Transactions on Artificial Intelligence
https://www.sciltp.com/journals/tai

Article

LLM-Prompting Driven AutoML: From Sleep Disorder—Classification to Beyond

Yutong Zhao [1,†], Jianye Pang [2,†], Xinjie Zhu [2,†] and Wenhua Shao [3,*]

[1] Beijing Guangqumen Middle School, Beijing 100062, China
[2] Alibaba International Digital Commerce Group, Shenzhen 518054, China
[3] School of Computer Science, Beijing University of Posts and Telecommunications, Beijing 100876, China
* Correspondence: shaowenhua@bupt.edu.cn
† These authors contributed equally to this work.

How To Cite: Zhao, Y.; Pang, J.; Zhu, X.; et al. LLM-Prompting Driven AutoML: From Sleep Disorder—Classification to Beyond. *Transactions on Artificial Intelligence* **2025**, *1*(1), 4. https://doi.org/10.53941/tai.2025.100004.

Received: 28 March 2025
Revised: 6 May 2025
Accepted: 9 May 2025
Published: 12 May 2025

Abstract: Traditional automated machine learning (AutoML) often faces limitations in manual effort, complexity management, and subjective design choices. This paper introduces a novel LLM-driven AutoML framework centered on the innovation of decomposed prompting. We hypothesize that by strategically breaking down complex AutoML tasks into sequential, guided sub-prompts, Large Language Models (LLMs) operating within a code sandbox on standard PCs can autonomously design, implement, evaluate, and select high-performing machine learning models. To validate this, we primarily applied our decomposed prompting approach to sleep disorder classification (illustrating potential benefits in healthcare). To assess the generalizability and robustness of our method across different data types, we subsequently evaluated it on the established 20 Newsgroups text classification benchmark. We rigorously compared decomposed prompting against zero-shot and few-shot prompting strategies, as well as a manually engineered baseline. Our results demonstrate that decomposed prompting significantly outperforms these alternatives. Our results demonstrate that decomposed prompting significantly outperforms alternatives, enabling the LLM to autonomously achieve superior classifier design and performance, particularly showing strong results in the primary sleep disorder domain and demonstrating robustness in the benchmark task. These findings underscore the transformative potential of decomposed prompting as a key technique for advancing LLM-driven AutoML across diverse application areas beyond the specific examples explored here, paving the way for more automated and accessible problem-solving in scientific and engineering disciplines.

Keywords: AutoML; LLM; prompt engineering; Sleep Disorder Newsgroups Classification

1. Introduction

Roughly 10 percent of the global populace is afflicted by sleep disorders, a figure increasing amidst rapid urbanization (WHO). Beyond daytime fatigue and diminished attention, sleep disorders can precipitate chronic ailments like cardiovascular diseases and diabetes. Consequently, the precise identification and classification of sleep disorders hold significant importance for enhancing public health.

Traditional machine learning algorithms such as decision trees, support vector machines (SVM) and random forests [1–3] face limitations. They require extensive manual operation in data preprocessing (handling missing values, outliers, standardization, normalization) and model training/tuning (algorithm selection, parameter setting, iterative optimization), consuming significant time and effort and being susceptible to human factors, thus limiting accuracy and reliability. Furthermore, their application heavily depends on expert experience for algorithm selection, feature engineering, and parameter adjustment. For complex classifications, varying expert opinions can lead to inconsistent results. These issues hinder the advancement and application of sleep disorder research, necessitating new technologies and methods to overcome these challenges.

Large language models (LLMs), a cutting-edge technology in natural language processing based on the Transformer [4], have shown remarkable progress through unsupervised pre-training on vast text data, demonstrating great potential in medical research [5–13]. However, the application of LLMs in sleep health research remains largely unexplored, particularly considering the limited accessibility of powerful GPU computing resources for many researchers. This gap highlights the importance of investigating LLM capabilities for sleep health analysis even on standard PC hardware.

Leveraging these capabilities, we hypothesize that LLMs, when equipped with a code execution sandbox and guided by structured prompts, can function as effective agents for *Automated Machine Learning* (*AutoML*). Specifically, they could possess the capability to autonomously design, generate, execute, and debug code for the entire machine learning pipeline–including data pre-processing, feature engineering, model selection, hyperparameter tuning, and evaluation–thereby streamlining the development process and enabling automated solutions, initially targeting challenging tasks like sleep disorder classification.

Furthermore, leveraging LLMs' superior natural language understanding and processing capabilities, we hypothesize that LLMs can directly analyze text-based data, such as sleep health and lifestyle data, effectively overcoming current limitations. Additionally, we hypothesize that the integration of LLMs with a code sandbox environment as an agent for AutoML, leveraging their robust knowledge reasoning capabilities [14], can accurately identify complex sleep disorder symptoms like insomnia and sleep apnea through reasoning (classification). The core of this research is to investigate and validate the effectiveness of different prompting strategies for LLM-driven AutoML, with a central focus on the proposed decomposed prompting technique. We hypothesize that decomposed prompting, by breaking down the complex AutoML workflow into manageable, guided steps, enables LLMs to autonomously achieve superior classifier design and performance compared to less structured approaches like zero-shot and few-shot prompting. To test this hypothesis, primarily focusing on the sleep disorder classification task, we apply our methodology. Furthermore, to evaluate the generalizability and robustness of decomposed prompting across different data types (structured vs. text), we extend our analysis to include the standard 20 Newsgroups text classification benchmark.

Our research demonstrates that prompt-driven LLMs with code execution sandbox possess robust capabilities for automating machine learning workflows across diverse domains. The innovation of decomposed prompting, in particular, significantly enhances their ability to tackle complex classification problems, as evidenced by our successful application to sleep disorder classification (a potential area of impact in healthcare, as illustrated conceptually in Figure 1) Furthermore, the effectiveness on the 20 Newsgroups benchmark demonstrates the broad applicability of our approach beyond specific domains like healthcare. Specifically, we hypothesize that their integration with sensor technology and wearable devices can further enhance the value of automated applications. Furthermore, we hypothesize that this integration will not only effectively address practical challenges in sleep medicine but, more importantly, will accumulate valuable experience for the broader application of LLMs in healthcare and Text data analysis.

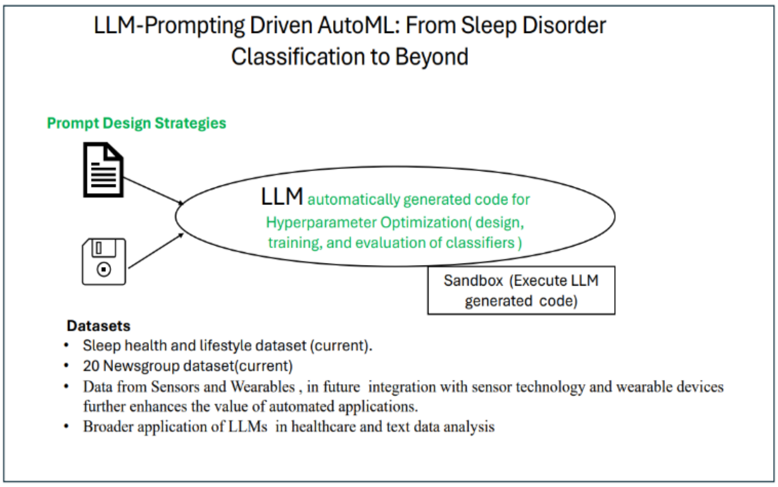


Figure 1. LLM-Prompting Driven AutoML: From Sleep Disorder Classification to Beyond.

Different prompting strategies significantly impact the effectiveness of LLMs in automated classification tasks, and decomposed prompting strategies is more suitable for complex classification tasks. we hypothesize that decomposed prompting strategies, due to their ability to guide LLMs towards deeper reasoning, will demonstrate unique advantages in text-based classification compared to manually created intuitive baseline, zero-shot and few-shot prompting. We design experiments to systematically evaluate the impact of different prompting strategies on classification accuracy, efficiency, and robustness to validate this hypothesis.

Moreover, by systematically evaluating these prompting strategies, motivated by the challenges in automated sleep disorder classification and subsequently testing their robustness on the newsgroups classification task we can clearly delineate their advantages and disadvantages. This in-depth analysis will provide clear guidance for subsequent optimization of prompting strategies and facilitate the continuous improvement of LLM performance in AutoML.

LLMs hold pioneering application value in automated Text based data classification. Introducing LLMs into the field of automated classification, particularly for challenging domains like sleep disorders, is a groundbreaking initiative. Our extension to the newsgroup classification benchmark serves to validate the broader potential of this approach. Specifically, compared to traditional methods, LLMs can automatically perform machine learning processes, enabling high-precision prediction, we will validate this hypothesis by experimentally comparing the classification performance of LLMs against traditional machine learning models.

2. Datasets

In this study, our primary focus utilizes the Sleep Health and Lifestyle dataset (Table 1), selected for its clinical relevance and the specific challenges it presents for AutoML (e.g., mixed data types, smaller sample size) as discussed earlier. This dataset allows us to test the LLM's ability to handle structured data effectively with limited samples. To assess the generalizability and robustness of our LLM-driven AutoML approach, particularly the prompting strategies, across different data modalities and complexities, we also employ the 20 Newsgroups dataset. This widely-used, high-dimensional text benchmark evaluates the LLM's capacity to manage unstructured data and tests the prompting strategies' effectiveness on a larger-scale text classification task.

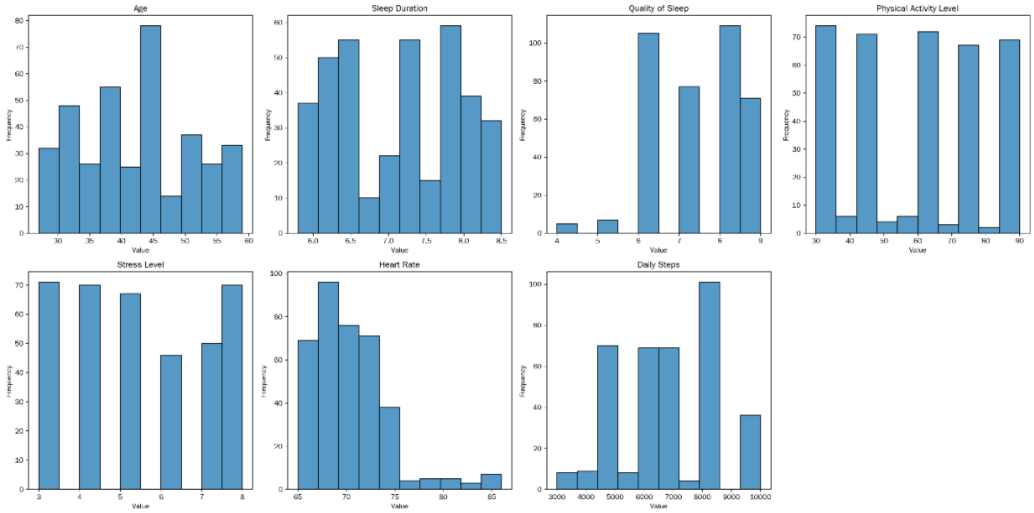
Table 1. Datasets.

Dataset	Overview	Key Features
Sleep Health and Lifestyle Dataset (Kaggle)	It comprises 400 rows and 13 columns, covering a wide range of variables related to sleep and daily habits. It includes details: "Person ID (identifier), Gender (Male/Female), Age (years), Occupation (text), Sleep Duration (hours), Quality of Sleep (scale 1–10), Physical Activity Level (minutes/day), Stress Level (scale 1–10), BMI Category (Underweight, Normal, Overweight, etc.), Blood Pressure (systolic/diastolic), Heart Rate (bpm), Daily Steps (count), Sleep Disorder (None, Insomnia, Sleep Apnea).	• Comprehensive Sleep Metrics: Explore sleep duration, quality, and factors influencing sleep patterns. • Lifestyle Factors: • Analyze physical activity levels, stress levels, and BMI categories • Cardiovascular Health: Examine blood pressure and heart rate measurements • Sleep Disorder Analysis: • Identify the occurrence of sleep disorders such as Insomnia and Sleep Apnea. Details about Sleep Disorder Column: • None: The individual does not exhibit any specific sleep disorder. • Insomnia: The individual experiences difficulty falling asleep or staying asleep, leading to inadequate or poor-quality sleep. • Sleep Apnea: The individual suffers from pauses in breathing during sleep, resulting in disrupted sleep patterns and potential health risks
20Newsgroup (Real world)	Widely used benchmark dataset in natural language processing (NLP) and machine learning for text classification, clustering, and information retrieval tasks. It was collected by Jason Rennie and consists of approximately 20,000 newsgroup	Data Structure: • 20 catategories (e.g., comp.graphics, rec.sport.hockey, talk.politics.mideast), each containing ~1000 text documents. • Raw data in plain text format, preserving email headers, signatures, and newsgroup metadata.

