Aviation Electronics, Information Technology, Telecommunications, Electricals, and Controls (AVITEC)
Vol. 8, No. 1, February 2026, pp. 151-161, p-ISSN 2685-2381, e-ISSN 2715-2626
http://dx.doi.org/10.28989/avitec.v8i1.3683

Research Article

Hybrid Food Recommendation System using Term Frequency–Inverse Document Frequency (TF-IDF), K-Nearest Neighbors (KNN), and Tag-Based Similarity

Juventania Sheva Mellany*, Kusnawi
Informatics Study Program, Faculty of Computer Science, Amikom University Yogyakarta, Indonesia

Article Info

Article history:
Submitted December 4, 2025
Accepted January 20, 2026
Published February 10, 2026

ABSTRACT

The rapid growth of the digital culinary industry increases the need for intelligent menu recommendation systems that can assist customers in making accurate and personalized choices. This study develops a hybrid food recommendation system that integrates three complementary approaches: popularity-based ranking, Term Frequency–Inverse Document Frequency (TF-IDF) with K-Nearest Neighbors (KNN) item similarity, and tag-based cosine matching. The system also incorporates a Content-Based Filtering component that leverages cosine similarity to strengthen similarity modeling across textual and tag-based representations. A total of 77,157 real transaction records from SR Cipali Restaurant, collected between April and December 2024, were used as the primary data source for system development and evaluation. Data preprocessing includes cleaning, category filtering, TF-IDF transformation for product names, One-Hot Encoding for tags, and price normalization to generate structured and comparable feature representations. Experimental results show that the TF-IDF KNN model achieves the best performance with an accuracy of 0.94, recall of 1.00, and F1-score of 0.89. The popularity-based model reaches an accuracy of 0.89 with balanced precision and recall of 0.80, while the tag-based model obtains a precision of 1.00 but lower recall due to tag inconsistency and ranking selectivity. The novelty of this study lies in the use of a hybrid lightweight framework evaluated on real-world restaurant transactions, which is rarely explored in previous research dominated by benchmark datasets. The proposed system demonstrates strong practicality for small and medium-sized restaurants that lack rating data and can be further improved by enhancing tag quality and incorporating more product attributes.

Keywords:
Hybrid recommendation system;
Content-Based Filtering;
Term Frequency–Inverse Document Frequency (TF-IDF);
K-Nearest Neighbors (KNN).

Corresponding Author:
Juventania Sheva Mellany,
Informatics Study Program, Faculty of Computer Science, Amikom University Yogyakarta, Indonesia.
Email: *shevamellany@students.amikom.ac.id

1. INTRODUCTION

Advancements in information technology have accelerated digital transformation across multiple sectors, including the culinary industry. In an increasingly competitive environment, restaurants are expected to deliver services that are not only fast but also aligned with individual customer preferences [1][2]. However, many restaurants still struggle to match menu offerings with customer tastes, particularly when user feedback is limited or preferences are not clearly expressed. Prior studies on food recommendation systems report similar challenges, with sparse interaction data and ambiguous user intent remaining persistent issues [3].

These challenges also appear in other application domains. In smartphone recommendation systems, model-based collaborative filtering often relies heavily on implicit behavior and limited user feedback, which can result in inaccurate preference modeling and encourage the adoption of feature-driven and transaction-based approaches [4]. From this perspective, recommendation systems rely on item attributes and historical user interactions to produce more relevant recommendations.

Restaurant *SR Cipali*, which serves as the case study in this research, is a local dining establishment located at Rest Area Kilometer 102, *Cikopo-Palimanan (Cipali)* Toll Road, Indonesia. This restaurant serves a variety of Indonesian specialties on a small to medium scale. The offers a range of Indonesian dishes and operates on a small-to-medium scale. Although SR Cipali has accumulated substantial historical transaction data, it has not yet implemented a digital recommendation system capable of suggesting menu items based on transaction patterns or product similarity. Consequently, there is a need for an automated and intelligent system that can generate menu recommendations using historical data and intrinsic food characteristics. Such a system is also expected to assist first-time customers in making informed choices without relying on prior purchase history [3][5].

In response to these conditions, this study proposes a hybrid food recommendation system that integrates popularity-based ranking, TF-IDF–based K-Nearest Neighbor (KNN) item similarity, and tag-based cosine similarity within a unified framework. The system utilizes product features, including menu names, tags, categories, and prices to identify item similarities and produce personalized recommendations for *SR Cipali* Restaurant. In addition, a manual tag-driven Content-Based Filtering component is incorporated to enable flexible, keyword-oriented recommendations. All components are implemented within a single hybrid architecture and evaluated using standard performance metrics to assess their effectiveness in supporting local restaurants and their potential applicability to other culinary MSMEs [6][7].

Rather than relying solely on an item-based K-Nearest Neighbors (KNN) approach, the proposed system integrates KNN with a content-based filtering mechanism derived from user-provided tags. This hybrid design allows the recommendation process to capture textual similarity between food items based on keywords such as “nasi ayam” (chicken rice) or “kuah” (soup), thereby improving recommendation flexibility and personalization. Comparable approaches have been explored by Verma et al. who applied TF-IDF and cosine similarity to recipe ingredients and titles from a 10K recipe dataset, matching user inputs to relevant cooking options [8]. Zhang demonstrated similar success using TF-IDF vectorization on Zomato restaurant reviews, preprocessing messy customer feedback to identify top similar dining places through cosine similarity[9]. Other research has applied TF-IDF and cosine similarity to movie synopses, demonstrating the effectiveness of textual feature representation for relevance matching in recommendation tasks [10]. Although not the most recent technique, subsequent studies continue to confirm that TF-IDF remains a practical baseline for transforming textual descriptions into numerical vectors for similarity computation, particularly in lightweight content-based recommendation pipelines [10][11].

