Knowledge Graph-Enhanced LLM for Food Recommendation through Question Answering

Fnu Mohbat, Mohammed J. Zaki

[1]Rensselaer Polytechnic Institute
110 8th St, Troy
NY 12180 USA
mohbaf@rpi.edu, zaki@cs.rpi.edu

Abstract

The rich online sources of food data has resulted in creation of structured food knowledge graphs (FoodKGs) which have been applied to various food computing tasks including food recommendation. The recent advancement in large language models (LLMs) have also led to their applications in recommendation systems. Despite several food recommendation systems utilizing KGs, there is limited research on employing FoodKGs to enhance LLMs for food recommendation. We propose food recommendation as question answer over a large food knowledge graph by leveraging the power of LLMs. Given a natural language question, the system extract entities and retrieves subgraphs from the KG which are fed into the LLM as context. The LLM then selects the recipes that satisfy all the constraints in the question. In our approach we fine-tune the LLM model to take in a question and the relevant subgraph from a FoodKG, to recommends relevant recipes. We also develop a benchmark dataset by curating recipe related questions, combined with constraints and personal preferences. We show via extensive comparison that our proposed LLM plus KG model significantly outperforms the other state-of-the-art (SOTA) LLM models for food recommendation.

Introduction

The importance of food for well being has created a need for employing machine learning for understanding food for a healthy lifestyle. Several online recipe sharing websites created rich resources of food data, attracting researchers to devise food computing methods from classification, retrieval, recipe generation to recommendation. Recommendation systems have been designed to improve user experience, however food recommendations systems are critical to human health and are more complex and multi-faceted. An effective food recommendation system should consider personal preferences, dietary constraints and health guidelines. In recent years, several food ontologies and knowledge graphs were generated to better organize the food data (Dooley et al. 2018; Haussmann et al. 2019; Razzaq et al. 2023). Subsequently, several food recommendation methods leveraged KGs for personalized food recommendation (Chen et al. 2021; Shirai et al. 2021; Ling et al. 2022; Li, Zaki, and Chen 2023; Kobayashi et al. 2024). Most of these approaches learn an embedding spaces for food recipes and then consider the most similar recipes in the embedding space as recommended recipes. One such method (Chen et al. 2021) proposed food recommendation as a question answering problem where they map questions and relevant recipes into a common embedding space.

The rapid advancements in large language models (LLMs) have driven efforts to optimize them domains like finance, medicine and food (Wu et al. 2023; Moor et al. 2023; Mohbat and Zaki 2024; Chhikara et al. 2024). However, these LLMs prone to hallucinations and outdated information (Xu, Jain, and Kankanhalli 2024). Retrieval-augmented generation (RAG) addresses these issues by providing contextual information to LLMs, leading to more accurate responses. The relevant context needs to be extracted from large data sources such as from text documents by identifying similar chunks of text or from knowledge graphs (KGs) by retrieving relevant subgraphs (Wang et al. 2021; Banerjee et al. 2023; He et al. 2024). The context retrieved as chunks of text documents may be more noisy whereas subgraphs may contains more precise information while also providing the relations among entities. Therefore, recently attention shifted toward utilizing KGs with LLMs to enhance their performance (He et al. 2024; Rangel et al. 2024) specifically for improving question answering over KGs, referred to as KGQA.

KGQA is a challenging task that requires understanding the natural language query, mapping it to the KG schema, and generating a graph query that can retrieve the correct answer from the KG (Shah and Tian 2024). LLM-based KGQA is a crucial research area offering innovation and improvements in question-answering systems (Shah and Tian 2024). KGQA models either retrieve relevant subgraphs and use reasoning to extract entities as answers (Wang et al. 2021), or use semantic parsing and thereby transform questions into SQL or SPARQL queries to get answers from the KG (Banerjee et al. 2023). Some efforts explored power of LLMs in zero or few shot setting (Taffa and Usbeck 2023; Avila et al. 2024) to generate SPARQL queries based on semantically parsed entities from a user question. However, this method needs access to the training data or use proprietary LLM such as ChatGPT. Despite several efforts using external knowledge (documents or KGs) with LLMs in sev-

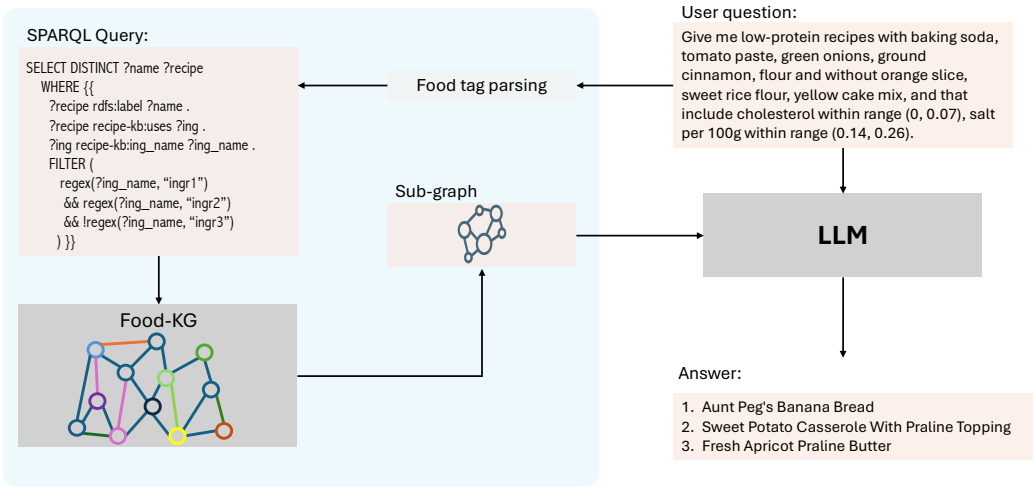

Figure 1: System overview: Given a natural language question (with all constraints), the system parses entities and generates a SPARQL query to retrieve a subgraph from the KG. The question and this subgraph as context, is then provided as input to the LLM, which generates a list of recipe names that satisfy all the constraints in the question.

eral domains, there is a lack of work on food recommendation using KGs and LLMs together while considering the health constraints and user preferences.

This paper proposes to food recommendation as constrained question answer over the open-source FoodKG knowledge graph (Haussmann et al. 2019). Our proposed method improves food recommendation by leveraging FoodKG as knowledge source and large language model for language processing. Given a user question, the system extract the entities to generate a SPARQL query to retrieve relevant subgraphs from the KG, which are input to LLMs as context along with the question. The overview of our proposed method is shown in Figure 1. To evaluate our model, first, we curated a benchmark dataset using questions from templates and combined them with simulated constraints and personal preferences. Next, we fine-tuned a large language model that takes a question and a subgraph from FoodKG as context, and recommends recipes that satisfy the constraints in the questions. We show via extensive experimental comparison that our proposed model outperforms the SOTA LLM models for food recommendation, showcasing the power of combining LLMs withe power of KGs. The code and benchmark dataset will be made publicly available.

