Received: November 30, 2023. Revised: March 13, 2024.

International Journal of
Intelligent Engineering & Systems

http://www.inass.org/

A Unique Query Processing Framework using Lexical-Cepstral Feature Extraction based B^{2}DT Classifier in Natural Language Processing

Ashlesha Kolarkar[1]* **Sandeep Kumar[1]**

[1] *Department of Computer Science and Engineering, Koneru Lakshmaiah Education Foundation Vaddeswaram, Vijayawada, India*
* Corresponding author's Email: ashlesha.pandhare@gmail.com

Abstract: The amount of data produced today is constantly increasing. With the advent of contemporary database tools and rising technology, we can store a lot of data. But, the problem is that a lot of people need to grow more adapted to the user interfaces and technological advancements that process data and show it in the manner desired by the user. It implies that a huge number of individuals require additional database management expertise. Thus, this work intends to implement a technology that will help users get precise data from databases without prior knowledge by converting natural language questions into SQL queries. In the existing works, several query processing frameworks are developed for retrieving data from the large database using advanced machine learning and deep learning techniques. But, the problems behind those works are computational burden, increased time consumption, high loss, lack of reliability, and accuracy. Therefore, the proposed work motivates to develop a simple as well as advanced feature extraction and machine learning models for an effective query processing. Here, the dictionary database (i.e. Text & Audio based SQL query formation) is created for system design and implementation. The proposed framework can handle both text and audio input data by extracting the features with the use of Lexical Text Data Analyzer (LTDA) and Mel-Frequency Cepstral Coefficients (MFCC) techniques respectively. Then, the extracted features are trained with the use of Bagged Bayesian Decision Tree (B2DT) classifier for an accurate query recognition. Finally, the performance of the proposed LTDA-MFCC integrated with B2DT model is validated and tested using several parameters. As we show, our model is able to improve an accuracy to 93.9% on the SQL query questions. In more detail, the accuracy, precision, recall, and F-score of instruction parsing reach 93.9%, 93.5%, 93.2%, and 93.4% respectively. Especially, it provides more convenient interactive means for accessing the databases through natural language statements in English language.

Keywords: Natural language processing (NLP), Machine learning (ML), Lexical text data analyzer (LTDA), Mel-frequency cepstral coefficients (MFCC), Bagged Bayesian decision tree (B^{2}DT), Query processing.

1. Introduction

In present days, the world has begun to be impacted by Machine Learning (ML) and Natural Language Processing (NLP) [1, 2]. The field of computer science known as NLP focuses on examining and comprehending how computers and human language interact. In other works, it is defined as a strategy for turning spoken or recorded natural language output into relevant insights. Typically, the NLP is a fascinating topic, because it involves both technology and human interaction to implement [3, 4]. Moreover, the ML is used to anticipate or ensure the successful implementation outcomes in a variety of contexts, including education, research, small-scale chip design, and quantum computing. Large datasets of any required design are now readily available due to advances in cloud computing technology. Typically, the designers have the ability to impart the knowledge they have gained via extensive training and growth. An algorithm has the capacity to learn from the facts at hand and provide highly efficient design and development [5]. The time required to complete such operations has lowered as well as the processes' effectiveness in achieving desired results. Syntax validation, semantic analysis,

DOI: 10.22266/ijies2024.0630.25

and parsing [6, 7] for compilers have all undergone a complete transformation as a result of NLP.

A. Background

In the actual world, people interact and ask questions using natural language. The Natural Language Interface to Database (NLIDB) is a program designed to assist users in querying databases using natural languages like English. It is based on the Natural Language Processing (NLP) approach. Query languages such as SQL are commonly used in relational database management systems (RDBMS) to extract information from the database. These systems also come with proprietary extensions of their own. Therefore, users must become fluent in the query language and informed about the database management system and database structure in order to build a query that gives the intended outcome of query results. Since most end users dealing with structured databases are not technical and are hesitant to write code in a query language, it might be challenging for non-technical end users to query relational databases without technical expertise. In relational database systems, it is crucial to implement a simple query interface that allows users to do flexible querying. It is important because companies or organizations want to lower the expenses of employee training while increasing customer accessibility to their services, and database users want to spend less time accurately searching the database. The natural language queries that users enter into the NLIDB system are converted into one of the query languages.

B. Challenges

When it comes to answering basic questions, NLIDBs have improved, but when it comes to handling complicated input queries, their adoption has been slower. When a question requires domain-specific expertise to be answered or has a complicated structure, it can be classified as complex. Ambiguous inquiries are included in the first category of queries, "complex structured queries." A query is deemed unclear if it is elliptical, has a complex syntactic or semantic structure, or both. Narrative queries and brief inquiries that require domain-specific knowledge to evaluate, deduce, or reason are included in the second category of queries referred to as "domain specific queries." These queries are derived from narratives or succinct phrases. If the difficulties surrounding the two sets of sophisticated inquiries are not addressed and the responses are not specifically recorded in the database, the queries will not be able to receive proper answers straight from the database.

End users frequently utilize incomplete, element-missing searches or formulate complex query structures. It is very difficult to match a complex query with any rule that the system supports. Because of this, the system's linguistic component must clearly interpret each ambiguous word or phrase in the input query. Inferences or reasoning procedures may be required to address complexity that cannot be handled by a rule-based system or articulated in a typical database query language.

C. Motivation

A huge quantity of data is kept in databases these days due to the increasing amount of information.The majority of web application systems, including web-based email, news portals, public scientific data repositories, source code repositories, and publication repositories across multiple disciplines, are built around database applications. It facilitates database access for novice users without the need to learn query languages.

D. Objectives

The majority of NLIDB systems aim to provide a user-friendly interface for data querying, assisting non-expert end users in submitting queries in natural language and obtaining database query responses. Typically, the NLIDB system consists of two main fundamental parts. When a natural language query is entered into the NLIDB system, its linguistic component converts it into an internal representation that is helpful, and then the system's second component converts it into database language. For a rule-based system, the NLP unit provides rules, or patterns, that enable the mapping of the query tree structure and the extraction of entities and relations from the input query. The database component of the system comes after the Natural Language Processing (NLP) unit.

E. Contributions

In the actual world, people interact and ask questions using natural language. The Natural Language Interface to Database (NLIDB) is a program designed to assist users in querying databases using natural languages like English. It is based on the Natural Language Processing (NLP) approach. Query languages such as SQL are commonly used in relational database management systems (RDBMS) to extract information from the database. These systems also come with proprietary extensions of their own. Therefore, in order to create a query that yields the intended output query results, users must become proficient in the query language and knowledgeable with the database management system and database structure. It can be difficult for non-technical end users to query relational databases without technical training, as end users who deal with structured databases frequently lack technical experience and are afraid to write code in a query language. In relational database systems, it is crucial to implement an intuitive query interface that allows users to do flexible querying. It is important because companies or organizations want to lower the expenses of employee training while increasing customer accessibility to their services, and database users want to spend less time accurately searching the database.The natural language queries that users enter into the NLIDB system are converted into one of the query languages.

