International Journal of Innovative Research and Scientific Studies, 8(2) 2025, pages: 1395-1407

IJIRSS
International Journal of Innovative Research & Scientific Studies

ISSN: 2617-6548

URL: www.ijirss.com

Powered SQL education: Automating SQL/PLSQL question classification with LLMs and machine learning

Naif Alzriqat[1*], Mohammad Al-Oudat[2]

[1]*Department of Software Engineering Faculty of Information Technology Philadelphia University Amman, Jordan.*
[2]*Department of Computer Science Faculty of Information Technology Philadelphia University Amman, Jordan.*

Corresponding author: Naif Alzriqat (*Email: 202220940@philadelphia.edu.jo*)

Abstract

Mastering Structured Query Language/Procedural Language (SQL/PLSQL) is considered challenging for academic students and industrial professionals, showing a significant gap between academic preparation and industrial demands that leads both to seek solutions on Stack Overflow (SO). This research presents a novel automated framework to classify SQL/PLSQL questions and shed light on learning challenges. A new dataset was collected from SO posts, totaling 10,266 questions, and categorized into five categories—Data Definition Language (DDL), Data Manipulation Language (DML), Data Query Language (DQL), Data Control Language (DCL), and Transaction Control Language (TCL)—using the LLM GPT-4o-mini API, followed by preprocessing and applying Machine Learning (ML) techniques like Random Forest and XGBoost. Results show that Data Query Language (DQL) and Data Manipulation Language (DML) are the most challenging areas, with Random Forest and XGBoost producing the highest classification accuracy at 85.57% and 85.13%, respectively, while DDL and DCL appear less often. This research bridges the gap between academic and industrial requirements, concluding that AI-driven analysis identifies the real challenges, suggesting that the academic curriculum enhance hands-on problem-solving to meet industry needs.

Keywords: Curriculum enhancement, Database education, Database skills, Industry-academic gap, LLM, AI, SQL categorization, Stack Overflow.

DOI: 10.53894/ijirss.v8i2.5467
Funding: This study received no specific financial support.
History: Received: 3 February 2025 / **Revised:** 6 March 2025 / **Accepted:** 11 March 2025 / **Published:** 18 March 2025

Competing Interests: The authors declare that they have no competing interests.
Authors' Contributions: All authors contributed equally to the conception and design of the study. All authors have read and agreed to the published version of the manuscript.
Transparency: The authors confirm that the manuscript is an honest, accurate, and transparent account of the study; that no vital features of the study have been omitted; and that any discrepancies from the study as planned have been explained. This study followed all ethical practices during writing.
Publisher: Innovative Research Publishing

1. Introduction

Database management is considered one of the important topics in computer education at all levels and careers. "Fundamentals of Databases" and "Database Management Systems" courses provide both theoretical and practical skills. In contrast, students during their studies and recent graduates face some challenges when applying their knowledge to real-world database tasks, specifically in SQL/PLSQL queries [1]. These challenges make students and database employees seek help using online programming community platforms such as Stack Overflow (SO) to resolve the problems they face [2]. The huge number of SQL/PLSQL posts on SO shows that there is a significant gap between academic and technical requirements in the industry. One of the most common employer concerns is that new graduates lack technical experience with databases, which leads to work inefficiencies with real-world database cases [3]. Despite the fact that corporations open positions for database career opportunities, many academic institutions fail to prepare students for the real-world technical problem-solving skills required by the industry [4]. This leads to a substantial gap between what the industry needs and what academia provides. To address this issue, this research provides a data-driven framework that integrates both LLM and ML to analyze and categorize SQL/PLSQL questions from SO posts. The proposed framework involves database collection, with LLM categorizing into five main categories: Data Definition Language (DDL), which defines and modifies database structures; Data Manipulation Language (DML), which modifies data in databases; Data Query Language (DQL), which retrieves and queries data; Data Control Language (DCL), which manages database access (GRANT, REVOKE); and Transaction Control Language (TCL), which controls transaction integrity. After that, several ML techniques are applied to the categorized dataset in order to extract patterns in database challenges. Random Forest and XGBoost emerge as the most effective classifiers, achieving the highest accuracy in SQL/PLSQL question categorization.

This study contributes to database education by:

- Developing an automated framework that classifies SQL/PLSQL-related learning difficulties using AI-based categorization.
- Providing a large-scale, real-world dataset that reflects the actual SQL challenges faced by students and professionals.
- Identifying the most challenging SQL/PLSQL domains, helping educators refine database curricula to focus on practical, hands-on learning.
- Bridging the gap between academic instruction and industry needs, ensuring students gain the skills necessary for professional database roles.

This study revolves around the main research question (MRQ) with additional research questions (RQs) that guide the analysis of the SO dataset, focusing on SQL/PL/SQL related questions. The MRQ is as follows:

- MRQ: How can the categorization of SO questions enhance the understanding of challenges faced by SQL developers?
- The main research question provides the final objective of our study, which is to categorize SQL/PLSQL-related questions in a meaningful way, aiding in the identification of key topics and challenges faced by developers. Based on the identified challenges in the SQL/PLSQL domain, the following research questions are formulated:
- RQ1: How can Large Language Models (LLMs) be leveraged to automatically categorize and interpret SQL/PLSQL related questions from SO?
- RQ2: Which categories (DDL, DQL, DML, DCL, and TCL) are most represented in SQL/PLSQL-related SO questions?

The rest of this paper is organized as follows. Section 2 presents the literature review, summarizing existing research related to our work. Section 3 describes the methodology, including data collection, preprocessing, and classification techniques. Section 4 discusses the results and key findings. Finally, Section 5 provides the conclusion and outlines directions for future research.

2. Literature Review

Recently, the curriculum and industry demand for skills in databases have acquired an increasingly widening gap, highlighting the need for new approaches to curriculum development. The literature emphasizes various strategies for closing this gap, including competency-based education, project-based learning, and formative assessments, which have been addressed in Mahardhani, et al. [5]. It becomes clear from this research that close interaction with all stakeholders, especially industry players, is vital for achieving a curriculum that is relevant in the job market. Similarly, Li and Hu [6] present the success of school-enterprise collaboration approaches, including on-the-job internships and joint R&D programs, in fostering practical, application-oriented skills. However, these approaches often lack scalability and fail to leverage large, community-driven datasets like SO to analyze emerging trends.

SQL education, in particular, faces its particular challenge in that students have difficulty with syntax rules, translating natural language questions into SQL queries, and debugging errors, as noted in Miedema [7]. There are tools such as SQLVis with graphical debugging capabilities, but such tools don't tackle the challenge of adaptive, real-time feedback tailored to individual knowledge gaps. Further innovation is demonstrated by Mittelhessen [8] which offers a hybrid static-dynamic evaluation model for delivering in-depth, individual feedback. While effective, this approach relies on predefined datasets and lacks the ability to adapt dynamically to new patterns and challenges emerging in real-world contexts such as SO.

Machine learning and data analytics are now very strong tools for curriculum development. To begin with, Yahya, et al. [9] uses machine learning classifiers to predict the skills that will be needed in future job markets, showing the potential of predictive analytics in mapping out curricula according to what is required by employers. In the same way, Nizar, et al. [10]

employs machine learning techniques in assessing students' technical, soft, and life skills. deficiencies. However, these methods require structured datasets that do not take advantage of user provided information from the internet which could be found on websites like SO. Utilizing SO as a data source has been found in studies to have potential in improving curriculum design. As exemplified in Venigalla, et al. [11] and Beyer, et al. [12] posts are classified by machine learning models into pre-defined categories. In automated categorization, these studies are not concerned with how these classifications may help aid in curriculum development. Similarly, Marcal, et al. [13] uses topic modeling to extract key concepts from SO posts to introduce in their teaching materials which is relevant to the industry. However, its reliance on static models limits adaptability to evolving trends in industry challenges.