A number of previous studies have empirically proven the effectiveness of these approaches. Verma et al. developed a recipe recommendation system using TF-IDF vectorization and cosine similarity on 10,000 recipes, successfully matching user ingredient inputs with relevant cooking options [8]. Febrywinata et al. implemented a recipe recommendation system using Content-Based Filtering and KNN that adapted recommendations based on ingredient availability, achieving accuracy levels of up to 80% [1]. Nuri and Senyurek highlighted that similarity-based methods are highly dependent on the completeness and representativeness of textual data, noting that user-dependent filtering becomes less reliable when user information is sparse or inconsistent [11]. Rifaldy and Setiawan showed that combining user–item filtering with KNN is effective in entertainment domains such as movie recommendation systems [12]. Furthermore, studies employing hybrid TF-IDF and KNN models indicate that integrating multiple techniques can improve recommendation precision and reduce the impact of sparse user data, making such approaches well suited for food recommendation scenarios [13].

Recent research also emphasizes that modern food recommendation systems increasingly adopt feature-based, tag-driven, and similarity-oriented representations to address the limitations of explicit user feedback [14]. These approaches prioritize item attributes, contextual factors, and user-interpreted semantics, which aligns with earlier discussions on sparse interaction data. Comprehensive reviews of food recommender systems confirm that Content-Based Filtering and KNN-based similarity methods remain highly effective in culinary applications, as both techniques can operate reliably under conditions of limited user interaction [15]. Other studies underscore the benefits of structured feature modeling, particularly when dietary rules, menu attributes, and contextual constraints are incorporated to improve recommendation quality [16]. While hybrid group-based recommendation systems have also been proposed to manage overlapping user preferences, their increased computational complexity limits their practicality for lightweight, restaurant-scale deployments such as the one addressed in this study [17].

Although more advanced recommendation models, such as neural architectures combining CNN and TF-IDF have been proposed, their performance improvements are closely tied to the availability of large, well-structured training datasets. Prior work shows that CNN-based approaches can enhance pattern extraction in text-intensive domains, but they also require higher computational resources and more complex preprocessing pipelines [18]. In practical environments characterized by simpler transactional or menu data, similarity-based methods such as KNN remain efficient and sufficiently accurate, particularly when supported by appropriate feature extraction and normalization. Other deep learning–based recommendation models have demonstrated improved accuracy in various domains, yet their computational overhead continues to make lightweight algorithms like KNN more suitable for resource-constrained settings such as culinary MSMEs [19].

Most existing studies on food recommendation systems rely on benchmark datasets such as MovieLens, Kaggle, or FoodRecSys-V1, or depend on explicit user ratings and questionnaire-based preference data [15], [20]. Only a limited number of studies evaluate TF-IDF, KNN, or hybrid content-based approaches using real restaurant transaction data, despite the fact that operational datasets often contain unstructured menu names, inconsistent tags, and heterogeneous product attributes. Additionally, prior research frequently examines each recommendation technique in isolation rather than integrating popularity-based, TF-IDF–based, and tag-based similarity into a single lightweight framework. As noted by Thongsri et al. [21], rating-centric systems often fail to capture actual customer behavior, reducing their applicability in real restaurant environments. These gaps indicate that lightweight hybrid recommendation approaches grounded in real transactional data, particularly within Indonesian culinary MSMEs remain underexplored.

Although TF-IDF and K-Nearest Neighbors have been widely adopted across various recommendation domains, this study focuses on their integration and evaluation within a real-world restaurant transaction context, where explicit customer preferences are unavailable and transaction records serve as the primary signals of user behavior. The research emphasizes a comparative analysis of popularity-based, content-based, and tag-based recommendation strategies under different preference scenarios. By examining their performance on sparse and imbalanced transactional data, this study offers insights into the strengths and limitations of hybrid recommendation pipelines for menu recommendation systems.

To address the identified gaps, this study makes three primary contributions. First, it proposes a hybrid recommendation framework that integrates popularity-based ranking, TF-IDF–based KNN item similarity, and tag-based cosine matching into a unified system tailored for food menu recommendations. Second, it evaluates the proposed model using 77,157 real transaction records from an Indonesian restaurant, providing empirical evidence beyond studies that rely on ratings or questionnaire-derived data. Third, it presents a comparative performance analysis of the three recommendation mechanisms to highlight their respective strengths, limitations, and suitability for practical deployment in culinary MSMEs.

2. RESEARCH METHODS

This study adopts a structured methodological framework consisting of data acquisition, exploratory analysis, feature preparation, model development, and performance evaluation. Each stage is designed to ensure that the recommendation outputs are generated from consistent, well-structured, and representative data. The overall workflow of the proposed hybrid food recommendation system is presented in Figure 1.