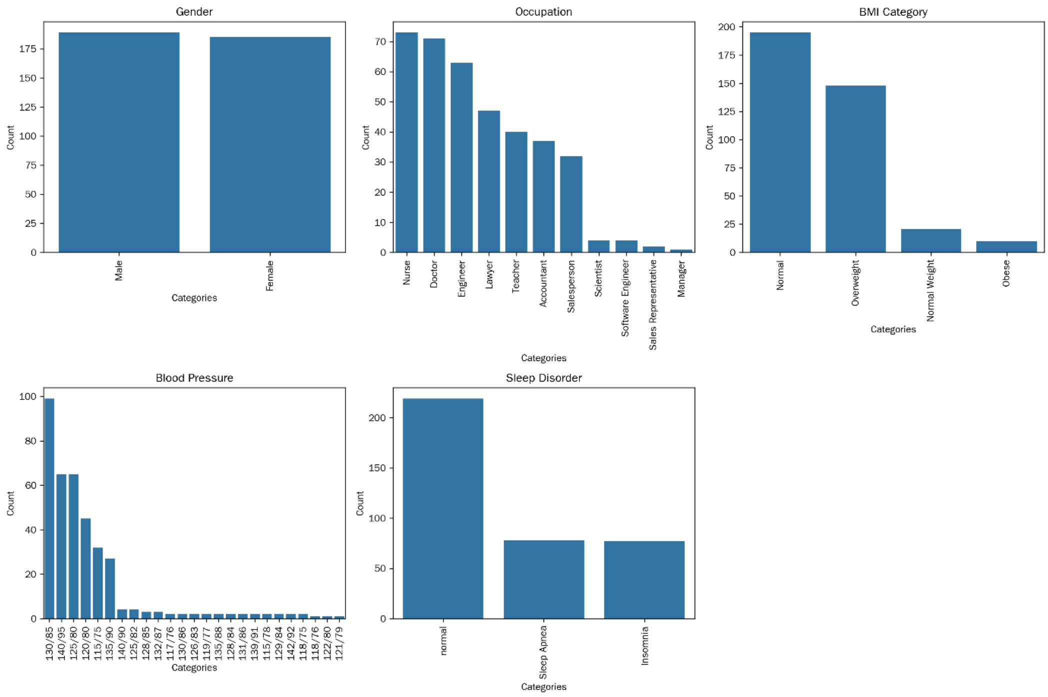
	posts, evenly distributed across 20 distinct topic categories, such as computer science, religion, politics, and hobbies. The dataset is publicly available at http://qwone.com/~jason/20Newsgroups/ (accessed on 10 May 2025) and has been a standard resource for evaluating text analysis algorithms since the late 1990s.	Typical Tasks: • Multi-class classification: Assigning documents to one of the 20 topics. • Multi-label classification (extension): Documents may belong to multiple categories (though the standard dataset is single-label). • Clustering, dimensionality reduction, and feature selection (e.g., using TF-IDF, word embeddings). Advantages: • Balanced class distribution, reducing bias toward dominant categories. • Diverse topics with varying linguistic styles (technical, conversational, argumentative), challenging model generalization. • Preprocessing flexibility: Users can apply tokenization, stopword removal, or stemming based on task requirements. Applications in Research The dataset is frequently used to validate the performance of text classifiers, such as: • Traditional machine learning models (Naive Bayes, SVM, Random Forests). • Deep learning approaches (neural networks, transformers). • Few-shot learning and zero-shot learning frameworks.

Figure 2a–c collectively enable us to formulate and evaluate hypotheses about The Sleep Health and Lifestyle dataset [15] characteristics, their relevance to sleep disorders, sample composition, and the underlying relationships between features, transitioning from descriptive observations to testing proposed connections.

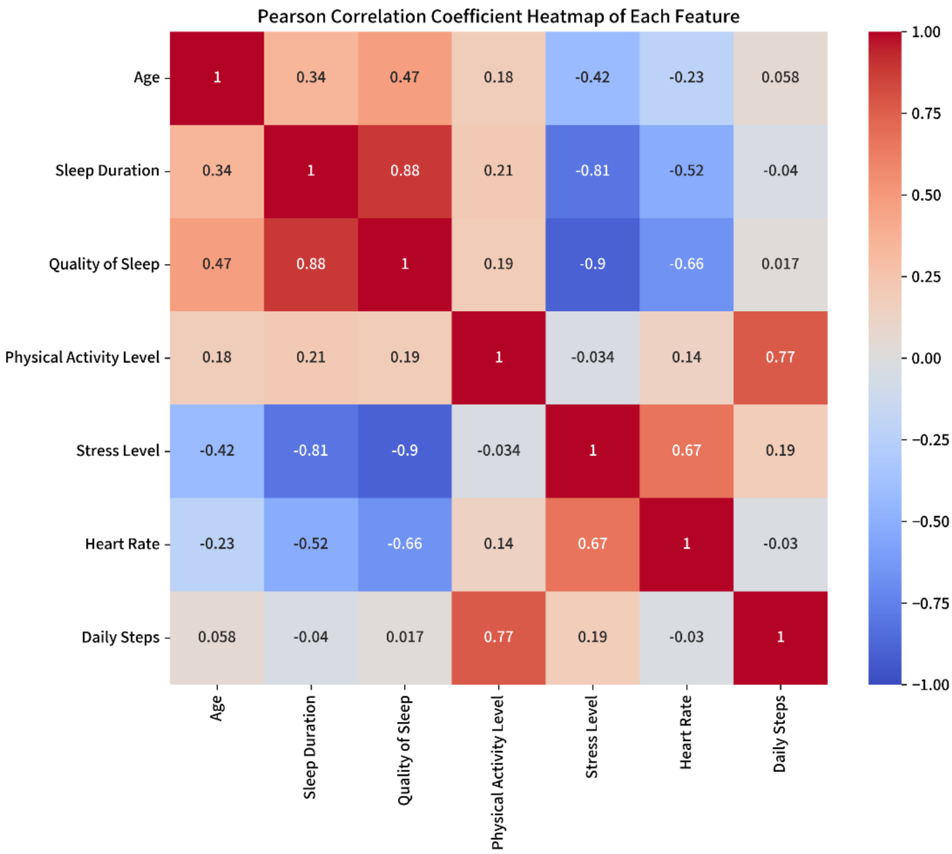
The 20Newsgroup dataset [16] was thoroughly analyzed to ensure the robustness of experimental results. A histogram illustrating the distribution of document counts across the 20 categories (Figure 3a) directly assessed class balance, verifying that no significant imbalance existed and mitigating potential bias in model performance. A pie chart (Figure 3b) explicitly detailed the training-test split ratio, outlining the experimental setup and data allocation for model development and evaluation phases. Box plots illustrating the distribution of text lengths (word count) within each category (Figure 3c) revealed text complexity differences, providing a data-driven rationale for pre-processing strategies like truncation or padding to optimize model input lengths. Finally, word cloud visualizations highlighting prominent keywords associated with each category (Figure 3d) visually represented topic distinctiveness, supporting the design logic of keyword-based rule classifiers and offering insights into class separability based on lexical content.



(**a**) Statistical chart of numerical features

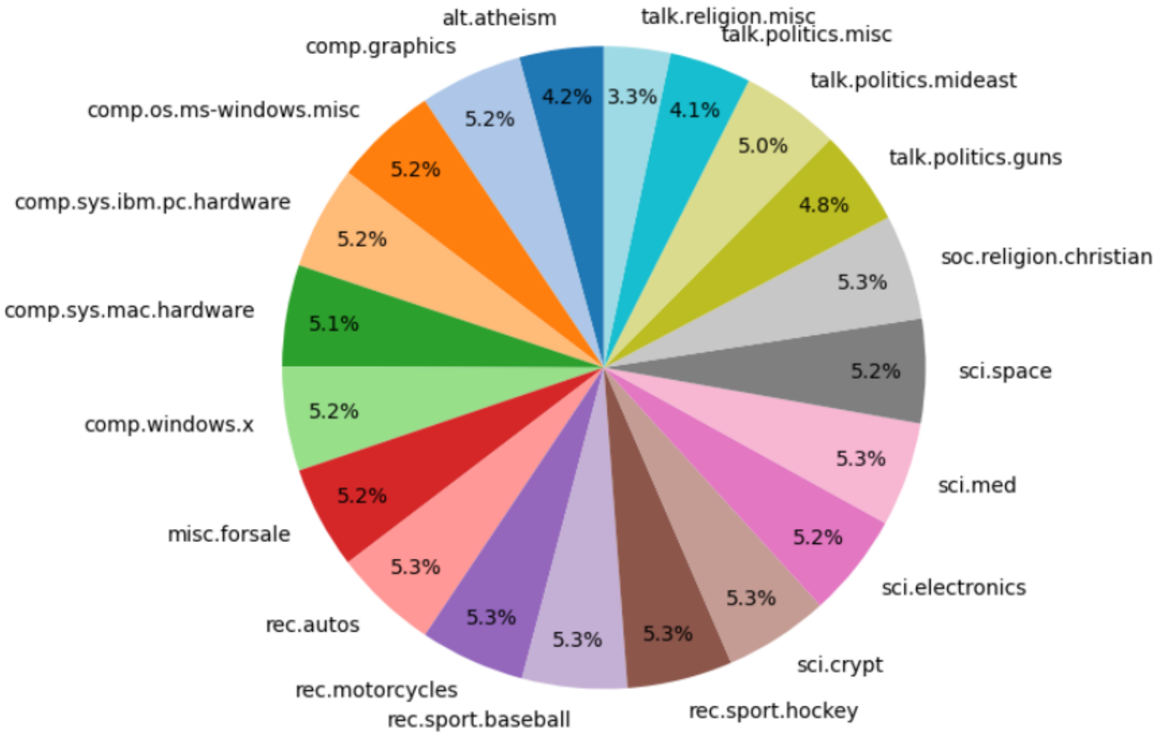


(**b**) Distribution chart of categorical features

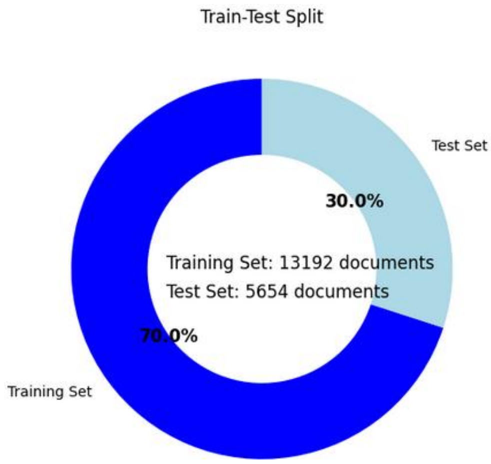


(**c**) Correlation coefficients among features

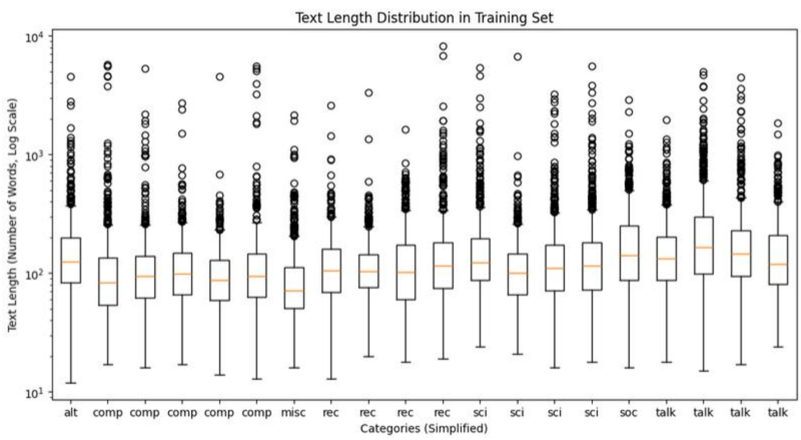
Figure 2. The Sleep Health and Lifestyle dataset characteristics.



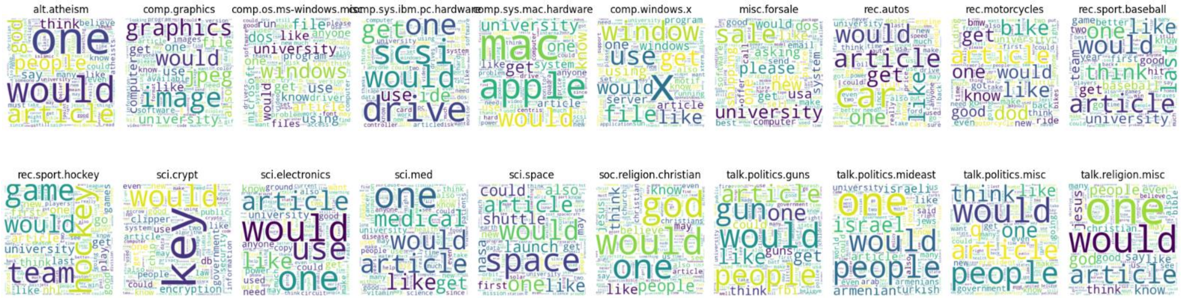
(**a**) Distribution of document counts across the 20 categories



(**b**) Training-test split ratio of text count



(**c**) Distribution of text lengths (word count) within each category



(**d**) word cloud visualizations highlighting prominent keywords associated with each category

Figure 3. 20 Newsgroup dataset dataset characteristics.

3. Research Methods

3.1. Selection of Large Language Model

Recognizing the constraint of conducting research on a PC without access to expensive GPU resources, we hypothesized that the selection of an LLM with a code sandbox would be a critical factor influencing the accuracy and efficiency of our automated sleep disorder classification and newsgroups classification research. Our ideal LLM would need to support: the ingestion of multiple Excel data files, code generation, code execution, code debugging, issue identification, code regeneration, and iterative execution until successful completion. We evaluated several LLMs based on these criteria. One reason for choosing these LLMs was their greater accessibility, offering features like free usage and unrestricted access within mainland China, bypassing network security limitations. However, certain LLMs (like Claude) were excluded; for instance, some are unavailable in mainland China, making registration with mainland mobile numbers impossible, or requires payment. Detailed comparisons are available in Appendix A, Gemini 2.0 Flash, while capable of generating code, did not support uploading Excel files and could not execute the generated code. Manual execution of its generated code encountered data format issues, leading to program interruptions near completion. Copilot supported the upload of multiple Excel files, but did not display the generated code. While it could execute the code, the final results were flawed; for example, it erroneously generated a ROC curve from a Confusion Matrix, with no ability to debug. Qwen2.5-Max supported only single Excel file uploads and could generate code but not execute it. DeepSeek-R1 (integrated with Baidu) also supported single Excel file uploads and code generation but lacked code execution capabilities. Kimi supported uploading multiple Excel files, displayed and executed code. After evaluating these available LLM options, we selected GPT4-O and Doubao [17] PC version (1.52.6). This decision was predicated on the hypothesis that GPT4-O and Doubao PC's inherent strengths in natural language processing (NLP), strong adaptability to sleep health data, newsgroups data, interactive capabilities, and robust technical features would collectively make it a highly compelling and justified choice for the automated classification of all Text related data in our research. Crucially, the chosen LLM needed to demonstrate proficiency with both the structured, tabular data typical of the sleep health domain and the unstructured text data of the newsgroups benchmark to enable a thorough evaluation of our prompting strategies' adaptability. We aimed to validate the following hypotheses:

Efficiency: user-friendly interface and technical features would streamline the implementation of our automated sleep disorder classification and newsgroup classification, leading to increased research efficiency. Specifically, its intuitive interface is capable of directly processing common data formats such as Excel files and scikit-learn dataset, simplifying data import and reducing pre-processing.