As is shown in recent works [14] using LLMs in educational tasks such as complex, multi-feature data extraction across various domains, their use in educational environments is picking up pace. The ability of GPT-4o-mini to classify SQL/PLSQL-related questions into educational categories provides an automated, efficient method for generating high-quality labeled datasets, reducing the reliance on manual labeling. Contrary to this, this advancement addresses the shortcomings of prior studies by offering a more comprehensive, dependable, and flexible way to dataset labeling, which is essential for the training of machine learning models in educational settings.

Our proposed framework addresses these limitations by integrating SO data with large language models and machine learning techniques. This approach enables the dynamic classification of SQL/PL/SQL related challenges, facilitating the identification of granular trends in student difficulties and emerging industry needs. By categorizing questions into meaningful educational categories such as DDL, DML, and TCL, the framework provides actionable insights for curriculum refinement. Unlike previous studies, it focuses explicitly on bridging the gap between academic education and real-world industry demands, ensuring curriculum enhancements remain relevant and responsive to current trends.

Table 1.
Summary of Reviewed Literature.

Source/Reference	**Source of Dataset**	**ML Algorithms Used**	**Task/Objective**	**Key Findings**	**Limitations**
A New Approach to Curriculum Development: The Relevance of the Higher Education Curriculum to Industry Needs	N/A	N/A	Analyzing the relevance of higher education curricula to industry needs	Project-based learning, competency-based approaches, and industry collaboration are essential for improving curriculum relevance.	Challenges in integrating industry needs into curricula and ensuring continuous updates.
Exploration and Practice of Software Curriculum Reform Based on the Integration of Industry and Education	N/A	N/A	Integration of industry and education for engineering education reform	Collaboration between universities and enterprises enhances student preparedness for industry challenges.	Limited scalability and resource-intensive implementation.
Toward a Fundamental Understanding of SQL Education	N/A	N/A	Understanding common challenges in SQL education	SQL learners struggle with syntax, semantic understanding, and error resolution due to the complexity of the language.	No experimental validation of proposed solutions.
ItsSQL: Intelligent Tutoring System for SQL	SQL student queries dataset	Hybrid static-dynamic evaluation approach	Developing an intelligent tutoring system for SQL education	Providing multiple reference solutions and individual feedback significantly improve learning outcomes.	Dependence on reference solution quality; self-learning capabilities require refinement.
Mapping Graduate Skills to Market Demands	3,831 computer science graduates` data from alumina surveys and hiring agencies	Support Vector Machines, Neural Networks	Predicting and aligning graduate skills with job market requirements	SVM and neural networks achieved high accuracy (82% and 88%) in predicting skill-job alignment.	Potential dataset bias; limited scope for emerging job roles.
A Random Forest Model for Prediction of Software Engineering Skill Set	Skill assessment dataset from computer science students	Random Forest, PCA, SHAP (Explainable AI)	Predicting software engineering skill sets among students	The PCA-enhanced Random Forest model achieved superior accuracy in skill prediction.	Potential overfitting; requires further validation with diverse datasets.

3. Methodology

The main objective of the research is to develop an automated framework that bridges the gap between academic and industrial requirements for SQL/PLSQL. The proposed framework contains a sequence of phases: data acquisition, preprocessing, categorization, model training, and performance evaluation, as illustrated in Figure 1.

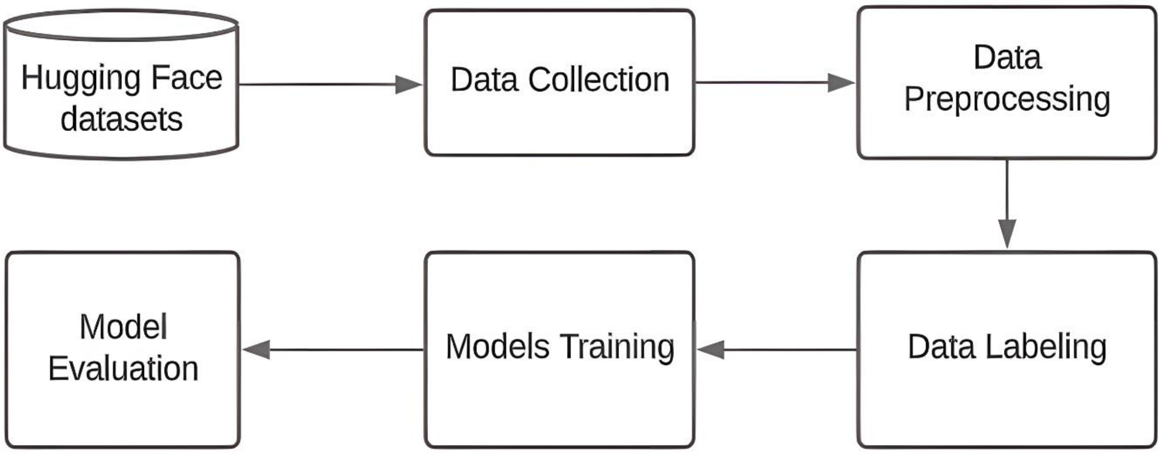


Figure 1.
Proposed system architecture.

3.1. Data Collection

A dataset of SO questions was collected from huggingface[1] ,, sourced from the Internet Archive StackExchange Data Dump. This collection encompasses every post submitted to SO prior to June 14, 2023, totaling approximately 60 million entries. The data was systematically processed and imported into a SQL Server database for optimal management and analysis. To exclude unnecessary details and present the relevant information, posts tagged with [sql], [plsql], and [database] were filtered to select only the posts posing different kinds of real situations faced by SQL/PLSQL programmers. This filtration process was designed to capture the typical issues and scenarios encountered by professionals working with SQL technologies. After this filtering process, 10,737 questions remained for examination. To guarantee their accurate representation of all five essential SQL categories—DDL (Data Definition Language), DQL (Data Query Language), DML (Data Manipulation Language), DCL (Data Control Language), and TCL (Transaction Control Language)—these questions were subjected to careful manual validation. Additionally, to refine the dataset, specific queries that contain keywords related to transactional control, such as COMMIT, ROLLBACK, and SAVEPOINT, were used to ensure the presence of suitable entries. This meticulous strategy was crucial for validating the dataset's comprehensiveness and ensuring its exact correspondence with SQL/PLSQL related subjects.

$$https://huggingface.co/datasets/mikex86/stackoverflow-posts^{1}$$

3.2. Preprocessing

In our data preprocessing, we give first priority to keeping the meaning of the context intact as it greatly enhances the performance of contextual word embeddings. The literature Peters, et al. [15] and Devlin, et al. [16] evaluates the significance of context preservation and its relevance to improving model performance, and consequently, to the accuracy of representation. Some of the major data preprocessing actions performed in order to have data suitable for the classification analysis are:

o Tokenization: The process of breaking down text into words or phrases.

o Stopword Removal: Elimination of common words like "and," "the" to concentrate on significant terms.

o Lemmatization: Converting words into their base or root forms, e.g., changing "queries" to "query" so that they may be treated uniformly throughout.

o Text Consolidation: Combining all parts (title and body) of a question together so that it can be better analyzed. By taking these steps, we ensure that the dataset is clean and standardized for subsequent classification and modeling.