Moreover, the combination of ML and NLP methodologies has increased the effectiveness of data mining tools [8, 9]. The processing speed of systems that parse and scan huge datasets has greatly improved. The goal of those who work with sophisticated algorithms and databases in the software sectors is to automate laborious operations. One area where automation is highly desired is database searching. For the purpose of query automation, many methods have been proposed. In the past few years, the field of databases and related IT has experienced tremendous growth. The intelligent database technology [10, 11] is a growing field in databases that will significantly alter how humans think and function. These days, it is more important than ever for non-expert users to query relational databases using natural language, semantics, etc., rather than relying solely on data point. The vision of using natural language to connect with computers provides an opportunity to smart human-computer interaction solutions. This opened up new opportunities for the development of Natural Language Interfaces for Databases (NLIDBs) [12-14]. Users of NLIDBs can access the database's data without any prior knowledge of query languages by expressing their queries. The primary goal of NLP is to make it possible for people who lack programming expertise to acquire and extract data from databases using natural language. The research communities have recently begun to pay attention to the study of NLIDB. The following are examples of the most popular natural language processing tools [15, 16]:

- *Stopwords:* The most frequently occurring words in a language are known as stop words, and there is no formal list or definition of stop words exists. Any group of words can be chosen as stop words depending on the stated objective. Stop words can provide crucial information about the relationships between various tokens for NLP. As a result, it is highly essential to recognize stop words, but leave them in as they are sometimes necessary for calculations.
- *Synonymy:* The challenge with synonymy is that a quick search or match is insufficient. The system must also take synonyms into consideration, and using a translation dictionary could be an option.
- *Tokenization:* An input question is divided into a sequence of tokens via tokenization. It can be as straightforward as a separator on whitespace, but it is more frequently based on many criteria or carried out using ML algorithms. If the input question contains punctuation, simple whitespace splitting tokenizers are sometimes insufficient.
- *Parts of Speech (PoS):* A group of words having related grammatical characteristics is referred to as a part of speech (PoS). The PoS tags noun and verb are present in almost all languages. PoS tagging refers to the procedure of assigning the appropriate PoS tag to each token in a document. The token itself as well as its context are used to tag the token. As a result, tokenizing the text and identifying end-of-sentence punctuation is required first. The data generated by the PoS tagger is used by more sophisticated NLP tools.
- *Stemming:* Both stemming and lemmatization aim to reduce a word's affixation and derivationally related forms to a basic form that is shared by all. By eliminating various word endings, stemming reduces related words to the same stem. Most stemming algorithms use a primitive heuristic technique that removes the ends of words to accomplish this. The lemma, which might be either the word's base form or the dictionary form, is obtained through lemmatization, which eliminates inflectional endings. Lemmatization algorithms often analyze the words' morphology and vocabulary in order to accomplish this.
- *Parsing:* Analyzing the grammatical structures (syntax) of a sentence is the process of parsing. The parser often uses context-free grammar. The first major path, dependency syntax, is one of two major ways to approach the syntax.

The major research contributions of this paper are given in below:

- The purpose of this paper is to develop a simple and novel query processing framework using an advanced feature extraction and machine learning algorithms.
- To obtain the features of the text data relevant to the given query, a Lexical Text Data Analyzer (LTDA) model is utilized.
- To obtain the features of the audio data relevant to the given query, Mel-Frequency Cepstral Coefficients (MFCC) mechanism is employed.
- To accurately predict the relevant data from the database according to the query, a novel Bagged

Bayesian Decision Tree (B^2DT) Classification algorithm is implemented.

- To validate and test the results and performance of the proposed query processing system, several parameters are used for analysis.

The remaining portions of the article are divided into the following sections: The pertinent literature studies for query processing and data retrieval using NLP are presented in Section 2. The proposed machine learning-based query processing system is briefly described in Section 3. The results of the suggested framework are validated and compared in Section 4 using a variety of metrics. In Section 5, the findings, results, and future scope of the work are summarized.

2. Related works

D. H. Maulud [17] presented a comprehensive review to examine the state-of-the-art methodologies used for semantic analysis. Here, the semantic interpretation has been carried out using NLP with reduced prediction error. The concept of sentiment analysis is the process of determining how people react to a specific topic and its characteristics. People utilize opinion mining services frequently because they wish to act responsibly. J. Du [18] introduced a new method, named as SentAugment for retrieving information from the web sentences. The suggested method is based on a large corpus of unsupervised sentences that was created from data that was web crawled. The sentence bank's size and diversity enable it to hold material from diverse domains and with various styles, allowing for the retrieval of pertinent information for numerous subsequent activities. Here, each phrase is encoded using a generic paraphrastic sentence encoder, whose embedding space does not depend on the tasks performed afterwards. Also, it will be utilized to extract portions of the sentence bank that are pertinent to specific activities. Moreover, embedding is constructed for each downstream task using the same paraphrastic paradigm. Furthermore, two semi-supervised learning strategies such as, self-training and knowledge distillation are integrated with the data augmentation technique. K. Affolter [19] investigated the performance of various natural language interfaces used in the academic sectors. Typically, the grammar-based systems are the most potent, although they rely heavily on their manually generated rules. The use of natural language interfaces for databases is one of the frequently discussed strategy that makes database queries easier even for casual users. Also, it enables users to search databases for information by typing questions that are stated naturally. Some natural language interfaces limit the use of the natural language to a language within the domain or to a dialect with grammatical restrictions. C. Rodriguez[20] investigated the temporal moment localization problem with the use of guided attention model. Identifying the beginning and conclusion of the temporal video segment that most closely matches a given natural language query is known as language-driven temporal moment localization. In essence, this refers to the use of natural language queries to localize actions in untrimmed videos. Although the language-based configuration permits an open set of activities, it also corresponds to a more natural query definition because it contains explicitly objects, their attributes, and connections between the concerned entities. An attention-based dynamic filter is used after both the input sentence and the input video have been encoded. The purpose of this is to enable the model to provide filters that can be applied over the film features and alter dynamically in response to the phrase query, responding to particular elements of the video embedding and thereby giving the model hints about the location.

W. E. Zhang [21] presented a detailed analysis to analyze the different types of deep learning techniques in NLP. Deep neural networks (DNNs) are substantial neural networks with an architecture made up of layers of neurons that act as separate computational units. The outcomes of a neuron's activation function on its inputs are transmitted to the neurons in the layer below via links with various weights and biases. To learn and accumulate knowledge from examples, deep neural networks attempt to resemble the organic neural networks of human brains. E. A. Olivetti [22] presented some of the data driven materials for information extraction using NLP. In this review, the authors examined text mining and NLP-based methods for fully and partially automating the acquisition and processing of scientific data. Analytics, data mining, and data visualization are made possible by the application of NLP on scientific language to create libraries of information for exploration. I. Spasic [23] suggested Text information can be summarized and made summative and interactive in order to illustrate similarities, gaps, or patterns. These are two of the main purposes of text extraction. The typical NLP processing framework includes the following stages: content acquisition & markup parsing, text preprocessing, document segmentation, entity recognition, and linking. Q. Su [24] suggested the initial stage is to create and compile a pertinent library of subject materials that will be used as a

Table 1. Literature Survey

Ref No.	**Technique**	**Data Set**	**Drawbacks**
2019, [2]	Natural Language Processing	Electronic Health Record (EHR)	There can be discrepancies in how the frames and related components are assigned to each publication.
2020, [5]	graph neural network.	SPIDER	Further investigation of using NL model combined with application domain knowledge or semantics needed in developing TTS models and algorithms of higher performance.
2020, [7]	neural network	TableQA	Concatenate on columns and contents of databases required to improve the difference of column expressions.
2020, [8]	NEL (Named Entity Linking)	AIDA-CoNLL	A lot of methods for elevating NL to KG are based on NER techniques from earlier generations, so new lifting methods that incorporate disambiguation and linkage to the best-of-breed NER techniques are required.
2020, [9]	Deep Learning	Press Ganey	For a machine learning or deep learning model to successfully and prospectively classify these data, the training set must be properly preprocessed.
2020, [12]	Deep Learning	Squad	The performance of the proposed method could vary, depending on the size of the training and testing datasets.
2020, [13]	Deep Neural Network	NUS-WIDE-Object	This approach does not directly learn from the raw data; instead, it treats various features derived from various extraction techniques in each type of raw data as a separate mode.
2021, [14]	OLAP Hypercube	OLAP DATASET	To use infinite-dimensional n-D cubes instead of 4-D cubes should be enabled for ingestion of big data
2020, [15]	Support Vector Machine	ADE-drug	There was a concept extraction constraint where the eliminated idea was a pair concept.
2020, [16]	Natural Language Processing	Legal AI	The three most important problems that LegalAI can solve by fusing symbol-based and embedding-based approaches can be the main focus.
2020, [24]	Deep Learning	LIAR, FEVER, Buzz-Face	Misinformation detection systems accuracy and dependability are limited by small data sets and pre-labelling.
2020, [27]	Machine Learning	Sacred Texts (Bible, Quran)	Because religious texts often contain unstructured corpora extracting knowledge from them can be difficult.
2023, [33]old	Natural Language Processing	Science Direct, Springer,WoS, IEEE	It ignores other businesses in favor of focusing mostly on automating consumer query responses in particular industries.
2021, [35]	Deep learning	ChineseQCI-TS	Proper study should be done of BERT, ELMO, and other pretraining models to encode keywords
2019, [36]	Deep learning	House purchase	Study joint learning of sequence labeling and relationship extraction should be improved.
2021, [37]	learning based extraction method	CaRB	Need to address entity disambiguation using popular knowledge bases
2022, [38]	Support Vector Machine	ESG data	It works well when examining the ESG performance of businesses that are regularly mentioned in news reports.