Automated Code Execution & Accuracy: This would enable the model to rapidly generate and execute code for implementing classification algorithms based on textual analysis, enhancing classification efficiency and mitigating potential coding errors, thus supporting accurate automated sleep disorder classification.

NLP Capabilities: extensive unsupervised pre-training on massive text corpora would enable it to acquire a deep and broad understanding of language, crucial for interpreting nuanced textual descriptions of sleep patterns and habits. Its inherent multi-head attention mechanism would allow the model to concurrently process different segments of input text, effectively capturing nuanced semantic relationships and contextual information relevant to sleep disorder classification. The multi-layer neural network architecture would facilitate deep feature extraction and information processing across multiple dimensions—lexical, sentential, and textual—enabling a comprehensive analysis of text content related to sleep disorders, and thus contributing to accurate classification.

3.2. LLM Prompt Design Strategies

To systematically investigate and optimize LLM performance in sleep disorder classification, and to rigorously test our hypotheses regarding prompting strategies, we curated three distinct prompting strategies. Each strategy was designed to test a specific hypothesis about how different levels of guidance and information influence LLM performance. These strategies are visualized in Flowchart of Prompting Strategies (Figure 4) and Manually created baseline (Appendix B) and detailed prompt examples are provided in Table 2.

Zero-shot prompting, not use training data, tests if LLMs can achieve basic performance using only clear task descriptions and their pre-existing knowledge from pre-training. This approach evaluates the model's inherent capabilities without task-specific examples, relying on explicit instructions to guide responses. The hypothesis is that LLMs' extensive pre-trained language understanding and knowledge are sufficient for basic classification tasks when given clear, rule-based prompts. Therefore, zero-shot prompting assesses the model's ability to extract relevant information and apply reasoning based on semantic cues in the task description, establishing a performance baseline to measure the effectiveness of more sophisticated prompting strategies. Within our comparative framework, the Zero-shot approach serves as the fundamental baseline for structured prompting, representing the performance achievable with clear instructions but without specific training examples or task decomposition guidance. The effectiveness of Few-shot and Decomposed prompting is evaluated relative to this baseline.

Few-shot prompting tests if exposing the LLM to a few data samples enhances classification accuracy by enabling it to learn feature-outcome relationships directly. We hypothesize that data patterns in a training set provide richer context, improving accuracy beyond the zero-shot baseline. The key hypothesis is that even a few examples are crucial for refining the LLM's understanding of the specific classification task and improving performance by learning from data patterns and feature-outcome relationships.

Decomposed prompting tests if breaking down a complex task into sub-tasks with step-by-step guidance improves efficiency and accuracy. In sleep disorder classification, the core hypothesis is that systematically guiding the LLM through sub-tasks will lead to a more structured, reasoned, and accurate approach, maximizing LLM potential for complex AutoML tasks. Decomposed prompting rigorously tests if structured, step-by-step guidance via task decomposition is the most effective way to achieve high performance in complex classification tasks using LLMs.

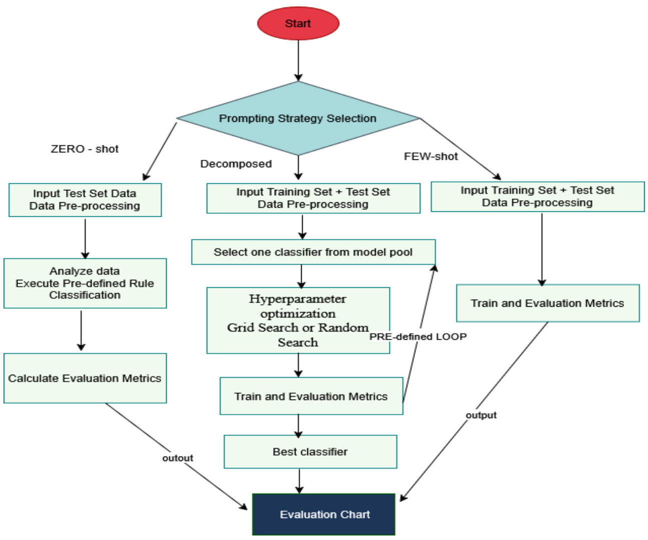


Figure 4. Flowchart of Prompting Strategies.

Table 2. Prompt examples.

Prompts	Sleep Health and Lifestyle Dataset	20Newsgroup
Manually Random forest Baseline	Implemented using scikit-learn with the following hyperparameters: n_estimators=100	Same
Zero-shot	• Task Overview	Use the fetch_20newsgroups dataset available on scikit-learn's official website and load the entire dataset. If

	– Test Dataset (‘Sleep_health_and_lifestyle_dataset_remaining_90_without_last_column.csv) – Real Label Dataset(Sleep_health_and_lifestyle_dataset_remaining_90.csv) – This task primarily consists of two sub-tasks: first, to perform rule_based_classifier on Test Dataset, second, to evaluate the classification results using specific evaluation metrics (sklearn.metrics in accuracy score, precision score, f1 score, recall score, confusion matrix, roc curve, auc and other evaluation metrics) and draw relevant charts.	the data is not pre–split into training and test sets, divide it in a 7:3 ratio (70% for the training set and 30% for the test set). Complete the following tasks in an efficient manner: 1. Design a rule–based classifier based on the test set data (not use train data) to perform full–category classification on the test set. 2. Evaluate the classification results using evaluation metrics from sklearn.metrics (accuracy score, precision score, F1 score, recall score, confusion matrix, ROC curve, AUC, etc.) and Draw relevant charts based on the evaluation results
Few-shot	• Task Overview – Test Dataset (‘Sleep_health_and_lifestyle_dataset_remaining_90_without_last_column.csv) – Real Label Dataset (Sleep_health_and_lifestyle_dataset_remaining_90.csv) – Training dataset (Sleep_health_and_lifestyle_dataset_selected_90.csv) – This task primarily consists of two sub-tasks: first, to perform classification (Random Forest) on Test Dataset, second, to evaluate the classification results using specific evaluation metrics (sklearn.metrics in accuracy score, precision score, f1 score, recall score, confusion matrix, roc curve, auc and other evaluation metrics) and draw relevant charts.	Use the fetch_20newsgroups dataset available on scikit-learn’s official website and load the entire dataset. If the data is not pre–split into training and test sets, divide it in a 7:3 ratio (70% for the training set and 30% for the test set). Complete the following tasks in an efficient manner: 1. Design Random Forest classifier based on the training set data to perform full–category classification on the test set. 2. Evaluate the classification results using evaluation metrics from sklearn.metrics (accuracy score, precision score, F1 score, recall score, confusion matrix, ROC curve, AUC, etc.) and Draw relevant charts based on the evaluation results
Decomposed	• Task Overview – Test Dataset (‘Sleep_health_and_lifestyle_dataset_remaining_90_without_last_column.csv) – Real Label Dataset (Sleep_health_and_lifestyle_dataset_remaining_90.csv) – Training dataset (Sleep_health_and_lifestyle_dataset_selected_90.csv) This task primarily consists of two sub-tasks: first, to perform classification (automatically design, train, and test various classifiers (Logistic Regression, Decision Tree, Random Forest, Gradient Boosting, SVM, MLP and search hyperparamters, select the BEST one for next step) on Test Dataset, second, to evaluate the classification results using specific evaluation metrics (sklearn.metrics in accuracy score, precision score, f1 score, recall score, confusion matrix, roc curve, auc and other evaluation metrics) and draw relevant charts.	Use the fetch_20newsgroups dataset available on scikit-learn’s official website and load the entire dataset. If the data is not pre–split into training and test sets, divide it in a 7:3 ratio (70% for the training set and 30% for the test set). Complete the following tasks in an efficient manner: 1. Design, train, various classifiers (Logistic Regression,Decision Tree, SVM, MLP) and search hyperparamters, select the BEST ONE to perform full–category classification on the test set. 2. Evaluate the classification results using evaluation metrics from sklearn.metrics (accuracy score, precision score, F1 score, recall score, confusion matrix, ROC curve, AUC, etc.) and Draw relevant charts based on the evaluation results

3.3. Environment and Experimental Procedure

The experimental environment was meticulously constructed to provide a robust and reliable platform for testing our hypotheses regarding LLM-driven AutoML for Text data classification. The hardware and software configurations were selected to ensure the accuracy and reproducibility of our experimental results, directly supporting the rigorous evaluation of our proposed methodologies. Specifically, the processor chosen for these experiments was an Intel(R) Core(TM) i5-6400T CPU @ 2.20GHz, featuring a main frequency of 2201 MHz, with 4 cores and 4 logical processors. This configuration was deemed sufficient to handle the computational demands of running the LLM and executing code within the sandbox environment, crucial for validating Hypotheses. The operating system was Microsoft Windows 10 Home Chinese Edition, version 10.0.19045 (build 19045), providing a stable and widely accessible software foundation. The Large Language Model utilized were GitHub Copilot GPT-4o, Doubao PC version 1.52.6, selected for its code execution sandbox capabilities, which are essential for testing Hypothesis and the AutoML workflows investigated in this study

The experimental procedure was carefully designed to systematically evaluate the efficacy of different prompting strategies and to demonstrate the feasibility of LLM-driven AutoML for classification. We establish two baselines for performance comparison. The first, a Traditional ML approach, employs a manually constructed, theoretically optimized Random forest. This baseline represents a model crafted with expert domain knowledge and designed to achieve near-optimal performance given the available resources and data. The second baseline is a Zero-Prompt (or Zero-Shot) approach, representing a no-training adaptation and usage of LLM on the task. As Traditional ML methods are inherently reliant on expert experience in feature engineering and model selection, we hypothesize that the relative performance of the Random forest compared to the Zero-Prompt approach will be highly case-dependent. That is, neither approach is inherently superior, and the optimal strategy is contingent upon the specific characteristics of the dataset and the task at hand.

Dataset partitioning and sample selection for both the primary sleep disorder task and the 20 Newsgroups generalization task followed controlled procedures (Table 3) to test prompting strategies:: creating controlled datasets for training and evaluation to test prompting Strategies: this step was designed to create distinct datasets for training (where applicable) and testing, allowing for a controlled assessment of each prompting strategy's performance.

Classifier design, training, and evaluation via Diverse Prompting Strategies: This core step involved prompting the LLM using zero-shot, 90-sample, and decomposed prompting strategies. For each strategy, the LLM was instructed to design, train (where applicable), and evaluate classifier using the provided datasets within its code execution sandbox. This systematic variation of prompting strategies allowed for a direct comparison of their effectiveness and a rigorous test of hypothesis regarding the influence of prompting on AutoML performance.

Table 3. Dataset Partitioning.

	Sleep Disorder Classification	**20Newsgroups Classification**
Training Set for Few-Shot Learning	To facilitate the 90-sample prompting strategy and test the hypothesis that data exposure enhances learning, we randomly selected 30 samples from each of the three categories (normal, sleep apnea, and insomnia) within the original dataset. This resulted in a total of 90 samples, designated as "prompts 90 examples", saved in CSV format for easy upload to the LLM	training set (11,314 documents) and a test set (7532 documents), with no overlap between categories in the splits.
Ground Truth Preparation for Evaluation	To ensure unbiased evaluation, the 90 samples used for training were removed from the original dataset. The remaining data constituted the "ground truth" dataset, representing unseen data for evaluating the generalization performance of the LLM-designed classifiers	Dataset contains Ground Truth
Test Set Creation for Blind Performance Assessment	To create a test set simulating a real-world scenario where the classifier predicts without knowing the true labels, a copy of the "ground truth" file was made. The last column, containing the "Sleep Disorder" labels (ground truth), was deleted, this dataset, devoid of labels, was used to assess the predictive accuracy of the LLM-generated classifiers on unseen, unlabeled data, mirroring a practical diagnostic setting	test set (7532 documents),
Data Ingestion into LLM	Manually uploading the three Excel files (training set, ground truth, and test set) to the Doubao platform ensured that the LLM had direct access to the necessary data within its environment. This step was	Prompt generated code to directly load from internet

	crucial for enabling the LLM to perform data-driven classifier design, training, and evaluation

4. Results

To quantitatively assess and compare the effectiveness of each prompting strategy, first in the primary sleep disorder task and subsequently in the 20Newsgroup classification benchmark, and thus provide empirical evidence for our hypotheses, we employed a suite of standard classification performance metrics. The results presented in this section reflect performance metrics obtained from a multiple execution runs for each prompting strategy and LLM combination within our controlled experimental environment. While the observed differences in performance, particularly the advantages of decomposed prompting, are substantial, Quantitative results are summarized in Figure 5 and Table 4, with detailed qualitative insights derived from the confusion matrices and ROC curves presented in Figure 6.

we present a quantitative overview of the classification performance achieved by each prompting strategy. The metrics provided—accuracy, precision, recall, F1-score, and AUC value—offer a direct numerical comparison of each strategy's effectiveness in sleep disorder classification. These metrics collectively serve as key indicators for evaluating the progressive improvement in model performance as we transition from basic zero-shot prompting to more sophisticated and data-informed strategies, directly addressing the comparative performance aspects of Hypothesis and the overall efficacy of LLM-driven AutoML.