3.3. Data Labeling

Accurate class labeling of SQL/PLSQL related questions is an integral part of knowing about problem encountered by SQL developers. Traditional methods overreliance on conventional labeling and conventional ML classifiers [17]. However, these approaches produced poor accuracy, which can be owed to native complexity and diversity in SQL/PLSQL related question, such as casual language and variable structures for contents. Beyer, et al. [12] employed manual classification and in contrast, Venigalla, et al. [11] employed NLP techniques such as LDA, LSA, TF-IDF, but yet, more work is yet to be developed in effective obtainment of labeling SQL/PLSQL related questions. In an attempt to counter such a challenge and ease labeling of information, GPT-4o-mini, a powerful large language model, was used in our work, and its use surprisingly eased overreliance on conventional labeling through an automation of label extraction, and in consequence, accelerated development of a high-quality and in-depth labeling for training machine learning models. This automated approach aligns with Caruana, et al. [18] who demonstrated that GPT-based model outshines in generating and recreating information through self-supervision, and in consequence, in improvement of accuracy and efficiency in labeling work. Besides, Rathje, et al. [14] confirms GPT-3 and GPT-4 model effectiveness in working with complex, multi-feature work in extraction in several languages, a fact supporting GPT-4o-mini suitability for use in our work. Each one of 10, 737 questions in our dataset underwent processing through GPT-4o-mini, with a specifically designed prompt used for extraction of 5 predefined categories critical for a thorough realization of each question. Prompt engineering involved iterative improvement and testing for accuracy and suitability in extracted information, and a demonstration of GPT-4o-mini prompt template is in Table 2.

Table 2.
GPT-4o-mini prompt template.

```
You are a highly knowledgeable SQL domain expert with a deep understanding of SQL concepts.
Your task is to analyze the given question and determine the **single most relevant SQL category** it belongs to base on its primary focus and intent.
Choose from the following categories:
- **DDL (Data Definition Language) **: Questions about defining or modifying database structures, such as CREATE, ALTER, or DROP operations.
- **DML (Data Manipulation Language) **: Questions related to manipulating data, including INSERT, UPDATE, DELETE, or MERGE operations.
- **DQL (Data Query Language) **: Questions focused on querying data, such as SELECT statements for data retrieval.
- **DCL (Data Control Language) **: Questions about managing database permissions and access control, such as GRANT or REVOKE statements.
- **TCL (Transaction Control Language) **: Questions involving transaction management, including COMMIT, ROLLBACK, or SAVEPOINT.
Carefully evaluate the question's content and intent, then select the **one category** that most closely aligns with the question.
If the question is unclear or does not belong to any category, respond with "Unclear".
Here is the question:
(text)
Provide your output as the name of the most appropriate category (e.g., "DDL"). Do not include any explanation or additional text in your response.
```

Source: The study eliminated 471 questions with unclear categories so the final dataset consisted of 10,266 questions.

3.4. Machine Learning Classification

The labeled dataset was used to train and evaluate multiple machine learning models. The following steps were performed:

3.4.1. Feature Representation

For feature representation of SQL/PLSQL-related questions, TF-IDF (Term Frequency-Inverse Document Frequency) was applied to convert the text into numerical features [19]. This method specifically proves to be valuable in text analysis as it promotes the terms that have more importance in the context of SQL queries, while dismissing those common words that do not contribute much to the classification task. This technique was used to transform the SQL/PLSQL-related questions into a matrix of numerical values, where each entry corresponds to the weighted importance of a word in a specific question. By applying TF-IDF, we ensured that the models could focus on the most informative terms, which are crucial for distinguishing between SQL categories such as DDL, DML, and DQL. The TF-IDF matrix was then inputted into several machine learning algorithms (Logistic Regression, Random Forest, SVM), giving them the chance to learn the subtleties in the data efficiently.

3.4.2. Model Selection

The study conducted an evaluation of multiple machine learning systems for categorizing questions regarding SQL into predefined categories. Below is a description of each model used in the experiments and their application to SQL question classification:

- Support Vector Machine (SVM): For SQL question classification, SVM is employed to find the optimal hyperplane that separates questions into their respective categories [20]. The model establishes maximum margin boundaries between classes by implementing Radial Basis Function (RBF) and other kernel functions for non-linear data separation. This allows the model to learn about the complex relationships that are typically found in the SQL query data. Regularization methods are applied for preventing overfitting thus, the model will generalize well on new and unseen SQL questions.
- Naive Bayes: Based on Bayes' Theorem, Naive Bayes assumes independence between features, which is effective for classifying SQL questions [20]. In training, the model computes conditional probability of each category given certain SQL/PLSQL related words are present. This probabilistic approach is efficient for large datasets, and it has the good property of fast classification, particularly for large vectors characterizing high-dimensional feature sets, as is the case with SQL question data.

- Logistic Regression: Logistic Regression is a linear model used to predict the probabilities of SQL questions belonging to different categories. The model applies a logistic function to map the input features to a categorical outcome [21]. It is particularly effective in cases where the relationship between the features and the categories is linear. The model was tested on various SQL categories proved efficient for classifying SQL questions with simpler structures.
- Random Forest: Random Forest creates number of decision trees, each of which is trained by random subset of the data, in order to prevent overfitted and achieve better generalization [20, 21]. By aggregating the results of all trees, Random Forest provides robust classification of SQL/PLSQL related questions. Randomness used during training is beneficial to capture various patterns in the data, which is capable of coping with high dimensional nature of SQL question datasets.
- XGBoost: XGBoost is a gradient boosting-based algorithm that uses decision trees to enhance SQL question classification and iteratively improve on the decision tree. It easily deals with vast data sets and provides high-dimensional SQL features with L1/L2 regularization, tree trimming, and also parallel processing [22]. XGBoost produces high precision and generalizability by alleviating residual errors, which turn out to be effective for categorizing complex SQL questions.

3.5. Handling Class Imbalance

We applied the hybrid resampling technique that combined the Synthetic Minority Over-sampling Technique (SMOTE) with Tomek Links under-sampling (SMOTE-Tomek) to handle class imbalance and enhance model performance across all categories [23]. The combination of SMOTE with Tomek Links helps improve class separability by generating new minority class samples and eliminating border samples that cause class overlap. The resampling procedure was limited to the training data so that the test data maintained its real-world distribution characteristics. We calculated class weights to reduce the remaining imbalance between classes because these weights allowed the models to learn from underrepresented classes without biasing their predictions toward majority classes. The technique produced substantial enhancements in both recall and F1-score measurements, thus guaranteeing proper representation of all categories in predicted results. The models showed excellent accuracy results but poor recall towards minority classes when no balancing methods were used, which indicated potential bias in learning. The combination of SMOTE-Tomek with class weighting produced balanced classification results across all categories, which improved the ROC-AUC scores and confusion matrices. The methodology we applied shows consistency with findings from previous research about how SMOTE-based techniques help imbalanced datasets [23], while Tomek Links filtering ensured that synthetic data added robustness instead of noise to the classification model.

3.6. Hyperparameter Tuning

Hyperparameter tuning is an important step that helps us to better optimize our machine learning models to increase their performance in terms of classification accuracy and generalization [24]. In our experiments, we used GridSearchCV to thoroughly scan for the ideal hyperparameter configuration for our models [25]. GridSearchCV was used for Logistic Regression, Random Forest, SVM, and XGBoost because it fully explores all the parameter combinations that are predefined with cross-validation. This approach ensures the selection of the hyperparameter set that has the best performance on the model, even though it is computationally expensive to do. Given that Naïve Bayes does not have many tunable hyperparameters, the default settings provided the best results, and further optimization was unnecessary. Through hyperparameter tuning, a significant increase in the accuracy, F1-score, and ROC-AUC of the model is observed. The best hyperparameters for each model are listed in Table 3.