source for information retrieval. The accessibility level, reservoir of specific topic articles, and document types all affect the content differently. Once the material has been collected, object extraction, item correlation, and attributes linking are employed as the three major operations to change the text's contents. The preprocessing of the study's text to identify the desired information is the first step in

Received: November 30, 2023. Revised: March 13, 2024.

this entire cycle. Preprocessing will differ depending on the sequence of things and the tools employed at each step. A simple method for identifying section headers is to match regular expressions against a set of text or regular expression code. Nevertheless, even this simple task can be complicated due to the numerous ways that authors apply headers. A. Silva [25] utilized a re-ranking strategy for enhancing query expansion in NLP. The authors of this study looked at how well word embedding represented queries and documents during query-document matching tasks. Additionally, a query vector representation is formulated for obtaining the informative terms with the use of IDF average word embedding. S. T. Y. Ramadan [26] proposed the MFCC which attempts to eliminate speaker subordinate features by preventing the essential recurrence and their music, simulating the logarithmic perspective of tumult and tone of a human sound-related framework. The MFCC includes the difference in the component vector over time as a key component of the element vector. The steps for getting the MFCC highlighting pattern are shown below:

- Pre-emphasis
- Framing
- Windowing
- Fourier transformation
- Mel-scaled filter banks

In order to enhance the sound waves and reduce the noise, a pre-emphasis filter is first applied to the data. The pre-emphasis filter regulates the recurrence range to prevent numerical issues during the Fourier transform operation since harmonic overtones possess lower magnitudes than lower frequencies.

The drawbacks related to some of the conventional techniques are shown in Table 1. It represents the limitations with other methods to clarify this work further.

3. Proposed methodology

The suggested NLP-based ML-based query processing framework is briefly explained in this section. The main contribution of this work is to develop an efficient as well as simple query recognition model with the use of feature extraction and machine learning classification algorithms. Here, the SQL database has been created with the employee and student record details, which includes both text and audio sequences of data. According to the given input data, the data relevant to the query has been retrieved from the database. After database generation, the query set is obtained as the input for processing. If it is the text data, the Lexical Text Data Analyzer (LTDA) model is used to extract the features from the dataset, where the standard operations like tokenization, PoS tagging, stop words removal, lemmatization, and stemming are performed. It helps to clean the raw dataset for

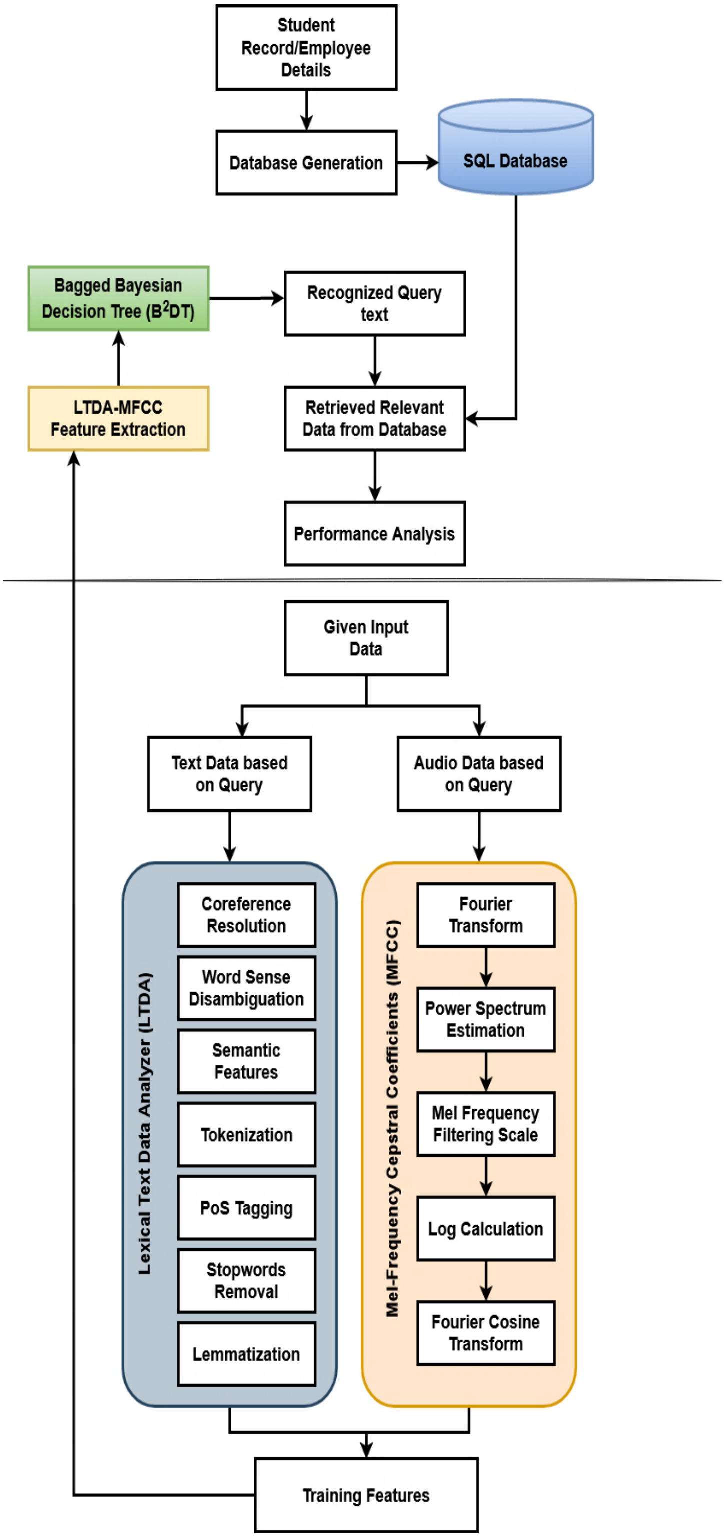
Student Record/Employee Details
Database Generation
SQL Database
Bagged Bayesian Decision Tree (B²DT)
Recognized Query text
LTDA-MFCC Feature Extraction
Retrieved Relevant Data from Database
Performance Analysis
Given Input Data
Text Data based on Query
Audio Data based on Query
Lexical Text Data Analyzer (LTDA)
Coreference Resolution
Word Sense Disambiguation
Semantic Features
Tokenization
PoS Tagging
Stopwords Removal
Lemmatization
Mel-Frequency Cepstral Coefficients (MFCC)
Fourier Transform
Power Spectrum Estimation
Mel Frequency Filtering Scale
Log Calculation
Fourier Cosine Transform
Training Features

Figure. 1 Work flow of the proposed system

Received: November 30, 2023. Revised: March 13, 2024.

improving the accuracy of recognition. Also, the new add-on features such as coreference, word sense disambiguation, and semantic are also extracted from the data. Consequently, the obtained features are maintained as the training set for classification. If it is an audio data, the Mel-Frequency Cepstral Coefficients (MFCC) model is deployed for extracting the features from the given voice. Similar to the text model, the audio features are also separately maintained for classifier training and testing operations. Moreover, the new ensemble learning model, named as, Bagged Bayesian Decision Tree (B2DT) is used to predict the results according to the extracted text and audio input data features. Finally, it produced the classified label for recognizing the query text, based on this the relevant data can be retrieved from the database. The workflow model of the proposed query recognition system is represented in Fig. 1.