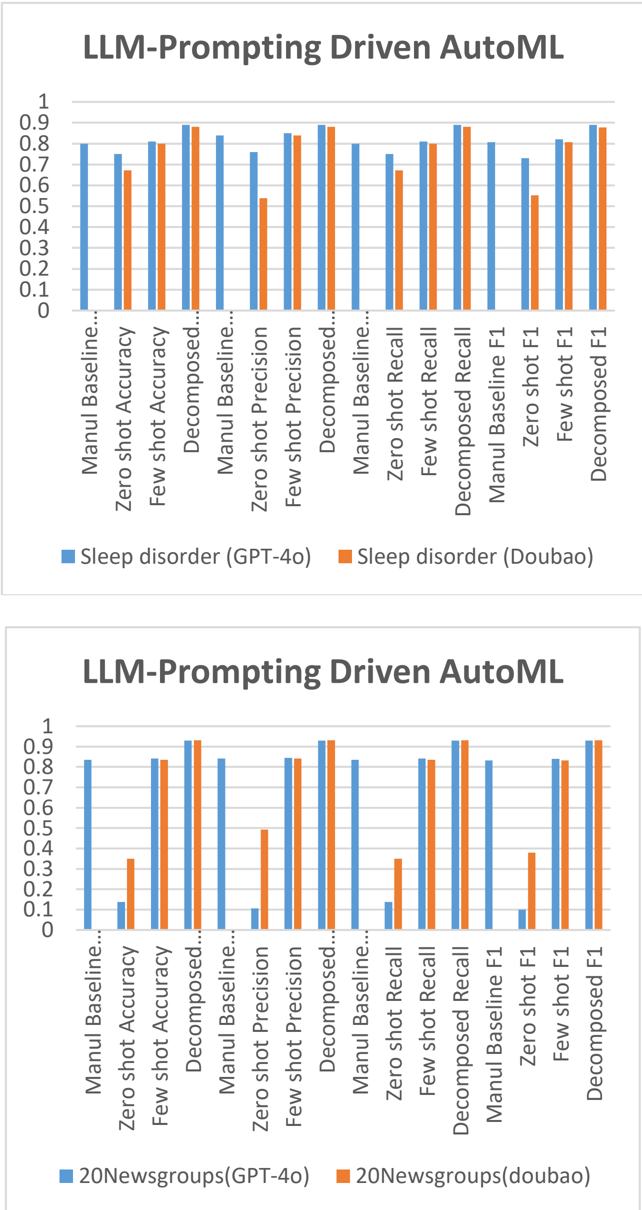


Figure 5. Performance of Prompting Strategies.

Table 4. Baseline vs. Best model from Decomposed Prompting.

	Accuracy	Precision	Recall	F1 Score
Manul baseline Random forest (Sleep disorder)	0.7993	0.8394	0.7993	0.8074
AutoML best SVM (Doubao PC Sleep disorder)	0.8803	0.8798	0.8803	0.8772
AutoML best SVM (GPT-4o Sleep disorder)	0.8873	0.8866	0.8873	0.8859
Manul Random forest baseline (20news group)	0.8349	0.8416	0.8349	0.8319
AutoML best MLP (Doubao PC 20news group)	0.9312	0.9317	0.9312	0.9313
AutoML best MLP (GPT-4o 20news group)	0.9264	0.9274	0.9264	0.9266

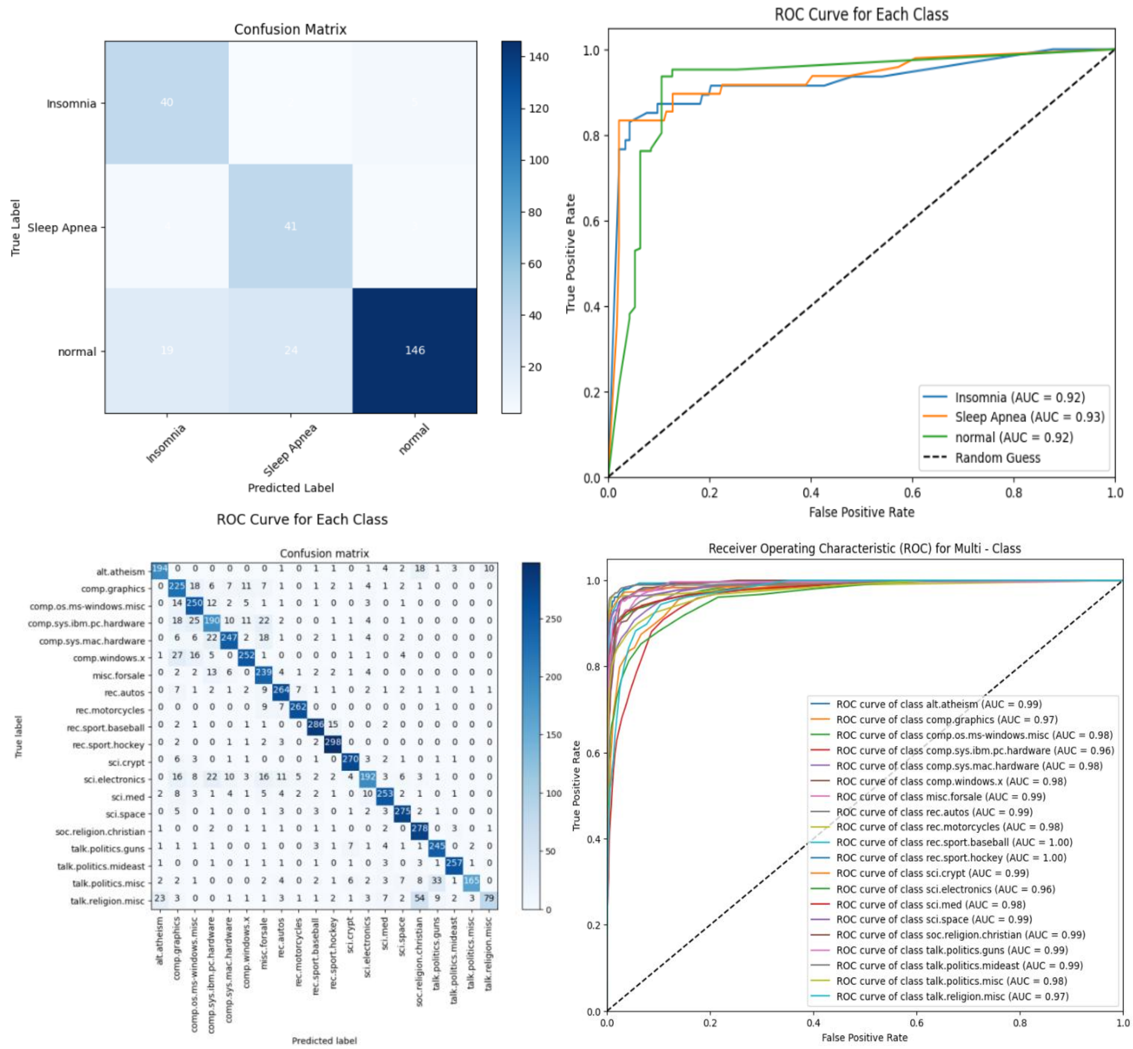


Figure 6. Performance of Manual Random Forest Baseline on Sleep Disorder (**Upper Panels**) and 20 Newsgroups (**Lower Panels**) Datasets.

The Manual Random forest baseline Figure 6 demonstrates strong binary classification for "normal" and "insomnia" in the sleep disorder task, achieving high accuracy and discriminative power. However, it struggles significantly with the "sleep apnea" category, frequently misclassifying it as "normal," indicating a major weakness. In the 20 Newsgroups multi-class task, the model shows a general capacity for categorization, evidenced by ROC curves above the diagonal. However, the 20 × 20 confusion matrix suggests varying performance across the 20 categories, with certain newsgroups likely being more difficult to distinguish and leading to higher misclassification rates.

The two confusion matrices in Figure 7 compare Doubao's (left) and GPT-4o's (right) zero-shot prompting performance on the three-class sleep disorder task. Both models excel at identifying "normal". However, Doubao struggles significantly with "sleep apnea," misclassifying many as "normal". GPT-4o demonstrates superior overall performance, with higher true positives and lower false negatives across all three classes, particularly for "sleep apnea." The two plots display Multi-Class ROC Curves for Doubao and GPT-4o. GPT-4o (orange lines) generally exhibits higher TPR at lower FPRs for "sleep apnea" and "insomnia" compared to Doubao (blue lines), reflected in its higher AUC values. Both models show strong performance for the "normal" class. The Rule-Based Classifier demonstrates good performance for "normal" (high AUC), but lower AUCs for "insomnia" and "sleep apnea" suggest limited effectiveness for these categories.

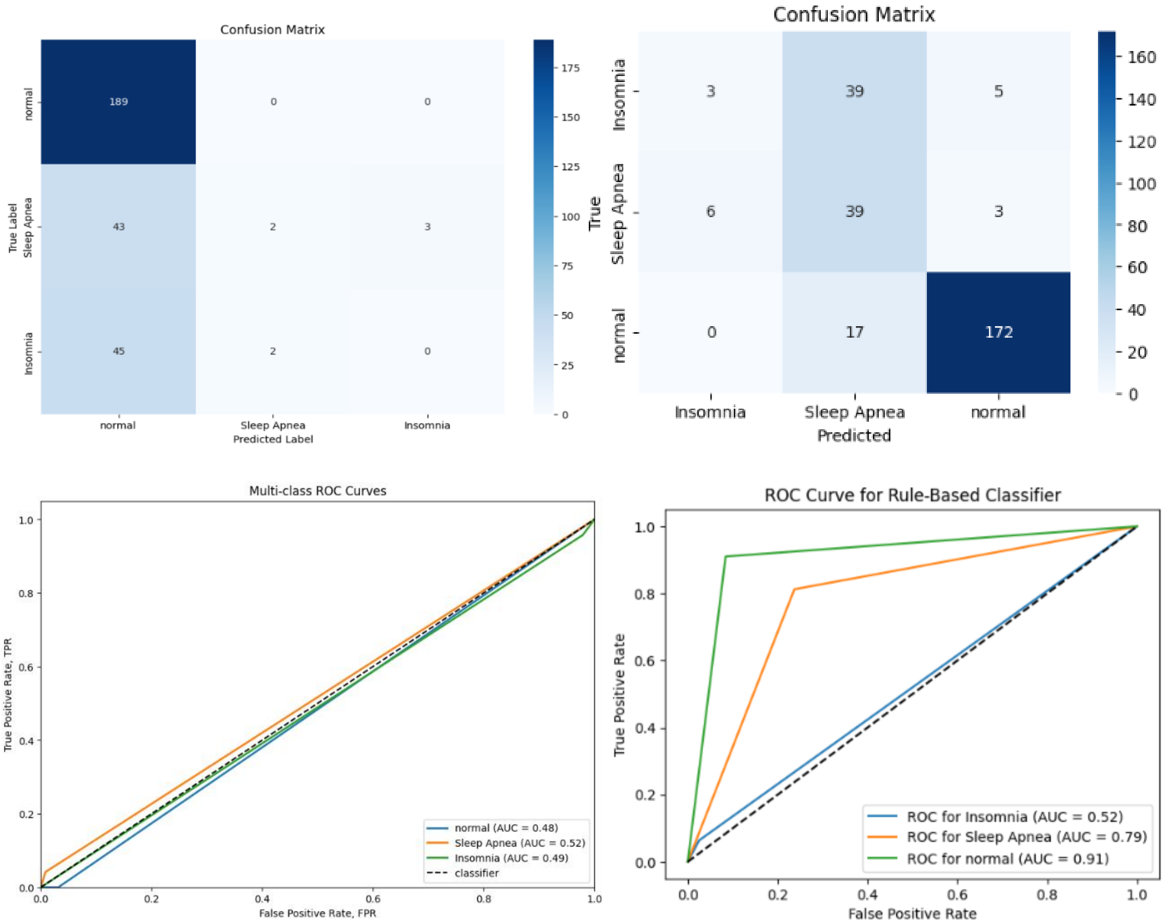
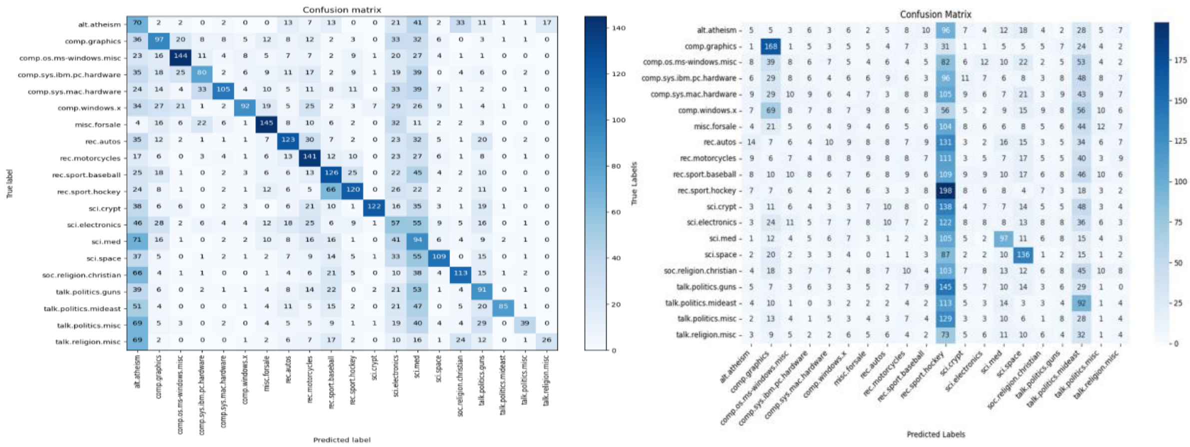


Figure 7. Zero-shot Prompting sleep disorder dataset doubao (**Left**), GPT-4o (**Right**).

Figure 8 shows zero-shot prompting on 20 newsgroups. Doubao's confusion matrix (left) shows more off-diagonal errors, indicating frequent misclassifications across various topics. GPT-4o's matrix has a clearer diagonal, suggesting better overall accuracy. The multi-class ROC curves generally favor GPT-4o (higher AUCs), demonstrating superior discrimination across most newsgroup categories compared to Doubao. The rightmost ROC curve likely represents an aggregate or specific class comparison.