Table 3.
Best Hyperparameters for Selected Models

Algorithm	Hyperparameters	Set Value
Logistic Regression	class_weight	'balanced'
	max_iter	1000
Random Forest	class_weight	'balanced'
	n_estimators	100
	random_state	42
SVM	class_weight	'balanced'
	probability	True
	kernel	'rbf'
	C	1.0
	random_state	42
XGBoost	learning_rate	0.1
	n_estimators	100
	eval_metric	'mlogloss'

GridSearchCV enabled us to optimize model parameters efficiently and was beneficial for enhancing classification performance, particularly with Random Forest and XGBoost, which maximize the hyperparameter space. Additionally, there are some other optimizations that also reflect the importance of parameter selection to gain SQL classification tasks and to develop a robust model.

3.7. Training and Testing

The preprocessed datasets for questions related to SQL\PLSQL split into training and testing datasets for model performance and generalizability testing. 80:20, 80 for training, and 20 for testing, 80:20 is a general practice in machine learning [26] providing an ideal balance between providing enough data to the model in a position to learn and having a portion for testing its performance with new, unseen data

3.8. Evaluation Metrics

Models were assessed using precision, recall, F1-score, and overall accuracy to ensure a comprehensive performance evaluation. To evaluate the performance of the classification models in categorizing SQL/PLSQL-related questions, a set of metrics specifically designed for multi-class classification was used. These evaluations present a thorough assessment of how well the models perform in classifying the correct SQL category (DDL, DML, DQL, DCL, TCL) for a given question.

Accuracy: Accuracy is the basic performance metric, in which it is an overall rate of correct prediction generated by a model. It can be also computed by dividing the number of correct predictions by the total number of predictions, like in Equation 1. For this paper, accuracy was taken as one of the significant factors for picking the finest model to be used for any type of SQL/PLSQL related question. In other words, it measures total performance within a certain model and the ability in terms of correct classification of a set of SQL/PLSQL questions.

$$Accuracy = \frac{Number\ of\ Correct\ Predictions}{Total\ Number\ of\ Predictions} \qquad (1)$$

Precision: Precision measures how well the model distinguishes positive samples. It is calculated as the ratio of true positives (TP) to all predicted positives (TP + FP). Given that false positives have a high cost, this metric is very important. For instance, in educational scenarios where misclassifying a question can result in inappropriate curriculum suggestions, precision is crucial. The precision formula is given in Equation 2.

$$Precision = \frac{TP}{TP + FP} \qquad (2)$$

Recall, also referred to as sensitivity, measures how well the model can detect all the positive classes from the dataset. It is defined as the ratio of true positive predictions (TP) over all actually positive items (TP + FN). This metric is crucial in education, as missing important questions will lead to knowledge gaps in the curriculum. The recall formula is given in Equation 3.

$$Precision = \frac{TP}{TP + FN} \qquad (3)$$

F1-Score: The F1-score combines the measures of precision and recall into one metric that is their harmonic mean, as shown in Equation 4. It is a balanced measurement of model performance, especially when there is an inequality of classes. In this research, the F1 score is regarded as very vital for assessing the effectiveness of the model to execute effectively across all categories to avoid having a prior choice or adding up to either precision or recall. This is especially important when dealing with multi-class classification of SQL/PLSQL related questions, where some categories could have fewer samples than others.

$$F1 - Score = 2 \times \frac{Precision \times Recall}{Precision + Recall} \qquad (4)$$

The combination of accuracy, recall, and precision, as well as the F1-score, provides a full picture of model capabilities. By testing for these metrics, we can confirm that our model is good overall (accuracy) and that it handles the subtleties of classifying SQL/PLSQL questions well, finding a balance between choosing relevant categories (recall) and avoiding irrelevant ones (precision).

The same has been used in testing for performance in each of the algorithms used in this work, including Logistic Regression, Random Forest, Support Vector Machines, and Naive Bayes. With the use of these metrics, we aim to identify the most effective model for use in the automation of the classification of questions related to SQL, ultimately facilitating the improvement of database education and curriculum development.

4. Results and Discussion

The main goal of this research was to analyze SQL/PLSQL-related questions from SO through machine learning models and classify them using a LLM. The objective was to find out what the key challenges for learners are and to provide insights for enhancing database education. This section presents the study’s findings in relation to the research questions and discusses their implications.

4.1. Model Performance

The evaluation of classification models included training and testing five machine learning algorithms which comprised Logistic Regression, Naïve Bayes, Random Forest, Support Vector Machine (SVM) and XGBoost. The performance metrics accuracy and F1-score along with ROC-AUC score for all models appear in Table 4.

Table 4.
Model Performance Comparison

Model	Accuracy	ROC AUC	F1-Score
Logistic Regression	84.16%	0.956	78.71%
Naïve Bayes	77.45%	0.923	72.41%
Random Forest	85.57%	0.962	81.59%
SVM	83.98%	0.954	78.43%
XGBoost	85.13%	0.960	81.02%

Among these, Random Forest obtained the highest accuracy (85.57%) and F1-score (81.59%), while XGBoost was the second best (85.13% accuracy, 81.02% F1-score). Naïve Bayes had the lowest performance in all metrics, confirming its limitations in text classification tasks that include contextual dependencies. The Model Accuracy Comparison is shown in Figure 2, serving as a visual representation of each model's accuracy. This reinforces once again that ensemble models (Random Forest and XGBoost) performed the best, while Naïve Bayes was the worst.

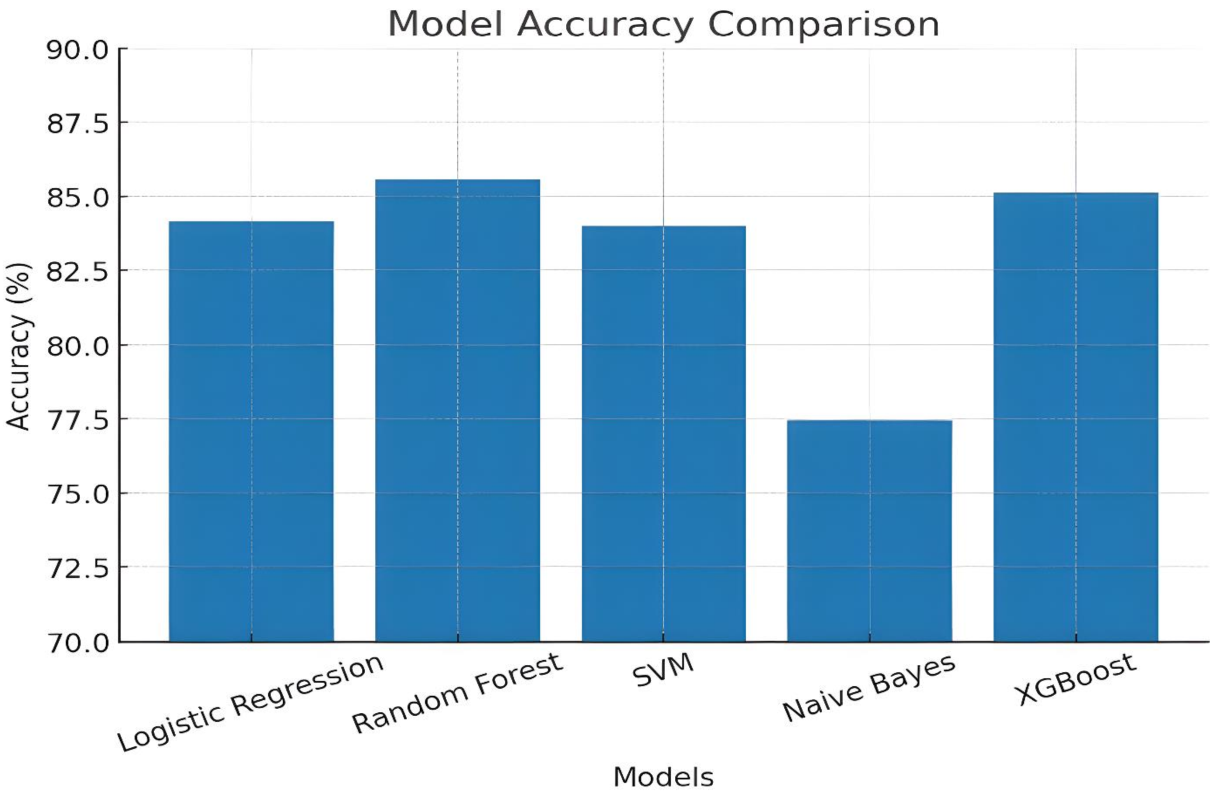
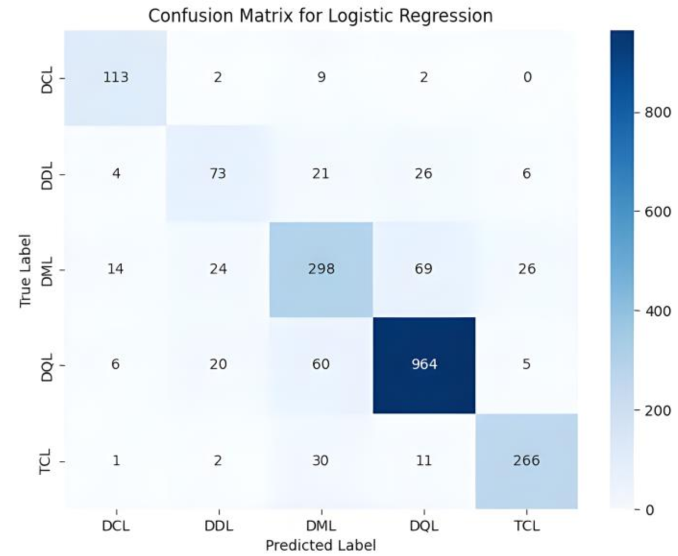


