real-world wording. With comprehension rooted in common expressions, interaction shifts closer to normal conversation.

B. System Architecture and Design

A structure built in separate levels operates through six connected phases. Starting with gathering incoming data, it moves next to transforming spoken words into written form when needed. Following that comes refinement of the textual input to prepare it. Feature selection then occurs using TF-IDF methods for numerical representation. Classification of purpose follows, handled by a LinearSVC model. Finally, replies are formed - guarded by one-time passcodes if actions involve private access. Shown visually in Fig. 1. is this arrangement laid out as blocks.

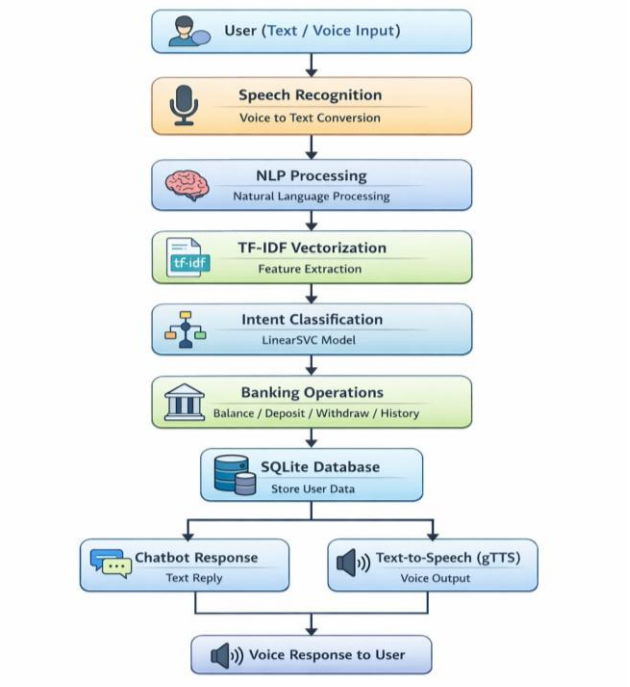

Fig. 1. Proposed System Architecture illustrating the six-stage processing pipeline

One path into the system comes through sound captured by a mic, another through typed words on a keyboard. When voice enters, it moves straight to Google's speech recognition service for conversion into written form. From there, regardless of origin, every message follows the same route ahead. Processing begins only after both types are shaped alike, ready for what follows. Uniformity at this phase allows consistent handling across modes without distinction. What matters most emerges later - how each query behaves once format differences fade. Even when sources differ, preparation makes them match before analysis starts. This alignment step ensures no bias toward one input type over another. Structure replaces source as the defining trait early in the pipeline. All entries appear identical once cleaned and standardized together.

First comes the shift to lowercase letters only. After that, punctuation marks along with special symbols get stripped away entirely. Tokens emerge next through segmentation of the cleaned text stream. Stopwords drop out afterward using NLTK's standard list for English. Then stems form by applying the Porter method instead of lemma reduction. This approach works better when handling verb variations common in bank-related searches - forms like 'transferring' become one base shape. Coverage improves across diverse word endings even without deep linguistic analysis tools present. Preference here ties directly to how such queries tend to appear naturally. [21] supports this choice within similar contexts.

Using Term Frequency–Inverse Document Frequency (TF-IDF) vectorization [22], feature extraction proceeds with an n-gram range set from one to two. A limit of ten thousand terms controls the vocabulary size during this process. Sublinear scaling applies to term frequencies before transformation. From here, the sparse output feeds into a LinearSVC model for classification.

C. Intent Classification Model

Starting with LinearSVC, the method applies a linear support vector approach fine-tuned for sorting text into multiple classes using one-against-all strategy [23]. Ten distinct intents in banking form the target groups: checking balances, moving funds, placing deposits, making withdrawals, reviewing transactions, opening accounts, asking about loans, handling card matters, submitting complaints, and raising general questions. Instead of default settings, the value for C - controlling regularization - emerged as 1.0 after testing options like 0.01, 0.1, 1.0, and 10.0 via systematic evaluation on held-out data.

