



NLP-BASED CHATBOT FOR FOOD WEBSITES

Krish¹, Harshit Sukhija², Nitish³, Vinay⁴, Ms.Aarti Chahal⁵

Department of CSE-AI&DS,

Panipat Institute of Engineering & Technology, Samalkha, India^{1,2,3,4,5}

⁵Mail ID: aarti.cse.ai-ds@piet.co.in

Abstract—Food websites provide a large number of menu items, but users often have difficulty identifying suitable dishes because conventional interfaces depend on manual browsing, filtering, and keyword search. This paper presents an NLP-based conversational chatbot designed for food websites to transform natural-language requests into personalized food recommendations. The proposed framework combines intent detection, named-entity and attribute extraction, semantic matching, dialogue-state tracking, and a recommendation layer connected to structured menu data. User requests such as “I want a spicy vegetarian meal under ₹300” are converted into constraints involving cuisine, dietary preference, price, taste, and dish type. A hybrid retrieval strategy combines lexical matching with sentence-level semantic similarity, while a dialogue manager asks clarification questions when required information is missing or ambiguous. The paper also defines an evaluation methodology based on intent accuracy, slot/attribute extraction F1-score, recommendation precision, response relevance, latency, and user satisfaction. The design emphasizes explainable recommendations and safeguards against hallucinated menu information by grounding responses in the website’s menu database. The proposed system can serve as a practical architecture for intelligent food ordering, discovery, and customer-support applications.

Keywords: Natural language processing, chatbot, food recommendation, conversational AI, intent detection, semantic search, restaurant websites.

INTRODUCTION

Online food ordering and restaurant discovery have changed the way consumers select meals. Although food websites expose menus, prices, categories, offers, and dietary information, the interaction is frequently organized around static menus and keyword-based search. This creates friction when a user has a goal expressed in natural language rather than as a precise filter. A user may know what they want semantically—“something light, vegetarian, spicy, and below ₹250”—without knowing the exact menu terms used by a website.

Conversational systems provide a natural alternative because they allow users to state requirements in their own words. Recent research on restaurant recommendation shows the value of textual information, sentiment analysis, and contextual preferences for personalized recommendations [1]–[4]. Food recommender systems have also increasingly incorporated machine learning, graph-based methods, and personal attributes, although the literature identifies data sparsity, personalization, and evaluation as continuing challenges [5], [6].

Large language models (LLMs) have expanded the capabilities of conversational interfaces, but direct generation is not sufficient for menu-oriented applications. An LLM can produce fluent text while



incorrectly inventing a dish, price, or ingredient. Retrieval-augmented generation (RAG) and grounding techniques address this problem by retrieving relevant evidence before response generation [7]. Therefore, a food website chatbot should combine NLP understanding with a structured menu knowledge source rather than relying only on generative output.

This paper proposes an NLP-based chatbot architecture specifically designed for food websites. The main contributions are: (1) a domain-oriented intent and attribute model for food queries; (2) a hybrid semantic and constraint-based recommendation method; (3) a dialogue manager for clarification and multi-turn interaction; (4) a grounded response-generation mechanism; and (5) an evaluation framework that measures both NLP performance and recommendation usefulness.

RELATED WORK

Food recommendation has traditionally been approached using content-based filtering, collaborative filtering, hybrid models, and context-aware techniques. Trattner and Elsweller surveyed food recommender systems and highlighted challenges involving nutrition, group recommendation, and healthy eating [6]. A systematic review by Bondevik et al. found that content-based methods remain common and that many systems rely on a single data source, while personal attributes are not always modeled adequately [5].

Textual information is particularly important for restaurant recommendation because reviews and food categories contain signals that are difficult to represent using only numerical ratings. Li et al. developed a restaurant recommendation model using textual information to estimate consumer preferences and reported improvements over comparison approaches [1]. Sentiment-aware approaches have similarly extracted food preferences from user comments and used semantic similarity to generate recommendations [2]. More recent work has employed BERT-based aspect-based sentiment analysis to model multiple restaurant attributes [3].