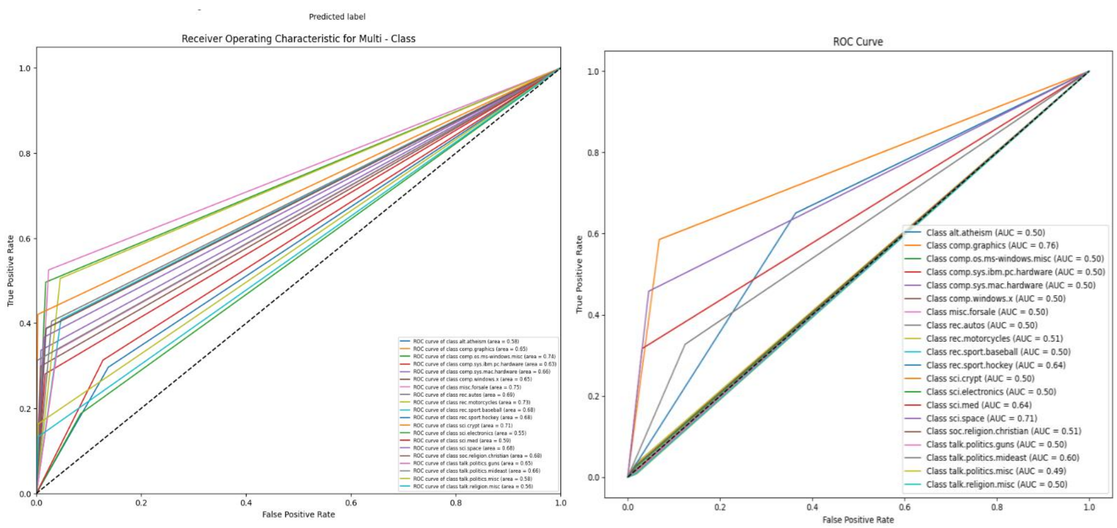
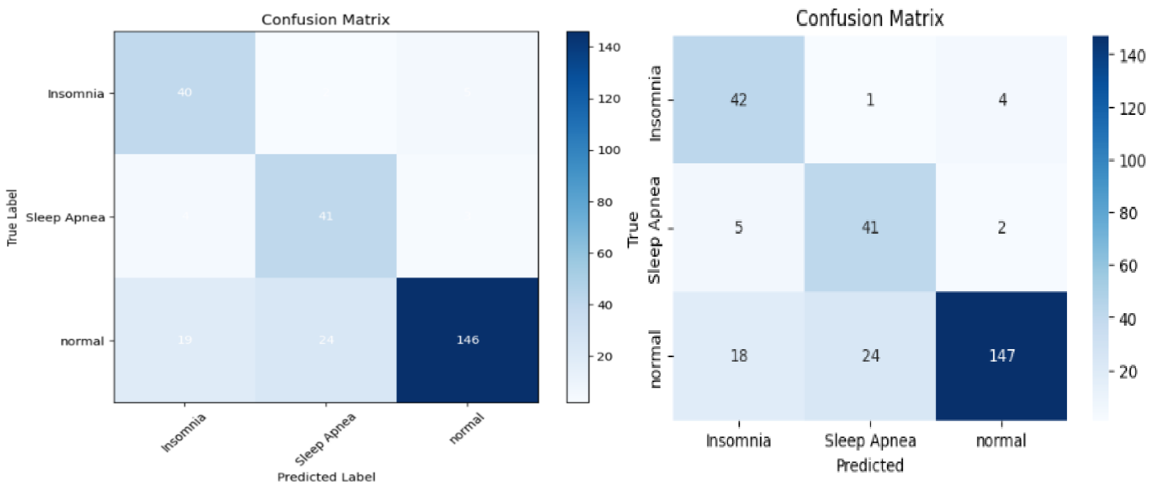


Figure 8. Zero-shot Prompting 20Newsgroup dataset doubao (**Left**), GPT-4o (**Right**).

Figure 9 shows few-shot prompting for sleep disorder. Doubao's matrix (left) shows confusion, particularly between "insomnia" and "normal." GPT-4o's matrix (right) has a stronger diagonal, indicating better accuracy across all three classes. The ROC curves show generally high AUC values for all classes (around 0.92), suggesting good performance with few-shot learning, and highlighting improvement over zero-shot results. GPT-4o appears more consistent and accurate in its predictions compared to Doubao.

Figure 10 shows few-shot prompting on 20 Newsgroups. Doubao's confusion matrix (left) still exhibits off-diagonal errors, indicating ongoing misclassifications. GPT-4o's matrix (right) shows a more prominent diagonal, suggesting improved accuracy with few-shot learning compared to zero-shot. The multi-class ROC curves show generally high AUCs for both models, with GPT-4o often achieving slightly better performance (higher AUCs in the legend), indicating enhanced discrimination between newsgroup categories with few-shot examples. The rightmost ROC curve likely represents a specific class comparison.



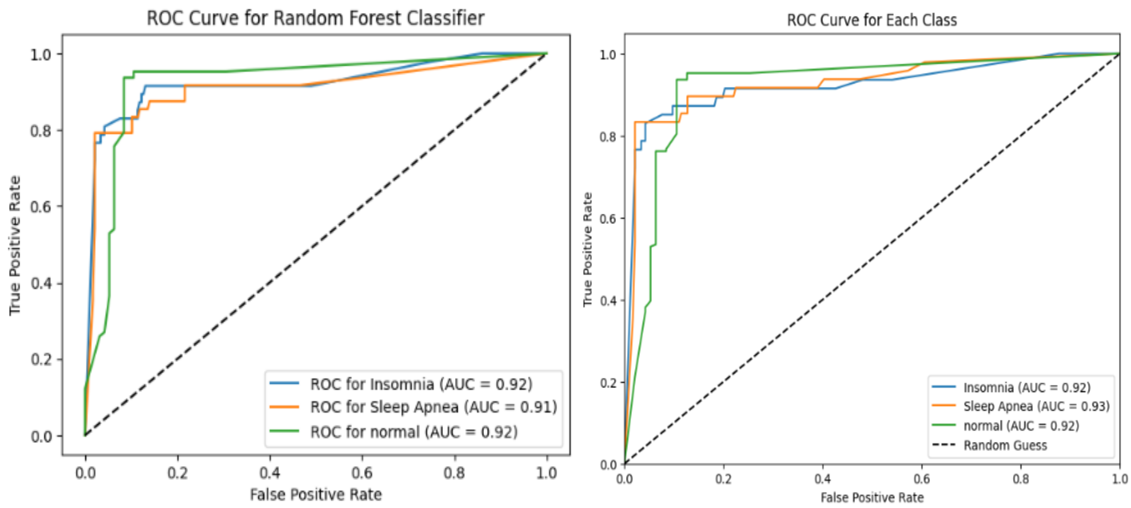


Figure 9. Few-shot Prompting sleep disorder doubao (**Left**), GPT-4o (**Right**).

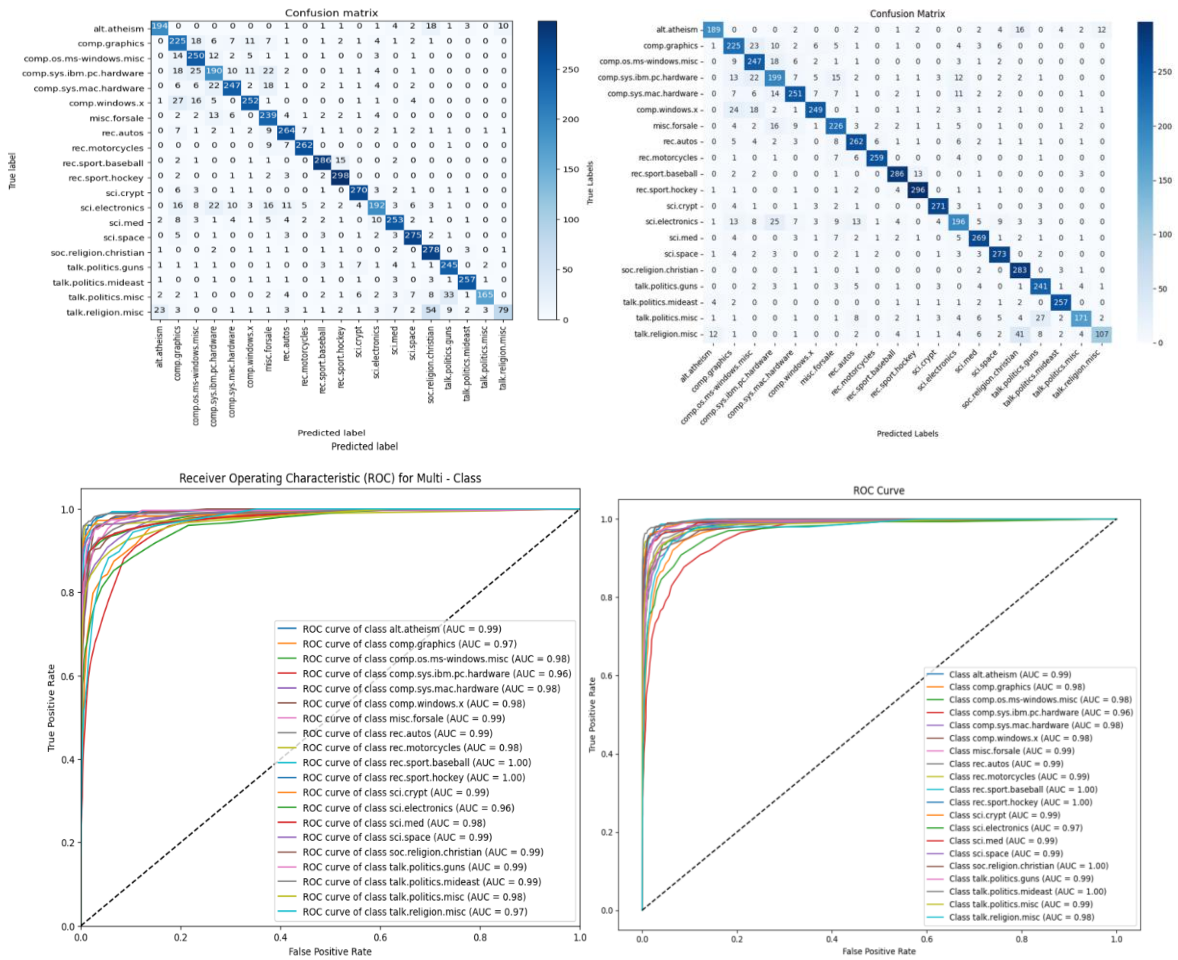


Figure 10. Few-shot Prompting 20Newsgroup doubao (**Left**), GPT-4o (**Right**).

Figure 11 displays decomposed prompting results for sleep disorder. Doubao's matrix (left) shows notable confusion, especially between "insomnia" and "normal." GPT-4o's matrix (right) exhibits a clearer diagonal, indicating improved accuracy over its zero-shot and few-shot performance. The ROC curves for the Best Model both show high AUCs for all classes, suggesting decomposed prompting is highly effective. Compared to zero-shot and few-shot prompting (Figure 6), both models, particularly GPT-4o, demonstrate significantly enhanced performance across all sleep disorder categories with decomposed prompting, achieving higher accuracy and better discrimination as reflected in the confusion matrices and elevated AUC scores. GPT-4o consistently outperforms Doubao.

Figure 12 presents decomposed prompting results for 20 Newsgroups. Doubao's confusion matrix (left) shows a more defined diagonal compared to zero-shot and few-shot, indicating improved accuracy across categories. GPT-4o's matrix demonstrates an even stronger diagonal with fewer off-diagonal errors, suggesting superior performance. The Best Model ROC curve both display very high AUC values (mostly 1.00 or very close) for nearly all newsgroup categories for both models. This signifies that decomposed prompting is highly effective for this task, leading to almost perfect classification. Compared to zero-shot and few-shot prompting, decomposed prompting achieves a substantial improvement, with both Doubao and GPT-4o demonstrating near-ideal discrimination between the different newsgroup topics. GPT-4o maintains a slight edge in overall consistency and accuracy.