Related Work

Food Recommendation

Most existing food recommendation methods suggest recipes based on various factors, such as recipe content, user behavior, dietary constraints, or a combination of these aspects (Chen et al. 2020; He and Zhu 2021; Li and Zaki 2022; Chen et al. 2023; Kobayashi et al. 2024). Initial food recommendation systems formulated recommendation as a classification or matrix completion problem (Chen et al. 2018; Salvador et al. 2017; He and Zhu 2021; Min et al. 2023). Several methods employed embedding based methods (Wahed et al. 2024; Li and Zaki 2022) by mapping recipe components such as title, ingredients and images into a common embedding space. During inference, these approaches use similarity search to retrieve recipes similar to the user's profile.

Later focus shifted towards using food knowledge graph (Li and Zaki 2022; Chen et al. 2023) or LLMs (Geng et al. 2022; Kirk, van Eijnatten, and Camps 2023; Yang et al. 2024; Rostami, Jain, and Rahmani 2024) for food recommendations systems. By creating a benchmark QA dataset, Chen et al. (2021) formulated food recommendation as knowledge base question answering through information retrieval based method where they mapped natural language questions and the possible answers into common embedding spaces using LSTM. To utilize user-recipe interactions in KGs for food recommendation, (Li and Zaki 2022; Gao et al. 2022) employed graph neural network based approaches. However, recent methods employ LLMs for food recommendations. Considering the popularity of ChatGPT to answering the questions, (Kirk, van Eijnatten, and Camps 2023) investigated it for nutrition questions. Whereas (Geng et al. 2022; Rostami, Jain, and Rahmani 2024) harness LLMs as a Language Processing engine in food recommendation system. Despite considerable efforts to leverage Knowledge Graphs and Large Language Models for developing food recommendation systems, there remains limited research on integrating Food KGs to augment LLMs for more personalized food recommendation. Specifically, individual preferences, health considerations, and nutritional constraints within a unified framework has not been extensively explored.

Question Answering Over Knowledge Graph (KGQA)

Question Answering Over Knowledge Graph (KGQA) refers to retrieving the knowledge from a KG to answer the questions. Initial studies parsed entities from a natu-

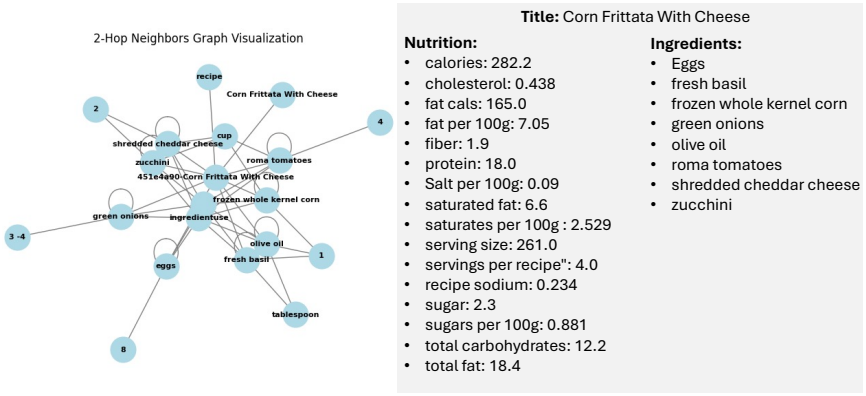

Figure 2: FoodKG Recipe sample: left panel shows a 2-hop KG subgraph for the recipe node shown on the right.

ral language question and generated SPARQL queries from templates to retrieve the answer (Shirai et al. 2021; Haussmann et al. 2019; Rangel et al. 2024). Another line of research focuses on leveraging graph neural networks (GNNs) or LSTM-based embeddings to formulate information retrieval based approaches (Chen et al. 2021; Gao et al. 2022; Forouzandeh et al. 2024; He et al. 2024). The recent advancements in integration of KG's with LLMs attracted their use for improving KGQA (Xu et al. 2024; Eppalapally et al. 2024; Hou and Zhang 2024; Ma et al. 2024). Several method explored the integration of KG to enhance LLM for reasoning (Luo et al. 2023; Sun et al. 2023), improving chat-bots for better customer service (Xu et al. 2024), product recommendations (Eppalapally et al. 2024), and food related tasks (Achiam et al. 2023; Min et al. 2022; Hou and Zhang 2024; Ma et al. 2024). FoodGPT (Qi et al. 2023) studies incremental pre-training to boost LLMs performance with KG in food domain. While numerous KGs and LLMs have been applied to food-related tasks, the full potential of combining KGs and LLMs in food science remains underexplored (Min et al. 2022; Ma et al. 2024). This represents a critical avenue for further research.

Food Recommendation as KGQA

We aim to develop a personalized food recommendation system by leveraging the food knowledge graph (FoodKG) (Haussmann et al. 2019) as contextual information, combined with a large language model serving as the generative recommendation engine. We hypothesize that enhancing LLMs with FoodKG for food recommendations will achieve superior performance compared to embedding-based retrieval methods. Therefore, we propose utilizing recipe subgraphs from FoodKG as context and LLMs to recommend recipes through a generative approach (see Figure 2 that shows an example KG subgraph, and the recipe text for the achor recipe node). We train the LLM to better understand the context in regards to the given question. During inference, given a natural language question, we use semantic parsing to extract the entities and then generate SPARQL queries to extract the relevant recipe subgraphs from the FoodKG. These subgraphs are input to LLM as context along with the question. The LLM selects the recipes from the context that meet the constraints in the question.

To this end, we leverage FoodKG to model a recipe recommendation system as a constraint question answering problem. We start with base (template) questions then incorporate ingredient constraints, i.e., the recipe should contain the given list of ingredients. To understand the negative constraints, we also add constraints that the recipe must not contain a given list of ingredients. For example *Recipe should contain Spinach and Butter but must not have Nuts*. Similarly, we also add constraint on nutritional values such as *recipe with more than 2 gram proteins and less than 500 calories*. Finally, we combine the base question with constraints to generate final constraint question. The resultant recipes after applying all filters on the KG are considered as ground truth recipes that meet the constraint in the question. An example final question could be: *Question: What very-low-carbs recipes use egg yolks, nutmeg, lemon wedges, blue cheese, orange juice and avoid ground red pepper, green onions, dried sweet basil leaves, and have cholesterol no more than 0.21?* More examples of base questions, constraints and final detailed questions can be found in Table 1. Subsequently, we generate a benchmark question answer dataset for food recommendations.