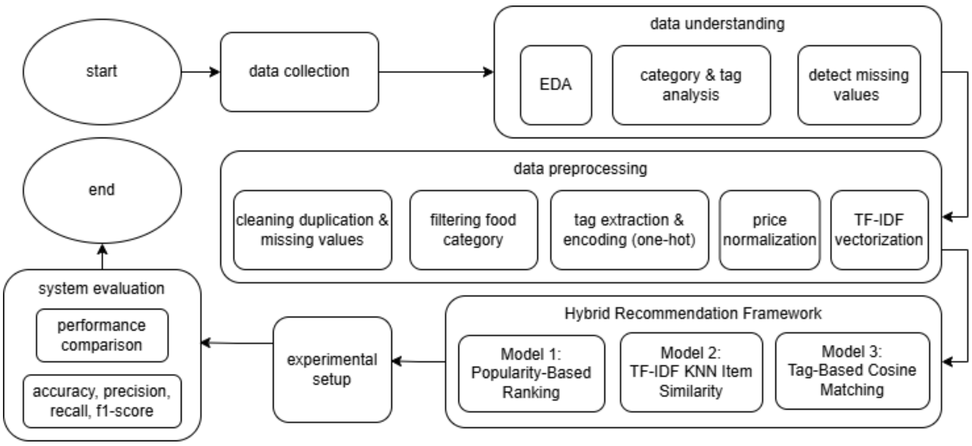

Figure 1. Hybrid food recommendation system workflow

2.1 Data Collection

The dataset used in this study was obtained directly from *SR Cipali* Restaurant in the form of Microsoft Excel files. The data spans transactions recorded between April and December 2024. The available attributes include transaction date, customer name, product name, product category, price, payment method, and product tags. This dataset is proprietary and has not been previously published.

2.2 Data Understanding

At this stage, Exploratory Data Analysis (EDA) was performed to examine the structure and characteristics of the transaction dataset, which consists of 77,157 records and 19 attributes. The analysis focused on identifying data types, examining category and tag distributions, detecting duplicate entries, and locating missing values. These steps are essential for determining which attributes are relevant for recommendation modeling, particularly Product Name, Category, Product Price, and Tags.

A category distribution analysis was also conducted to ensure that only items classified under the "Food" category were included in the modeling process, in accordance with the scope of this study. Prior to model development, an initial inspection was carried out to understand the overall data structure, identify incomplete records, and detect potential anomalies. Data cleaning was performed as part of this stage, as it plays a critical role in ensuring optimal model performance [7].

2.3 Data Preprocessing

This stage focuses on ensuring data quality and consistency before the dataset is used across the three hybrid recommendation models. Preprocessing begins with handling missing values in key attributes, namely Product Name, Category, Product Price, and Tags, followed by filtering the dataset so that only products labeled under the "Food" category are retained.

After cleaning, the dataset is prepared in two different forms to accommodate the requirements of each model. The first dataset, referred to as *"df_penjualan"*, preserves all transaction records, including duplicate menu items. This format is required for Model 1 (Popularity-Based Ranking), which determines menu popularity based on actual sales frequency. The second dataset, *"df_bersih"*, removes duplicate entries so that each menu item appears only once. This dataset is used for Model 2 (TF-IDF KNN Item Similarity) and Model 3 (Tag-Based Cosine Matching), ensuring that similarity calculations are based on feature characteristics rather than purchase volume.

Feature transformation is then applied. For Model 3 (Tag-Based Cosine Matching), the Tag attribute is decomposed into structured keyword lists and converted into numerical representations using One-Hot Encoding. Each tag is represented as a binary feature indicating its presence in a product. In addition, the price attribute is normalized using the StandardScaler method to ensure comparability with other features. The normalization process is defined by Equation (1).

$$x' = \frac{(x - \mu)}{\sigma} \quad (1)$$

with: x : original price value
μ : mean price
σ : standard deviation
x' : normalized value.

For Model 2 (TF-IDF KNN Item Similarity), textual feature representation is applied to the Product Name attribute using the Term Frequency–Inverse Document Frequency (TF-IDF) method. This technique converts menu names into numerical vectors that reflect the relative importance of terms across the dataset, enabling more accurate similarity-based comparisons. This preprocessing approach aligns with previous studies [18], which emphasize the role of structured text representation in enhancing Content-Based Filtering performance.

2.4 Hybrid Recommendation System Modeling

All recommendation components in this study are implemented within a hybrid framework, where each method addresses different user scenarios and data characteristics.

2.4.1 Popularity-Based Ranking

This model generates recommendations based on sales frequency derived from historical transaction data. Similar to the approach described in [6], the total number of orders is used as an indicator of menu popularity. Menu items with the highest sales volumes are treated as customer favorites and are prioritized in the recommendation list.

2.4.2 TF-IDF KNN Item Similarity

In this model, users provide a product name as input, and the system returns a set of menu items with similar characteristics. Similarity is computed using the K-Nearest Neighbor (KNN) algorithm with cosine similarity as the distance metric, as expressed in Equation (2).

$$x' = cos(\theta) = \frac{A \cdot B}{|A|\,|B|} \quad (2)$$

Equation (2) measures the similarity between two feature vectors A and B based on the cosine of the angle between them. The resulting value ranges from 0 to 1, where values closer to 1 indicate stronger similarity. Consequently, higher cosine similarity scores reflect closer feature alignment between products.

The selection of cosine similarity is supported by [2], which demonstrates its effectiveness in capturing relationships between menu items in restaurant datasets. Similar findings are reported in [22], where item-to-item similarity calculations contribute to improved recommendation relevance in collaborative filtering–based restaurant systems. These studies reinforce the suitability of KNN in this research, as the method prioritizes retrieving items that are most similar to the user's query.

The features used in similarity computation include TF-IDF representations of product names and tags, product categories, and normalized price values, as described in the preprocessing stage.

2.4.3 Tag-Based Cosine Matching

This model applies a Content-Based Filtering strategy in which users provide keyword-based inputs or tags, such as "*nasi ayam*". User input is transformed into a vector representation using the same encoding scheme applied to product tags in the dataset. The system then computes cosine similarity between the input vector and each product's tag vector. Items with the highest similarity scores are ranked at the top of the recommendation list.

