위치 기반 맞춤형 추천을 위한 RAG 챗봇 시스템 개발*

박시연○, 이해영, 홍수정, 이기용
숙명여자대학교 컴퓨터과학과
{sy.park, hae081128, redcree, kiyonglee}@sookmyung.ac.kr

Development of an RAG-based Chatbot System for Location-Based Personalized Recommendations

Siyeon Park○, Hae Young Lee, Sujeong Hong, Ki Yong Lee
Department of Computer Science, Sookmyung Women's University

요 약

본 논문에서는 사용자의 위치 정보를 기반으로 맞춤형 장소 추천을 제공하는 챗봇 시스템을 제안한다. 제안된 시스템은 실시간으로 업데이트되는 사용자의 현재 위치 정보를 반영하여 최적의 장소를 추천하는 데 중점을 둔다. 1) 초기 단계에서는 사용자가 위치한 행정구역을 기준으로 검색 범위를 축소하고, 2) K-최근접 이웃(KNN) 알고리즘을 이용하여 사용자로부터 가까운 장소들을 선별한다. 3) 이후, 코사인 유사도를 통해 사용자 요구에 가장 적합한 장소를 추천한다. 또한, Retrieval-Augmented Generation(RAG) 기법을 이용하여 외부 데이터베이스에서 입력된 프롬프트와 관련된 장소 정보를 검색하고 이를 답변에 반영함으로써 사용자에게 더욱 정확하고 관련성 높은 추천을 제공한다. 실험 결과, 제안 시스템은 기존 챗봇 시스템의 한계를 극복하고 사용자의 위치와 요구에 맞춤화된 장소 추천 서비스를 제공함을 확인하였다.

1. 서 론

현재 챗봇 시스템은 대형 언어 모델(Large Language Model, LLM)[1]을 기반으로 사용자의 질의를 이해하고, 그 문맥에 맞는 적절한 응답을 생성하는 능력을 갖추고 있다. 그러나 LLM은 고정된 데이터에 의존하기 때문에 학습 시점 이후의 새로운 정보나 동적인 데이터를 반영하지 못하는 한계를 지닌다. 특히 사용자의 위치나 특정한 요구 사항처럼 개인화된 데이터를 처리하는 데 어려움을 겪는다. 이로 인해 LLM 기반 시스템은 사용자의 실시간 요구를 충분히 반영하지 못할 가능성이 있으며, 사용자 만족도가 저하될 위험이 있다. 예를 들어, 특정 위치에서 이용할 수 있는 서비스나 맞춤형 장소 추천에 대한 요구를 충족하지 못하는 경우가 있다.

이러한 제약을 극복하기 위해, 본 논문은 RAG(retrieval-augmented generation) 기반의 장소 추천 챗봇 시스템을 제안한다. RAG는 맥락에 근거하지 않은 허구의 정보를 답변으로 생성(hallucination)해 내는 LLM의 한계를 보완하고, 외부 정보를 실시간으로 검색하여 더 정확하고 적절한 응답을 생성할 수 있는 구조를 제공한다 [2]. 제안 시스템은 사용자의 위치와 요구 사항을 반영하여 최적의 장소를 추천하며, 맞춤형 응답을 제공하는 것을 목표로 한다. 이를 구현하기 위해 제안 시스템은 먼저 외부 데이터의 검색 범위를 사용자가 위치한 행정구역 내로 줄이는 1차 필터링 작업을 진행하고, 이어서 K-최근접 이웃(KNN) 알고리즘을 사용하여 사용자의 위치와 가장 가까운 K개의 장소를 선택한다. 이후 코사인 유사도를 활용해, 사용자가 입력한 질의 내용과 가장 유사한 장소를 K개의 후보 장소 중에서 최종적으로 선택하여 추천한다. 이러한 방법은 사용자의 요구와 특성에 맞춘 최적의 장소 추천을 가능하게 한다.

본 논문의 구성은 다음과 같다. 2장에서는 관련된 선행 연구를 검토하고, 3장에서는 제안한 RAG 기반 챗봇 시스템의 설계와 구현 방법을 다룬다. 4장에서는 제안된 시스템의 성능을 평가하기 위해 기존 챗봇 시스템과의 실험 결과를 비교·분석하며, 5장에서는 결론과 향후 연구 방향을 제시한다.