Conversational recommendation extends these ideas by making the interaction interactive rather than one-shot. ChatDiet demonstrates the potential of LLM-augmented conversational food recommendation while also emphasizing the need for personalization beyond standalone language generation [4]. Research on restaurant recommendation chatbots has further examined continued adoption and identified effort expectancy, hedonic motivation, price value, and habit as relevant factors [8].

Modern NLP components can support the proposed architecture. Word embeddings represent semantic relationships [9], transformer architectures provide attention-based sequence modeling [10], and BERT enables contextual language representations [11]. Sentence-transformer models are effective for semantic similarity and retrieval [12]. Intent classification and slot filling can therefore be implemented using a combination of transformer encoders, entity extraction, and rule-based constraints.

PROBLEM FORMULATION

Let a food website contain a menu $M = \{m_1, m_2, \dots, m_n\}$, where each item m_i is represented by attributes such as name, category, ingredients, cuisine, dietary tags, taste descriptors, price, availability, and textual description. A user query q is a natural language sequence. The chatbot must infer an intent I and a set of attributes A from q , then return a ranked subset R of M .

The recommendation task can be expressed as $R = \text{Top-}k(m \in M \mid C(A, m))$, where C is a constraint function. For soft preferences, a relevance score can be defined as $S(m|q) = \alpha S_{\text{sem}} + \beta S_{\text{attr}} + \gamma S_{\text{pref}} + \delta S_{\text{text}}$, where S_{sem} is semantic similarity between the query and menu description, S_{attr} measures



attribute compatibility, S_{pref} represents known user preferences, and S_{text} represents lexical overlap. The weights α , β , γ , and δ can be tuned on a validation dataset.

The system must also distinguish between hard constraints and soft preferences. For example, “vegetarian” can be treated as a hard dietary constraint, whereas “preferably spicy” may be a soft preference. This distinction reduces unsuitable recommendations and makes the ranking process interpretable.

PROPOSED SYSTEM ARCHITECTURE

The proposed system follows a modular pipeline consisting of the user interface, NLP preprocessing, intent and attribute extraction, dialogue-state tracking, menu retrieval, recommendation ranking, and grounded response generation. The modular design allows the NLP model, database, or response generator to be replaced independently.

A. USER INTERFACE AND QUERY PROCESSING

The chatbot is embedded in a food website as a conversational widget. The user can type requests in natural language, including incomplete or comparative queries. Preprocessing includes normalization, tokenization, spelling correction where appropriate, and detection of numerical expressions such as prices. The system should preserve important food entities and negations; for example, “not spicy” must not be reduced to the token “spicy” alone.

B. INTENT DETECTION

The intent layer categorizes requests into domain actions. Example intents include `food_search`, `recommendation`, `menu_question`, `dietary_filter`, `price_filter`, `ingredient_question`, `availability_check`, `order_support`, and `clarification`. A transformer-based classifier can be fine-tuned on annotated food-domain utterances. A lightweight baseline can use TF-IDF with logistic regression, while the transformer model provides contextual understanding.

C. ATTRIBUTE AND ENTITY EXTRACTION

The attribute extractor identifies entities such as cuisine, dish, ingredient, dietary preference, taste, meal type, price range, calorie preference, quantity, and location. Slot values are normalized to the vocabulary used by the menu database. For example, “no meat” and “vegetarian” may be mapped to a vegetarian constraint, subject to the restaurant's defined dietary metadata. Named-entity recognition can be implemented using a transformer sequence-labeling model, while domain dictionaries provide coverage for restaurant-specific dish names.

D. DIALOGUE-STATE TRACKING

Multi-turn interaction requires the chatbot to maintain a compact state. A state can be represented as $D_t = \{I, A, H\}$, where I is the current intent, A is the current attribute set, and H contains relevant dialogue history. If a user says “show me something under 300” after asking for vegetarian food, the system should retain the vegetarian constraint rather than treating the second message as an independent search.