This approach is consistent with the findings of Febrywinata et al. [1] , who demonstrated that content-based attribute matching using cosine similarity and KNN can generate relevant and preference-aligned recipe recommendations.

2.5 Experiment Configuratiom

The experimental implementation was conducted using Google Colab with Python version 3.10. The primary libraries employed include Scikit-learn for model construction and similarity computation, Pandas and NumPy for data processing, and Matplotlib for visualization.

Feature transformation was performed using TF-IDF Vectorizer for textual representation of product names, One-Hot Encoding for tag features, and StandardScaler for normalizing product prices to ensure comparable feature scales.

Model performance was evaluated using Accuracy, Precision, Recall, and F1-Score metrics to assess the relevance and suitability of the generated recommendations.

2.6 Recommendation System Evaluation

The recommendation systems were evaluated based on their ability to produce relevant menu suggestions under predefined preference scenarios. The popularity-based ranking model was assessed using historical transaction data, where frequently purchased items were interpreted as indicators of general customer preference. In contrast, the TF-IDF KNN item similarity and tag-based cosine matching models were evaluated using simulated preference inputs designed to reflect plausible customer selection behaviors.

This evaluation strategy enables controlled and consistent comparison across different recommendation approaches. The outputs generated by each model were systematically analyzed to assess their relative performance.

Four evaluation metrics were employed. Accuracy measures the proportion of correct recommendations, as defined in Equation (3).

$$Accuracy = \frac{\{TP+TN\}}{TP+TN+FP+FN} \tag{3}$$

Precision evaluates the proportion of recommended items that are relevant, as shown in Equation (4).

$$Precision = \frac{\{TP\}}{TP+FP} \tag{4}$$

Recall measures the system's ability to identify all relevant items, as expressed in Equation (5).

$$Recall = \frac{\{TP\}}{TP+FN} \tag{5}$$

The F1-Score represents the harmonic mean of Precision and Recall, providing a balanced performance measure, as shown in Equation (6).

$$F - Score = \frac{2\times(Precision\times Recall)}{Precision+Recall} \tag{6}$$

In these equations, TP (True Positive) denotes relevant items correctly recommended, FP (False Positive) represents irrelevant items incorrectly recommended, FN (False Negative) refers to relevant items not recommended, and TN (True Negative) indicates irrelevant items correctly excluded by the system.

3. RESULTS AND DISCUSSION

3.1 Preprocessing Results

The preprocessing stage yielded several key outcomes related to data quality and structural readiness for recommendation modeling. Missing values in critical attributes, namely Product Name, Category, Product Price, and Tags. were systematically identified and removed, ensuring that all retained records were suitable for subsequent analysis. Following this step, the dataset was filtered to include only items classified under the "Food" category. This filtering ensured that the modeling process remained aligned with the scope and objectives of the menu recommendation framework.

Given the transactional nature of the original dataset, preprocessing produced two distinct dataset structures to accommodate different modeling requirements. The first dataset, *df_penjualan* preserves all transaction records, including duplicate product entries. This dataset is specifically used in Model 1 (Popularity-Based Ranking), where purchase frequency serves as a direct indicator of menu popularity. In contrast, *df_bersih* is a deduplicated dataset in which each menu item appears only once. This structure is employed in Model 2 (TF-IDF KNN Item Similarity) and Model 3 (Tag-Based Cosine Matching) to ensure that similarity computations are driven by product features rather than transaction volume. The use of *df_bersih* thus provides a more objective basis for measuring item-to-item similarity.

Preprocessing also generated the feature representations required for the modeling stage. For Model 2, the Product Name attribute was transformed using the Term Frequency–Inverse Document Frequency (TF-IDF) method. This transformation produced a vectorized representation of menu names, capturing the relative importance of each term within the dataset and forming the basis for semantic similarity calculations.

For Model 3, the tag extraction and cleaning process resulted in a structured keyword list (*Tag_List)*, which was subsequently converted into numerical form using One-Hot Encoding. This process generated a binary matrix indicating the presence or absence of specific tags for each menu item. An illustrative example of this transformation is presented in Table 1.

Table 1. Example of tag transformation results

No	Product Name	Tag_List	Vector One-Hot
1	Nasi Pecel Ayam	[nasi, ayam, lalapan]	[1, 0, 1, 1, 0, 0, 0, 0, 0]
2	Nasi Soto Ayam	[nasi, kuah, ayam]	[1, 1, 0, 0, 0, 0, 0, 0, 0]
3	Nasi Pecel Lele	[nasi, lele, lalapan]	[1, 0, 1, 0, 0, 0, 1, 0, 0]
4	Bakso	[kuah, daging, mie, sapi]	[0, 1, 0, 0, 1, 1, 0, 1, 0]
5	Nasi Telur	[nasi, telur, lalapan]	[1, 0, 1, 0, 0, 0, 0, 0, 1]

The ordering of values in the One-Hot vectors follows the predefined tag sequence: [*nasi, kuah, ayam, lalapan, daging, mie, lele, sapi, telur*]. A value of 1 indicates the presence of a tag, while 0 denotes its absence. For instance, the vector [1, 0, 1, 1, 0, 0, 0, 0, 0] for *Nasi Pecel Ayam* reflects the presence of the tags *nasi, ayam,* and *lalapan*. This example illustrates how qualitative tag information is converted into a structured numerical format suitable for similarity-based recommendation modeling.

Following One-Hot Encoding, the Product Price attribute was normalized using StandardScaler to ensure compatibility with other features. This normalization step prevents price values from disproportionately influencing similarity calculations in Model 2 and Model 3, allowing all features to contribute more evenly to the recommendation process.