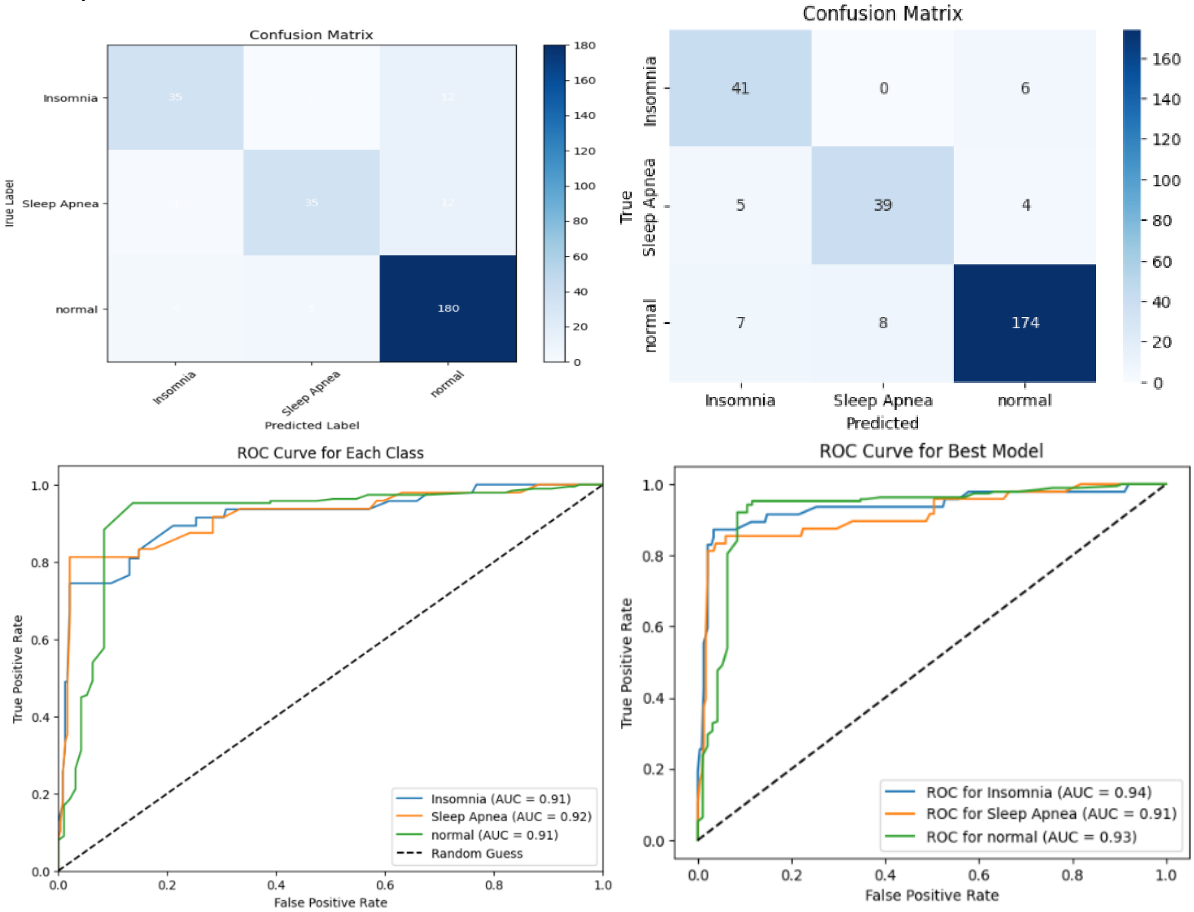
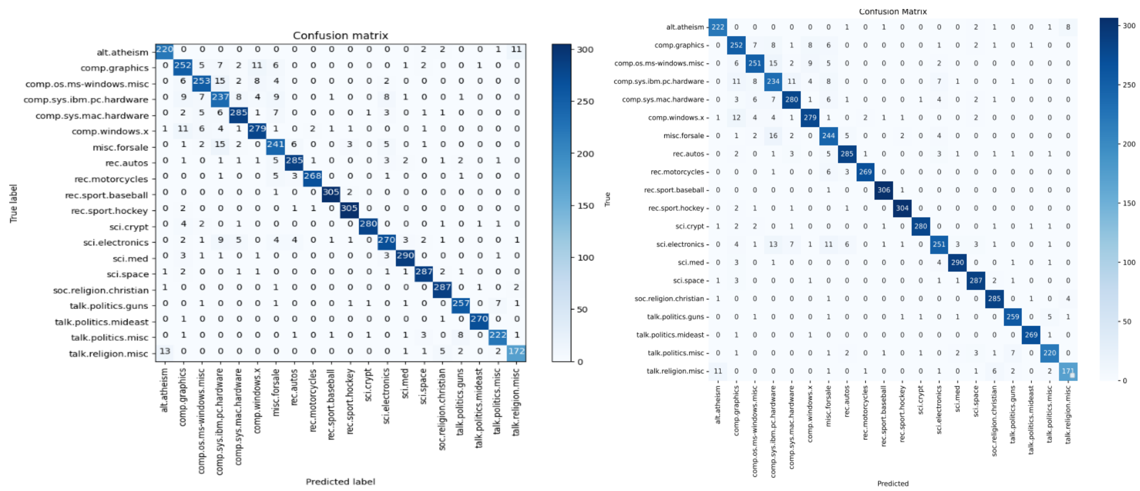


Figure 11. Decomposed Prompting sleep disorder dataset doubao (**Left**), GPT-4o (**Right**).



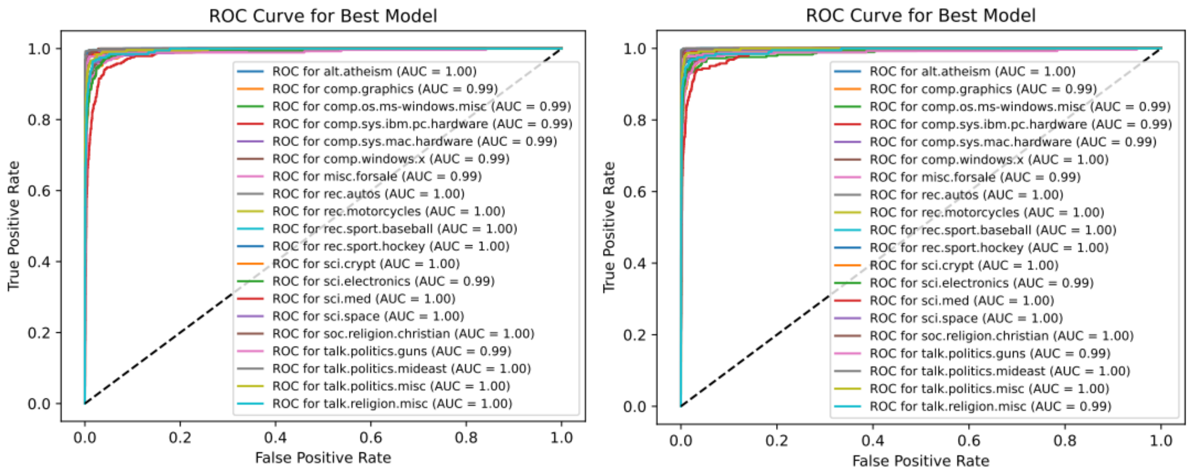


Figure 12. Decomposed Prompting 20Newsgroup dataset doubao (**Left**), GPT-4o (**Right**).

5. Discussion

Our comparative analysis of zero-shot, few-shot, and decomposed prompting strategies highlights the critical role of structured guidance in LLM-driven AutoML. The superior performance of decomposed prompting across both the primary sleep disorder task and the 20 Newsgroups generalization benchmark underscores the effectiveness of strategic task decomposition and automated classifier selection in achieving high-performing models. While zero-shot prompting offers simplicity, its limited performance in these complex classification scenarios emphasizes the need for more sophisticated prompting techniques to fully leverage LLM capabilities for AutoML. This outcome partially refutes the initial assumption that solely relying on LLMs' pre-training knowledge and basic heuristics is sufficient to effectively address the nuanced complexities, emphasizing that explicit, task-relevant instructions and data guidance are crucial for fully realizing the potential of LLMs in such complex tasks. The consistent and significant performance gains achieved with decomposed prompting in both evaluated domains firmly establish its efficacy as a powerful strategy for tackling the complexities of AutoML. The success of decomposed prompting directly addresses key challenges in traditional AutoML, such as the need for expert intervention in pipeline design and hyperparameter tuning. By enabling the LLM to systematically explore different classifiers and optimize their parameters through guided sub-tasks, this strategy effectively automates complex decision-making processes within the AutoML workflow, leading to the observed performance gains. This approach provides valuable methodological insights and a robust framework for future research aimed at advancing LLM-driven AutoML across a wide range of applications, extending beyond the specific classification tasks explored in this study.

The subpar performance of zero-shot prompting in Text classification stems from its limited data utilization and insufficient model learning capacity. Relying solely on predefined heuristics, zero-shot approaches cannot leverage training data for in-depth pattern recognition. Text classification tasks are complex, determined by intricate, often non-linear interactions among various variables. Devoid of training data exposure, zero-shot prompting struggles to discern these subtle interdependencies between features and classification outcomes, failing to capture the underlying data patterns crucial for accurate identification.

The few-samples prompting strategy significantly enhances performance by exposing the LLM to a limited set of training examples, allowing the model to discern potential correlations between feature combinations and classification target types, learning data patterns and establishing a correspondence between input features and classification results, effectively mimicking data-driven learning. Furthermore, a specific machine learning model, when coupled with essential preprocessing steps, including encoding categorical features and standardizing numerical features, demonstrably enhances data quality and optimizes the learning process. This leads to more accurate classifications, particularly for common sleep disorder types and newsgroup type. However, the inherent capacity limitations of the specified model preclude it from fully capturing the intricate non-linear relationships that characterize text classification tasks and their multifaceted determinants. Consequently, residual classification errors persist, suggesting the potential utility of more complex models capable of representing higher-order interactions within the data.

The decomposed prompting strategy emerges as the most effective approach, achieving the highest performance metrics due to its strategic task decomposition and the optimized selection of a classifier tailored to the data characteristics through automated machine learning (AutoML) techniques. It breaks down the classification task into manageable stages, systematically exploring and rigorously model parameter-tuning a diverse suite of classifiers through hyperparameter optimization techniques such as Grid Search or Random Search, followed by comparative evaluation. This meticulous AutoML process identifies the most suitable model architecture and hyperparameters for the specific dataset. This automated model selection process yields significant performance gains in both the sleep disorder and newsgroup classification. The success of the decomposed prompting strategy lies in its ability to leverage AutoML to automatically discover the classification algorithm and hyperparameter configuration best suited to the inherent structure and complexities of each dataset. By exhaustively searching the model and hyperparameter space, the decomposed prompting strategy transcends the limitations of manually selected models, identifying solutions that capture the intricate relationships between features and target variables more effectively. This automated approach allows the system to adapt to the specific characteristics of each problem domain, enabling superior classification performance compared to strategies relying on pre-defined model choices and static parameter settings.

6. Conclusions and Future Directions

Through a thoughtfully designed experimental framework and an in-depth, multifaceted analysis, we successfully demonstrate prompting LLMs to autonomously generate code for machine learning tasks, achieving strong results in our primary application area of sleep disorder classification. We convincingly validated our initial hypothesis that LLMs equipped with a code execution sandbox possess significant potential for AutoML. Furthermore, the successful application of our methods, particularly decomposed prompting, to the 20 Newsgroups benchmark validates the generalizability of this approach to standard text classification tasks. Our findings clearly highlight the pivotal influence of prompting strategies, especially task decomposition, on LLM-driven AutoML efficacy for both structured health data and unstructured text data.

A limitation of the present study lies in its focus on textual task classification using a specific set of Large Language Models (LLMs). Future research should investigate the generalizability of these findings across a wider range of models, tasks, and data modalities.

Future work necessitates comprehensive comparative experiments and rigorous external validation across diverse multimodal models, encompassing a range of NLP tasks and extending to other multimodal tasks. This is essential to enhance generalization on external datasets and mitigate overfitting. To further explore the generalizability of decomposed prompting as a core technique for LLM-driven AutoML, future research should also investigate its application to other fundamental machine learning tasks beyond classification, such as regression, clustering, and dimensionality reduction. Furthermore, examining its effectiveness with different types of data beyond text and the structured data explored in our sleep health application, including images, audio, and time series data, would provide valuable insights into the versatility and broad applicability of this approach. Key areas for future investigation include:

- Exploring multimodal integration: Investigating the integration potential of LLMs, guided by decomposed prompting, with diverse data modalities through case studies and proof-of-concept experiments. This includes analyzing their application value in real-world scenarios to validate their efficacy and utility across a broader spectrum of tasks and data types.
- Refinement of prompting strategies: Investigating advanced techniques like meta-chain of thought [18,19] and contextual prompting [20] for improved reasoning within the decomposed prompting framework.
- Dataset expansion and balancing: addressing limited diversity and imbalanced distribution via techniques like oversampling e.g., SMOTE [21] in the context of LLM-driven AutoML.
- Exploration of advanced classifier models: prioritizing sophisticated deep learning architectures, CNNS, RNNS, LSTMs, GRUs [22–26] for automatic feature learning in complex data, potentially integrated within the decomposed prompting workflow for more nuanced model selection.

Author Contributions

Y.Z.: Literature review, methodology, original draft preparation, methodology, writing, and revising. J.P.: reviewing and editing. X.Z.: reviewing and editing. W.S.: supervision, reviewing, and editing. All authors have read and agreed to the published version of the manuscript.

Funding

This research received no external funding.

Data Availability Statement

The data used in this study are from the publicly available Sleep Health and Lifestyle Dataset, which can be accessed at https://www.kaggle.com/datasets/uom190346a/sleep-health-and-lifestyle-dataset. 20Newsgroup Dataset. Available online: http://qwone.com/~jason/20Newsgroups.

Conflicts of Interest

The authors declare no conflict of interest.

Appendix A. Large Language Models Selection and Evaluation

Gemini 2.0 Flash, while capable of generating code, did not support uploading Excel files and could not execute the generated code. Manual execution of its generated code encountered data format issues, leading to program interruptions near completion.