The dialogue manager also identifies missing information. If the query “recommend a pizza” produces many candidates, the system can ask a focused question such as “Would you prefer vegetarian or non-vegetarian?” Clarification should be minimized because excessive questioning increases interaction cost.

E. MENU KNOWLEDGE BASE

The menu database is the authoritative source for factual information. Each item should have a stable identifier and structured fields for name, description, category, ingredients, dietary labels, price, availability, cuisine, and optional nutrition values. A preprocessing pipeline converts menu text into



searchable representations. The database can be implemented using a relational database for transactional consistency and a vector index for semantic retrieval.

F. HYBRID RECOMMENDATION ENGINE

The recommendation engine combines hard filtering and soft ranking. First, hard constraints remove incompatible menu items. Second, semantic retrieval identifies dishes that are conceptually related to the query. Sentence embeddings can be generated for menu descriptions and the user query, and cosine similarity can be used for retrieval [12]. Finally, the ranking layer combines semantic similarity, attribute compatibility, user preferences, popularity, and optional business rules.

G. GROUNDED RESPONSE GENERATION

The final response should be generated only from retrieved menu records. A template-based response can be used for highly structured answers, while an LLM can paraphrase retrieved facts. The generator receives a constrained context containing the selected dishes, prices, and relevant attributes. This follows the grounding principle of retrieval-augmented generation [7] and reduces the risk of hallucinating menu information.

and mobile food-ordering environments.

NLP AND RECOMMENDATION METHODS

A. BASELINE NLP PIPELINE

A baseline implementation can use TF-IDF vectors with a linear classifier for intent recognition and dictionary-based entity matching for attributes. TF-IDF provides an interpretable reference point, but it is limited when two phrases have similar meaning but little lexical overlap.

For example, “I need a meat-free dish” and “show vegetarian food” should map to a similar intent even though their surface forms differ. Word embeddings address part of this problem by representing words in a continuous vector space [9]. Transformer encoders further improve contextual representation by using attention mechanisms [10].

B. TRANSFORMER-BASED INTENT CLASSIFICATION

BERT-style contextual representations can be fine-tuned for food-domain intent classification [11]. Given an utterance q , the encoder produces contextual token representations and a pooled representation h . A softmax layer estimates $P(I|q)$. The predicted intent is selected using $\text{argmax}_I P(I|q)$. Class weighting or focal loss can be considered when rare intents have insufficient training examples.

C. SEMANTIC RETRIEVAL

Sentence-level embeddings are used to compare user queries with menu descriptions. Sentence-BERT demonstrates that siamese and triplet architectures can produce efficient sentence embeddings for semantic similarity [12]. For a query embedding e_q and menu embedding e_m , cosine similarity is computed as:

$$\text{sim}(q,m) = (e_q \cdot e_m) / (||e_q|| ||e_m||).$$

The top candidates are retrieved from a vector index. To avoid recommending an item that is semantically similar but violates a hard requirement, retrieval is followed by constraint validation.

D. RANKING FUNCTION

A practical ranking score can be defined as:

$$\text{Score}(m) = \alpha S_{\text{sem}}(m) + \beta S_{\text{attr}}(m) + \gamma S_{\text{pref}}(m) + \delta S_{\text{pop}}(m) - \lambda P(m).$$



Here, Ssem is semantic similarity, Sattr is attribute compatibility, Spref represents historical preferences, Spop is optional popularity, and P penalizes violations or undesirable properties. Hard constraints can be represented by setting $\text{Score}(m) = -\infty$ when an item fails a mandatory requirement.

E. CONVERSATIONAL PERSONALIZATION

Personalization should be explicit and controllable. The system can use preferences stated in the current session and, where consent is available, longer-term preferences. A preference vector may contain cuisine, dietary pattern, spice level, preferred price range, and frequently selected categories. The system should allow users to override or clear stored preferences.

