An Adaptive Dining Advisor Using Hybrid Language Models and Online Learning

Yuyan Wang
Department of Computer Science
University of Virginia
Charlottesville, VA, USA
email@virginia.edu

***Abstract*—Personalized recommendation systems often struggle with evolving user preferences. This project presents an Adaptive Dining Advisor that predicts an individual user's satisfaction with dining hall menus by combining large language model scoring with online supervised learning. The system operates in a human-in-the-loop setting, collecting user feedback incrementally and updating predictions in real time. To address the cold-start problem, the model integrates heuristic language-based scoring with a gradually learned regression model trained on user ratings. Due to limited menu availability near the end of the academic term and time constraints, a comprehensive quantitative evaluation was not completed. The complete implementation and experimental code are available in a Git Repo.**

***Index Terms*—personalized recommendation, online learning, large language models**

I. Introduction

A. Problem Definition

The University of Virginia's Observatory Hill Dining Room (O-Hill) [1] serves thousands of students daily with rotating menus across breakfast, lunch, and dinner. While staple items such as pizza, burgers, and fries are consistently available, students' dining decisions are largely driven by the changing entrées offered each day, including stir-fry options, themed cuisines, and special dishes. These variable menu items strongly influence whether a student will enjoy the meal and whether a trip to the dining hall is worthwhile.

Currently available dining apps and websites primarily provide static menu listings without any personalized assessment of how appealing a given menu may be to an individual user. As a result, students must rely on guesswork or prior experience when deciding whether to visit the dining hall, eat elsewhere, or cook at home. This project addresses this gap by developing an AI-driven system that predicts how appealing a dining hall menu is for a specific user, using a numerical rating scale from 1 (not worth visiting) to 5 (strongly recommended).

B. Importance and Interest

O-Hill is one of the most frequently visited dining halls at UVA and plays a central role in students' daily routines. However, students often spend time checking menus or walking to the dining hall only to discover that the available options do not match their preferences. This uncertainty leads to frustration, wasted dining swipes, and unnecessary food waste.

Food satisfaction has a meaningful impact on students' well-being and overall campus experience. Dining preferences are highly personal and shaped by factors such as cultural background, dietary habits, and individual taste. A personalized dining recommendation system can reduce decision-making effort, discourage unappealing visits, and improve overall satisfaction. To achieve this, the proposed system integrates a local large language model (LLM), implemented using the Ollama framework, to contextually evaluate menu descriptions, along with a supervised online learning model that continuously adapts to individual user preferences through feedback.

II. Novelty

This project introduces a personalized dining recommendation system that differs from traditional menu browsing tools and food recommendation approaches in several key ways.

First, the system adopts a hybrid reasoning-and-learning framework that combines semantic understanding from a large language model with a supervised regression model trained on an individual user's feedback. While conventional recommendation systems typically rely on either heuristic rules or historical interaction data, this approach integrates language-based reasoning with data-driven personalization.

Second, the proposed system explicitly addresses the cold-start problem. A locally deployed language model provides meaningful menu evaluations even in the absence of any training data, allowing the system to generate useful recommendations from the first interaction. As user ratings are collected, the system gradually shifts emphasis toward the learned model.

Third, the system is designed with privacy and autonomy in mind. All inference is performed locally using a lightweight language model deployed via the Ollama framework, ensuring that user profiles, dietary preferences, and feedback remain on-device rather than being transmitted to external services.

Finally, the system supports continuous online adaptation through incremental updates of a supervised regression model after each user interaction. This enables the model to track evolving preferences without requiring batch retraining or large datasets.

Overall, this project demonstrates how a hybrid LLM-based reasoning component and an online supervised learning model can be combined into a practical, privacy-preserving, and adaptive recommendation system tailored to a single user's preferences.

III. Approach

The Adaptive Dining Advisor is designed as an online learning system composed of three main components: menu acquisition, hybrid prediction, and incremental model updating.