3.2 Hybrid Recommendation System Modeling

The hybrid recommendation system was implemented using three complementary models: popularity-based ranking, TF-IDF KNN item similarity, and tag-based cosine matching. Each model was constructed using preprocessed data and tailored to address different recommendation scenarios.

3.2.1 Popularity-Based Ranking

Model 1 applies a popularity-driven approach using the *df_penjualan* dataset, which retains all transaction duplicates. In this model, the frequency of product occurrences is treated as a proxy for customer preference. Purchase counts are computed for each menu item and ranked in descending order, with the most frequently purchased items recommended as top choices. Despite its simplicity, this approach is effective in environments with high transaction volumes and relatively stable consumption patterns.

3.2.2 TF-IDF KNN Item Similarity

Model 2 employs a TF-IDF–based similarity mechanism using the *df_bersih* dataset. Users input a specific product name, and the system identifies other menu items with the highest semantic similarity. Product names are first transformed into TF-IDF vectors, capturing the relative importance of terms across the dataset.

Cosine similarity is then applied to measure the proximity between vectors, and the K-Nearest Neighbor (KNN) algorithm is used to retrieve the most similar items. The output consists of a ranked list of menu items that are semantically closest to the user's input, reflecting shared textual characteristics in menu naming.

3.2.3 Tag-Based Cosine Matching

Model 3 is designed to accommodate preference-based inputs in the form of keywords or tags, such as *"nasi ayam"* or *"nasi kuah"*. Product tags are extracted from the *df_bersih* dataset and represented as binary vectors through One-Hot Encoding.

User input is converted into the same vector representation, after which cosine similarity is calculated between the input vector and each product's tag vector. The final recommendation score combines cosine similarity values with the number of matching tags. Products with higher overall scores are ranked more prominently, as they are considered more aligned with the user's stated preferences.

3.3 Hybrid Recommendation System Results

3.3.1 Popularity-Based Ranking

Model 1 generates recommendations based on purchase frequency derived from the *df_penjualan* dataset. This approach assumes that frequently ordered menu items reflect overall customer preferences. Products are ranked according to their occurrence in historical transaction records, and items with the highest frequencies are presented as top recommendations.

Table 2 summarizes the top five menu items with the highest sales volumes, illustrating the model's ability to capture dominant ordering patterns from real transaction data.

Table 2. Popularity-based ranking

No	Product Name	Purchase Amount
1	Nasi Pecel Ayam B	7296
2	Nasi Soto Ayam	5926
3	Nasi Pecel Ayam SP	4887
4	Nasi Ayam Goreng	4826
5	Bakso	3499

3.3.2 TF-IDF KNN Item Similarity

Model 2 processes user input in the form of product names and identifies other menus with the highest similarity scores. Product names are transformed into TF-IDF vectors, and cosine similarity is used within the KNN framework to compute distances between items in the feature space.

The recommendation output consists of the top five items with the smallest distance values relative to the user's input. Table 3 presents an example output, demonstrating how the model retrieves products semantically related to the keyword "*ayam*", along with their corresponding similarity distances.

Table 3. TF-IDF KNN item similarity

No	Product Name	Distance/Similarity Score
1	Nasi Pecel Ayam B	0.3780
2	Nasi Soto Ayam	0.4105
3	Nasi Pecel Ayam K	0.4532
4	Nasi Ayam Goreng SP	0.5051
5	Nasi Pecel Lele	1.000

Distances in Model 2 are computed as $1-$ cosine similarity, meaning that lower values indicate stronger similarity. A distance value of 1 signifies no detectable similarity between the input product and the dataset items.

3.3.3 Tag-Based Cosine Matching

Model 3 implements a content-based recommendation approach that relies on user-provided tag inputs to retrieve menu items with similar semantic characteristics. Users specify one or more descriptive keywords *"nasi kuah"*, which are then matched against the predefined tag attributes associated with each product in the dataset. This mechanism enables the system to generate recommendations that align closely with user preferences expressed in natural language terms.

To support this process, the Tag attribute of each product is first transformed into a structured list (*Tag_List).* Each unique tag is then encoded using One-Hot Encoding, resulting in a binary feature vector that represents the presence (1) or absence (0) of specific tags for every menu item. This representation ensures that all products and user inputs are projected into a common vector space, allowing similarity calculations to be performed consistently.

User input tags undergo the same encoding procedure and are converted into a binary query vector. The similarity between the user input vector and each product vector is subsequently computed using cosine similarity, which measures the angular similarity between two vectors rather than their absolute magnitude. In addition to the cosine similarity value, the system records the number of tags that directly match the user input to provide an interpretable indicator of overlap between user preferences and product attributes.

The recommendation results for the input tag *"nasi kuah"* are presented in Table 4, which lists the five menu items with the highest relevance scores. The ranking reflects a combination of cosine similarity values and the number of matching tags, with higher-ranked items exhibiting stronger proportional alignment with the input keywords.

Table 4. Tag-based cosine matching

No	Product Name	Tag_List	Tag Matching Input	Similarity Score
1	Nasi Soto Ayam	[ayam, nasi, kuah]	[nasi, kuah] (2)	0.8165
2	Nasi Sop Iga	[iga, kuah, nasi]	[nasi, kuah] (2)	0.8165
3	Nasi Sop SP	[iga, kuah, nasi]	[nasi, kuah] (2)	0.8165
4	Nasi Soto Daging	[daging, kuah, nasi, sapi]	[nasi, kuah] (2)	0.7071
5	Nasi Putih	[nasi]	[nasi] (1)	0.7071

Table 4 demonstrates that the similarity score is influenced not only by how many tags match the user input, but also by the total number of tags assigned to each product. Since cosine similarity evaluates the angle between vectors, products with fewer overall tags tend to have shorter vector lengths. As a result, when a large proportion of a product's tags match the input, the resulting vector direction is closer to the query vector, producing a higher similarity score.