F. HANDLING AMBIGUITY AND NEGATION

Food language is highly ambiguous. “Light food” may refer to low calories, small portions, or mild taste. Similarly, “without cheese” is a negative ingredient constraint. The chatbot should not guess when ambiguity materially changes the recommendation. Instead, it can ask a clarification question or provide clearly labeled alternatives.

DATASET AND EXPERIMENTAL DESIGN

A research prototype should be evaluated using a combination of menu records and annotated user queries. The menu dataset can be constructed from publicly available or institutionally authorized restaurant menus, subject to licensing and privacy constraints. Each menu item should be normalized into a common schema. User utterances should be annotated for intent and attribute slots.

A representative dataset may contain 8–12 intent categories and 10–15 attribute types. The data should be divided into training, validation, and test sets using a stratified split. To avoid leakage, near-duplicate utterances should remain in the same partition.

The evaluation should compare at least three configurations: (1) keyword-based search, (2) semantic retrieval without dialogue state, and (3) the proposed hybrid conversational system. The comparison allows the contribution of semantic understanding and multi-turn context to be quantified.

Conversational performance can additionally be measured using task completion rate, number of turns to successful recommendation, average response latency, and user satisfaction. Human evaluation should assess relevance, clarity, personalization, and factual grounding on a Likert scale.

IMPLEMENTATION DETAILS

A reference implementation can use a web frontend with a chat component, a backend API, a relational menu database, and a vector retrieval service. Python-based NLP frameworks can be used for preprocessing and transformer inference, while a REST API connects the chatbot to the food website. The architecture is model-agnostic and can use either a locally hosted transformer or a commercial language model, provided that the deployment satisfies privacy and data-governance requirements.

A. MENU INGESTION

Menu data is first validated for missing values, duplicate identifiers, inconsistent prices, and unavailable items. Text fields are normalized, and controlled vocabularies are used for cuisine and dietary categories. Each item is converted into a searchable document containing both structured fields and a natural-language representation.

B. QUERY UNDERSTANDING

The query-understanding service produces an intent label and structured attributes. A sample output is: `intent=recommendation; diet=vegetarian; cuisine=Indian; taste=spicy; price_max=300`. Confidence



values should be retained so that low-confidence interpretations can trigger clarification rather than an incorrect recommendation.

C. RETRIEVAL AND FILTERING

The system first evaluates hard constraints against structured menu fields. The remaining records are embedded and retrieved using semantic similarity. A hybrid index can combine BM25-style lexical retrieval with dense vector retrieval. The two candidate sets are merged and re-ranked. This approach is useful because food names can contain exact domain terms while user requests can also be highly paraphrastic.

D. EXPLAINABLE RECOMMENDATION

Each recommendation should include a concise reason, such as “Recommended because it is vegetarian, spicy, and within your ₹300 budget.” Explanation fields are generated from the same structured attributes used by the ranking model. This avoids post-hoc explanations that are inconsistent with the actual recommendation logic.

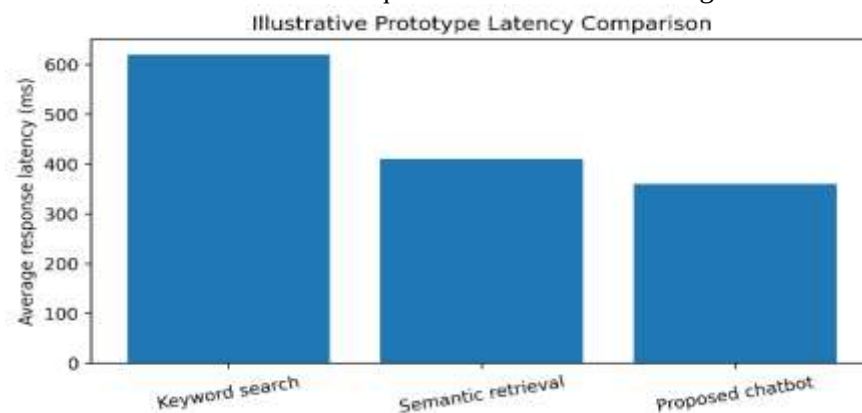
E. SAFETY AND FACTUALITY

The chatbot should not invent allergen information or make unsupported medical claims. If allergen information is unavailable, the system should state that the website does not provide verified information. Nutrition-oriented suggestions should be framed as general information rather than medical advice. Menu availability and price should be retrieved from the current website data.