1) Menu data is obtained either through synthetic menu generation during early system use or by scraping real dining hall menus from the dining website using automated browsing tools. To keep the system simple and robust, the scraper is limited to menu items listed under the *Hearth* and *True Balance* sections, which consistently contain the primary hot entrées and dietary-restricted options that most strongly influence dining decisions. Other sections with constant (ex. pizza and fries) or less informative items are excluded to reduce noise and complexity. The collected menu text is then cleaned and standardized before being passed to the prediction pipeline.
2) The system employs a hybrid prediction strategy. A local large language model produces an initial heuristic score for each menu based on the user's dietary profile and preferences. In parallel, a supervised regression model uses text-based features extracted from the menu to predict a user rating. These two predictions are combined using a weighted scheme, allowing the system to rely more heavily on the language model during early stages and gradually shift toward the learned model as training data increases.
3) The system operates in an online learning loop. After each recommendation, the user provides a numerical rating. This feedback is logged to persistent storage and immediately used to update the supervised model via partial fitting. The model is saved and reloaded across sessions, enabling continuous learning over time.

IV. Key Components

A. Menu Text Preprocessing

Dining hall menus are provided as structured dictionaries containing lists of dish names grouped by serving stations or categories. To enable consistent and effective learning from textual menu descriptions, the raw menu text is first normalized and cleaned.

All dish names are converted to lowercase to ensure case consistency. Punctuation and special characters are removed using regular-expression-based tokenization, producing a sequence of individual words. Common stop words that do not convey meaningful information about food content (e.g., *and*, *with*, *the*) are removed, and very short tokens are filtered out to reduce noise. The remaining words from all menu categories are aggregated into a single normalized text string representing the daily menu, where each word is an element of the menu text list. This cleaned representation preserves semantic content while reducing sparsity and irrelevant variation.

B. Text Feature Engineering

After preprocessing, the standardized menu text is converted into numerical features suitable for online supervised learning. A hashing-based text vectorization approach is used to map menu descriptions into a fixed-dimensional feature space. Specifically, a HashingVectorizer [2] with 1024 dimensions is employed to transform each menu into a sparse feature vector.

The hashing strategy eliminates the need to maintain an explicit vocabulary and allows the system to process previously unseen menu items without retraining or feature reconfiguration. This property is particularly important in an online learning setting where new dishes and menu variations appear frequently. Each dimension of the resulting feature vector represents a hashed combination of token occurrences, capturing informative patterns in menu text while maintaining constant memory usage.

C. Language Model Scoring Component

To provide a strong prior for menu evaluation and mitigate the cold-start problem, the system incorporates a local large language model (LLM) as an initial scoring mechanism. The model used in this project is `llama3.1:8b`, deployed locally via the Ollama [3] framework. Running inference locally enables low-latency predictions while preserving user privacy, as no personal data or preferences are sent to external servers.

The LLM is prompted with two primary inputs: a structured user profile describing dietary needs, allergies, and food preferences, and a cleaned textual representation of the daily dining hall menu. The model is instructed to evaluate the menu as a whole and return a numerical score on a 1–5 scale, where 1 indicates that the user should avoid the dining hall and 5 indicates a strong recommendation. In addition to the score, the model generates a short textual rationale explaining its assessment.

To ensure consistency and safety, the system prompt explicitly instructs the model to ignore protected attributes and to output its response in strict JSON format. This structured output constraint simplifies parsing and reduces ambiguity during downstream processing. In cases where the model response cannot be parsed correctly or inference fails, a neutral fallback score is used to maintain system robustness.

D. Supervised Regression and Hybrid Prediction Strategy

The personalized prediction component of the system is implemented using a supervised regression model based on the SGDRegressor [4] from the `scikit-learn` library. This model is well suited for online learning due to its support for incremental updates via stochastic gradient descent and `partial_fit`, allowing the system to learn continuously from user feedback without retraining from scratch.

The regression model takes the 1024-dimensional hashed text feature vector and the LLM score as input and predicts a numerical rating representing the user's expected satisfaction with the menu. After each user interaction, the model is updated using the newly provided rating, enabling adaptation to evolving preferences over time.

To address the cold-start problem and balance general reasoning with personalized learning, the system employs a hybrid scoring strategy that combines the output of a local large language model (LLM) with the supervised model's prediction. The final predicted score $\hat{y}$ is computed as

$$\hat{y} = (1 - \alpha)\, y_{\text{LLM}} + \alpha\, y_{\text{model}}, \tag{1}$$

where y_{LLM} denotes the score generated by the language model and y_{model} denotes the prediction produced by the supervised regression model. The weighting factor α is defined as

$$\alpha = \min\left(\frac{N}{50}, 1\right), \tag{2}$$

where N is the number of user-provided ratings collected so far. This formulation ensures that the system relies primarily on the language model when little or no training data is available and gradually shifts emphasis toward the personalized regression model as more feedback is collected.