This effect can be observed in menu items such as *Nasi Soto Ayam, Nasi Sop Iga*, and *Nasi Sop SP,* each of which contains three primary tags, two of which correspond directly to the input *"nasi kuah"*, hese items achieve a cosine similarity value of 0.8165, indicating a strong proportional match. In contrast, *Nasi Soto Daging* includes four tags in total. Although it also shares two tags with the input, the additional tag increases the vector length, reducing the cosine similarity value to 0.7071.

These results highlight that cosine similarity captures the relative proportion of shared attributes, rather than relying solely on the absolute count of matching tags. This characteristic is particularly beneficial in recommendation scenarios where products vary in descriptive richness, as it prevents items with excessive or less-focused tag assignments from being unfairly prioritized. Consequently, the tag-based cosine matching model provides recommendations that are not only relevant but also proportionally aligned with the user's stated preferences.

3.4 Model Evaluation

The model evaluation phase was designed to assess how effectively each component of the proposed hybrid recommendation system produces menu recommendations that align with user preferences. Rather than relying on subjective judgments, the evaluation framework compares the system-generated recommendations with reference data derived from historical transaction records and predefined evaluation scenarios. In this study, user interactions are treated as representations of real customer behavior, where preferences are expressed through specific product names or descriptive tags. The relevance of the system output is then determined by how well the recommended items correspond to these simulated yet realistic preference inputs.

To provide a comprehensive assessment, four standard performance metrics, Accuracy, Precision, Recall, and F1-Score were employed, as introduced in Chapter 2. These metrics were selected because they collectively capture different aspects of recommendation quality, including overall correctness, relevance of retrieved items, and the system's ability to identify all relevant options. The evaluation in this section focuses on analyzing the metric outcomes for each recommendation approach, enabling a structured comparison across models.

Because the three models operate under different recommendation paradigms, the evaluation procedures were adapted to reflect the underlying logic of each method while maintaining a consistent measurement framework. For Model 1 (Popularity-Based Ranking), performance is assessed by examining whether the most frequently purchased menu items, as indicated by historical sales data, appear within the Top-N recommendation list. A higher ranking of popular items indicates better alignment with general customer preferences, which is the primary objective of this model.

In Model 2 (TF-IDF KNN Item Similarity), evaluation focuses on the semantic proximity between products. Recommendations are considered relevant when the suggested items belong to the nearest neighbor set of the query product based on cosine similarity of TF-IDF feature vectors. Items that exceed a predefined similarity threshold or appear consistently within the closest neighbors are treated as correct predictions, reflecting the model's ability to retrieve items with comparable textual and attribute-based characteristics.

For Model 3 (Tag-Based Cosine Matching), performance is evaluated based on the compatibility between user-input tags and product tag representations. Products that achieve the highest cosine similarity scores while simultaneously exhibiting strong tag overlap with the input are classified as relevant recommendations. This approach ensures that both proportional similarity and explicit attribute matching are taken into account when determining prediction correctness.

By tailoring the evaluation procedure to each model's operational principles while applying a unified set of performance metrics, this evaluation strategy enables a fair and meaningful comparison among the three recommendation approaches. The resulting metric values are subsequently analyzed to identify which method, or combination of methods, delivers the most accurate and preference-aligned menu recommendations within the hybrid system.

3.5 Evaluation Results

The evaluation results of the three recommendation models are summarized in Table 5. To complement the numerical results shown in Table 5, Figure 2 presents a visual comparison of the evaluation metrics for the three models, enabling clearer observation of performance differences.

Table 5. Example of Tag Transformation Results

Metric	Popularity-Based Ranking	TF-IDF KNN Item Similarity	Tag-Based Cosine Matching
Accuracy	0.882	0.941	0.47
Precision	0.8	0.8	1.0
Recall	0.8	1.0	0.357
F1-score	0.8	0.888	0.526

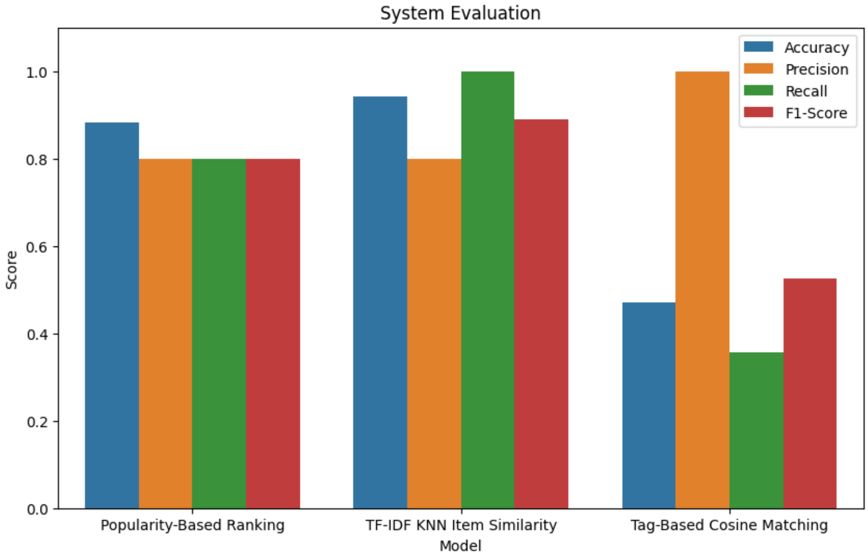

Figure 2. Evaluation chart

Based on the table, Model 2, TF-IDF KNN Item Similarity, showed the most stable performance with the highest accuracy value of 0.94 and a perfect recall of 1.00. This indicates that the KNN approach using TF-IDF vectors is capable of identifying almost all products relevant to user input, while providing consistent results in displaying similar products. This system is particularly effective when users enter menu names that are very similar to other products.