EXPECTED RESULTS AND DISCUSSION

The proposed architecture is expected to outperform keyword-only search on queries containing paraphrases, multiple constraints, and conversational references. Transformer-based intent classification should improve contextual interpretation, while semantic retrieval should improve recall for conceptually related menu items. The hybrid ranking stage is expected to reduce irrelevant results by enforcing explicit dietary and price constraints.

However, performance depends strongly on menu-data quality. If dietary labels, ingredient descriptions, or availability fields are incomplete, even a strong NLP model cannot guarantee correct recommendations. Similarly, rare dishes and restaurant-specific terminology may reduce entity extraction accuracy. Continuous evaluation is therefore required when the menu changes.





The literature supports the value of combining textual information and personalized attributes for restaurant recommendation [1], [3], while recent LLM-based work highlights the potential of conversational personalization [4]. At the same time, retrieval-grounded architectures are important because generative models can otherwise produce unsupported information [7]. The proposed system combines these directions in a website-oriented architecture that keeps menu records as the factual source.

LIMITATIONS

The proposed framework has several limitations. First, high-quality intent and slot annotations require domain-specific labeling effort. Second, semantic retrieval may be computationally expensive for very large menus unless embeddings are indexed efficiently. Third, user preferences can be sparse for new users. Fourth, multilingual and code-mixed food queries may require additional training data. Finally, evaluation based only on offline metrics cannot fully represent real user experience.

FUTURE WORK

Future work can investigate multilingual and code-mixed conversational food search, including Hindi-English queries commonly used in Indian food-ordering contexts. A multimodal extension could incorporate food images, while a knowledge graph could represent relationships among dishes, ingredients, cuisines, allergens, and dietary constraints. Recent work on ontology-enhanced RAG suggests another promising direction for reducing hallucination and improving structured reasoning in food advisory systems [13]. Further research can also examine online learning from explicit feedback, contextual recommendations based on time and occasion, group food recommendation, and sustainable food choices. User studies should measure whether conversational interaction actually reduces search effort and increases satisfaction compared with conventional website filtering.

CONCLUSION

This paper presented an NLP-based chatbot architecture for food websites that integrates intent detection, attribute extraction, dialogue-state tracking, semantic retrieval, constraint filtering, recommendation ranking, and grounded response generation. The central design principle is to combine conversational flexibility with structured menu facts. Unlike a purely generative chatbot, the proposed approach keeps the restaurant menu as the source of truth and uses NLP models to interpret user needs. The evaluation framework provides measurable criteria for assessing both language understanding and recommendation quality. The architecture offers a practical foundation for intelligent food discovery and ordering systems and can be extended with multilingual NLP, multimodal retrieval, personalization, and ontology-enhanced RAG.

REFERENCES

- [1] Q. Li, D. Jang, D. Kim, and J. Kim, "Restaurant recommendation model using textual information to estimate consumer preference: Evidence from an online restaurant platform," *Journal of Hospitality and Tourism Technology*, vol. 14, no. 5, pp. 857–877, 2023, doi: 10.1108/JHTT-01-2023-0019.
- [2] A. A. A. Alian, et al., "Restaurant recommender system based on sentiment analysis," *Machine Learning with Applications*, vol. 6, Art. no. 100114, 2021, doi: 10.1016/j.mlwa.2021.100114.
- [3] Q. Li, D. Jang, D. Kim, and J. Kim, "Deep learning mechanism and big data in hospitality and tourism: Developing personalized restaurant recommendation model to customer decision-making,"