3.1 Lexical text data analyzer (LTDA) model based feature extraction

The information extraction plays a vital role in many real time applications like text mining, business intelligence, knowledge management, and so on. The process of data retrieval in the NLP framework is heavily depends on the features used for training and testing. Here, the Lexical Text Data Analyzer (LTDA) model [27] is used to extract the text features from the given input data.

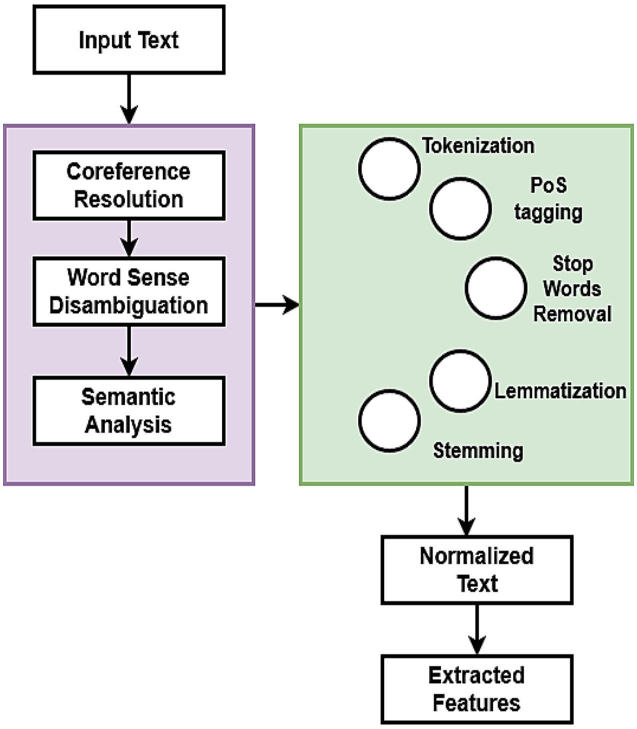

Figure. 2 Text feature extraction

The context of a language must be understood in order to fully comprehend the language, which is accomplished via comprehending the grammar rules and other linguistic characteristics. Similar to this, the linguistic characteristic in natural language processing enables language understanding. Due to the frequent adherence of review sentences in each category to a specific structural pattern, linguistic aspects can be helpful.

The pipeline model of the proposed LTDA model is shown in Fig. 2. The Coreference resolution constitutes one of the core NLU operations. When entities are mentioned in texts in any language, it makes it easier to understand the relationship between them. Discourse, as used in NLP, is a collection of statements that follow one another.

When a word in a phrase or discussion refers to some other word, subject, or element, is known as reference in natural language processing. Such references are resolved through the resolution procedure. Typically, a coreference occurs when two references to the same real-world object are made. Finding textual mentions and grouping them according to the entity they refer to is known as coreference resolution. In this case, an entity designates a physical object, and mention is a referencing expression that defines that physical object. In this step, all occurrences from the input dataset are extracted and relationships between entities and references, or nouns and pronouns, are built. Understanding how words and phrases are used in context is helpful in determining their logical and accurate interpretation. For instance, “bark” can apply to both the bark of a tree and the sound a dog makes. With the aid of general knowledge, this step determines whether or not each statement or word has any significance.

Due to the ambiguity of the Natural Language, the same exact words frequently have multiple meanings. The problem of determining a word’s meaning when it is employed in various contexts is resolved by word sense disambiguation. The way that machines can take a large number of words and transform them into useful data is exciting to developers. Because natural language’s laws are imprecise and undefined, dealing with it is challenging for machines. To handle textual data, natural language processing algorithms need to be trained with rules and knowledge of the semantic aspects of the language. The meaning of individual words and word combinations are the two fundamental components of the semantic features analysis. The group of words is divided into four categories: context, lexical structure, word formation, and relationship between words. It also includes the following processes:

- Tokenization
- PoS tagging
- Stopwords removal
- Lemmatization

The text is tokenized to retrieve its tokens, and a tokenizer can be thought of as a classifier that divides tokens into several categories. Unlabelled words in natural language are marked with Part-of-Speech labels, such as noun, verb, adjective, and preposition. Lexical likelihood and context probability are two elements that affect a word's tag [28]. Word-to-word dependencies between pairs of words are analyzed by dependency parsers. There is a category for each dependent that corresponds to its grammatical purpose. Each arc in the graph representing a grammatical dependency connecting the words of the sentence to one another is represented by dependency parsers, which model language as a collection of relationships between words. Moreover, the standard stop words removal and lemmatization processes are also performed. Finally, the extracted set of features are passed to the machine learning classifier for query recognition.

3.2 Mel-frequency cepstral coefficients (MFCC) based feature extraction for audio data

Typically, the Mel-Frequency Cepstral Coefficients (MFCC) [29] is the component extraction method that is most frequently used in many applications for speech recognition. The stream must be divided into short timescales after pre-emphasis. This trend is justified by the fact that frequencies in a signal change with time; as a result, if indeed the Fourier transform is used on the whole signal, the signal's dynamic frequency ranges will be destroyed. In order to avoid this problem, it is required to trust the frequencies for a small period of time. By windowing each frame separately, the signal discontinuities for each frame's beginning and end will be reduced through spectral distortion. Since, it offers a reasonable balance between frequency resolution and a Hamming window. Then, the discrete fourier transformation is performed for each frame by using the following model:

$$X_i(h) = \sum_{k=1}^{K} x_i(k)h(k)e^{-\frac{j2\pi mk}{K}} 1 \leq h \leq H \qquad (1)$$

Where, $h(k)$ indicates the hamming window having K samples, and m indicates the length of DFT. Then, the periodogram of power spectrum $G_i(m)$ is estimated for the given speech frame as shown in below:

$$G_i(m) = \frac{1}{K}[X_i(m)]^2 \qquad (2)$$

Triangular channels on a Mel-scale are used to split recurrence groups in order to determine the filter banks. The Melscale applies smaller adjustments in pitch at low frequency range compared to those at high frequencies in order to mimic the non-straight human ear's experience of audio. The addition of this scale causes the highlights to more closely match what listeners perceive. During this process, the user's text input in natural language is retrieved. To create words, the input is divided based on white space. If more than one sentence is found, it is separated at the '.' to create separate sentences, and then each sentence is tokenized to create words. Sentences that have been tokenized are loaded into a list data structure. The words in a list are automatically hashed to enable quicker access. The list object makes a collective reference to these words, maintaining the sentence's intended meaning. For more accurate context-sensitive analysis, the words are rearranged once the word set has been produced. Despite the fact that words have meaning on their own, context-sensitive facts give them an entirely distinct meaning. Word orders must be taken into account while performing semantic analysis. The relationship between the words offers the information needed to select a database and the appropriate relational schema. The proper words for the selection criteria based on the database and table names are chosen using weight values. Following dependency analysis, the selected words are segregated based on their weights. All of the bags are searched for isolated words, and any matches are collected. For streamlined and clear processing, matches are kept in a new list. The list is iterated over, one by one, to find all the identical item sets. The pre-generated code template library is cross-referenced with the index component after it has been extracted. Moreover, the query is built on a set of specific keywords that are created. Then, the computed query is stored for further use, and is passed to the machine learning classifier for recognition.