E. Dataset and Data Collection

The dataset used in this project is constructed incrementally through user interaction and is designed to support online learning. Each data instance corresponds to a single daily menu evaluation and consists of three primary columns.

- The first column contains the preprocessed menu text, which represents the cleaned and normalized textual description of the dining hall menu for a given day.
- The second column records the score generated by the local large language model.
- The third column stores the user-provided rating, which reflects the user's true satisfaction with the menu on a 1–5 scale. This value serves as the ground-truth label for training the supervised regression model.

The dataset is stored in a comma-separated values (CSV) format and grows over time as new feedback is collected. This design supports incremental model updates and persistence across sessions, allowing the system to learn continuously from a single user's evolving preferences.

V. Evaluation

Evaluating personalized, human-in-the-loop systems presents inherent challenges due to limited data availability and the subjective nature of user feedback. In this project, evaluation is conducted in an online setting using incremental error metrics computed over user interactions.

A. Evaluation Metrics

Model performance is assessed using Mean Absolute Error (MAE) and Root Mean Squared Error (RMSE), which measure the deviation between predicted menu scores and user-provided ratings. After each interaction, the system updates cumulative error statistics and records the current MAE and RMSE. The prediction error for a single interaction is defined as the difference between the predicted menu score and the user-provided rating. Formally, the error e is computed as

$$e = \hat{y} - y_{\text{user}}, \tag{3}$$

where $\hat{y}$ denotes the predicted menu score produced by the hybrid model and y_{user} denotes the user's ground-truth rating. This error formulation is used to compute aggregate evaluation metrics over time.Tracking these metrics over time allows observation of learning behavior as additional feedback is collected. MAE provides an interpretable measure of average prediction error, while RMSE penalizes larger errors more heavily and highlights instability during early learning stages. Both metrics are well-suited for evaluating continuous-valued predictions in an online learning context.

B. Visualization

To visualize learning behavior over time, the system generates a plot of evaluation metrics after each user interaction. The plot displays both Mean Absolute Error (MAE) and Root Mean Squared Error (RMSE) as functions of the number of samples observed. The horizontal axis represents the cumulative number of user-provided ratings, while the vertical axis shows the corresponding error values.

C. Experimental Constraints

The scope of quantitative evaluation in this project is limited by practical constraints. Due to the end of the academic semester, only a small number of daily menus (till December 18th) from the Observatory Hill Dining Room were available for data collection. Additionally, the system relies on real user interaction, which restricts the volume of labeled data that can be collected within a short time frame.

VI. Conclusion

This project presents an adaptive dining recommendation system that combines a local large language model with an online supervised learning approach to generate personalized menu evaluations. By integrating semantic reasoning with incremental learning from user feedback, the system is able to produce meaningful recommendations from the first interaction and gradually adapt to individual preferences over time.

Due to time constraints and limited availability of dining hall menus near the end of the academic semester, a comprehensive quantitative evaluation was not possible. The dataset contains a small number of samples collected from real user interactions, which restricts large-scale analysis. During experimentation, practical limitations related to system latency were also observed. Specifically, waiting for the web scraper to retrieve the daily menu and for the language model to generate an initial score introduced noticeable delays in the user experience. While acceptable for prototyping and research purposes, these delays highlight the need for optimization, such as menu caching, asynchronous scraping, or lighter-weight language models, in future iterations.

Overall, this work demonstrates the feasibility of combining local language models with online learning for personalized recommendation in a privacy-preserving manner. Future work will focus on improving efficiency, extending evaluation across longer time horizons and multiple users, and exploring alternative learning strategies to further enhance adaptability and responsiveness.

References

[1] University of Virginia Dining Services, "Observatory Hill Dining Room (O-Hill)," Available online: https://virginia.campusdish.com/LocationsAndMenus/ObservatoryHillDiningRoom

[2] F. Pedregosa *et al.*, "Scikit-learn: Machine Learning in Python," *Journal of Machine Learning Research*, vol. 12, pp. 2825–2830, 2011. [Online]. Available online: https://scikit-learn.org/stable/modules/generated/sklearn.feature_extraction.text.HashingVectorizer.html

[3] Ollama, "Ollama: Run Large Language Models Locally," Available online: https://ollama.com

[4] F. Pedregosa et al., "Stochastic Gradient Descent," Scikit-learn Documentation. Available online: https://scikit-learn.org/stable/modules/generated/sklearn.linear_model.SGDRegressor.html