Model 1, Popularity-Based Ranking also performs quite well with an accuracy of 0.89 and balanced precision and recall values of 0.80. These findings illustrate that menus with high sales frequency tend to match the preferences of most customers. However, due to its popularity-based nature, this system is not responsive to specific or contextual requests, it is only optimal as a general recommendation based on purchasing trends.

Unlike the other two models, Model 3, Tag-Based Cosine Matching, obtained lower evaluation scores, particularly in terms of accuracy and recall, even though precision reached 1.00. High precision indicates that all products appearing in the recommendations are indeed relevant to the user's tag input. However, the low recall is due to the ranking-based system, where only the Top-N products with the highest scores are displayed as recommendations. As a result, other products that are actually relevant but not included in the Top-N are counted as negative predictions, thereby decreasing recall and accuracy.

In addition, the quality of tags in the dataset also affects the evaluation results. Many products have incomplete or inconsistent tags, causing them to fail to compete in similarity score calculations, even though they are actually relevant. Model 3 also works very selectively because the score is determined from a combination of cosine similarity and the number of matching tags. This selectivity results in very high precision but sacrifices the system's ability to capture all relevant items, resulting in lower F1-Scores and accuracy compared to the other two models.

Overall, the differences in evaluation scores show that each system has advantages in specific user contexts. Model 1 excels in general recommendations based on sales trends, Model 2 is optimal for specific searches based on menu names, while Model 3 provides accurate recommendations in keyword-based use cases, although its performance is affected by the completeness and consistency of tags in the dataset.

A number of previous studies provide useful reference points for interpreting the performance of the proposed models. Verma et al. [8] reported that a TF-IDF recipe recommender performed well for ingredient matching, although its effectiveness depended solely on text similarity. The TF-IDF KNN model in this study achieved a higher accuracy of 0.94, indicating that combining TF-IDF vectorization with additional hybrid cues yields more stable results when applied to real transaction data

Similarly, Febrywinata et al. [1] obtained an accuracy of 0.80 using TF-IDF and KNN on recipe datasets driven by ingredient text. The higher accuracy observed in this research suggests that incorpo8rating structured attributes such as tags, categories, and prices generates stronger similarity signals than text-only representations.

Zhang [9] found that TF-IDF and cosine similarity perform effectively for restaurant review matching but are sensitive to messy textual data, a pattern reflected in this study's tag-based model, which achieved perfect precision but weaker recall due to inconsistent tagging. The hybrid design used here reduces this limitation by supplementing tag similarity with popularity cues and TF-IDF KNN matching, resulting in more robust overall performance.

Taken together, these comparisons show that lightweight hybrid configurations, when supported by multi-feature representations tend to outperform single-technique approaches documented in earlier work, especially in real-world culinary MSME environments.

4. CONCLUSION

This study successfully developed a menu recommendation system for *SR Cipali* Restaurant using three different approaches, namely popularity-based recommendations, product name similarity using TF-IDF and KNN, and tag matching using cosine similarity. The pre-processing results produced clean, structured, and relevant data, thereby supporting the modeling process optimally. Evaluation of the three models showed that each approach had its own characteristics and advantages. The popularity-based system was able to capture actual ordering patterns and provide recommendations in line with general purchasing trends. The product name-based system performed best with the highest accuracy and recall, indicating that TF-IDF representation was effective in identifying menus with semantic similarities to each other. Meanwhile, the tag-based system offers highly precise recommendations when users have clear preferences, although its performance is greatly influenced by the completeness and consistency of tags in the dataset. Overall, these three systems complement each other and show that different recommendation approaches can address diverse user needs, ranging from general recommendations to specific keyword-based searches. These findings confirm that integrating simple but targeted methods can produce effective recommendation systems for restaurant scenarios with a wide variety of menus. This research provides practical contributions in the utilization of transaction data and product content to support customer and restaurant manager decision-making. In the future, research can be improved by enriching product attributes, improving tag quality, and combining hybrid approaches to increase the accuracy and relevance of recommendations in various contexts of use.

REFERENCE

[1] E. Febrywinata, R. Rahmatika Az-Zahra, and K. Nurfitri, "Sistem rekomendasi masakan berdasarkan ketersediaan bahan menggunakan metode content-based filtering dengan algoritma K-Nearest Neighbors [Food recipe recommendation system based on ingredient availability using content-based filtering and the K-Nearest Neighbors algorithm] (in Indonesian)," *Jurnal Teknologi Informasi dan Terapan (JUTITI)*, vol. 5, no. 2, pp. 131–140, 2023. https://doi.org/10.55606/jutiti.v5i2.5690

[2] R. Singh and P. Dwivedi, "Food recommendation systems based on content-based and collaborative filtering techniques," in *Proc. 14th Int. Conf. on Computing Communication and Networking Technologies (ICCCNT)*, 2023, pp. 1–6. https://doi.org/10.1109/ICCCNT56998.2023.10307080

[3] J. N. Bondevik, K. E. Bennin, Ö. Babur, and C. Ersch, "A systematic review on food recommender systems," *Expert Systems with Applications*, vol. 236, Art. no. 122166, Mar. 2024. https://doi.org/10.1016/j.eswa.2023.122166