Despite comparable performance, LinearSVC emerged as the chosen method instead of transformer models because it aligns better with limited computing resources. Running on standard equipment - an Intel Core i5, 8 gigabytes of memory, no graphics processing unit - the model finishes learning in under half a minute. Each prediction takes fewer than fifty thousandths of a second, meeting strict timing demands during operation. In contrast, systems built on BERT require specialized hardware to respond quickly,

something absent in the intended kiosk setup where such components cannot be installed.

D. Authentication Mechanism

A security step involving a unique code becomes required before actions like moving money, taking out funds, or starting new accounts. When such an activity is detected, the platform creates a temporary six-number passcode. This code travels to the phone on file for the verified customer. Inputting the delivered digits follows next. Only after matching the entered code exactly does entry into protected records occur. When verification fails, a record is made. Following three failed tries one after another, the system ends the session. Such an approach reduces risks tied to repeated attack methods. Access by unintended users stays blocked, especially where devices are left open for public use.

IV. METHODOLOGY

1. Dataset Construction and Preparation

A total of 8,000 queries related to banking formed the basis of this research. Originating from expert design, one portion reflected typical user intents within the field. Instead of automation, real specialists crafted these initial examples by hand. To broaden word choice and phrasing variation, some underwent rewording followed by translation into another language and back again. Historical records from past support conversations contributed further entries. These logs had personal details removed strictly under current privacy laws. Two separate reviewers assigned exactly one purpose tag per entry. Despite differences in interpretation that may arise, their choices aligned closely most of the time. Measured via Cohen's kappa statistic, consistency between them reached 0.91 - a level considered almost complete. While minor discrepancies occurred, they did not disrupt overall reliability. The process concluded only after both raters finalized every label independently.

Category	Sample Count	Total Samples	Set Split
Category 1	800-1000	6400	Training
Category 2	800-1000	1600	Test

TABLE I: Dataset Class Distribution

Ten intent classes form the dataset. Roughly equal in size, every group holds from 800 up to 1,000 entries. Further details on these counts appear in Table I. Written entirely in English, each query stays consistent in language. Division into subsets followed a fixed split: 80% assigned to training, 200 more than 1,500 reserved for testing. Stratified sampling ensured proportional representation throughout both portions. Despite variation in sample count, balance remains evident across categories.

B. Preprocessing and Feature Extraction

Every query went through preparation following the method described in Section III-B. After that step, the training data saw term frequency transformed using inverse document frequency weighting. That produced a word list kept unchanged for later use on test samples to avoid information spill. Matrix dimensions for these operations stood at 8,000 rows by 784 columns.

3. Model Training and Hyperparameter Selection

Training used the LinearSVC model alongside a TF-IDF matrix built from training data. Through five rounds of validation, settings adjusted - macro F1-score guided selection. One thousand steps capped the run; hinge loss applied when C stood at 1.0. Other models followed: Complement Naive Bayes entered, known effective for documents [8]. Logistic Regression joined - with L2 penalty, set to C = 1.0, using lbfgs method. Each approach learned from matching samples, structured alike, measured uniformly.

4. System Workflow

Processing begins at request arrival, either spoken or written. When speech occurs, sound waves convert into digital format first. This digitized stream moves toward a transcription service immediately after capture. From there, converted words enter an editing phase before further steps unfold. Edited phrases transform next through a weighting system learned during earlier training cycles. Weighted outputs feed directly into a decision algorithm calibrated on labelled examples. Whenever that label matches certain restricted actions - sending money, removing funds, starting accounts - a verification code becomes required unexpectedly. Access to stored records only proceeds once identity confirmation completes successfully. From user input onward, data flows into the processing module where relevant account details emerge or transaction steps initiate. Response delivery follows via written output along with audio generated using Google's Text-to-Speech interface. As shown in Figure 2, every stage connects in sequence. At completion, feedback reaches the user through dual channels.