Dataset Generation

We curated a large benchmark dataset using FoodKG by leveraging the different components of recipes for constrained question answer generation. Each sample in our dataset contains a user query, ingredient preferences, nutritional constraints and ground truth answers. We combine the user query, ingredient preferences and nutritional constraints to generate input questions for the model. The ground truth answers are those recipes that satisfy all the preferences and constraints in the input question. To generate each sample, we used recipe attributes such as tags, ingredients, nutritional values. The examples of tags could be *American*, *Healthy*, *Easy to cook*, etc. Nutritional values include contents of calories, fat, protein, sugar, and so on.

FoodKG contains 1 Million recipes with ingredients, nutritional information and tags, organized into 65 Million

Base question: Give me $\{tag\}$ recipes with $\{ingredients\}$ and without $\{not\_have\_ingredients\}$	**Base question:** What are the $\{tag\}$ dishes that contain $\{ingredients\}$ but do not contain $\{not\_have\_ingredients\}$
Template constraints: have $\{nutrition\}$ no more than $\{limit\}$, $\{nutrition\}$ within range $\{limit\}$	**Template constraints:** have $\{nutrition\}$ at least $\{limit\}$, and $\{nutrition\}$ less than $\{limit\}$
Personal preferences: tag: low-protein	**Personal preferences: tag:** vegetarian
Likes: baking soda, tomato paste, green onions, ground cinnamon, flour	**Likes:** margarine, frozen peas, shredded cheddar cheese, baking soda, vinegar
Dislikes: orange slice, sweet rice flour, yellow cake mix	**Dislikes:** cracked wheat, chili pepper, fresh pepper
Nutrition constraints: cholesterol no more than 0.07, salt per 100g (0.14, 0.26)	**Nutrition constraints:** fiber at least 4.24, saturated fat less than 6.49
Question: Give me low-protein recipes with baking soda, tomato paste, green onions, ground cinnamon, flour and without orange slice, sweet rice flour, yellow cake mix, and have cholesterol no more than 0.07, salt per 100g within range (0.14, 0.26).	**Question:** What are the top vegetarian recipes containing margarine, frozen peas, shredded cheddar cheese, baking soda, vinegar and excluding cracked wheat, chili pepper, fresh pepper, and meeting the fiber at least 4.24, saturated fat less than 6.49 condition?
Answer: Aunt Pegś Banana Bread, Sweet Potato Casserole With Praline Topping, Fresh Apricot Praline Butter.	**Answer:** B. B. Kingś German Chocolate Cake, Apple Bread, Momś Raisin Rock Cookies

Table 1: Examples of constraints, corresponding questions, and relevant recipe names as ground truth answers. Ingredient preferences specify whether certain ingredients should or should not be included in the recipes. Nutritional constraints are numerical conditions applied to nutrient values, defined by limits such as less than, greater than, or within a specified range for different nutrients.

triplets. Let t_j be a tag in FoodKG, and $R(t_j)$ the set of tagged recipes with tag t_j, which form the potential context for an LLM. Now, let $I(t_j)$ be the set of ingredients in $R(t_j)$, then we define $I^+(t_j) \subset I(t_j)$ as the set of ingredients that the user likes to have and $I^-(t_j) \subset I(t_j)$ as the set of ingredients that user wants to avoid. Recipes that satisfy all the constraints are considered as ground truth answers or positive set of recipes $R^+(t_j)$ and the remaining $R^-(t_j) = R(t_j) - R^+(t_j)$ are considered as negative set of recipes.

Generating personal preferences

To personalize the recommendation of recipes, individualized information regarding a person's likes, dislikes, and other personal choices are important. This personalized data can be leveraged to tailor food recommendations to align more closely with the individual's preferences. In this study, we simulate the personalization by incorporating both ingredient preferences and nutritional constraints that one may wish to prioritize. This allows us to generate personalized food recommendations that consider both taste preferences and dietary needs.

Ingredient Preferences: We simulate a person's ingredient preferences by randomly sampling two mutually exclusive sets of ingredients $I^+(t_j)$ and $I^-(t_j)$ from the list of ingredients $I(t_j)$ such that $I^+(t_j) \cap I^-(t_j) = \emptyset$. One set $I^+(t_j)$ is treated as person's preferred ingredients, while the other set $I^-(t_j)$ is considered as disliked ingredients that one may wish to avoid in the recipes. This approach allows us to model a person's likes and dislikes of ingredients.

Nutritional constraints: To account for user preferences related to nutrition, such as calorie intake, salt content, or other dietary factors, we introduce nutritional constraints by sampling nutrients and their limits. We also select one of the three filters: 'less than', 'greater than', or 'fall within a defined range'. Moreover, during data generation, we want few recipes as valid answers, therefore, we design nutritional constraints such that there is a subset of recipes that meet the constraints.

Let x_i be the nutrient in a recipe. Then, let $\mu(R(t_j), x_i)$ and $\sigma(R(t_j), x_i)$ denote the mean and standard deviation of nutrient x_i in the set of tagged recipes $R(t_j)$, respectively. To generate the nutritional constraints, first we randomly select one of the three filters: 'less than', 'greater than', or 'fall within a defined range'. Then, we define a threshold for the nutrient x_i^{thresh} by sampling a random number in range of $\mu(R(t_j), x_i) \pm 2\sigma(R(t_j), x_i)$ and apply the selected filters (i.e., less than, greater than, or within two standard deviations). Finally, all selected nutritional constraints along with filters are combined with the base question. This approach enhances the diversity of the questions, incorporating both conditional logic and negations, which are crucial for generating more complex and realistic queries.

KGQA benchmark dataset

The simulated preferences and constraints are combined with a user query to create a detailed question, which is then fed into the LLM. The examples of base question, template constraints, simulated nutritional limits and final detailed questions are given in Table 1. Similar to (Chen et al. 2021), we generated the base template queries with placeholders for an individual's ingredient preferences and dietary restrictions. These placeholders are replaced with their respective values. The simulation of the preferences and constraints ensures that the generated questions are realistic and can yield a few recipes as ground answer to the question, avoiding impossible requests. The recipes that meet all the conditions in the detailed final question are considered as recommended recipes.

We used FoodKG as our knowledge base, which contains

over 1 million recipes labeled with 490 unique tags with each recipe potentially having multiple tags. For this dataset, questions are generated using 15 health-related categories such as *dairy-free*, *low-fat*, and *high-fiber*, leaving the remaining tags for potential future work. A full list of the tags used for data generation can be found in Appendix . Notably, the recipes associated with these 15 health-related tags also include a total of 472 tags, covering the majority of the tags in the FoodKG.