[4] F. A. Prayoga and K. Kusnawi, "Smartphone recommendation system using model-based collaborative filtering method," *Jurnal Teknik Informatika (JUTIF)*, vol. 3, no. 6, pp. 1613–1622, Dec. 2022. https://doi.org/10.20884/1.jutif.2022.3.6.413

[5] L. Xia, C. Huang, Y. Xu, J. Zhao, D. Yin, and J. Huang, "Hypergraph contrastive collaborative filtering," in *Proc. 45th Int. ACM SIGIR Conf. on Research and Development in Information Retrieval*, 2022, pp. 70–79. https://doi.org/10.1145/3477495.3532058

[6] H. H. Wang, M. Zhao, X. Xie, W. Li, and M. Guo, "Knowledge graph convolutional networks for recommender systems," in *Proc. The Web Conference (WWW)*, 2019, pp. 3307–3313. https://doi.org/10.1145/3308558.3313417

[7] C. N. Sánchez, J. Domínguez-Soberanes, A. Arreola, and M. Graff, "Recommendation system for a delivery food application based on number of orders," *Applied Sciences*, vol. 13, no. 4, Art. no. 2299, 2023. https://doi.org/10.3390/app13042299

[8] V. P. Verma, "Recipes recommendation system using machine learning," *International Journal of Research in Applied Science and Engineering Technology*, vol. 13, no. 5, pp. 3583–3587, May 2025. https://doi.org/10.22214/ijraset.2025.71001

[9] S. Zhang, "Restaurant recommendation system based on TF-IDF vectorization: Integrating content-based and collaborative filtering approaches," in *Proc. Int. Conf. on Data Science, Advanced Algorithm and Intelligent Computing (DAI)*, 2024, pp. 610–618. https://doi.org/10.2991/978-94-6463-370-2_62

[10] P. V. Snigdha, M. Naveen, S. Rahul, C. N. Sujatha, and P. Pradeep, "Movie recommendation system using TF-IDF vectorization and cosine similarity," *International Journal of Research in Applied Science*

and Engineering Technology, vol. 10, no. 8, pp. 1128–1134, Aug. 2022. https://doi.org/10.22214/ijraset.2022.46367

[11] Y. Nuri and E. Senyurek, "Research abstracts similarity implementation using the TF-IDF algorithm," *IOSR Journal of Computer Engineering*, vol. 27, no. 1, pp. 4–10, 2025. https://doi.org/10.9790/0661-2701040410

[12] Z. T. Yap, S. C. Haw, and N. E. Binti Ruslan, "Hybrid-based food recommender system utilizing KNN and SVD approaches," *Cogent Engineering*, vol. 11, no. 1, Art. no. 2436125, 2024. https://doi.org/10.1080/23311916.2024.2436125

[13] R. S. Rao, "Personalized book recommendation system using a TF-IDF and KNN hybrid," *International Journal of Research in Applied Science and Engineering Technology*, vol. 10, no. 7, pp. 3872–3874, Jul. 2022. https://doi.org/10.22214/ijraset.2022.45736

[14] R. Yera, A. A. Alzahrani, L. Martínez, and R. M. Rodríguez, "A systematic review on food recommender systems for diabetic patients," *International Journal of Environmental Research and Public Health*, vol. 20, no. 5, Art. no. 4248, 2023. https://doi.org/10.3390/ijerph20054248

[15] Y. Y. Chow, S. C. Haw, P. Naveen, E. A. Anaam, and H. Bin Mahdin, "Food recommender system: A review on techniques, datasets and evaluation metrics," *Journal of System and Management Sciences*, vol. 13, no. 5, pp. 153–168, 2023. https://doi.org/10.33168/JSMS.2023.0510

[16] Z. Patel and H. J. Chokshi, "Food recommender system: Methods, challenges, and future research directions," *International Journal of Engineering Trends and Technology*, vol. 73, no. 5, pp. 123–130, 2025. https://doi.org/10.14445/22315381/IJETT-V73I5P123

[17] R. N. Bin Rais, M. Noman, O. Khalid, and I. A. Khan, "A hybrid group-based food recommender framework for handling overlapping memberships," *Applied Sciences*, vol. 14, no. 13, Art. no. 5843, 2024. https://doi.org/10.3390/app14135843

[18] M. Li, "Recommendation system building based on CNN and TF-IDF approaches," in *Proc. Int. Conf. on Artificial Intelligence and Data Engineering*, 2024, pp. 215–220.

[19] L. N. Luyen, M. H. Abel, and P. Gouspillou, "Enhancing context-aware recommender systems through deep feature interaction learning," *Journal of Multi-Criteria Decision Analysis*, vol. 32, no. 1, pp. 45–58, 2025. https://doi.org/10.1002/mcda.70012

[20] R. Alabduljabbar, "Matrix factorization collaborative-based recommender system for Riyadh restaurants: Leveraging machine learning to enhance consumer choice," *Applied Sciences*, vol. 13, no. 17, Art. no. 9574, 2023. https://doi.org/10.3390/app13179574

[21] N. Thongsri, P. Warintarawej, S. Chotkaew, and W. Saetang, "Implementation of a personalized food recommendation system based on collaborative filtering and the knapsack method," *International Journal of Electrical and Computer Engineering*, vol. 12, no. 1, pp. 630–638, 2022. https://doi.org/10.11591/ijece.v12i1.pp630-638

[22] M. J. Awan *et al.*, "A recommendation engine for predicting movie ratings using a big data approach," *Electronics (Switzerland)*, vol. 10, no. 10, Art. no. 1215, May 2021. https://doi.org/10.3390/electronics10101215