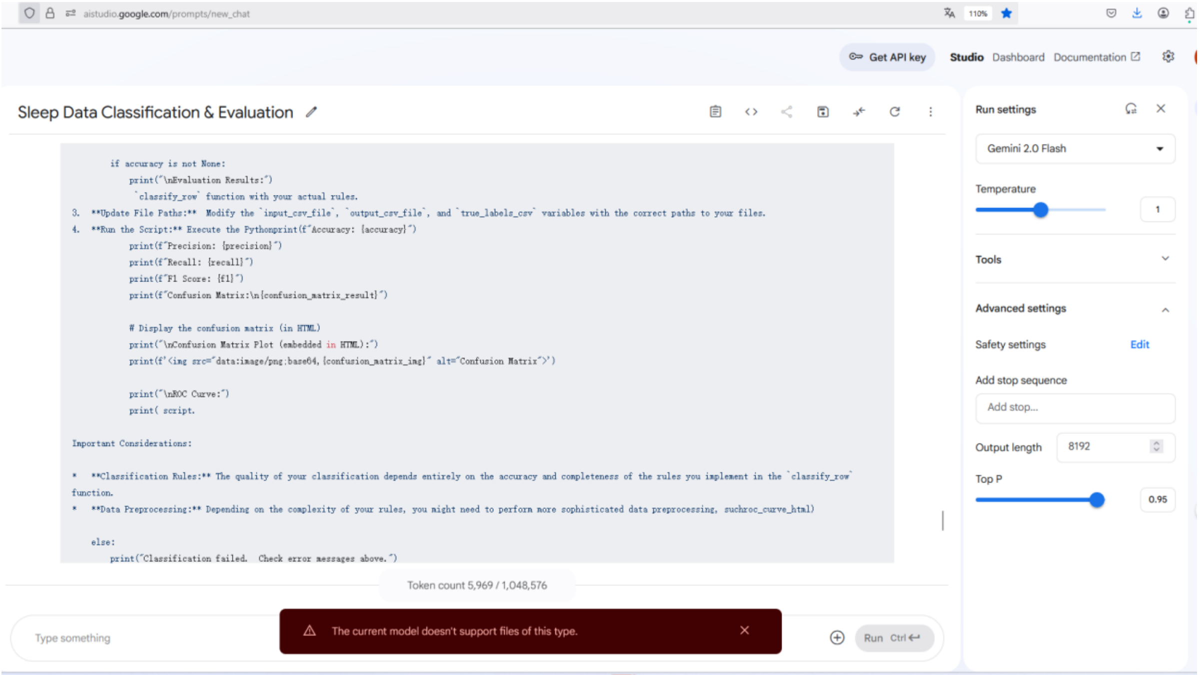


Figure A1. Gemini 2.0 Flash (a).

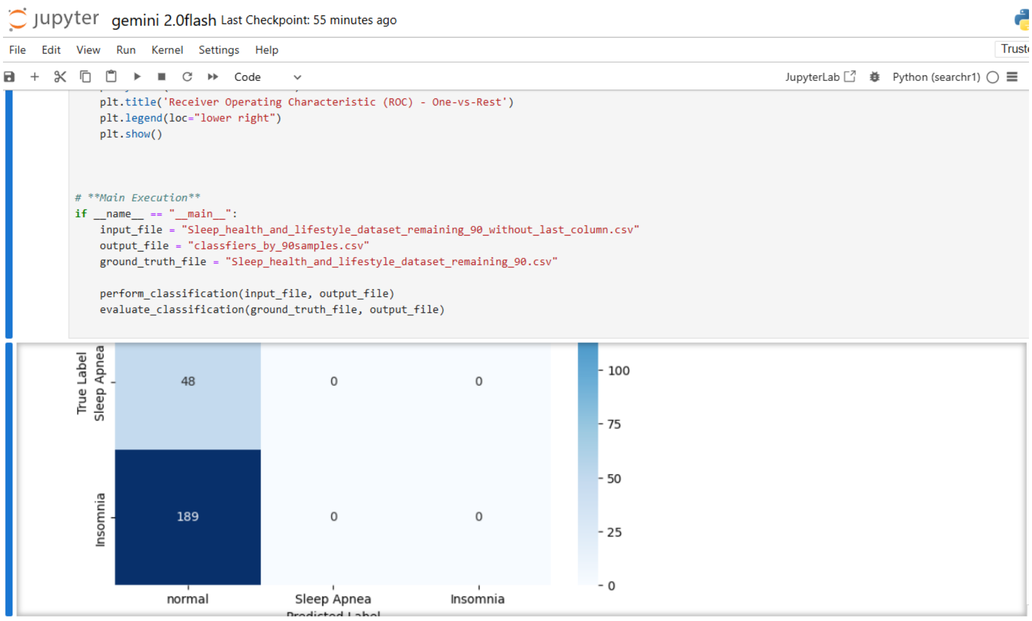


Figure A2. Gemini 2.0 Flash (b).

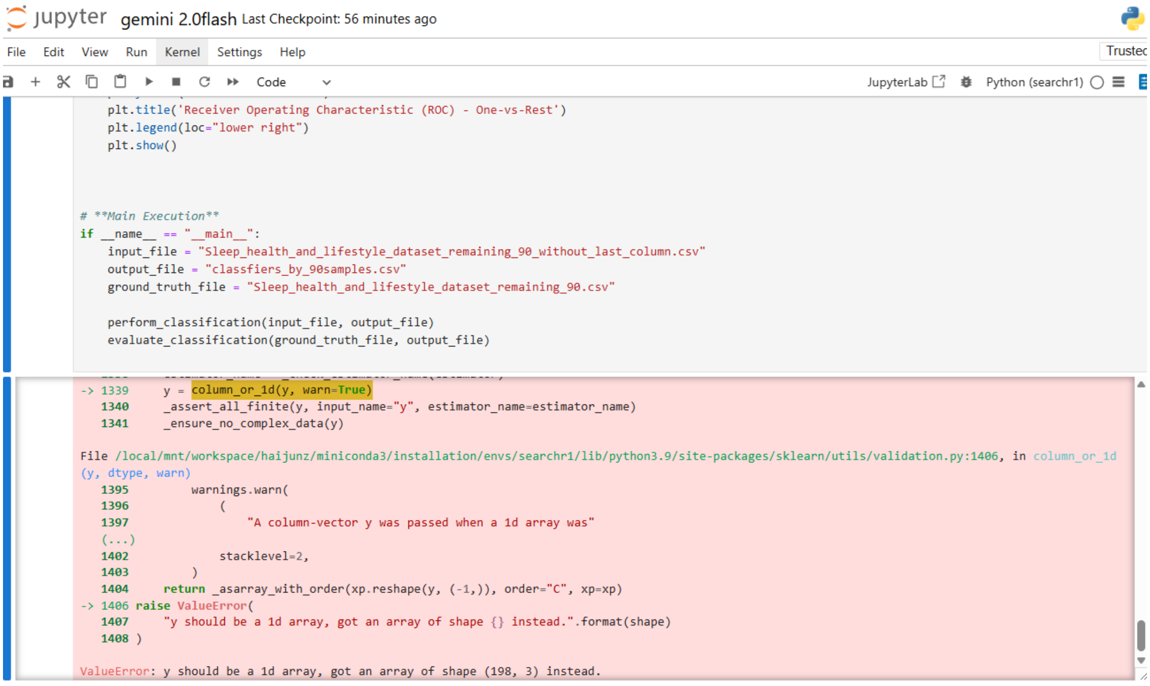
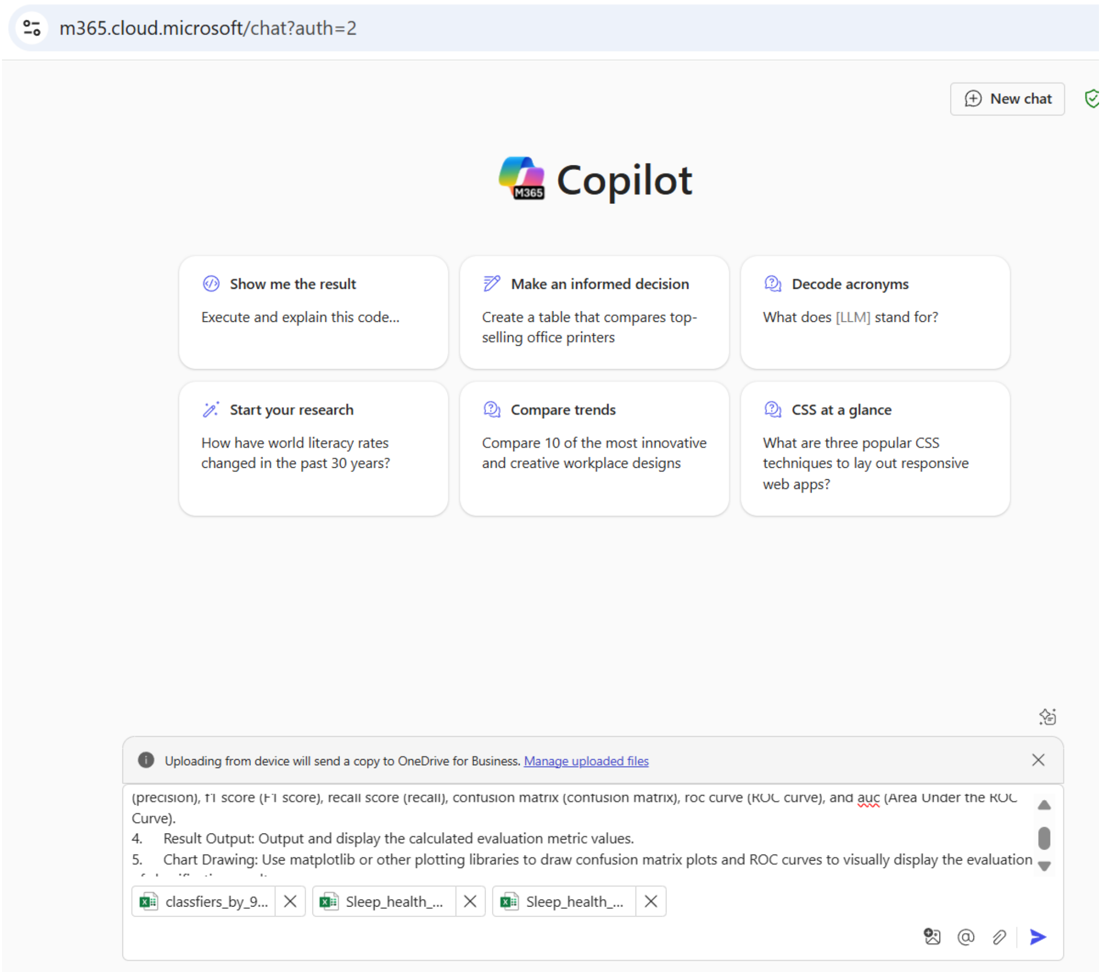


Figure A3. Gemini 2.0 Flash (c).

Copilot supported the upload of multiple Excel files, but did not display the generated code. While it could execute the code, the final results were flawed; for example, it erroneously generated a ROC curve from a Confusion Matrix, with no ability to debug.



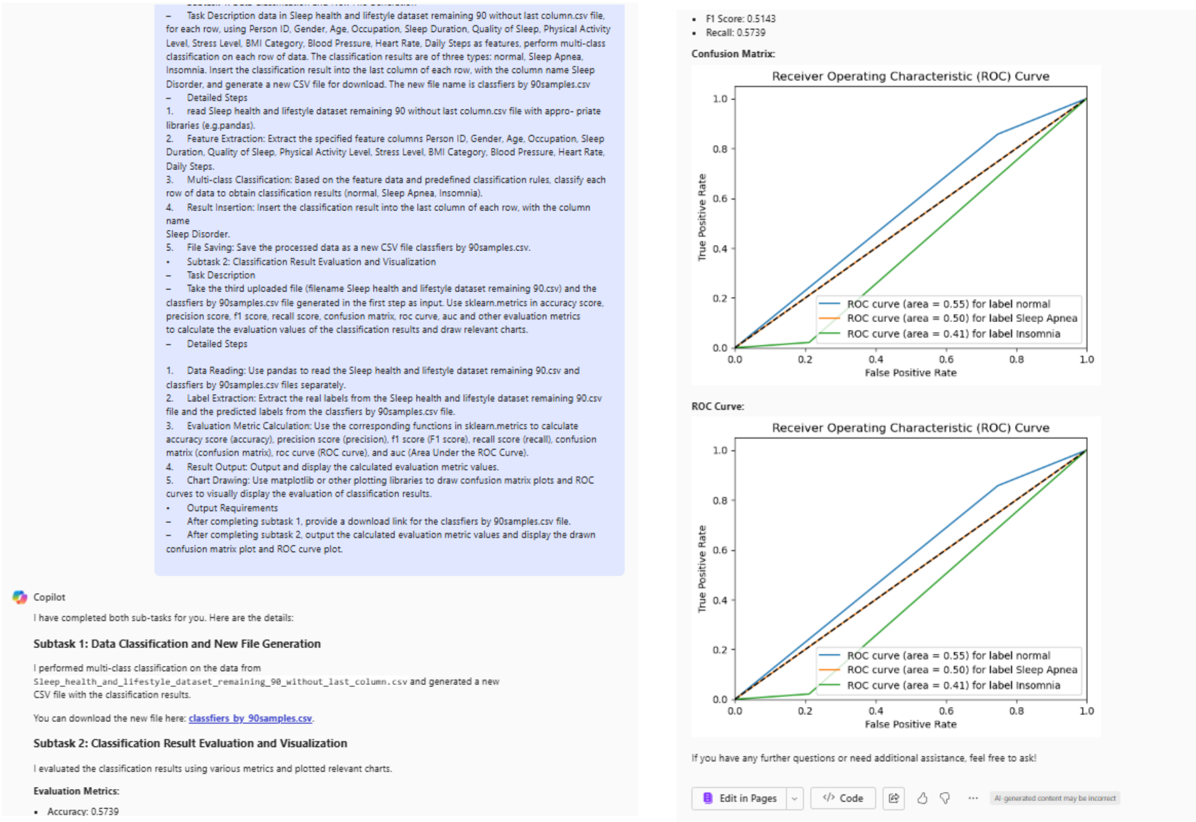


Figure A4. Copilot.

Qwen2.5-Max supported only single Excel file uploads and could generate code but not execute it.