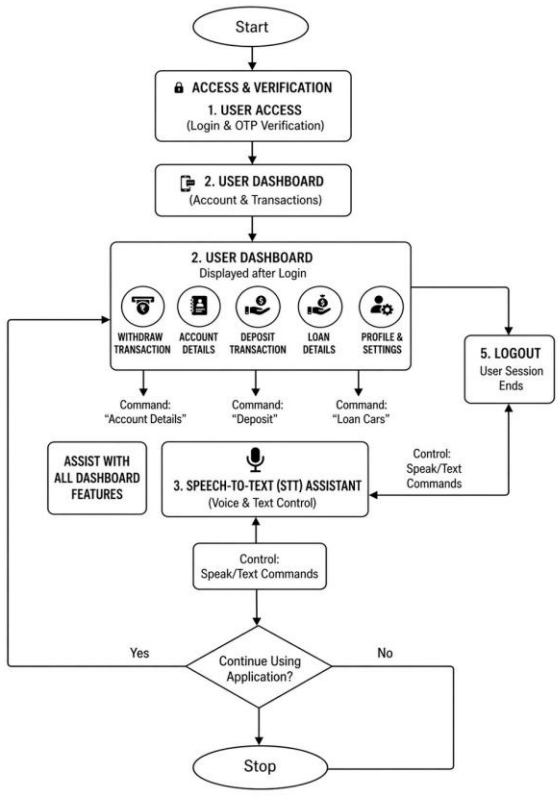

Fig. 2. End-to-end system workflow for voice and text input modalities

V. RESULT AND DISCUSSION

A. Intent Classification Performance

A total of 1,600 samples formed the separate test group used to evaluate the fitted LinearSVC system. Accuracy across these instances reached 91.67 percent. Precision, averaged uniformly across classes, stood at 0.912; recall followed closely behind at 0.908. The balanced measure combining both - F1-score - recorded a value of 0.910. Individual class-level F1 outcomes appear in Table I. Among the 62 incorrect classifications, mistakes between deposit and fund transfer appeared most often - 23 instances in total. Similar wording used by customers when describing payments may explain this pattern. The structure of transaction labels possibly contributes to such overlaps. Errors clustered here suggest subtle differences in phrasing influence model decisions.

Performance comparisons among the three classifiers appear in Table II. Across every measure, LinearSVC showed higher results than the two reference methods. At 83.2% accuracy, Naive Bayes performed modestly - its rigid assumption of independent features struggles when applied to real linguistic patterns. Reaching 88.4%, Logistic Regression delivered solid outcomes, yet still trailed behind the SVM variant designed to maximize classification margins. To assess whether the gap between LinearSVC and Logistic Regression mattered beyond chance, a McNemar's test was applied; findings indicated strong statistical support ($p < 0.01$). Though numerically close in some cases, only one model maintained consistent superiority.

Classifier	Accuracy (%)	Precision	Recall	F1-Score
Naive Bayes	83.2	0.831	0.824	0.827
Logistic Regression	88.4	0.882	0.879	0.880
LinearSVC (Proposed)	91.67	0.912	0.908	0.910

TABLE II: Comparative Model Performance On 1,600-Sample Test Set

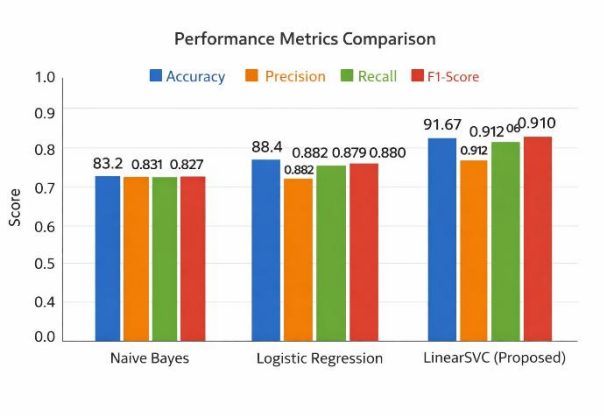

Fig. 3. Per-Metric Performance Comparison of Three Classifiers on Test Sets

B. System Response Latency

From start to finish, response delay measured across 100 trials involving both input types showed consistent patterns. Text-based interactions took, on average, 312 milliseconds between request initiation and output visibility - fluctuations typically within 48 milliseconds. When speech entered the process, conversion via Speech-to-Text added roughly 680 extra milliseconds, varying by about 95 milliseconds each time. This brought total delivery time for spoken requests close to 992 milliseconds per instance. Despite differing paths, neither method crossed beyond two seconds - the widely recognized boundary where responsiveness begins to feel sluggish in live systems.