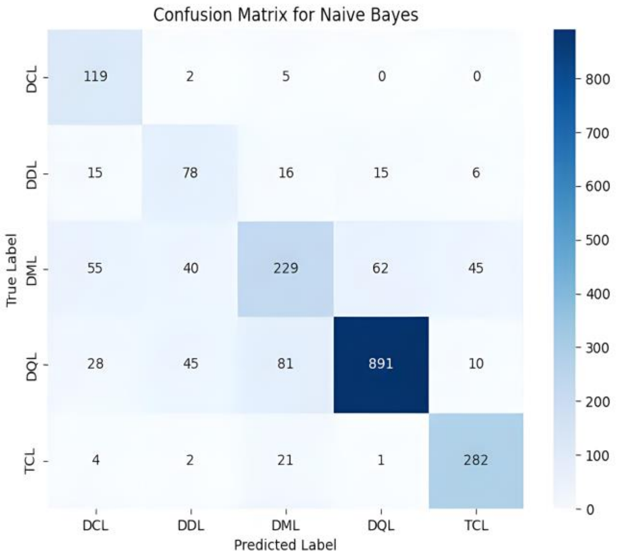
Figure 2.
Model Accuracy Comparison.

4.2. Classification Performance Analysis

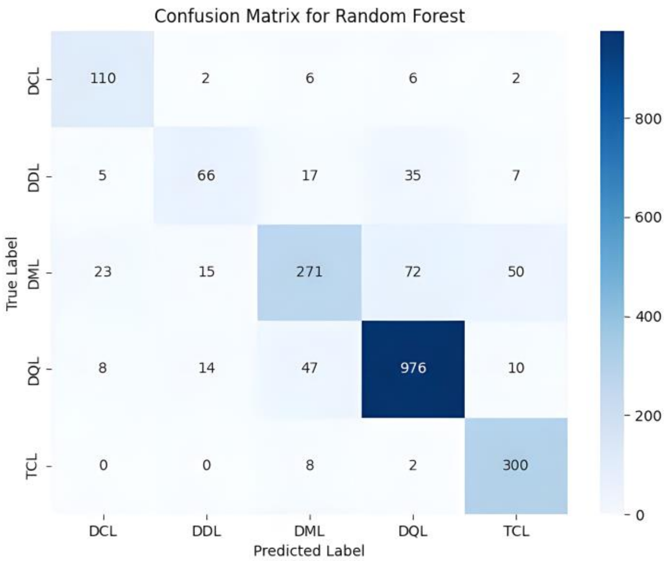
To have a better understanding of the classification behavior of the models, we have looked through their confusion matrices (Figure 3) and precision-recall curves (Figure 4). The confusion matrices show the performance of each model with discrimination of the five different SQL types: DDL, DML, DQL, DCL, and TCL. The precision-recall curves provide additional insight into classification effectiveness across varying decision thresholds. Among all categories, the DQL category showed the highest accuracy at its level in the confusion matrix among all classification techniques applied. Logistic Regression showed 92.3% correct results, which was surpassed by 94.8% and 94.3% accuracy achieved by Random Forest and XGBoost, respectively. The Random Forest and XGBoost models showed top performance at distinguishing DQL from DML since they demonstrated strong accuracy measurements and F1-scores. In contrast, Naïve Bayes struggled the most, often misclassifying DML as DQL. As a result, it managed only 72.4% accuracy when distinguishing DML-related questions. The DDL produced more errors due to its smaller number of related questions, not because of the understanding of SQL concepts. Logistic Regression made 24.6% errors when assigning DDL questions to the DML category, yet Random Forest's error rate was 19.8% in DDL. However, this does not necessarily suggest that students find schema design more challenging. Instead, it likely indicates that DDL-related queries are less commonly encountered in practice. The models SVM and Logistic Regression showed equal results when classifying between DDL and DML statements but frequently mixed up DCL and TCL records. The two models showed good classification results for DQL tasks but faced minor misjudgments between DDL and DML categories. These data patterns repeat through both sets of precision-recall charts. Random Forest and XGBoost demonstrated the most balanced precision-recall curves, highlighting their ability to maintain high classification performance across all categories. Specifically, Random Forest achieved a precision of 88.2% and recall of 85.5%, while XGBoost had a precision of 87.5% and recall of 84.9%. On the other hand, Naïve Bayes had the weakest precision-recall performance, which reflects its difficulty in differentiating between SQL categories with overlapping features. Throughout all models, DDL always produced lower accuracy results. The Logistic Regression model reached 74.3% successful results for DDL tasks, but the SVM algorithm achieved 76.1%. Models perform poorly with DDL tasks mainly because the dataset does not offer enough examples to teach this kind of database change. The questions indicate that when students study DDL, they have less practical difficulty than they do with DML and DQL. By combining insights from the confusion matrices and precision-recall curves, we can conclude that machine learning models—particularly ensemble-based methods like Random Forest and XGBoost—are highly effective in classifying SQL/PLSQL related questions from Stack Overflow. However, to better align database education with areas where students struggle the most, there should be a greater focus on query formulation and data manipulation, as these remain the most common sources of difficulty.