Measure	PFoodReq (Chen et al. 2021)		KGQA (our)	
	Train set	Test set	Train set	Test set
Number of questions	4613	2305	62320	7790
Ground truth answer (min)	1	1	1	1
Ground truth answer (max)	296	178	1776	954
Ground truth answer (avg)	2.94	2.84	10.67	9.77
Tagged recipes (min)	2	2	7	7
Tagged recipes (max)	2486	2485	4445	4445
Tagged recipes (avg)	408.4	377.99	3167	3163

Table 2: Dataset Statistics: This table shows the number of questions, along with the corresponding number of recipes in the set of tagged recipes $R(t_j)$ as overall context and ground truth answer $R^+(t_j)$.

Overall, our dataset contains $77,900$ question-answer pairs, split into 80%, 10%, and 10% for training, validation, and testing, respectively. The test set includes $7,790$ questions where the relevant context for each question could encompass all tagged recipes. Table 2 shows that the number of recipes in ground truth $R^+(t_j)$ vary from 1 to 954 whereas recipes relevant to entities $R(t_j)$ in questions (without constraints) range from 7 to as many as 4445, showing the complexity and variety of questions.

Food recommendation system

We formulate personalized food recommendation as LLM based constrained question answering over the food knowledge graph. Given a natural language user query, our system extracts entities from the query and retrieves the relevant subgraph from KG. The LLM is trained to identify the recipes from the KG subgraph that satisfy the user query. Overall, over system consists of three modules: retrieval of subgraphs from KG, finetuning the model on KGQA benchmark and finally model inference the over KG for generating the response as recommended recipes.

Subgraph Retrieval

Given natural language question, the first step is to parse entities such as tags and then generate SPARQL query to retrieve subgraphs from KG. For now, we assume that each question is based on only one tag which also belong to known set of tags. Each recipe graph contains the name of the dish, list of ingredients and nutritional information. The recipe subgraphs are serialized into a text sequence and given as context to the LLM. Note that there could be more than 1,000 tagged recipes as subgraphs returned by SPARQL query whereas LLMs accept only a limited sequence length. Therefore, it is not feasible to pass the entire set of tagged recipes $R(t_i)$ as a context to the LLM in a single call. Instead, we provide a subset of recipes as context during each forward pass of the LLM.

Model Optimization

We want the model to be able to select the recipes that meet the constraints in the question, so during training we sample K recipe subgraphs from $R(t_j)$ as the context set C_j for a single forward pass. Note that C_j may contain recipes from $R^+(t_j)$ and $R^-(t_j)$ sets, but dataset statistics in Table 2 show that $|R^-(t_j)| \gg |R^+(t_j)|$, therefore in practice we sample at most $K/2$ positive recipes from $R^+(t_j)$ and the remaining from $R^-(t_j)$ and combine both to get context set C_j. The recipes sampled from $R^+(t_j)$ are considered as ground truth answer Y. This allows the model to learn to select from vast distribution of context (positive or negative recipes). The model was trained on standard cross-entropy loss which is defined as.

$$L_{CE} = CE(p(Y), p(\tilde{Y})) \tag{1}$$

Where, $p(Y)$ is probability of ground truth recipes tokens as one hot vector and $p(\tilde{Y})$ is the predicted probability of the recipe tokens generated by the model.

Inference over KG

As discussed above, all tagged recipes $R(t_j)$ could potentially serve as the overall context for a question generated for a tag t_j. Dataset statistics in Table 2 indicate that the maximum number of tagged recipes for a tag t_j can reach up to 4,445, resulting in a total token count significantly exceeding the model's sequence length, which may also lead to GPU memory overflow. Therefore, similar to the training, we can pass limited number of recipe subgraphs as context during each call to the LLM. We iterate over all the subgraphs by passing a subset of subgraphs as context to LLM along with question and combine the answers from multiple calls to LLM to generate the final response. This approach allows us to perform inference and evaluate the model on variable number of recipe subgraphs.

Experiments

We used a generative model where we iterate over the subset of subgraphs as context to the LLM and the LLM selects recipes that satisfy the constrains in questions. Finally, we combine answers from multiple calls to the LLM as final answer. Unlike retrieval methods where order of the retrieved recipe matters, our setup is inherently order agnostic. We report performance on standard retrieval metrics such as accuracy, mean average precision (mAP), overall precision, recall and F1 formally defined in Appendix . All the experiments were conducted on four NVIDIA RTX A6000 GPUs. For baseline comparison, we select several open source LLMs with 10B parameters based on their performance reported on Huggingface open LLM leader board (https://huggingface.co/spaces/open-llm-leaderboard/open_llm_leaderboard; Oct 1, 2024). These include internLM2 (Cai et al. 2024), Mistral (Jiang et al. 2023), Llama series (Llama-2 (Touvron et al. 2023), Llama3.1 (Meta 2024)) and Phi series (Phi-2 (Javaheripi et al. 2023), Phi-3-mini (Abdin et al. 2024)).

Model	Model Size	Sequence Length	Acc	mAP	P	R	F1
internLM2	7B	4k	5.55	0.06	0.024	0.055	0.034
Mistral*	7B	4k	55.82	0.214	0.536	0.558	0.547
Phi-2	2.7B	1K	37.77	0.271	0.084	0.378	0.137
Llama-2	7B	4k	62.72	0.557	0.825	0.627	0.713
Llama-3.1	8B	4k	40.58	0.146	0.28	0.406	0.332
Phi-3-mini-4K	3.8B	4K	4.4	0.047	0.192	0.044	0.071
Phi-3-mini-128K	3.8B	16K	27.81	0.275	0.778	0.278	0.41
Our	3.8B	16K	**96.83**	**0.96**	**0.978**	**0.969**	**0.973**

Table 3: KGQA test set: Our model (Phi-3-mini-128K fine-tuned on 16K sequence length) vs. pre-trained LLMs.

Results on our KGQA benchmark

Open source LLMs: Table 3 shows the performance on recent state-of-the-art LLMs. Despite internLM2 and Llama-3.1 claim about out performance on several benchmarks, both failed to understand the complex numerical constraints in KGQA benchmark questions. On the other hand, Llama-2 seems to perform better. The capability of handing larger sequence length by Phi-3-mini-128K helps it perform better than Phi-3-mini-4K, which also aligns with their performance on huggingface open LLM leaderboard.

We selected Phi-3-mini-128K for finetuning on our KGQA benchmark dataset due to its compact size and relatively strong performance. Given the constraints of limited GPU memory, we reduced the sequence length to 16K for training. During testing, the model performed equally well when 4K or 16K sequence length was selected as evident from Table 4. The primary distinction was that a 4K sequence length accommodated fewer subgraphs in the context compared to a 16K sequence length. The evaluation of our model on KGQA test set shown in Table 3 clearly improves 60 points on accuracy compared to Phi-3-mini-128K. Compared to Mistral and Llama-2, our model demonstrates higher scores on all metrics. Specifically, our model is almost 26 points better than Llama-2 on F1 score. Overall, our model significantly outperforms the other models.