C. OTP Authentication Performance

During testing, the One-Time Password system underwent two hundred trial logins. Delivery delay averaged 1.8 seconds per code. First-time success occurred in 94 out of every 100 tries. When errors appeared, a second try

followed - typing lag being the usual reason. Zero incorrect approvals took place throughout analysis. Despite occasional input slowness, performance remained consistent across runs.

D. Sample System Outputs

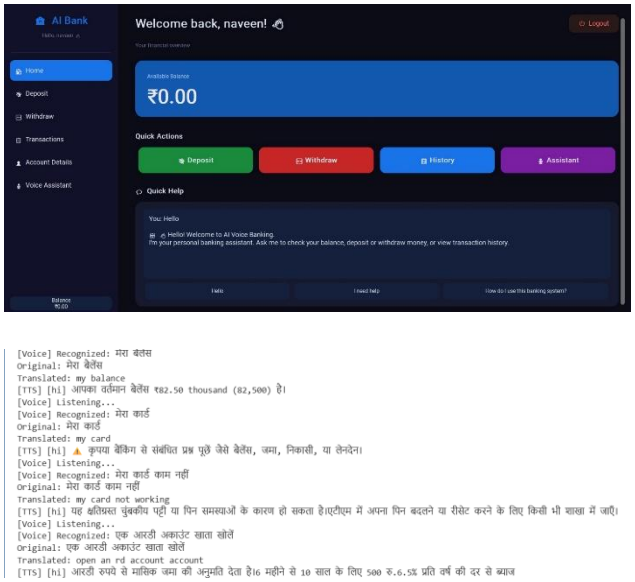

Fig. 4. Sample System Dashboard

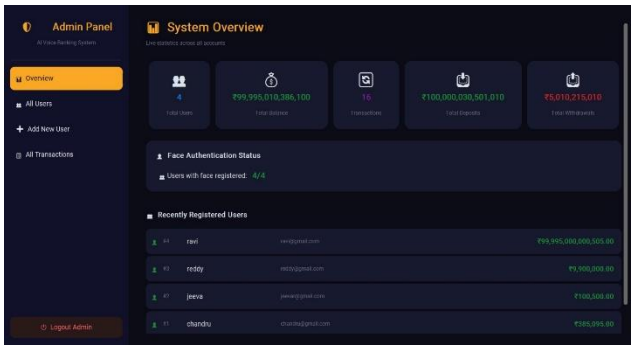

Fig. 5. Sample System Admin Dashboard

Fig. 6. Physical Hardware Setup of the Prototype Kiosk

E. Limitations

One should observe the limitations tied to current conditions. Owing to emphasis only on English entries, mention of multilingual capacity has been set aside temporarily in the heading. Without systematic review of misrecognized terms during voice interpretation, consequences for later classification stages remain unclear. Despite consistency and scale, information was gathered outside live finance operations - suggesting possible changes under real-world acoustic complexity. Progress will require attention to these points.

VI. CONCLUSION AND FUTURE SCOPE

One second marked the upper limit for text responses. Accuracy reached 91.67 percent in identifying user intents. A structured flow brought together speech processing and secure login. Performance stood above earlier models by a clear margin. Inputs spoken aloud took under two seconds to process fully. Classification relied on support vectors instead of simpler probabilistic methods. Security formed around one-time passwords embedded within sessions. Earlier attempts used less precise ways to weigh terms. 8,000 samples shaped the foundation of testing material. 91.67 percent - this figure repeated often - showed consistency. Statistical tests confirmed gains were not due to chance. Voice interactions demanded tighter timing than typed ones. The framework handled multiple tasks without external tools. 1,600 cases remained separate for final evaluation. Precision across categories averaged just below the top mark. Terms mattered more when weighted by occurrence and rarity. Response speed met expectations for live usage scenarios. Older techniques fell short when compared directly.