2. 사전 지식 및 관련 연구

2.1 RAG(retrieval-augmented generation)

그림 1은 RAG(retrieval-augmented generation)의 일반적인 구조를 나타낸 것이다. RAG는 LLM의 한계를 보완하는 방법으로 제안된 모델로, 외부 데이터베이스에서 정

* 이 논문은 정부재원(과학기술정보통신부 여대학원생 공학연구팀제 지원사업)으로 과학기술정보통신부와 한국여성과학기술인 육성재단의 지원을 받아 연구되었습니다. (No.WISET-2024-063호)

보를 검색하는 방식이다. [3]에서는 RAG가 LLM이 생성한 응답의 신뢰성을 높이고, 고정된 지식만으로는 처리하기 어려운 정보 갱신 문제를 해결할 수 있으며, LLM만 단독으로 사용한 기존 모델에 비해 집약적인 질의에서 더 나은 성능을 보였음을 밝혔다.

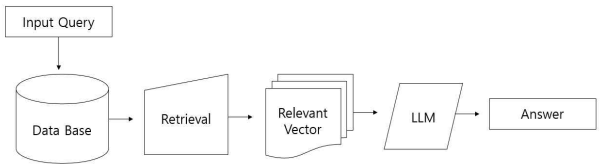

그림 1. RAG의 일반적인 구조

그러나 기존 연구들은 주로 문서 검색이나 QA 시스템에 RAG를 적용하는 데 그쳤으며, 사용자의 위치와 같은 개인 정보에 적용한 사례는 드물다. 본 논문은 이러한 점에서 RAG를 사용하여 실시간 위치 기반의 맞춤형 추천을 제공하는 새로운 접근 방식을 제안하며, 이를 통해 RAG의 활용 범위를 확장하고자 한다.

2.2 임베딩(Embedding)

임베딩(Embedding)은 고차원의 데이터를 저차원의 벡터로 변환하는 기법으로, 데이터의 의미를 벡터 공간에 효율적으로 표현하여 기계 학습 모델이 쉽게 처리할 수 있도록 돕는다.

임베딩 벡터는 주로 word2vec [4], BERT [5]와 같은 기계 학습 기법을 통해 학습되며 이러한 벡터 표현은 벡터 간의 사칙연산(덧셈, 뺄셈 등)과 내적 연산을 통해 단어 사이의 유사성과 의미적 관계를 분석할 수 있게 한다.

예를 들어, 벡터연산 $\overrightarrow{king} - \overrightarrow{man} + \overrightarrow{woman} \approx \overrightarrow{queen}$은 벡터 공간에서 "king"과 "man"의 차이를 "woman"에 더해 “queen”이라는 결과를 도출하여, 단어 간의 의미적 관계를 수치적로 설명하는 방법을 보여준다. 이러한 벡터 연산은 단순한 단어 간의 유사성뿐만 아니라 복잡한 의미적 관계까지도 반영할 수 있다.

이러한 임베딩 기법은 텍스트 기반의 추천 시스템에서 사용자의 질의를 처리하는 부분에 널리 사용되고 있으며, 본 연구 역시 임베딩을 통해 사용자 질의와 장소 간의 유사성을 계산하여 추천 성능을 개선하는 방법을 제안한다.

3. 제안하는 방법

3.1 문제 정의

사용자의 위치 좌표를 (x_u, y_u)라하고, 사용자 질의의 임베딩 결과를 $E_q = (q_1, q_2, ..., q_i)$라 하자. 각 장소의 위치 좌표는 (x_{ip}, y_{ip})로 나타내며, 해당 장소에 대한 정보의 임베딩 결과를 $E_p = (p_1, p_2, ..., p_i)$라고 정의한다. 본 논문에서는 사용자의 위치$(x_u, y_u)$를 기준으로 K-최근접 이웃(KNN) 알고리즘을 통해 가장 가까운 K개의 장소를 찾고, 이 장소들의 임베딩 벡터 E_p와 사용자 질의의 임베딩 벡터 E_q 간의 코사인 유사도를 계산하여, 사용자 위치 근처의 가장 적합한 장소를 추천하는 시스템을 제안한다.

3.2 장소 추천 매커니즘

그림 2는 제안된 시스템의 전체 아키텍처와 데이터 흐름을 시각적으로 나타낸 것이다. 이 시스템은 세 가지 주요 구성 요소로 이루어져 있으며, 각 구성 요소는 상호작용하여 사용자의 요청을 처리하고 최적의 장소를 추천한다.