3.3 Bagged bayesian decision tree (B^2DT) classification

The text features and audio features are separately given to the classifier for training and testing operations. In the existing works, various classification techniques are implemented to for query recognition. Typically, the supervised

Table 2. List of symbols and its descriptions

Variable	Descriptions
$P(.)$	Probability function
$P\left(\frac{Q_i}{d_i}\right)$	Posterior probability
$aP(Q_i)$	Prior probability
$P(F_{ij}/Q_i)$	Class conditional probability
S	Training dataset
$d_1, d_2 \dots d_n$	Tuples
$F_1, F_2 \dots F_n$	Attributes
$\{Q_1, Q_2 \dots Q_n\}$	Class label
d_k	Value of attribute

classification techniques are regarded as crucial tools for decision-making in the field of machine learning. For instance, the Naïve Bayes (NB), Support Vector Machine (SVM), K-Nearest Neighbor (KNN), and Random Forest (RF) are the most commonly used techniques in the NLP applications. However, it has the problems of computational burden in prediction, more time consumption, and low accuracy. Therefore, the proposed work intends to implement a new Bagged Bayesian Decision Tree (B2DT) classification model for query recognition. It obtains the features as the input for training, and produced the classified label as the output. Here, the proposed B2DT is developed based on the ensemble of decision tree and NB. For the set of extracted features from the given dataset $S = \{d_1, d_2 \dots d_n\}$, the samples $d_i = \{d_i1, d_i2 \dots d_im\}$, variables $\{F_1, F_2 \dots F_n\}$, and class labels $Q = \{Q_1, Q_2 \dots Q_n\}$ are initialized at first. For each class in the dataset, the prior probability is computed, and consequently, the class conditional probability is estimated for each variable by using the following models:

$$P\left(\frac{Q_i}{d}\right) = \frac{P\left(\frac{d}{Q_i}\right)P(Q_i)}{P(d)} \tag{3}$$

If the prior probabilities of the classes are unknown, it is assumed that the distribution of the classes is equally probable.

$$P\left(\frac{d}{Q_i}\right) = \prod_{k=1}^{n}\left(\frac{d_k}{Q_i}\right) \tag{4}$$

Moreover, the sample belongs to the class with the highest posterior probability after each training sample is classified using these probabilities. As a result, the dataset no longer contains any training cases that were incorrectly classified. The NB classifier eliminates these misclassified samples since they are the problematic occurrences that have a negative impact on classification accuracy. The best splitting attribute with the maximum information gain is found for constructing the tree. In addition, by enhancing the tree parameters such tree size and split parameter, the suggested model's accuracy can be further improved. Bagging is an ensemble-based technique that combines the predictions of various classifiers to create a single classifier. Each of the individual classifiers that make up the ensemble are less accurate than the final single classifier. A better ensemble can only be created when the individual learners that make up the ensemble are correct and source their errors on different parts of the input space. Stability is one of the main characteristics that enhances the performance and predictive accuracy of the bagging approach, while instability refers to minor changes in the dataset that have a significant impact on the outcomes of the predictions. By generating a variety of training datasets for each individual learner, bagging produces precise ensembles based on resampling approaches, often known as bootstrapping methods. This method is more effective when used with learning algorithms that are inherently unstable, like neural networks and decision trees. While not true for stable machine learning algorithms, bagging increases the classification accuracy of unstable methods. It lessens variance and prevents the classifier from being overfit.

Algorithm 1 - Bagged Bayesian Decision Tree (B^{2}DT) Classification

```
Input:  Training dataset, class labels, and number of samples;
Output: Predicted label;
For i = 1 to n do
    1. Create $n$ number of models;
    2. For each class
         Estimate the prior probabilities $P(D)$;
       End for;
    3. For each attribute value in the dataset
         Compute the class conditional probabilities $P(Q_i)$;
       End for;
    4. For each training instance in the dataset
         Estimate the posterior probabilities;
         If $d_i$ is mispredicted
           Remove $d_i$ from the dataset;
         End if;
       End for;
    5. Create a tree with $N$ number of nodes;
    6. Identify the best splitting criterion $T_Q$ and label according to the attribute value;
    7. For each outcome of the splitting criterion
```

Received: November 30, 2023. Revised: March 13, 2024.

```
        Consider the set of instances $S_k$ in the
        dataset for the outcome;
        If $S_k$ is empty
            Insert a leaf node with the majority
            class designated on it there;
        Else
            Insert the node return by $(S_k, F_i, T_Q)$;
    End for;
    For each testing instance
        $Q^*(d) = \underset{y}{\arg max} \sum_i \omega(Q_i(d) = y)$    (5)
    End for;
8.  Obtain the best prediction result;
```

4. Results and discussion

This section presents the results and discussion of the proposed query recognition framework using several performance measures. Here, the dictionary database (i.e. Text & Audio based SQL query formation) is created with the set of samples. According to the given query, the relevant data is retrieved from the database with the use of feature extraction + machine learning techniques. Fig. 3 validates the performance of the proposed model based on the parameters of training accuracy, testing accuracy, wrongly identified instances, not identified instances, precision, and recall. These parameters are specifically used for validating the effectiveness of the classifier, and are calculated as shown in below:

$$Accuracy = \frac{Tp+Tn}{Tp+Tn+Fp+Fn} \tag{6}$$

$$Precision = \frac{Tp}{Tp+Fp} \tag{7}$$

$$Recall = \frac{Tp}{Tp+Fn} \tag{8}$$

$$F1-Score = \frac{2\times Precision\times Recall}{Precision+Recall} \tag{9}$$

Where, Tp - true positives, Tn - True negatives, Fp - False positives, and Fn - False negatives. The improved values of these parameters assure the better performance of the classifier. According to the outcomes from Fig. 3, it is observed that the combination of proposed LTDA+MFCC+B^{2}DT model provides an increased performance outcomes in query recognition.

Figs. 4-7 compares the prediction performance of the proposed and existing classifiers integrated with the feature extraction algorithms such as Term Frequency - Inverse Document Frequency (TF-IDF)[35], unigram, bigram, and trigram.

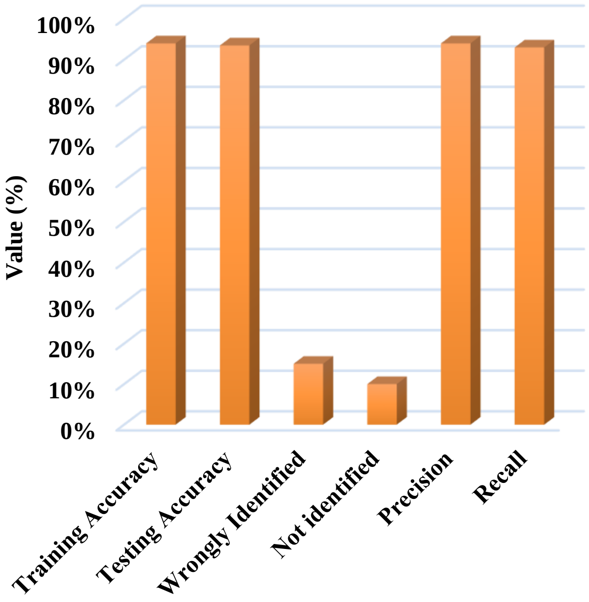

Figure. 3 Performance analysis

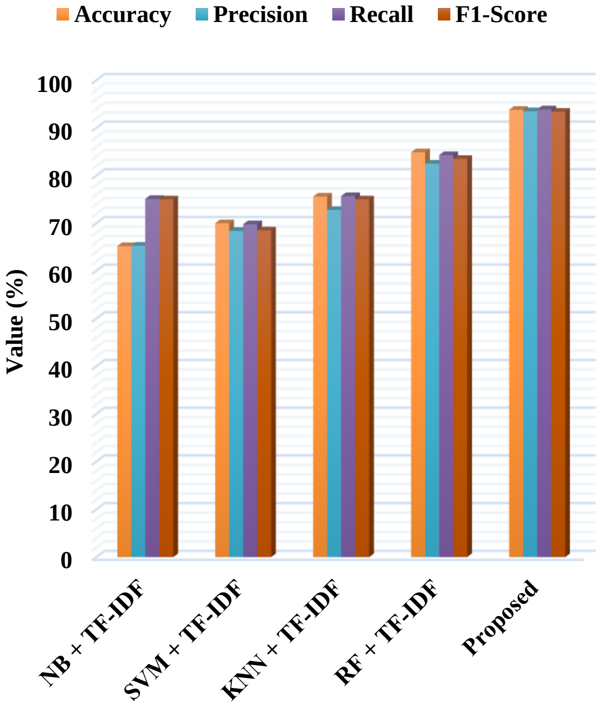

Figure. 4 Comparative analysis among the proposed and existing machine learning models incorporated with TF-IDF feature extraction techniques