Clearly, basic language analysis works well with dedicated code checks in specialized financial systems. Despite simple computing setups, precise categorization remains possible, especially where equipment capabilities are restricted. The way techniques align matters more than size, once conditions allow little room to adapt.

Ahead of timeline, movement splits - not evenly - into three paths. One path widens the intent classifier, adding Hindi, Tamil, then Telugu, aiming at improved reach among bank users in India. At the same time, samples in those tongues are gathered; testing also begins on script variety, along with blended input forms. Instead of full setups, another line focuses only on voice decoding, measuring shifts in precision via word-level mistakes. Later phases introduce limited field tests: advanced arrangements, including adjusted BERT versions, run under matched speed limits versus simpler options.

VII. REFERENCES

1] G. Caldarini, S. Jaf, and K. McGarry, "A Literature Survey of Recent Advances in Chatbots," *Information*, vol. 13, no. 1, pp. 1–19, 2022.

[2] J. Gao, M. Galley, and L. Li, "Neural Approaches to Conversational AI: Recent Trends," *Foundations and Trends in Information Retrieval*, vol. 16, no. 2, pp. 103–230, 2022.

[3] S. Minaee *et al.*, "Deep Learning-Based Text Classification: A Comprehensive Review," *ACM Computing Surveys*, vol. 55, no. 3, pp. 1–40, 2022.

[4] A. Chowdhery *et al.*, "PaLM: Scaling Language Modeling with Pathways," *JMLR*, vol. 24, pp. 1–113, 2023.

[5] Y. Liu *et al.*, "Advancements in Transformer-Based Models for NLP," *IEEE Trans. Neural Networks*, vol. 33, no. 8, pp. 3456–3468, 2022.

[6] X. Zhang *et al.*, "Recent Advances in Intent Detection for Conversational Systems," *IEEE Access*, vol. 11, pp. 56789–56805, 2023.

[7] H. Chen and L. Liu, "CNN–LSTM Hybrid Models for Text Classification," *Expert Systems with Applications*, vol. 213, 2023.

[8] M. Gupta *et al.*, "Lightweight NLP Models for Resource-Constrained Environments," *IEEE Internet of Things Journal*, vol. 10, no. 4, pp. 2987–2998, 2023.

[9] N. Shrivastava *et al.*, "Natural Language Processing for Conversational AI: Chatbots and Virtual Assistants," *IEEE IATMSI*, 2025.

[10] S. Zhang *et al.*, "A Survey on Chatbot Design, Architectures, and Applications," *IEEE Trans. Artificial Intelligence*, vol. 5, no. 1, pp. 12–28, 2024.

[11] R. Smith and K. Williams, "Retrieval-Based vs Generative Chatbots: A Comparative Study," *IEEE Access*, vol. 12, pp. 44567–44580, 2024.

[12] "Recent Deep Learning-Based NLP Techniques for Chatbot Development: An Exhaustive Survey," *IEEE Conference Publication*, 2023.

[13] H. K. Jain *et al.*, "Conversational AI: A Comprehensive Study on Building and Enhancing Chatbot Systems," *IJISAE*, vol. 12, no. 3, pp. 2431–2437, 2024.

[14] E. Nouri *et al.*, "Proceedings of NLP for Conversational AI Workshop," *ACL*, 2024.

[15] A. K. Richardson and L. Heck, "Commonsense Reasoning for Conversational AI," *arXiv*, 2023.

[16] M. Liu *et al.*, "CA-BERT: Context-Aware BERT for Multi-Turn Chat Interaction," *arXiv*, 2024.

[17] M. Perera *et al.*, "A Survey of Conversational Question Answering Systems," *arXiv*, 2025.

[18] E. C. Acikgoz *et al.*, "Conversational Agents: Capabilities and Future Directions," *arXiv*, 2025.

[19] "A Contemporary Review on Chatbots and ChatGPT Applications," *Computer Science Review*, vol. 52, 2024.

[20] S. Šarčević *et al.*, "Enhancing Conversational AI Systems Using Generative Models," *IEEE MIPRO Conference*, 2024.