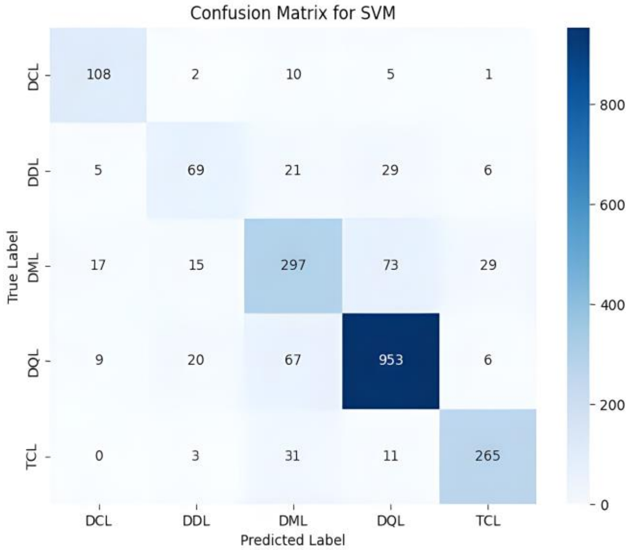
Confusion Matrix for Logistic Regression



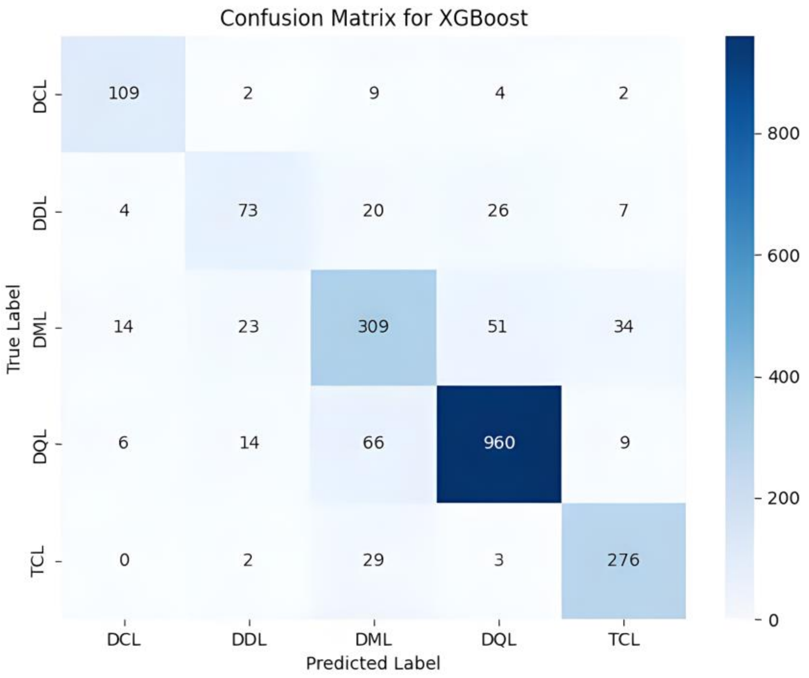
Confusion Matrix for Naive Bayes



Confusion Matrix for Random Forest

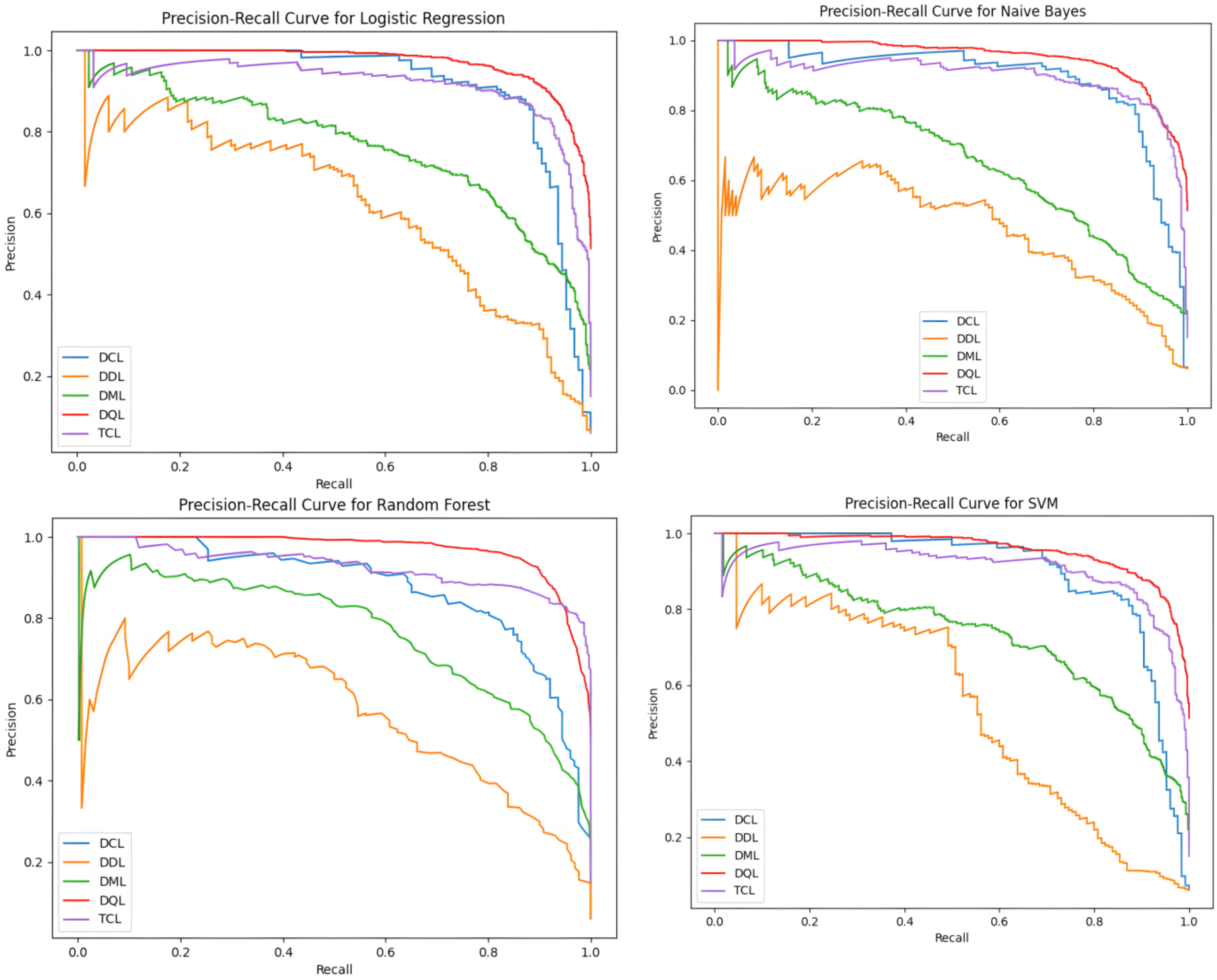


Confusion Matrix for SVM



True Label \ Predicted Label	DCL	DDL	DML	DQL	TCL
DCL	109	2	9	4	2
DDL	4	73	20	26	7
DML	14	23	309	51	34
DQL	6	14	66	960	9
TCL	0	2	29	3	276

Figure 3.
Confusion Matrix.