The standard Decision Tree(DT),Logistic Regression(LR),Random Forest(RF) ,Support Vector Machine(SVM), and Naïve Bayes(NB) are all the widely used models in NLP for feature extraction. Due to the inclusion of feature extraction, the training

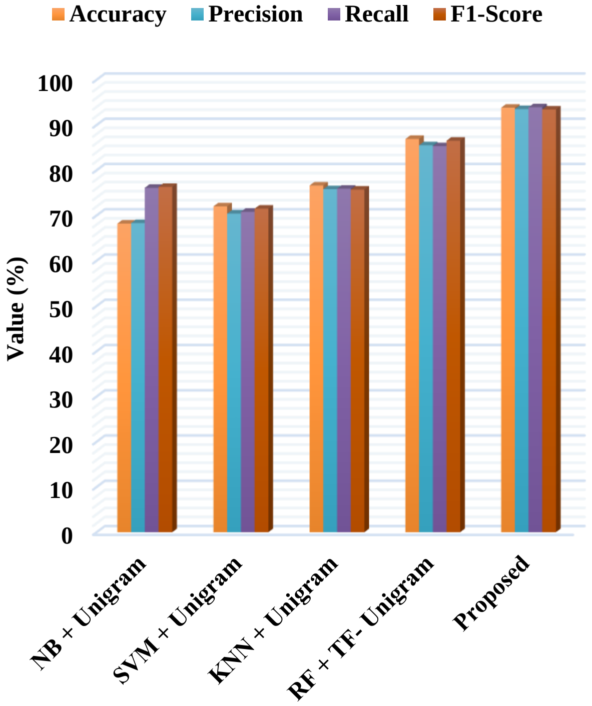

Figure. 5 Comparative analysis among the proposed and existing machine learning models incorporated with Unigram feature extraction techniques

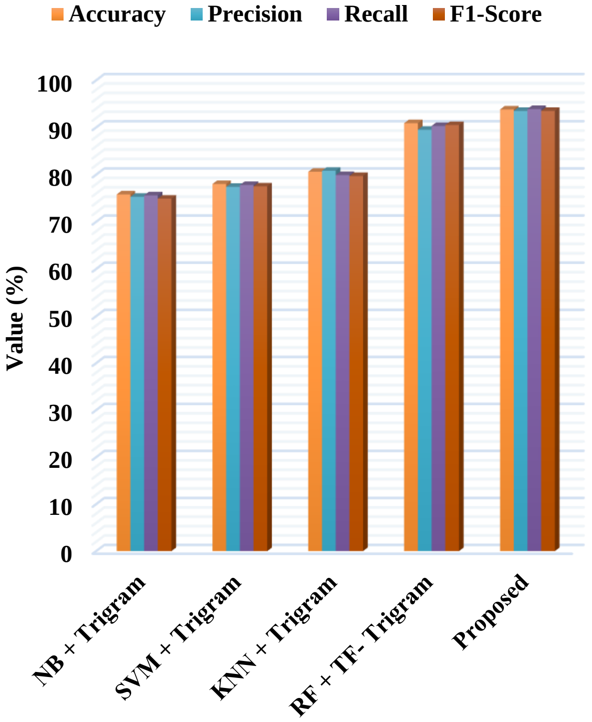

Figure. 7 Comparative analysis among the proposed and existing machine learning models incorporated with Trigram feature extraction techniques

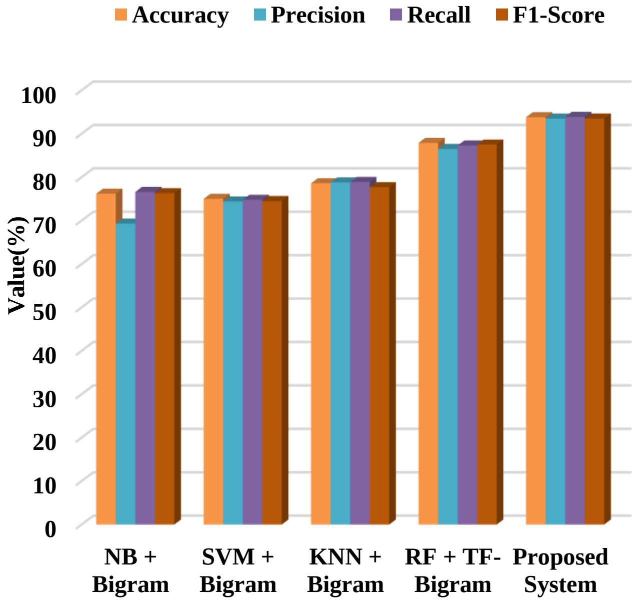

Figure. 6 Comparative analysis among the proposed and existing machine learning models incorporated with Bigram feature extraction techniques

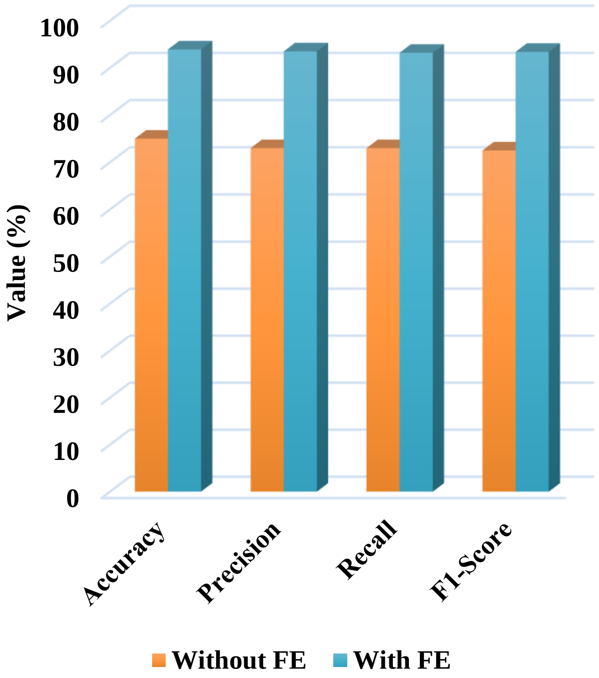

Figure. 8 Performance analysis of the proposed model with and without feature extraction

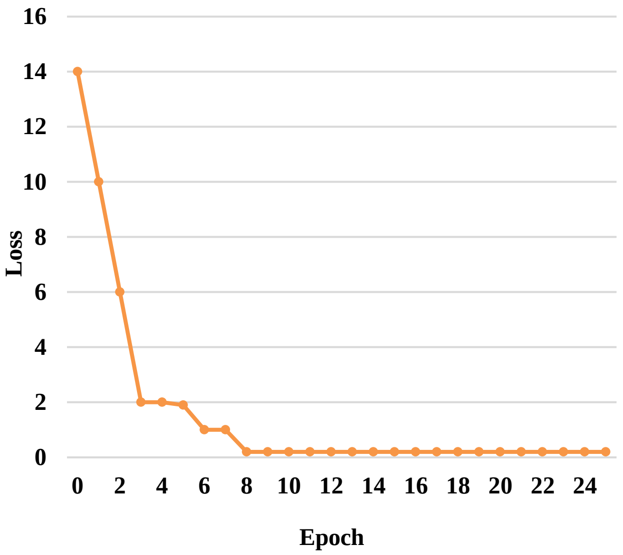

Figure. 9 Training loss

Table 3. Comparative analysis

Ref.No.	Techniques	Dataset	Accuracy
2021, [5]	graph neural network.	SPIDER	70.8%
2020, [7]	neural network	TableQA	90%
2020, [12]	Deep Learning	Squad	80%
2020, [13]	PCA+SVM	NUS-WIDE-Object	83%
2020, [15]	Support Vector Machine	ADE-drug	91.4%
2020, [16]	Natural Language Processing	Legal AI	86.8%
2020, [18]	Supervised Learning	STS-Benchmark	80.8%
2020, [22]	Machine Learning	STM datasets	89%
2021, [30]	Neural network	WikiSQL,Table QA	88%
2021, [31]	Machine Learning	IMDb,Company sales	91.7%, 94%
2021, [32]	Deep Learning	SPIDER	60%
2019, [36]	Deep Learning	House Purchase	75% to 80%
2022, [38]	Natural Language Processing	ESG data	81%