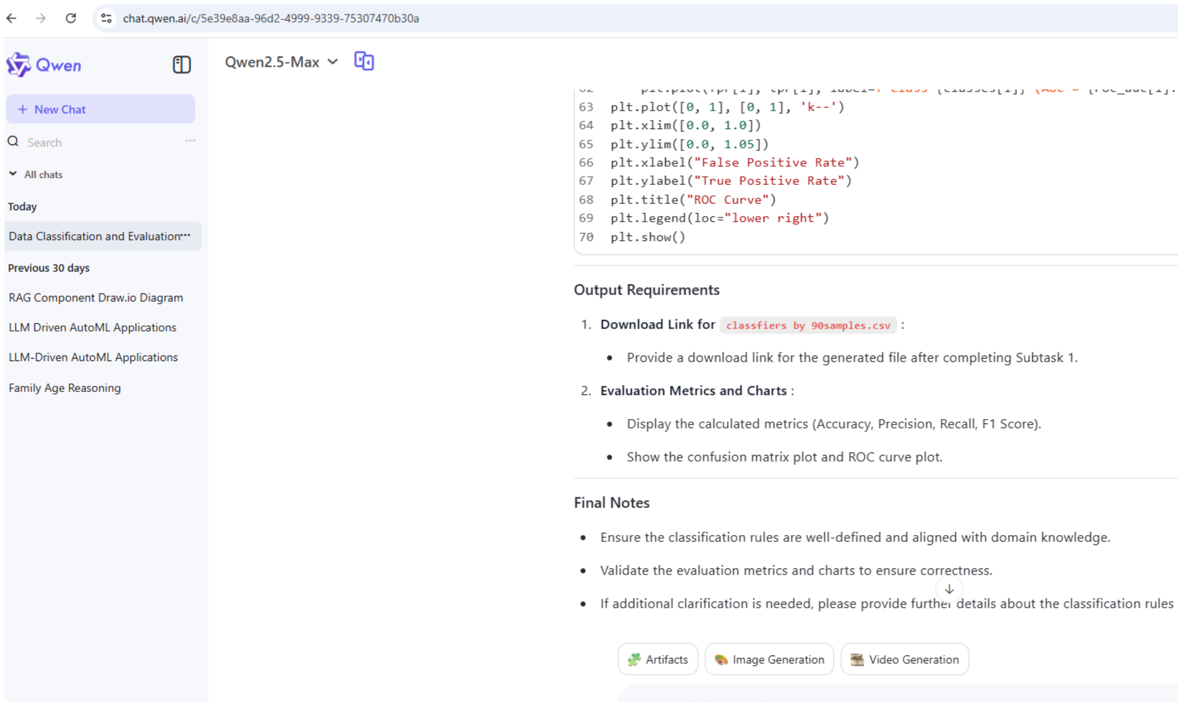


Figure A5. Qwen2.5-Max.

DeepSeek-R1 (yuanbao) supported csv files uploads and code generation.

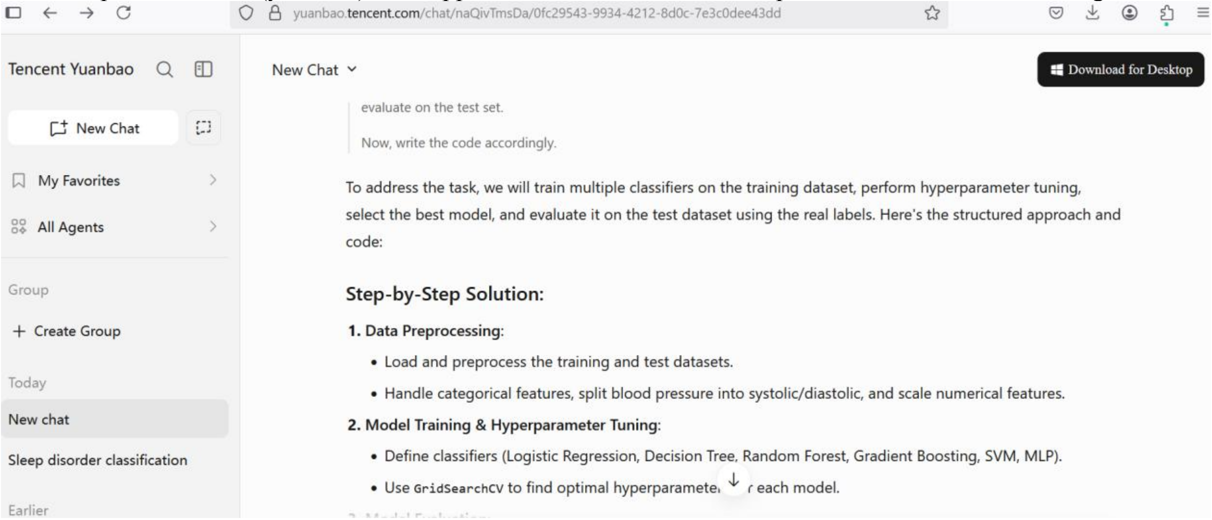


Figure A6. DeepSeek-R1.

Appendix B. Establishing Prompt Engineering Baselines: An Exploration from Naïve/Enhanced Contextual Prompts to Manually Aligned LLM-Generated Classifiers and Rule-Based Strategies

Naive Prompt generated qualitative textual analysis and descriptions of potential disorder categories but did not produce executable code for a classification model. This served as a valuable initial baseline. Enhanced Contextual Prompt auto-generated Python code that utilized a RandomForestClassifier and included the requested evaluation metrics and visualizations (Table A1).

Table A1. Naïve Prompt vs Enhanced Contextual Prompt GPT-4o output.

Prompts	GPT-4o Outputs
Naïve Prompts Which disorder category do you think these features belong to?	Based on the dataset features, the information presented is predominantly related to lifestyle, physical activity, sleep patterns, health indicators, and stress levels. This type of data is often used in the analysis of sleep disorders or health conditions that are influenced by such factors. Here's how the features relate to potential sleep or health-related disorders: Likely Disorder Categories: Sleep Disorders: Sleep Duration, Quality of Sleep, Sleep Disorder directly relate to sleep health. Disorders such as insomnia, sleep apnea, or hypersomnia could be analyzed. Lifestyle-Induced Disorders: Physical Activity Level, Stress Level, Daily Steps may contribute to conditions like chronic fatigue syndrome, obesity, or cardiovascular issues. Metabolic and Cardiovascular Disorders: BMI Category, Blood Pressure, Heart Rate are indicators for disorders like hypertension, obesity, or diabetes. Mental Health Disorders: Stress Level connects to anxiety or depression, which can also manifest as sleep disturbances. Occupational Health Disorders:

	Occupation might be explored for links to occupational stress or burnout, which in turn affect sleep and health. Each of these features contributes to understanding how lifestyle factors and health indicators interplay with sleep and related disorders. If you have a specific category in mind, I can help narrow down the analysis further!

Enhanced Contextual Prompt
Features: all columns of file (Sleep_health_and_lifestyle_dataset_remaining_90_without_last_column.csv)
Label: Sleep Disorder column of file (Sleep_health_and_lifestyle_dataset_remaining_90.csv)
Tasks: which disorder category do you think these features belong to? and to evaluate the classification results using specific evaluation metrics (sklearn.metrics in accuracy score, precision score, f1 score, recall score, confusion matrix, roc curve, auc and other evaluation metrics) and draw relevant charts.
	…
	features_file = "../dataset/Sleep_health_and_lifestyle_dataset_remaining_90_without_last_column.csv"
	labels_file = "../dataset/Sleep_health_and_lifestyle_dataset_remaining_90.csv"
	
	features_df = pd.read_csv(features_file)
	labels_df = pd.read_csv(labels_file)
	
	# Extracting features and target label
	X = features_df.drop(columns=["Person ID"]) # Drop Person ID as it's not a feature
	y = labels_df["Sleep Disorder"]
	# Split the dataset into training and test sets
	X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.3, random_state=42)
	# Train a classification model (Random Forest)
	model = RandomForestClassifier(random_state=42)
	model.fit(X_train, y_train)
	# Predict on test data
	y_pred = model.predict(X_test)
	y_proba = model.predict_proba(X_test)
	# Compute evaluation metrics
	accuracy = accuracy_score(y_test, y_pred)
	# Compute evaluation metrics
	accuracy = accuracy_score(y_test, y_pred)
	precision = precision_score(y_test, y_pred, average="weighted")
	recall = recall_score(y_test, y_pred, average="weighted")
	f1 = f1_score(y_test, y_pred, average="weighted")
	…
	# Plot Confusion Matrix
	plt.figure(figsize=(10, 6))
	sns.heatmap(conf_matrix, annot=True, fmt='d', cmap='Blues', xticklabels=model.classes_, yticklabels=model.classes_)
	plt.title("Confusion Matrix")
	plt.xlabel("Predicted")
	plt.ylabel("Actual")

An interesting observation from the LLM-generated code (from the Enhanced Contextual Prompt) was that it independently split the provided test dataset again into its own internal training and testing sets. This behavior, while demonstrating the LLM's capability, rendered its direct quantitative results incomparable with our other experiments that used a predefined, consistent train-test split. To address this and provide a robust comparison framework focused on prompt strategies as you advised, we have made the following significant modifications to our experimental baselines in the main body of the paper:

- **Manually Aligned Baseline (Derived from Enhanced Prompt):** We took the RandomForestClassifier code generated by the LLM (from the "Enhanced Contextual Prompt"). We then **manually modified this code to ensure it utilized our standard, predefined training and testing datasets.** This allowed us to evaluate the core logic of the LLM-generated model on a comparable basis with other methods. This now serves as one of our primary baselines.
- **Rule-Based Zero-Shot Prompt Baseline:** To further explore the spectrum of prompt engineering, we have also introduced and evaluated a "Rule-Based Zero-Shot Prompt" as another baseline. This prompt is

designed to guide the LLM towards a classification decision using more explicit, rule-like instructions without relying on code generation for a traditional ML model.

References

1. Ibomoiye Domor Mienye, N.; Jere, N. Survey of Decision Trees: Concepts, Algorithms, and Applications. IEEE Xplore. Available online: https://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=10562290 (13 May 2024).
2. Kumari, A.; Akhtar, M.; Shah, R.; et al. Support matrix machine: A review. *arXiv* **2023**, arXiv:2310.19717.
3. Curth, A.; Jeffares, A.; van der Schaar, M. Why do random forests work? Understanding tree ensembles as self-regularizing adaptive smoothers. *arXiv* **2024**, arXiv:2402.01502.
4. Vaswani, A.; Shazeer, N.; Parmar, N.; et al. Attention is all you need. *Adv. Neural Inf. Process. Syst.* **2017***, 30.*
5. Kim, Y.; Xu, X.; McDuff, D.; et al. Health-LLM: Large language models for health prediction via wearable sensor data. *arXiv* **2024**, arXiv:2401.06885.
6. Nori, H.; Lee, Y.T.; Zhang, S.; et al. Can generalist foundation models outcompete special-purpose tuning? Case study in medicine. *arXiv* **2023**, arXiv:2311.16452.
7. Saab, K.; Tu, T.; Weng, W.-H.; et al. Capabilities of Gemini models in medicine. *arXiv* **2024**, arXiv:2404.18416.
8. Singhal, K.; Tu, T.; Gottweis, J.; et al. Towards expert-level medical question answering with large language models. *arXiv* **2023**, arXiv:2305.09617.
9. McDuff, D.; Schaekermann, M.; Tu, T.; et al. Towards accurate differential diagnosis with large language models. *arXiv* **2023**, arXiv:2312.00164.
10. Wang, G.; Zhao, W.; Han, J.; et al. MedFound: The first medical large language model passing the physician qualification examination. *J. Artif. Intell.* **2024**, *5*, 1–12.
11. McDuff, D.; Xu, X.; Kim, Y.; et al. Personal health large language model (PH-LLM): Leveraging large language models for personalized health insights. *arXiv* **2023**, arXiv:2311.17133.
12. Zhang, Y.; Maziarka, P.; Klicpera, J.; et al. DiffSBDD: Equivariant diffusion for structure-based drug design. *arXiv* **2024**, arXiv:2403.14338.
13. Liu, X.; Uchiyama, M.; Okawa, M.; et al. Prevalence and correlates of insomnia in the Japanese general population: Results from the Japan epidemiological sleep study. *Sleep* **2000**, *23*, 497–506.
14. Katz, D.M.; Bommarito, M.J.; Gao, S.; et al. GPT-4 passes the bar exam. *Philos. Trans. R. Soc. A* **2024**, *382*, 20230254
15. Sleep Health and Lifestyle Dataset. Available online: https://www.kaggle.com/datasets/uom190346a/sleep-health-and-lifestyle-dataset (accessed on 10 May 2025).
16. 20Newsgroup Dataset. Available online: http://qwone.com/~jason/20Newsgroups/ (accessed on 10 May 2025).
17. Doubao–Your Intelligent AI Assistant. doubao.com, ByteDance. Available online: doubao.com/download/desktop (accessed on 13 May 2024).
18. Wei, J.; Wang, X.; Schuurmans, D.; et al. Chain-of-thought prompting elicits reasoning in large language models. *arXiv* **2022**, arXiv:2201.11903.
19. Xiang, V.; Snell, C.; Gandhi, K.; et al. Towards system 2 reasoning in LLMs: Learning how to think with meta chain-of-thought. *arXiv* **2025**, arXiv:2501.04682.
20. Zhou, X.; Huang, M.; Wang, H.; et al. Contextual prompting for few-shot text classification. In Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing, Abu Dhabi, UAE, 7–11 December 2022; pp. 9312–9327.
21. Chawla, N.V.; Bowyer, K.W.; Hall, L.O.; et al. SMOTE: Synthetic minority over-sampling technique. *J. Artif. Intell. Res.* **2002**, *16*, 321–357.
22. LeCun, Y.; Bengio, Y.; Hinton, G. Deep learning. *Nature* **2015**, *521*, 436–444.
23. LeCun, Y.; Bottou, L.; Bengio, Y.; et al. Gradient-based learning applied to document recognition. *Proc. IEEE* **1998**, *86*, 2278–2324.
24. Graves, A.; Mohamed, A.-R.; Hinton, G. Speech recognition with deep recurrent neural networks. In Proceedings of the 2013 IEEE International Conference on Acoustics, Speech and Signal Processing, Vancouver, BC, Canada, 26–31 May 2013; pp. 6645–6649.
25. Hochreiter, S.; Schmidhuber, J. Long short-term memory. *Neural Comput.* **1997**, *9*, 1735–1780.
26. Cho, K.; Van Merriënboer, B.; Bahdanau, D.; et al. On the properties of neural machine translation: Encoder–decoder approaches. *arXiv* **2014**, arXiv:1409.1259.