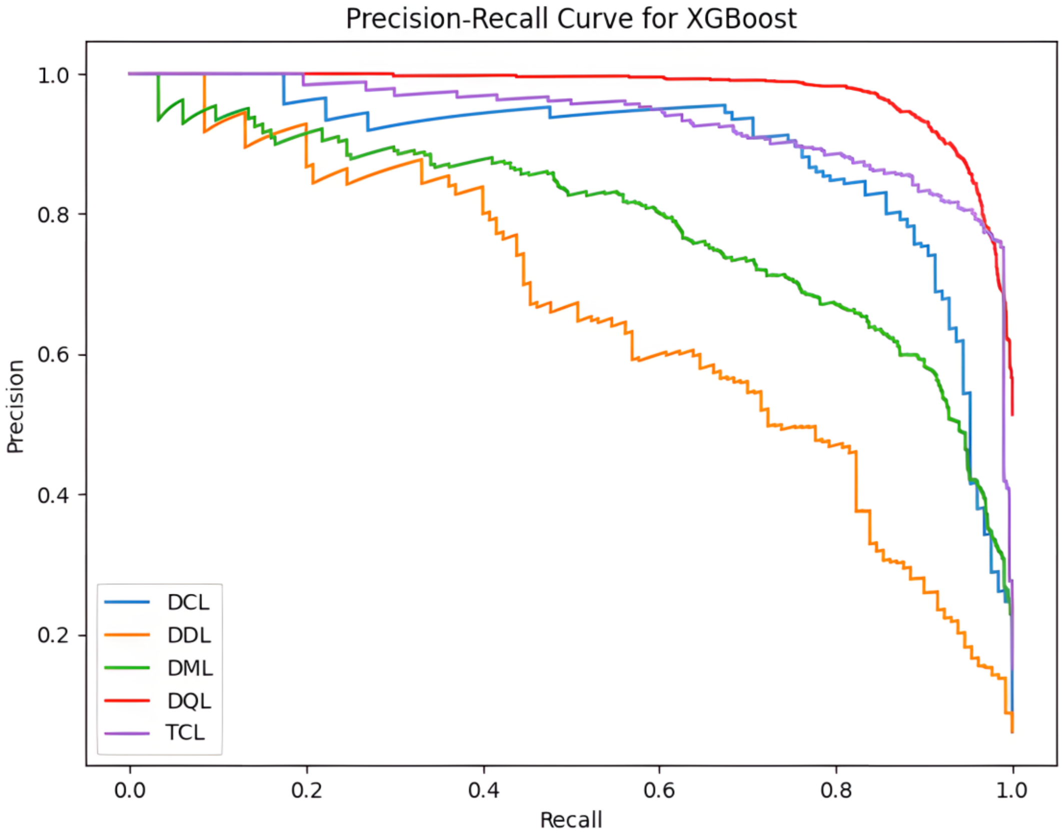


Figure 4.
Precision-Recall Curves.

4.3. Addressing the Research Questions

MRQ: How can the categorization of SO questions enhance the understanding of challenges faced by SQL developers?

The classification of SQL/PLSQL-related questions makes it possible to analyze the learning difficulties faced by SQL developers through an organized framework. Our research used LLM-based classification to detect specific challenges within each SQL category before conducting a machine learning evaluation. The results demonstrate that:

Students face their greatest SQL difficulties in performing Data Query Language (DQL) and Data Manipulation Language (DML) operations according to their questions on Stack Overflow.

Students and professionals are asking more questions about DDL which indicates schema design and database structuring presents greater difficulties to them.

Using LLM models together with machine learning algorithms creates an efficient system for SQL question classification, which enables educators to detect educational patterns for curriculum improvement.

The developed categorization system generates practical information that helps instructors identify difficult SQL concepts for academic use to enhance database education methods.

RQ1: How can Large Language Models (LLMs) be leveraged to automatically categorize and interpret SQL/PLSQL related questions from SO?

LLMs, particularly GPT-4o-mini, were used to label SQL/PLSQL-related questions into five predefined categories: DDL, DML, DQL, DCL, and TCL. The automatic classification process led to the creation of a labeled dataset for training machine learning models. The research outcome proves that integrating LLM-produced labels with machine learning systems leads to substantial improvements in classification precision. The Random Forest and XGBoost models achieved exceptional precision in SQL category identification through their accurate performance, which reached 85% and above, and their ROC-AUC scores exceeded 0.96. The results demonstrate that LLMs establish a solid base for SQL question classification, but machine learning models effectively increase classification performance. The efficiency of LLMs provides scalable analysis of educational research datasets through categorization processes that are essential for large-scale research. The feasibility of using LLMs as part of automated educational tools for enhanced curriculum assessment has been validated by these test results.

RQ2: Which categories (DDL, DQL, DML, DCL, and TCL) are most represented in SQL/PLSQL related SO questions?

The analyses show that DQL (SELECT queries) stands as the most common category because all models achieve their best performance in this area. The analysis demonstrates that data modification challenges through DML commands (INSERT, UPDATE, DELETE) occur frequently after DQL. As shown in Figure 5, the distribution in DQL, then DML, then TCL, then DDL, and then DCL confirms that learners prioritize issues related to data retrieval and modification over schema design and access control.

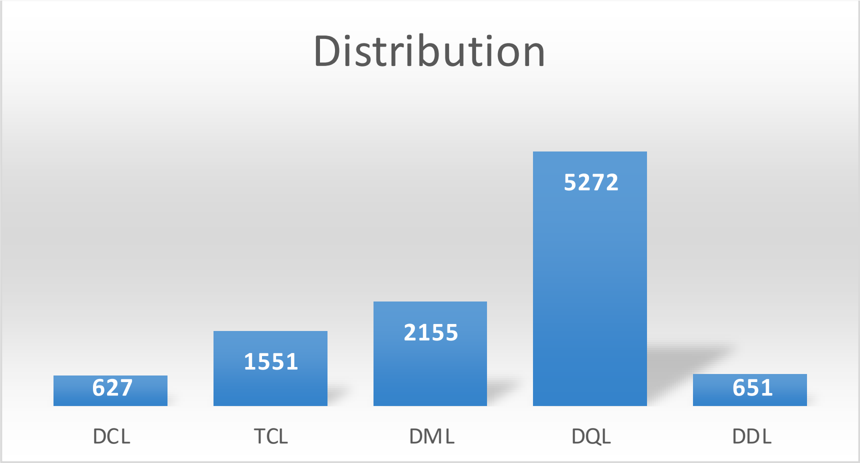


Figure 5.
Final distribution of categories.

Students, along with experts, experience minimal difficulties regarding schema design and modification tasks, which are reflected in the limited number of questions related to DDL. The scarcity of DDL-related questions indicates that such problems occur infrequently in real-world practice since students prioritize query writing and data management (DML and DQL). Students struggle most with DML and DQL concepts based on the majority of questions in our dataset. DDL remains important, but the need for troubleshooting its tasks may be lower. Database courses need to provide practical exercises for all database languages, including DDL, together with DML and DQL, for students to develop a complete understanding of the subject.

4.4. Implications for Database Education

The findings indicate that database education currently faces multiple knowledge gaps. Higher education institutions should enhance their implementation of practical SQL problem-solving tasks, with a strong focus on DML and DQL operations, as these areas present the most learning difficulties. While other domains remain essential for database management, the lower number of issues reported suggests that students may encounter fewer challenges with them. However, incorporating more hands-on exercises can still be beneficial in reinforcing foundational database skills. Academic institutions should align their SQL instruction with industry requirements by integrating practical case studies and project-based learning. This study applies machine learning techniques to analyze data generated from LLM-based classification, identifying key challenges in SQL education. The results indicate that DQL and DML are the primary areas where students struggle, emphasizing the need for practical, application-driven database coursework. Developing adaptive learning systems that adjust lessons based on individual student needs can further improve SQL education.

5. Conclusion and Future Work

SQL skills remain a crucial yet difficult skill for students shifting from the academic to the industrial field, as can be seen from the huge SQL queries on SO. This study proposes a new framework based on data that combines Large Language Models (LLMs) and machine learning to analyze and categorize them in order to understand the particular nature of the difficulties to which the learners are confronted. With the help of classifying SQL questions into five main areas—DDL, DML, DQL, DCL, and TCL—this research provides empirical proof of knowledge gaps in SQL education. The results suggest that the students find data retrieval (DQL) and data modification (DML) the most difficult, although it seems that schema definition (DDL) and access control (DCL) are less difficult. These results of research uncover a disconnect between academic programs and the working industry, so there is a need for targeted improvements in SQL instruction. The research contributes to the field by testing the effectiveness of AI-driven classification in educational analysis. Using GPT-4o-mini for automated labeling and machine learning for pattern detection, the research develops a scalable process for discovering, identifying, and addressing SQL learning challenges. Among the machine learning models experimented with, Random Forest and XGBoost resulted as the most accurate classifiers in categorizing SQL questions. These findings indicate that AI and machine learning should be integrated into curriculum design to allow educators to better use instruction and learning in terms of data. The end result of this research is to provide improvement in database education in that instruction is brought in line with requirements in the industry. By identifying specific areas where students struggle, this research builds a foundation for more targeted, focused learning experiences. However, though this system is able to accurately assess areas that need curricular adjustment, applying and affirming proposed reforms is a crucial next step.