and testing efficiency of the classifier has been highly improved. Since, the prediction performance of any classifier is greatly influenced by the set of features. Hence, it is more essential to abstract the suitable and relevant features from the given data for classification. Similar to this, the analysis demonstrates that the suggested algorithm performs much better than the competing algorithms.

Fig. 8 validates the performance of the proposed query processing framework with and without feature extraction. For analyzing the major effects of applying LTDA and MFCC feature extraction models, the performance measures are assessed with and without feature extraction techniques. It is obvious that the proposed query recognition framework provides an effective results with the inclusion of feature extraction model. The effectiveness of the proposed method can be represented by comparing with the existing state-of-the-art methods or techniques.

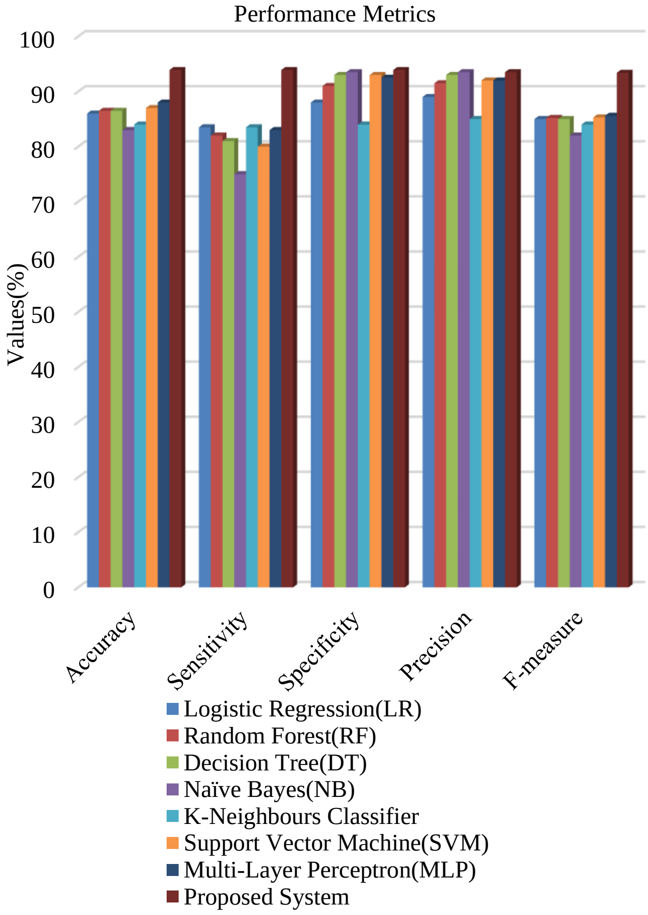

Figure. 10 Overall comparative analysis

Table 4. Performance of proposed system

Sr. No.	Performance factors	Proposed system
1	Accuracy	93.9%
2	Precision	93.5%
3	Recall	93.2%,
4	F score	93.4%

Fig. 9 validates the training loss of the proposed classification model with respect to different epochs. This parameter is mainly used to determine that how classifier fits the training model. In other words, it is defined that the error value of the training set can be assessed by using the loss measure. Hence, it must be reduced to the maximum for assuring the better classifier's performance. According to the analysis, it is noted that the training loss of the proposed B2DT is greatly reduced with the use of LTDA-MFCC models. In addition, the overall performance of the proposed and existing query recognition models is validated and compared by using several measures as shown in Fig. 10.

For this comparison, some of the standard machine learning classifiers such as LR, RF, DT, NB, KNN, SVM and MLP have been considered in this work. The results stated that the proposed B^{2}DT incorporated with the LTDA-MFCC model outperforms the existing classifiers by accurately recognizing the queries.

5. Conclusion and future work

This paper presents a new query processing framework for retrieving the relevant data from the database with the use of advanced LTDA-MFCC feature extraction and B2DT classification models. Here, the SQL dictionary database has been created with the employee and student record details, which includes both text and audio sequences of data. After database creation, the input data is obtained for processing, if it is text data, the LTDA model is applied to extract the features. During this process, the standard operations like tokenization, PoS tagging, stop words removal, lemmatization, and stemming are performed. It helps to clean the raw dataset for improving the accuracy of recognition. Also, the new add-on features such as coreference, word sense disambiguation, and semantic are also extracted from the data. If it is an audio data, the MFCC technique is applied for feature extraction. Then, the obtained features are maintained as the training set for classification. Here, the B2DT classification approach is deployed for recognizing the query according to the trained features. Furthermore, the performance of the proposed query recognition framework is validated using several parameters like ROC, training loss, accuracy, precision, recall and so on. Also, the results are compared with the standard machine learning algorithms extensively used in NLP applications. According to the analysis, it is noted that the LTDA + MFCC incorporated with the B2DT technique provides an improved outcomes with accuracy of 93.9% over other approaches. In future, the present work can be enhanced by implementing a new feature extraction and deep learning algorithms for improving the overall performance of the query recognition framework.

6. Limitations

Although the suggested methodology was successful in creating NLIDB, it is important to recognize its limits. As with other machine learning techniques, the suggested approach's performance may differ based on the quantity of the training and testing datasets. Only the provided dataset is used to evaluate the methodology. As a result, in the future, the sample size can be increased to better confirm and enhance the effectiveness of the suggested strategy.

Author Contributions

Conceptualization, Ashlesha Kolarkar, Sandeep Kumar; Data collection, Ashlesha Kolarkar; Methodology, Ashlesha Kolarkar; Writing—original draft preparation, Ashlesha Kolarkar; Review and Editing, Sandeep Kumar; Evaluation and validation of results, Sandeep Kumar; Final verification, Sandeep Kumar.

References

[1] S.Adhikari, “Nlp based machine learning approaches for text summarization”, In: *Proc. of Fourth International Conference on Computing Methodologies and Communication (ICCMC)*, pp. 535-538, 2020.

[2] S. Datta, E. V. Bernstam, and K. Roberts, “A frame semantic overview of NLP-based information extraction for cancer-related EHR notes”, *Journal of biomedical informatics,* Vol. 100, pp. 103-301, 2019.

[3] M. J. Bommarito II, D. M. Katz, and E. M. Detterman,”LexNLP: Natural language processing and information extraction for legal and regulatory texts”, *Research Handbook on Big Data Law*, ed: Edward Elgar Publishing, pp. 216-227, 2021.

[4] K. Ahkouk and M. Machkour, “Towards an interface for translating natural language questions to SQL: a conceptual framework from a systematic review”, *International Journal of Reasoning-based Intelligent Systems*, Vol. 12, No. 4, pp.264-275, 2020.

[5] T. Bai, Y. Ge, S. Guo, Z. Zhang, and L.Gong, “Enhanced Natural Language Interface for Web-Based Information Retrieval”, *IEEE Access*, Vol. 9, pp.4233-4241, 2021

[6] F. Amato, G. Cozzolino, V. Moscato, and F. Moscato, "Analyse digital forensic evidences through a semantic-based methodology and NLP techniques", *Future Generation Computer Systems*, Vol. 98, pp. 297-307, 2019.