Sequence Length	Acc	mAP	P	R	F1	TP	FP	FN
4K	87.0	0.997	0.994	0.875	0.931	118263	761	16898
16K	87.0	0.997	0.994	0.875	0.931	118310	742	16874

Table 4: Results on the KGQA test set when using the full context graph. The LLM is iteratively provided with subset of recipe subgraphs. LLM select the relevant recipes from given context during each iteration and the final answer is formed by combining all the selected recipes at the end of the process.

Impact of recipe types: Recipes in FoodKG are tagged with one or more tags based on the nature of the dish. We evaluated all models on questions related to various types of tagged recipes and report F1 scores in Figure 3 and Table 10 (see Appendix) for our model versus baseline LLMs. Compared to other models, our model consistently showed higher F1 scores than others. For *dairy-free* recipes, Llama-2 (overall second best) and our model could recommend few correct recipes as evident by the F1 scores. Table 5 shows performance of our model on different types of recipes based on associated tags. We can see that for most of the tags, our model shows similar scores except *dairy-free* and *gluten-free*, suggesting that the model has generalized across various types of dishes. Relatively lower accuracy and F1 scores for *dairy-free* recipes is due to fewer samples in training data.

Tag	Acc	mAP	P	R	F1
lactose	91.64	0.898	0.955	0.916	0.935
vegan	97.49	0.964	0.988	0.975	0.981
vegetarian	97.58	0.966	0.987	0.976	0.981
dairy-free	66.67	0.667	0.667	0.667	0.667
gluten-free	77.4	0.922	0.992	0.779	0.873
nut-free	1.0	1.0	0.909	1.0	0.952
egg-free	95.13	0.939	0.982	0.951	0.966
low-carb	96.52	0.952	0.983	0.965	0.974
low-fat	96.55	0.964	0.956	0.966	0.961
low-protein	98.08	0.981	0.988	0.981	0.984
low-sodium	98.17	0.982	0.978	0.982	0.98
low-cholesterol	95.07	0.924	0.98	0.951	0.965
high-protein	96.72	0.992	0.944	0.967	0.956
high-calcium	95.28	0.937	0.981	0.953	0.967
high-fiber	93.33	0.938	1.0	0.933	0.966

Table 5: Results on KGQA test set per tag

context size	Acc	mAP	P	R	F1	TP	FP	FN
6	93.97	0.888	0.881	0.94	0.91	19637	2641	1259
10	75.53	0.74	0.903	0.755	0.823	23323	2496	7558
20	70.22	0.717	0.91	0.702	0.793	30483	3010	12926
40	68.6	0.718	0.88	0.686	0.771	34949	4752	15996
100	65.38	0.714	0.898	0.654	0.757	39109	4425	20706

Table 6: KGQA Results: Impact of the size of negative context. By increasing the number of graphs in context (i.e., negative graphs in context), the performance goes down probably due to increase in false positives compared true positives.

Impact of context size: The context for each question consists of relevant subgraphs retrieved from the FoodKG, which include both positive and negative recipes. The number of retrieved subgraphs or tagged recipes may vary for each question. The model may become vulnerable to accepting false positives when number of negative recipes increases in the context. To evaluate this, we limited the maximum number recipes included in the context to K by sampling at most $K/2$ recipe subgraphs from each positive set

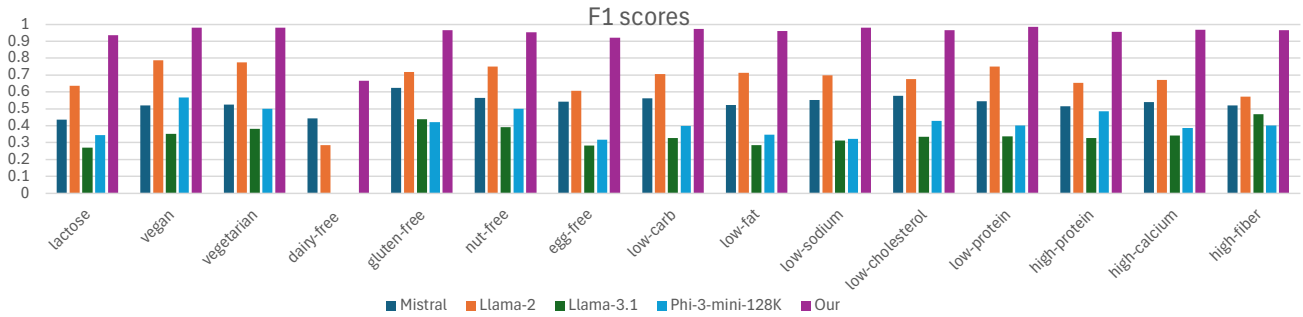

Figure 3: F1 score for different models on various types of recipes.

of tagged recipes $R^+(t_j)$ and negative set of tagged recipes $R^-(t_j)$. Table 6 demonstrates that increasing the size of context has negative impact on overall performance. This can be attributed to to the increase in negative subgraphs relative to positives, which results in more false positives and false negatives even though true positives also increase.

Model	Acc	mAP	P	R	F1
P-MatchNN	-	0.455	-	0.451	0.412
pFoodReq	-	0.627	-	0.618	0.637
Llama-2-7B	49.83	0.322	0.204	0.498	0.289
Our	**74.5**	**0.769**	**0.825**	**0.885**	**0.854**

Table 7: PFoodReq Results: Our model compared to baseline shows better scores.

Results on PFoodReq benchmark

State of the art: Due to close resemblance of our benchmark with PFoodReq (Chen et al. 2021), we compare our approach on PFoodReq dataset (See Table 2 for the statistics). Table 7 shows that compared to the previous methods, our method using LLMs with KGs outperforms with a margin of 0.14 points on mAP and 0.22 points on F1 metrics. This suggests that our method by utilizing the power of generative modeling grounded in KG facts generalizes better and outperforms the classical embedding based methods.

context size	Acc	mAP	P	R	F1	TP	FP	FN
M	74.5	0.769	0.825	0.885	0.854	10380	2196	1349
6	98.5	0.989	0.99	0.995	0.993	7032	68	38
10	62.9	0.644	0.675	0.903	0.772	7682	3271	726
20	62.4	0.63	0.666	0.907	0.768	8104	4061	832
40	50.3	0.482	0.524	0.927	0.696	9612	8763	758

Table 8: PFoodReq Results: Impact of the size of negative context. M: When all $R^+(t_j)$ and an equal number subgraphs from $R^-(t_j)$ are taken as potential context.