Looking ahead, the practical use of this research will be dependent on future work to apply and test its outcomes. The short-term next steps involve collaboration with universities to implement proposals of the framework in a trial program, using modified SQL instructions based on the outcomes of the research, and measuring its effects using student surveys and outcomes. In addition, structured workshops with educators, database professionals, and students will be conducted to further develop the curriculum so that it is accurate to both academic requirements and industry needs. A long-term evaluation is also important to learn the degree and length of time these curriculum changes help to prepare students for professional careers, such as monitoring student performance over time and collecting feedback from employers. Additionally, expanding the data scope to include extra platforms like GitHub discussions, academic forums, and corporate training materials will offer deeper insight into SQL challenges across different contexts. By exploring these future directions, this research seeks to close the disconnect between academic education on SQL and the expectations of the industry, ultimately leading to better teaching of databases in education.

References

[1] R. Alkhabaz, Z. Li, S. Yang, and A. Alawini, *Student's learning challenges with relational, document, and graph query languages*. United States: Association for Computing Machinery, 2023, pp. 30–36.

[2] S. Kim, M. Di Penta, and T. Zimmermann, "Mining software repositories," in *10th Working Conference on Mining Software Repositories (MSR): Proceedings, May 18-19, 2013, San Francisco, CA, USA, IEEE*, 2013.

[3] P. Orr, L. Forsyth, C. Caballero, C. Rosenberg, and A. Walker, *A systematic review of Australian higher education students' and graduates' work readiness*. Australia: Routledge, 2023.

[4] O. S. Parihar and K. Singh, "Knowledge and skills for data science professionals," 2023.

[5] A. J. Mahardhani, B. Nadeak, I. M. Hanika, I. Sentryo, and R. Kemala, "A new approach to curriculum development: The relevance of the higher education curriculum to industry needs," *International Journal of Educational Research Excellence,* vol. 2, no. 2, pp. 501-509, 2023. https://doi.org/10.55299/ijere.v2i2.620

[6] Y. Li and H. Hu, "Exploration and practice of the integration of industry and education in high-level research universities and local application-oriented universities in China," presented at the 2023 2nd International Conference on Educational Innovation and Multimedia Technology (EIMT 2023), Atlantis Press, 2023.

[7] D. Miedema, *Toward a fundamental understanding of SQL education*. United States: Association for Computing Machinery, 2023, pp. 64–68.

[8] T. H. Mittelhessen, "ItsSQL: Intelligent tutoring system for SQL ItsSQL," Intelligent tutoring system for SQL Working Paper No. 20, 2023.

[9] A. E. Yahya, W. M. Yafooz, and A. Gharbi, "Mapping graduate skills to market demands: A holistic examination of curriculum development and employment trends," *Engineering, Technology & Applied Science Research,* vol. 14, no. 4, pp. 14793-14800, 2024. https://doi.org/10.48084/etasr.7454

[10] J. Nizar, R. Sharmila, and K. Jaseena, "Prediction and assessment of software engineering skillset among computer science students using convolutional neural networks through explainable Ai," *Journal Of Theoretical And Applied Information Technology,* vol. 102, no. 21, pp. 1-21, 2024.

[11] A. S. M. Venigalla, C. S. Lakkundi, and S. Chimalakonda, "SOTagger - Towards classifying stack overflow posts through contextual tagging," in *Proceedings of the International Conference on Software Engineering and Knowledge Engineering, SEKE, Knowledge Systems Institute Graduate School*, 2019, pp. 493–496, doi: https://doi.org/10.18293/SEKE2019-067.

[12] S. Beyer, C. MacHo, M. Pinzger, and M. Di Penta, "Automatically classifying posts into question categories on stack overflow," in *Proceedings - International Conference on Software Engineering, IEEE Computer Society*, 2018, pp. 211–221, doi: https://doi.org/10.1145/3196321.3196333.

[13] I. Marcal, R. E. Garcia, D. Eler, and R. C. M. Correia, "A strategy to enhance computer science teaching material using topic modelling: Towards overcoming the gap between college andworkplace skills," in *SIGCSE 2020 - Proceedings of the 51st ACM Technical Symposium on Computer Science Education*, 2020, pp. 366–371, doi: https://doi.org/10.1145/3328778.3366858.

[14] S. Rathje, D.-M. Mirea, I. Sucholutsky, R. Marjieh, C. E. Robertson, and J. J. Van Bavel, "GPT is an effective tool for multilingual psychological text analysis," *Proceedings of the National Academy of Sciences*, vol. 121, no. 34, p. e2308950121, 2024. https://doi.org/10.1073/pnas.2308950121

[15] M. E. Peters, M. Neumann, M. Gardner, C. Clark, K. Lee, and L. Zettlemoyer, "Deep contextualized word representations," Retrieved: http://allennlp.org/elmo. [Accessed 2022.

[16] J. Devlin, M.-W. Chang, K. Lee, K. T. Google, and A. I. Language, "BERT: Pre-training of deep bidirectional transformers for language understanding," Retrieved: https://github.com/tensorflow/tensor2tensor, 2023.

[17] K. Alreshedy, D. Dharmaretnam, D. M. German, V. Srinivasan, and T. A. Gulliver, "Predicting the programming language of questions and snippets of stackoverflow using natural language processing," Retrieved: http://arxiv.org/abs/1809.07954. [Accessed 2018.

[18] R. Caruana, L. Pratt, and S. Thrun, *Multitask learning*. United States: Kluwer Academic Publishers, 1997.

[19] R. Çekik, "Effective text classification through supervised rough set-based term weighting," *Symmetry,* vol. 17, no. 1, pp. 1-29, 2025. https://doi.org/10.3390/sym17010090

[20] P. D. Joshi, S. Pocker, R. A. Dandekar, R. Dandekar, and S. Panat, "HULLMI: Human vs LLM identification with explainability," Retrieved: http://arxiv.org/abs/2409.04808. [Accessed 2024.

[21] T. Tojima and M. Yoshida, "Zero-shot classification of art with large language models," *IEEE Access,* vol. 13, pp. 17426–17439, 2025. https://doi.org/10.1109/ACCESS.2025.3532995

[22] K. Kim and J. B. Kim, "Two-step model based on XGBoost for predicting artwork prices in auction markets," *International Journal of Knowledge-based and Intelligent Engineering Systems,* vol. 28, no. 1, pp. 133-147, 2024. https://doi.org/10.3233/KES-230041

[23] H. Hairani, A. Anggrawan, and D. Priyanto, "Improvement performance of the random forest method on unbalanced diabetes data classification using SMOTE-Tomek Link," *International Journal on Informatics and Visualization,* vol. 7, no. 1, pp. 258–264, 2023. https://doi.org/10.30630/joiv.7.1.1069

[24] B. Bischl, *Hyperparameter optimization: Foundations, algorithms, best practices, and open challenges*. United States: Wiley and Sons Inc, 2023.

[25] Y. Zhao, W. Zhang, and X. Liu, "Grid search with a weighted error function: Hyper-parameter optimization for financial time series forecasting," *Applied Soft Computing,* vol. 154, p. 111362, 2024. https://doi.org/10.1016/j.asoc.2024.111362

[26] I. O. Muraina, "Ideal dataset splitting ratios in machine learning algorithms: General concerns for data scientists and data analysts," Retrieved: https://www.researchgate.net/publication/358284895, 2022.