[7] X. Zhang, F. Yin, G. Ma, B. Ge, and W. Xiao, "M-SQL: Multi-Task Representation Learning for Single-Table Text2sql Generation", *IEEE Access*, Vol. 8, pp. 43156- 43167, 2020.

[8] T. Al-Moslmi, M. G. Ocaña, A. L. Opdahl, and C. Veres, "Named entity extraction for knowledge graphs: A literature overview", *IEEE Access*, Vol. 8, pp. 32862-32881, 2020.

[9] K. Nawab, G. Ramsey, and R. Schreiber, "Natural language processing to extract meaningful information from patient experience feedback", *Applied Clinical Informatics*, Vol. 11, pp. 242-252, 2020.

[10] E. E. B. Adam, "Deep learning based NLP techniques in text to speech synthesis for communication recognition", *Journal of Soft Computing Paradigm (JSCP)*, Vol. 2, pp. 209-215, 2020.

[11] R. K. Dwivedi, M. Aggarwal, S. K. Keshari, and A. Kumar, "Sentiment analysis and feature extraction using rule-based model (RBM)", In: *Proc. of International Conference on Innovative Computing and Communications: Proceedings of ICICC 2018*, Vol. 2, pp. 57-63,2018.

[12] B. Zhong, W. He, Z. Huang, P.E.D. Love, J. Tang, and H. Luo, "A building regulation question answering system: A deep learning methodology", *Advanced Engineering Informatics*, pp.1-11, 2020.

[13] D. Wang, J. Su, and H. Yu, "Feature extraction and analysis of natural language processing for deep learning English language", *IEEE Access*, Vol. 8, pp. 46335-46345, 2020.

[14] F. H. Hazboun , M. Owda, and A.Y.Owda, "A Natural Language Interface to Relational Databases Using an Online Analytic Processing Hypercube", *MDPI AI 2*, pp.720-737,2021.

[15] Y. Kim and S. M. Meystre, "Ensemble method-based extraction of medication and related information from clinical texts", *Journal of the American Medical Informatics Association*, Vol. 27, pp. 31-38, 2020.

[16] H. Zhong, C. Xiao, C. Tu, T. Zhang, Z. Liu, and M. Sun, "How does NLP benefit legal system: A summary of legal artificial intelligence", *arXiv preprint arXiv:2004.12158*, 2020.

[17] D. H. Maulud, S. R. Zeebaree, K. Jacksi, M. A. M. Sadeeq, and K. H. Sharif, "State of art for semantic analysis of natural language processing", *Qubahan Academic Journal*, Vol. 1, pp. 21-28, 2021.

[18] J. Du, E. Grave, B. Gunel, V. Chaudhary, O. Celebi, and M. Auli, "Self-training improves pre-training for natural language understanding", *arXiv preprint arXiv:2010.02194*, 2020.

[19] K. Affolter, K. Stockinger, and A. Bernstein, "A comparative survey of recent natural language interfaces for databases", *The VLDB Journal*, Vol. 28, pp. 793-819, 2019.

[20] C. Rodriguez, E. Marrese-Taylor, F. S. Saleh, H. Li, and S. Gould, "Proposal-free temporal moment localization of a natural-language query in video using guided attention", In: *Proc. of the IEEE/CVF winter conference on applications of computer vision*, pp. 2464-2473, 2020.

[21] W. E. Zhang, Q. Z. Sheng, A. Alhazmi, and C. Li, "Adversarial attacks on deep-learning models in natural language processing: A survey", *ACM Transactions on Intelligent Systems and Technology (TIST)*, Vol. 11, pp. 1-41, 2020.

[22] E. A. Olivetti, J. M. Cole, E. Kim, O. Kononova, G. Ceder, and T. Y.-J. Han, "Data-driven materials research enabled by natural language processing and information extraction", *Applied Physics Reviews*, Vol. 7, pp. 041317, 2020.

[23] I. Spasic, and G. Nenadic, "Clinical text data in machine learning: systematic review", *JMIR medical informatics*, Vol. 8, p. e17984, 2020.

[24] Q. Su, M. Wan, X. Liu, and C.-R. Huang, "Motivations, methods and metrics of misinformation detection: an NLP perspective", *Natural Language Processing Research*, Vol. 1, pp. 1-13, 2020.

[25] A. Silva, and M. Mendoza, "Improving query expansion strategies with word embeddings", In: *Proc. of the ACM Symposium on Document Engineering 2020*, pp. 1-4, 2020.

[26] S. T. Y. Ramadan, T. Sakib, M. A. Rahat, M. M. Hossain, R. Rahman, and M. M. Rahman, "An Integrated Embedded System Towards Abusive Bengali Speech and Speaker Detection Using NLP and Deep Learning", In: *Proc. of 25th International Conference on Computer and Information Technology (ICCIT)*, pp. 698-703, 2022.

[27] N. Varghese, and M. Punithavalli, "Lexical and semantic analysis of sacred texts using machine learning and natural language processing", *International Journal of Scientific & Technology Research*, Vol. 8, pp. 3133-3140, 2019.

[28] D. Deepa, and A. Tamilarasi, "Sentiment analysis using feature extraction and dictionary-

based approaches", In: *Proc. of Third International conference on I-SMAC (IoT in Social, Mobile, Analytics and Cloud)(I-SMAC)*, pp. 786-790, 2019.

[29] S. Islam, M. Hasan, and M. I. Jabiullah, "An Implementation of Advanced NLP for High-Quality Text-To-Speech Synthesis", *Advancement of Computer Technology and its Applications,* Vol. 4, pp.1-12, 2022.

[30] A. Guo, X. Zhao, and W. Ma, "ER-SQL: Learning enhanced representation for Text-to-SQL using table contents", *Neurocomputing 465*, pp. 359-370, 2021.

[31] A. Prasad, S. S. Badhya, Yashwanth YS, S. Rohan, Shobha G, and Deepamala N, "Enhancement of Natural Language to SQL Query Conversion using Machine Learning Techniques", *(IJACSA) International Journal of Advanced Computer Science and Applications*, Vol. 11, No. 12, pp.-494-503, 2020.

[32] D. Chandarana, M. Mathkar, A. Patil, and D. Dubey, "Natural Language Sentence to SQL Query Converter", *International Journal of Engineering Research & Technology (IJERT)*, Vol. 9, No. 3, pp.467-472, 2021.

[33] A. Kerkar , H. Ali Shaikh , P. Satam, and J. E. Nalavade, "Natural Language Query Processing for SQL And NOSQL Queries", *International Journal of Scientific & Engineering Research*, Vol. 11, No. 3, pp.1584-1586, 2020.

[34] P.Munde, S. Tambe, A. Shaikh, P. Sawant, and D. Mahajan, "Voice Based Natural Language Query Processing", *International Research Journal of Engineering and Technology (IRJET)*, Vol. 7, No. 3, pp.4696-4699, 2020.

[35] Haowu, C. Shen, Z. He,Y.Wang, and X. Xu, "SCADA-NLI:A Natural Language Query and Control Interface for Distributed Systems", *IEEE ACCESS*, Vol. 9, pp. 78108 -78127, 2021.

[36] W. An, Q. Dou, X.Zhou, and P.Jiang, "Natural Languagre-To-SQL based on Relationship Extraction", *IEEE*, pp.1219-1215, 2019.

[37] E. Smith, D. Papadopoulos, M. Braschler, and K. Stockinger, "LILLIE: Information extraction and database integration using linguistics and learning-based algorithms", *Information Systems*, Vol. 105, pp.1-15, 2022.

[38] J. Fischbach, M. Adam, V. Dzhagatspanyan, D. Mendez, J. Frattini, and O. Kosenkov, "Automatic ESG Assessment of Companies by Mining and Evaluating Media Coverage Data: NLP Approach and Tool", *arXiv preprint arXiv:2212.06540*, 2022.