Impact of context size: Table 8 demonstrates the impact of context size for PFoodReq benchmark dataset on our model. Consistent with findings in KGQA, an increase in context size leads to a decline in performance, which is associated with a increase in number of false positives. This suggests that if the context has more number of negative subgraphs, the model may select recipes that do not fully meet all the specified constraints.

Conclusion

This paper presents a food recommendation system that combines the power of KGs with LLMs in a question answering framework. We also create a large-scale QA benchmark dataset using FoodKG. After evaluation of several open source LLMs, we selected Phi-3-mini-128K to adapt it for food recommendations by training it to understand the subgraph from FoodKG to answering the complex constrained questions regarding personalized food recommendations. The evaluation results show that our model outperforms the baseline models. In future, we plan to improve tag identification and SPARQL query generation employing Chain-of-thoughts reasoning with LLMs. Incorporating ingredients substitution, person's health information and cultural preferences could could further improve the system's ability. Finally, combining food recommendation system and recipe generation would not only suggest suitable recipes but also generate customized recipes, offering a seamless solution for meal planning and cooking.

Limitations

- The recipes recommended by our food recommendation systems rely on the recipe subgraphs retrieved from FoodKG (Haussmann et al. 2019). The subgraphs serve as context for LLM for suggesting the recipes relevant to the user queries. Therefore our system may not suggest accurate recipe if incorrect context information is provided.
- Our food recommendation system uses the nutritional information from FoodKG, whereas for a realist application, the simulated nutritional constraints are assumed to be provided by the user. While, our system can recommend sugar-free recipes, it may not accurately recommend correct recipes for a diabetic person. In other words, our system do not directly establish the relationship between person's health conditions and the corresponding dietary constraints. This capability is left for future research.

References

Abdin, M.; Jacobs, S. A.; Awan, A. A.; Aneja, J.; Awadallah, A.; Awadalla, H.; Bach, N.; Bahree, A.; Bakhtiari, A.; Behl, H.; et al. 2024. Phi-3 technical report: A highly capable language model locally on your phone. *arXiv preprint arXiv:2404.14219*.

Achiam, J.; Adler, S.; Agarwal, S.; Ahmad, L.; Akkaya, I.; Aleman, F. L.; Almeida, D.; Altenschmidt, J.; Altman, S.; Anadkat, S.; et al. 2023. GPT-4 technical report.

Avila, C. V. S.; Vidal, V. M.; Franco, W.; and Casanova, M. A. 2024. Experiments with text-to-SPARQL based on ChatGPT. In *2024 IEEE 18th International Conference on Semantic Computing (ICSC)*, 277–284. IEEE.

Banerjee, D.; Awale, S.; Usbeck, R.; and Biemann, C. 2023. Dblp-quad: A question answering dataset over the dblp scholarly knowledge graph. *arXiv preprint arXiv:2303.13351*.

Cai, Z.; Cao, M.; Chen, H.; Chen, K.; Chen, K.; Chen, X.; Chen, X.; Chen, Z.; Chen, Z.; Chu, P.; et al. 2024. Internlm2 technical report. *arXiv preprint arXiv:2403.17297*.

Chen, J.-J.; Ngo, C.-W.; Feng, F.-L.; and Chua, T.-S. 2018. Deep understanding of cooking procedure for cross-modal recipe retrieval. In *Proceedings of the 26th ACM International Conference on Multimedia*.

Chen, M.; Jia, X.; Gorbonos, E.; Hoang, C. T.; Yu, X.; and Liu, Y. 2020. Eating healthier: Exploring nutrition information for healthier recipe recommendation. *Information Processing & Management*, 57(6): 102051.

Chen, Y.; Guo, Y.; Fan, Q.; Zhang, Q.; and Dong, Y. 2023. Health-aware food recommendation based on knowledge graph and multi-task learning. *Foods*, 12(10): 2079.

Chen, Y.; Subburathinam, A.; Chen, C.-H.; and Zaki, M. J. 2021. Personalized food recommendation as constrained question answering over a large-scale food knowledge graph. In *Proceedings of the 14th ACM international conference on web Search and data mining*, 544–552.

Chhikara, P.; Chaurasia, D.; Jiang, Y.; Masur, O.; and Ilievski, F. 2024. Fire: Food image to recipe generation. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, 8184–8194.

Dooley, D. M.; Griffiths, E. J.; Gosal, G. S.; Buttigieg, P. L.; Hoehndorf, R.; Lange, M. C.; Schriml, L. M.; Brinkman, F. S.; and Hsiao, W. W. 2018. FoodOn: a harmonized food ontology to increase global food traceability, quality control and data integration. *npj Science of Food*, 2(1): 23.

Eppalapally, S.; Dangi, D.; Bhat, C.; Gupta, A.; Zhang, R.; Agarwal, S.; Bagga, K.; Yoon, S.; Lipka, N.; Rossi, R. A.; et al. 2024. KaPQA: Knowledge-Augmented Product Question-Answering. *arXiv preprint arXiv:2407.16073*.

Forouzandeh, S.; Rostami, M.; Berahmand, K.; and Sheikhpour, R. 2024. Health-aware food recommendation system with dual attention in heterogeneous graphs. *Computers in Biology and Medicine*, 169: 107882.

Gao, X.; Feng, F.; Huang, H.; Mao, X.-L.; Lan, T.; and Chi, Z. 2022. Food recommendation with graph convolutional network. *Information Sciences*, 584: 170–183.

Geng, S.; Liu, S.; Fu, Z.; Ge, Y.; and Zhang, Y. 2022. Recommendation as language processing (rlp): A unified pretrain, personalized prompt & predict paradigm (p5). In *Proceedings of the 16th ACM Conference on Recommender Systems*, 299–315.

Haussmann, S.; Seneviratne, O.; Chen, Y.; Ne'eman, Y.; Codella, J.; Chen, C.-H.; McGuinness, D. L.; and Zaki, M. J. 2019. FoodKG: a semantics-driven knowledge graph for food recommendation. In *The Semantic Web–ISWC 2019: 18th International Semantic Web Conference, Auckland, New Zealand, October 26–30, 2019, Proceedings, Part II 18*, 146–162. Springer.

He, J.; and Zhu, F. 2021. Online continual learning for visual food classification. In *Proceedings of the IEEE/CVF international conference on computer vision*.

He, X.; Tian, Y.; Sun, Y.; Chawla, N. V.; Laurent, T.; LeCun, Y.; Bresson, X.; and Hooi, B. 2024. G-Retriever: Retrieval-Augmented Generation for Textual Graph Understanding and Question Answering. *arXiv preprint arXiv:2402.07630*.

Hou, Y.; and Zhang, R. 2024. Enhancing Dietary Supplement Question Answer via Retrieval-Augmented Generation (RAG) with LLM. *medRxiv*, 2024–09.