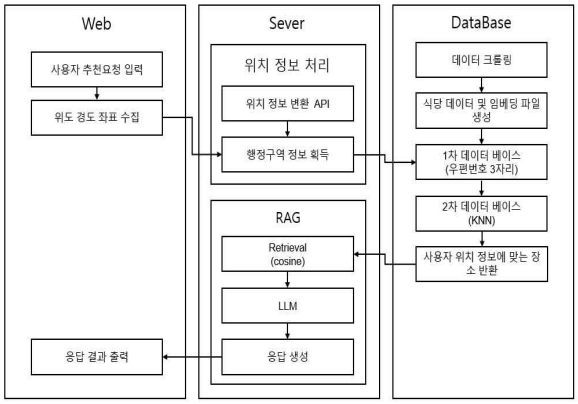

그림 2. 제안 시스템의 플로우차트

첫 번째 구성 요소는 ‘웹(Web) 인터페이스’로, 사용자는 웹 인터페이스의 프롬프트에 특정 조건을 만족하는 장소 추천 요청을 입력하며, 이때 사용자의 현재 위치 정보가 함께 수집된다. 두 번째 구성 요소는 ‘서버’이다. 사용자의 요청과 위치 정보가 서버로 전달되면, 서버는 위치 정보 처리 모듈을 통해 사용자의 위치 좌표를 분석한다. 이 모듈은 API를 호출하여 사용자의 위치 좌표를 행정구역 정보로 변환하고, 변환된 행정구역 정보는 서버에서 외부 데이터베이스의 검색 범위를 좁히는 데 사용된다. 세 번째 구성 요소인 ‘데이터베이스’는 세 단계의 필터링 과정을 통해 최적의 장소를 반환한다.

필터링 단계는 다음과 같다:

- 첫 번째 필터링 단계에서 서버는 사용자의 위치 좌표를 주소로 변환하고, 해당 주소를 우편번호와 매핑하여 우편번호의 앞 3자리를 기준으로 장소를 필터링한다. 여기서 우편번호 앞 3자리는 해당 장소가 속한 구나 동을 의미하며, 이 방식은 외부 데이터베이스의 검색 범위를 줄여 처리 속도를 향상시키는 역할을 한다.
- 두 번째 필터링 단계에서는 유클리디안 거리(수식 1)를 사용하여 1차로 필터링한 데이터베이스 내에서 사용자의 위치에 가장 가까운 K개의 장소를 선택한다.

$$distance_i = \sqrt{(x_u - x_{ip})^2 + (y_u - y_{ip})^2} \quad (1)$$

- 세 번째 필터링 단계에서는 선택된 장소들의 임베딩 벡터 $E_p = (p_1, p_2, \ldots, p_k)$와 사용자 질의에 대한 임베딩 벡터 $E_q = (q_1, q_2, \ldots, q_i)$간의 코사인 유사도(수식 2)를 계산하여, 사용자의 요청에 가장 부합하는 최적의 장소를 최종적으로 선정한다.

$$similarity = \frac{E_q \cdot E_p}{\|E_q\| \, \|E_p\|} = \frac{\sum_{i=1}^{k} q_i \times p_i}{\sqrt{\sum_{i=1}^{k} (q_i)^2} \times \sqrt{\sum_{i=1}^{n} (p_i)^2}} \quad (2)$$

첫 번째와 두 번째 필터링 단계는 데이터베이스에서 이루어지며, 세 번째 필터링 단계는 RAG 모듈에서 수행된다. RAG 모듈은 프롬프트와 문맥적으로 유사한 장소 정보를 검색(retrieval)하고, 이를 자연어 형태로 변환하여 프롬프트에 다시 전달함으로써, 대형 언어 모델(LLM)이 적절한 응답을 생성할 수 있도록 한다. 이 시스템은 사용자의 위치와 요청의 특성에 부합하는 맞춤형 장소 추천을 통해 사용자에게 더욱 정확하고 구체적인 결과를 제공한다.

4. 실험 결과

본 연구는 제안한 챗봇 시스템이 요청에 부합하는 장소를 얼마나 정확하게 추천하는지 검증하기 위한 실험을 진행하였다. 동일한 질문을 기존 시스템(ChatGPT-4o mini)과 제안한 챗봇 시스템에 입력하여, 두 모델의 답변을 비교한다. 각 질문에는 사용자가 원하는 특정 장소에 대한 요구 조건이 포함되어 있으며, 두 시스템에 동일한 위치 좌표를 제공하여 30개의 질문에 대한 정확도를 분석하였다.