-
- International Journal of Hospitality Management, vol. 121, Art. no. 103803, 2024, doi: 10.1016/j.ijhm.2024.103803.
- [4] Z. Yang, E. Khatibi, N. Nagesh, M. Abbasian, I. Azimi, R. Jain, and A. M. Rahmani, “ChatDiet: Empowering personalized nutrition-oriented food recommender chatbots through an LLM-augmented framework,” *Smart Health*, 2024, Art. no. 100465, doi: 10.1016/j.smhl.2024.100465.
- [5] J. N. Bondevik, K. E. Bennin, Ö. Babur, and C. Ersch, “A systematic review on food recommender systems,” *Expert Systems with Applications*, vol. 238, Art. no. 122166, 2024, doi: 10.1016/j.eswa.2023.122166.
- [6] C. Trattner and D. Elswiler, “Food recommender systems: Important contributions, challenges and future research directions,” *arXiv:1711.02760*, 2017.
- [7] P. Lewis et al., “Retrieval-augmented generation for knowledge-intensive NLP tasks,” in *Proc. Advances in Neural Information Processing Systems (NeurIPS)*, vol. 33, 2020, pp. 9459–9474.
- [8] A. M. Al-Emran, et al., “Customer intentions toward the adoption of WhatsApp chatbots for restaurant recommendations,” *Journal of Hospitality and Tourism Technology*, vol. 16, no. 4, pp. 784–816, 2025, doi: 10.1108/JHTT-01-2024-0024.
- [9] T. Mikolov, K. Chen, G. Corrado, and J. Dean, “Efficient estimation of word representations in vector space,” *arXiv:1301.3781*, 2013.
- [10] A. Vaswani et al., “Attention is all you need,” in *Proc. Advances in Neural Information Processing Systems (NeurIPS)*, vol. 30, 2017.
- [11] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova, “BERT: Pre-training of deep bidirectional transformers for language understanding,” in *Proc. NAACL-HLT*, 2019, pp. 4171–4186.
- [12] N. Reimers and I. Gurevych, “Sentence-BERT: Sentence embeddings using Siamese BERT-networks,” in *Proc. EMNLP-IJCNLP*, 2019, pp. 3982–3992.
- [13] “Ontology-enhanced RAG for a personalised and sustainable food advisory system,” *Data & Knowledge Engineering*, vol. 164, Art. no. 102579, 2026, doi: 10.1016/j.datak.2026.102579.
- [14] M. A. H. M. Al-Smadi, et al., “A survey of conversational recommender systems,” *ACM Computing Surveys*, 2024.
- [15] F. Ricci, L. Rokach, and B. Shapira, Eds., *Recommender Systems Handbook*, 3rd ed. New York, NY, USA: Springer, 2022.
- [16] C. C. Aggarwal, *Recommender Systems: The Textbook*. Cham, Switzerland: Springer, 2016.
- [17] D. Jurafsky and J. H. Martin, *Speech and Language Processing*, 3rd ed., draft, Stanford University, 2024.
- [18] Y. Koren, R. Bell, and C. Volinsky, “Matrix factorization techniques for recommender systems,” *Computer*, vol. 42, no. 8, pp. 30–37, 2009, doi: 10.1109/MC.2009.263.
- [19] F. Pedregosa et al., “Scikit-learn: Machine learning in Python,” *Journal of Machine Learning Research*, vol. 12, pp. 2825–2830, 2011.
- [20] T. Chen and C. Guestrin, “XGBoost: A scalable tree boosting system,” in *Proc. 22nd ACM SIGKDD Int. Conf. Knowledge Discovery and Data Mining*, 2016, pp. 785–794.
- [21] K. Cho et al., “Learning phrase representations using RNN encoder-decoder for statistical machine translation,” in *Proc. EMNLP*, 2014, pp. 1724–1734.
- [22] M. Lewis et al., “BART: Denoising sequence-to-sequence pre-training for natural language generation, translation, and comprehension,” in *Proc. ACL*, 2020, pp. 7871–7880.
-