Javaheripi, M.; Bubeck, S.; Abdin, M.; Aneja, J.; Bubeck, S.; Mendes, C. C. T.; Chen, W.; Del Giorno, A.; Eldan, R.; Gopi, S.; et al. 2023. Phi-2: The surprising power of small language models. *Microsoft Research Blog*, 1(3): 3.

Jiang, A. Q.; Sablayrolles, A.; Mensch, A.; Bamford, C.; Chaplot, D. S.; Casas, D. d. l.; Bressand, F.; Lengyel, G.; Lample, G.; Saulnier, L.; et al. 2023. Mistral 7B. *arXiv preprint arXiv:2310.06825*.

Kirk, D.; van Eijnatten, E.; and Camps, G. 2023. Comparison of answers between ChatGPT and human dieticians to common nutrition questions. *Journal of Nutrition and Metabolism*, 2023(1): 5548684.

Kobayashi, A.; Mori, S.; Hashimoto, A.; Katsuragi, T.; and Kawamura, T. 2024. Functional Food Knowledge Graph-based Recipe Recommendation System Focused on Lifestyle-Related Diseases. In *2024 IEEE 18th International Conference on Semantic Computing (ICSC)*, 261–268. IEEE.

Li, D.; and Zaki, M. J. 2022. Food Knowledge Representation Learning with Adversarial Substitution. In *Proceedings of the 2nd Conference of the Asia-Pacific Chapter of the Association for Computational Linguistics and the 12th International Joint Conference on Natural Language Processing*.

Li, D.; Zaki, M. J.; and Chen, C.-h. 2023. Health-guided recipe recommendation over knowledge graphs. *Journal of Web Semantics*, 75: 100743.

Ling, Y.; Nie, J.-Y.; Nielsen, D.; Knäuper, B.; Yang, N.; and Dubé, L. 2022. Following good examples-health goal-oriented food recommendation based on behavior data. In *Proceedings of the ACM Web Conference 2022*, 3745–3754.

Luo, L.; Li, Y.-F.; Haffari, G.; and Pan, S. 2023. Reasoning on graphs: Faithful and interpretable large language model reasoning. *arXiv preprint arXiv:2310.01061*.

Ma, P.; Tsai, S.; He, Y.; Jia, X.; Zhen, D.; Yu, N.; Wang, Q.; Ahuja, J. K.; and Wei, C.-I. 2024. Large Language Models in Food Science: Innovations, Applications, and Future. *Trends in Food Science & Technology*, 104488.

Meta, A. 2024. Introducing meta llama 3: The most capable openly available llm to date. *Meta AI*.

Min, W.; Liu, C.; Xu, L.; and Jiang, S. 2022. Applications of knowledge graphs for food science and industry. *Patterns*, 3(5).

Min, W.; Wang, Z.; Liu, Y.; Luo, M.; Kang, L.; Wei, X.; Wei, X.; and Jiang, S. 2023. Large scale visual food recognition. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 45(8): 9932–9949.

Mohbat, F.; and Zaki, M. J. 2024. LLaVA-Chef: A Multimodal Generative Model for Food Recipes. In *ACM International Conference on Information and Knowledge Management*.

Moor, M.; Huang, Q.; Wu, S.; Yasunaga, M.; Dalmia, Y.; Leskovec, J.; Zakka, C.; Reis, E. P.; and Rajpurkar, P. 2023. Med-flamingo: a multimodal medical few-shot learner. In *Machine Learning for Health (ML4H)*. PMLR.

Qi, Z.; Yu, Y.; Tu, M.; Tan, J.; and Huang, Y. 2023. Foodgpt: A large language model in food testing domain with incremental pre-training and knowledge graph prompt. *arXiv preprint arXiv:2308.10173*.

Rangel, J. C.; de Farias, T. M.; Sima, A. C.; and Kobayashi, N. 2024. SPARQL Generation: an analysis on fine-tuning OpenLLaMA for Question Answering over a Life Science Knowledge Graph. *arXiv preprint arXiv:2402.04627*.

Razzaq, M. S.; Maqbool, F.; Ilyas, M.; and Jabeen, H. 2023. EvoRecipes: A Generative Approach for Evolving Context-Aware Recipes. *IEEE Access*.

Rostami, A.; Jain, R.; and Rahmani, A. M. 2024. Food Recommendation as Language Processing (F-RLP): A Personalized and Contextual Paradigm. *arXiv preprint arXiv:2402.07477*.

Salvador, A.; Hynes, N.; Aytar, Y.; Marin, J.; Ofli, F.; Weber, I.; and Torralba, A. 2017. Learning cross-modal embeddings for cooking recipes and food images. In *Proceedings of the Conference on Computer Vision and Pattern Recognition*. IEEE.

Shah, M.; and Tian, J. 2024. Improving LLM-based KGQA for multi-hop Question Answering with implicit reasoning in few-shot examples. In *Proceedings of the 1st Workshop on Knowledge Graphs and Large Language Models (KaLLM 2024)*, 125–135.

Shirai, S. S.; Seneviratne, O.; Chen, C.-H.; Gruen, D. M.; and McGuinness, D. L. 2021. Healthy Food Recommendation and Explanation Generation using a Semantically-Enabled Framework? In *International Semantic Web Conference: Posters, Demos, and Industry Tracks*. CEUR-WS.

Sun, J.; Xu, C.; Tang, L.; Wang, S.; Lin, C.; Gong, Y.; Shum, H.-Y.; and Guo, J. 2023. Think-on-graph: Deep and responsible reasoning of large language model with knowledge graph. *arXiv preprint arXiv:2307.07697*.

Taffa, T. A.; and Usbeck, R. 2023. Leveraging LLMs in Scholarly Knowledge Graph Question Answering. In *QALD/SemREC@ ISWC*.

Touvron, H.; Martin, L.; Stone, K.; Albert, P.; Almahairi, A.; Babaei, Y.; Bashlykov, N.; Batra, S.; Bhargava, P.; Bhosale, S.; et al. 2023. Llama 2: Open foundation and fine-tuned chat models.

Wahed, M.; Zhou, X.; Yu, T.; and Lourentzou, I. 2024. Fine-Grained Alignment for Cross-Modal Recipe Retrieval. In *Proceedings of the Winter Conference on Applications of Computer Vision*, 5584–5593. IEEE.

Wang, H.; Zhou, L.; Zhang, W.; and Wang, X. 2021. LiteratureQA: A Qestion Answering Corpus with Graph Knowledge on Academic Literature. In *Proceedings of the 30th ACM International Conference on Information & Knowledge Management*, 4623–4632.

Wu, S.; Irsoy, O.; Lu, S.; Dabravolski, V.; Dredze, M.; Gehrmann, S.; Kambadur, P.; Rosenberg, D.; and Mann, G. 2023. BloombergGPT: A large language model for finance. *arXiv preprint arXiv:2303.17564*.