프롬프트 : 제로페이를 사용할 수 있는 한식당을 추천해 줘		
분류	**답변 내용**	**정답**
제안 시스템	'신화쭈꾸미'와 '예향정 노원점'은 주변에 위치한 제로페이가 가능한 한식 식당입니다.	O
기존 시스템	감자탕집: 감자탕을 제공하는 식당입니다. 한식뷔페: 다양한 한식을 뷔페로 제공합니다.	X

표 1. 제안 시스템과 기존 시스템에서의 답변 비교

표 1은 특정 위치 좌표와 질의에 대해 두 시스템이 생성한 답변을 비교한 것이다. 기존 시스템에 질의를 할 때에는 사용자의 위치 좌표도 함께 제공하였다. 제안된 챗봇 시스템은 사용자의 위치와 요구 조건을 고려하여 적절한 장소를 추천하는 것으로 나타났다. 반면, 기존 챗봇 시스템은 동일한 좌표와 질의에 대해 추상적이고 모호한 답변을 생성하며, 사용자의 요구 사항에 부합하지 않는 장소를 추천하였다. 이 차이를 명확히 하기 위해 같은 방식의 실험을 30개의 질문으로 확장하여 진행하였고, 그 결과는 그림 3의 그래프와 같다.

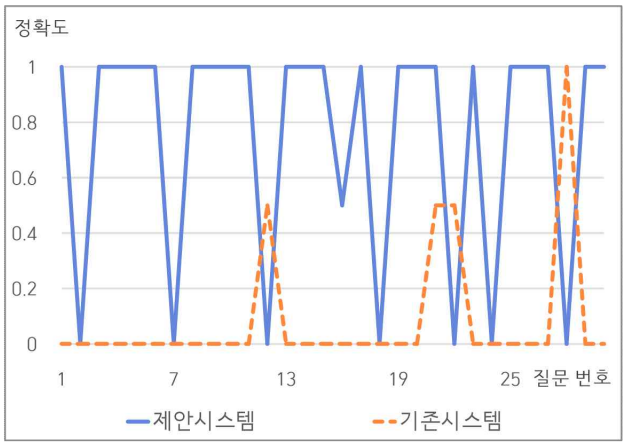

그림 3. 30개의 질문에 대한 두 시스템의 정확도 비교

그림 3은 두 시스템이 각 질문에 대해 생성한 답변의 정확도를 문항별로 나타낸 그래프이다. 모든 답변이 정확할 경우 1점을 부여하였고, 일부만 정확한 경우에는 답안의 개수와 관계없이 0.5점을 부여하였다. 이는 동일한 질문이어도 질의 방식에 따라 시스템이 반환되는 답변의 개수가 달라질 수 있기 때문에, F1 값과 같은 전통적인 평가 지표를 적용하는 데 한계가 있어 이를 보완하기 위한 조치이다. 특히 답변의 개수를 사전에 지정하지 않은 상황에서는, 하나의 답변이라도 생성된 답안이 사용자의 요구 사항을 얼마나 충족했는지가 더 중요하기 때문에 이러한 평가 방법을 적용하였다.

제안된 시스템은 사용자의 요청에 대한 답변 정확도가 높았던 반면, 기존 시스템은 다양한 오류가 포함된 답변을 생성하여 낮은 정확도를 보였다. 이는 제안된 시스템이 사용자의 위치와 요구 조건을 더욱 정확하게 반영하였음을 나타낸다.

시스템	평균 정확도 (Accuracy, %)
제안 시스템	75
기존 시스템	8.33

표 2. 두 시스템의 평균 정확도

표 2는 위 실험 결과를 바탕으로 제안된 시스템과 기존 시스템의 평균 정확도를 비교한 것이다. 제안 시스템의 평균 정확도는 0.75로, 기존 시스템의 0.0833에 비해 월등히 높은 성능을 보였다. 이는 제안 시스템이 사용자의 요구에 적합한 장소를 더 정확하게 추천할 수 있음을 보여준다.

그림 4는 두 시스템에서 발생한 주요 오류 유형과 그 빈도를 나타낸 것이다. 제안 시스템은 기존 시스템에서 자주 발생했던 모호한 장소 추천(예: 김치찌개 집)과 사용자의 위치와 관계없이 전국에 분포된 체인점 추천, 허구 장소 추천 등의 오류를 효과적으로 개선하였다. 이는 RAG 기반의

구조가 LLM에서 흔히 발생하는 환각 현상을 효과적으로 억제하고, 모호한 답변을 줄이며, 사용자의 위치 정보를 반영하여 보다 정확한 추천을 가능하게 했음을 보여준다.