Xu, Z.; Cruz, M. J.; Guevara, M.; Wang, T.; Deshpande, M.; Wang, X.; and Li, Z. 2024. Retrieval-augmented generation with knowledge graphs for customer service question answering. In *Proceedings of the 47th International ACM SIGIR Conference on Research and Development in Information Retrieval*, 2905–2909.

Xu, Z.; Jain, S.; and Kankanhalli, M. 2024. Hallucination is inevitable: An innate limitation of large language models. *arXiv preprint arXiv:2401.11817*.

Yang, Z.; Khatibi, E.; Nagesh, N.; Abbasian, M.; Azimi, I.; Jain, R.; and Rahmani, A. M. 2024. ChatDiet: Empowering personalized nutrition-oriented food recommender chatbots through an LLM-augmented framework. *Smart Health*, 32: 100465.

Appendix

Recipe tags

We selected recipes with the following tags for dataset generation, so that question answers pairs inherently focus on health constraints.

dairy-free	vegan	vegetarian
gluten-free	lactose	nut-free
egg-free	low-carb	very-low-carbs
low-fat	low-protein	low-sodium
low-cholesterol	high-protein	high-calcium
high-fiber	healthy	

Figure 4: Tags representing various dietary preferences and nutritional constraints.

Dataset Details

Tag	Tagged recipe	Train set	Test set
dairy-free	7	18	3
vegan	968	2287	314
vegetarian	3392	8142	979
gluten-free	565	1328	187
lactose	366	874	112
nut-free	45	107	13
egg-free	440	1078	124
low-carb	4239	10248	1244
very-low-carbs	1005	2411	288
low-fat	2202	5296	639
low-protein	3320	7944	1032
low-sodium	4445	10561	1407
low-cholesterol	3710	8990	1059
high-protein	690	1642	219
high-calcium	540	1318	162
high-fiber	33	76	8

Table 9: Our dataset: Number of tagged recipes for each tag and questions for each tag in test set.

Foundational Models

Here we provide a list of baseline LLMs we compare with in our empirical evaluation.

internLM2 (Cai et al. 2024) is the second generation internLM model, trained to capture long-term dependencies. It outperforms on 30 benchmarks in long context modeling and open-ended subjective evaluations.

Mistral (Jiang et al. 2023) is engineered for superior performance and efficiency. It is the second best model on Hugging face leader board (Dec 27, 2023); its 7B model can outperforms LLaMA-2 13B model.

LLama-2 (Touvron et al. 2023) is a collection of foundation language models ranging from 7B to 70B. Due to popularity of llama series, we select Llama-2-7B model in our study.

Llama-3.1 (Meta 2024) is a set of large scale very powerful open source LLM that improves upon Llama-2, and is comparable to the flagship models like GPT-4, GPT-40 and Claude 3.5 Sonnet. Therefore, it became an obvious choice for our study.

Phi-2 (Javaheripi et al. 2023) is a 2.7B parameter LLM designed for efficient and high-performing natural language processing tasks. It has demonstrated better performance than LLaMA-2 (13B) and Mistral (7B) models on a range of benchmark tasks, showcasing its effectiveness in various NLP domains.

Phi-3 (Abdin et al. 2024) has improved over Phi-2; even its mini version at 3.8B parameters outperforms several 7B and 13B models. We employed Phi-3-mini-4k and Phi-3-mini-128K in our study for their performance despite the smaller size.

Metrics

We used standard retrieval metrics and provide their formal definitions considering order agnostic evaluation of all the models. Let Y be a list of recipes as ground truth answer and $\tilde{Y}$ a list of recipes recommended by the model. Then, we define true positive (TP), false positive (FP) and false negative (FN) as follows:

$$\begin{aligned} TP &= Y \cap \tilde{Y} \\ FP &= \tilde{Y} - Y \\ FN &= Y - \tilde{Y} \end{aligned} \tag{2}$$

Then, accuracy, precision (P), recall (R) and F1 scores are computed as follows:

$$\begin{aligned} Acc &= \frac{|TP|}{|Y|} \\ P &= \frac{|TP|}{|TP| + |FP|} \\ R &= \frac{|TP|}{|TP| + |FN|} \\ F1 &= \frac{2PR}{P + R} \end{aligned} \tag{3}$$

Additional Results

Table 10 shows performance of open sources LLMs and our model on KGQA benchmark for recipes tagged with few tags including lactose, vegan, vegetarian, gluten-free and nut-free.

Model	Tag	Acc	mAP	P	R	F1
internLM2	lactose	1.67	0.015	0.008	0.017	0.011
Mistral		36.79	0.14	0.538	0.368	0.437
Llama-2		51.42	0.477	0.838	0.514	0.637
Llama-3.1		32.08	0.152	0.233	0.321	0.27
Phi-3-mini-128K		21.74	0.202	0.812	0.217	0.343
Our		91.64	0.898	0.955	0.916	0.935
internLM2	vegan	7.89	0.09	0.038	0.0079	0.051
Mistral		49.16	0.201	0.549	0.492	0.519
Llama-2		70.97	0.669	0.885	0.71	0.788
Llama-3.1		42.95	0.161	0.296	0.43	0.351
Phi-3-mini-128K		42.11	0.404	0.873	0.421	0.568
Our		79.49	0.964	0.988	0.975	0.981
internLM2	vegetarian	7.78	0.085	0.034	0.078	0.048
Mistral		52.01	0.201	0.531	0.52	0.526
Llama-2		70.84	0.639	0.856	0.708	0.775
Llama-3.1		46.23	0.179	0.325	0.462	0.381
Phi-3-mini-128K		35.2	0.361	0.871	0.352	0.501
Our		97.58	0.966	0.987	0.976	0.981
internLM2	gluten-free	5.85	0.062	0.027	0.058	0.037
Mistral*		67.31	0.26	0.581	0.673	0.624
Llama-2		60.94	0.559	0.87	0.609	0.717
Llama-3.1		54.85	0.208	0.364	0.548	0.438
Phi-3-mini-128K		28.46	0.282	0.811	0.285	0.421
Our		95.13	0.939	0.982	0.951	0.966
internLM2	nut-free	6.67	0.103	0.019	0.067	0.029
Mistral*		59.09	0.224	0.542	0.591	0.565
Llama-2		68.18	0.628	0.833	0.682	0.75
Llama-3.1		45.45	0.243	0.345	0.455	0.392
Phi-3-mini-128K		36.67	0.385	0.786	0.367	0.5
Our		100	1.0	0.909	1.0	0.952

Table 10: Results on KGQA test set for reported for several tags. Overall, over model performs for questions relevant to the tags.