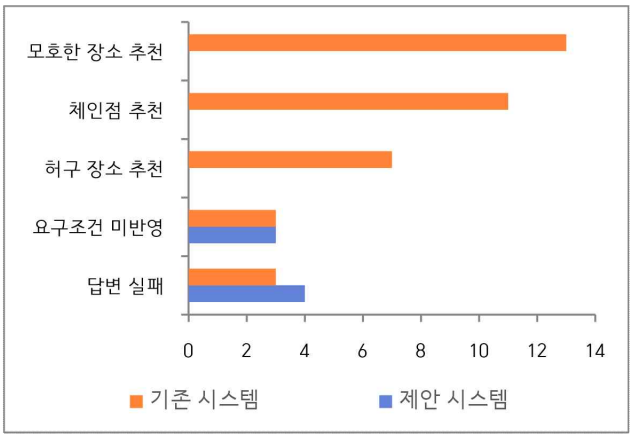

그림 4. 두 시스템에서 발생한 주요 오류 유형 및 개수

그러나 제안 시스템도 여전히 몇 가지 한계를 드러냈다. 첫째, 답변 실패 사례는 제안 시스템에서 4건 발생하였고, 이는 기존 시스템의 3건보다 많았다. 이 결과는 사용자의 특수한 요청과 해당 위치에 대한 데이터가 부족하여 발생한 것으로 분석된다. RAG 시스템이 외부 데이터에 의존하기 때문에, 데이터가 부족할 경우 답변을 생성하지 못하는 한계가 드러난 것이다.

둘째, 요구 조건 미반영 오류는 제안 시스템과 기존 시스템에서 동일하게 3건씩 발생하여, 두 시스템 모두 복잡한 요구 사항을 처리하는 데에 한계가 있음을 확인할 수 있었다. 특히, 제안 시스템의 경우 사용자가 요청한 조건의 일부만 반영된 사례가 있었다. 예를 들어, 사용자가 “주차를 할 수 있는 양식집을 추천해줘”라고 요청했음에도 불구하고, 시스템은 “주차장이 있는 한식집”을 추천한 경우가 있었다. 이는 제안 시스템이 일부 요구 사항을 제대로 반영하지 못하였음을 보여주며, 다중 조건을 처리할 수 있도록 외부 데이터를 임베딩할 때 가중치를 부여하는 방식과 같은 개선이 필요함을 시사한다.

5. 결 론

본 논문에서는 사용자의 위치와 세부 요청을 고려한 맞춤형 장소 추천을 위해 RAG 기반 챗봇 시스템을 제안하였다. 제안된 시스템은 사용자의 위치 정보를 우편번호로 변환하고, 해당 우편번호와 일치하는 행정구역 내의 장소에 대해 1차 필터링을 한다. 이후 KNN 알고리즘을 활용해 사용자의 위치와 가까운 장소들을 선별하여 2차 필터링을 수행한다. 마지막으로, 사용자 질의와 필터링된 장소들 간의 유사도를 분석하여 사용자의 요청에 가장 적합한 최적의 답안을 생성하는 방식으로 구성된다.

실험 결과, 제안 시스템은 기존 챗봇 시스템에 비해 더 높은 정확도로 답변을 생성하였으며, 위치 정보를 반영한 맞춤형 추천을 효과적으로 수행함으로써, 사용자의 요구에 더욱 부합하는 결과를 제공함을 확인하였다.

향후 연구에서는 다양한 도메인에 적용할 수 있는 가능성을 검토하고, 더욱 다양한 변수와 데이터를 통합하여 시스템의 정확도 및 활용도를 높이는 방안을 모색할 것이다. 또한, 사용자의 실시간 피드백을 반영하거나, 기존의 장소 데이터를 삭제하는 언러닝 기법을 도입하여 시스템의 유연성과 정확성을 더 높일 수 있을 것으로 기대된다.

참고문헌

[1] W. X. Zhao et al, "A survey of large language models," *arXiv Preprint,* arXiv:2303.18223, 2023.

[2] P. Sánchez and A. Bellogín, "Point-of-interest recommender systems based on location-based social networks: a survey from an experimental perspective," *ACM Computing Surveys (CSUR)*, vol. 54, (11s), pp. 1-37, 2022.

[3] P. Lewis, E. Perez, et al. "Retrieval-augmented generation for knowledge-intensive NLP tasks." *Advances in Neural Information Processing Systems (NeurIPS)*, 2020.

[4] T. Mikolov, K. Chen, et al. "Efficient Estimation of Word Representations in Vector Space." *Proceedings of the International Conference on Learning Representations (ICLR)*, 2013.

[5] J. Devlin, "Bert: Pre-training of deep bidirectional transformers for language understanding," *arXiv Preprint,* arXiv:1810.04805, 2018.