

Context Tamil Llama-QA: A Question and Answering Framework for Semantic Information Retrieval Using Tamil Large Language Model

Jeevit Davidson S. and M. Murali

Department of Computing Technologies, SRM Institute of Science and Technology, India

Article history

Received: 30-09-2025

Revised: 25-04-2026

Accepted: 06-05-2026

Corresponding Author:

M. Murali

Department of Computing
Technologies, SRM Institute of
Science and Technology, India
Email: muralim@srmist.edu.in

Abstract: The Question Answering System (QA) is essential for effectively retrieving and providing relevant information, and its performance is heavily reliant on the efficiency of information retrieval mechanisms. As the volume of information within the corpus continues to grow, the complexity of generating accurate and pertinent answers also increases substantially. This challenge is particularly significant when designing QA systems for regional languages such as Tamil, which present unique linguistic nuances. Currently, the primary retrieval mechanism depends largely on token matching, which identifies relevant words within user queries. While this approach may be effective for straightforward questions, it tends to falter when faced with inquiries that require deeper contextual understanding. As a result, the reliance on token matching can lead to inconsistencies or irrelevant answers, ultimately undermining the quality of the QA system. To overcome these challenges and enhance the consistency and relevance of answers, we propose a novel methodology that combines generative models with context-based retrieval techniques. This approach integrates an ensemble classifier with the Tamil Llama model, significantly improving both the speed and accuracy of information retrieval from the vector store. By leveraging the strengths of both models, we ensure that the retrieval process is context-driven rather than solely reliant on superficial word matches. Additionally, the CHAI dataset, specifically crafted for handling inquiries in Tamil, undergoes classification and labeling using advanced real-time classifier models. This rigorous classification enhances the system's ability to grasp the nuances of user queries, thereby improving response accuracy. Our evaluations demonstrate that this approach not only elevates the quality of the retrieved answers but also substantially reduces the response time for generating accurate answers, leading to a more efficient and user-friendly QA experience.

Keywords: Large Language Models (LLMs), Retrieval Augmented Generation (RAG)

Introduction

The Question Answering (QA) system is designed to comprehend and respond to questions in natural language through automation. It is tailored for specific domains such as healthcare, literature, sports, and customer service. Document retrieval plays a crucial role in the performance of the QA system as it aids in finding pertinent answers, which can be presented as sentences, paragraphs, or n-grams. This study holds significance as it can be applied to diverse domain-specific QA systems.

For example, a subject-specific QA system can greatly benefit students by enhancing their understanding of the subject matter. Closed-domain QA systems offer their own set of advantages, with their performance being contingent on the accuracy of the provided answers (Badugu and Manivannan, 2020; Alzubi et al., 2023).

The system must recover closely related information to furnish pertinent answers. Information retrieval is a fundamental element of QA systems, a realm in which OpenAI's ChatGPT and Large Language Models (LLMs) have shown improvement. However, LLMs encounter

limitations in domain-specific systems due to their restricted knowledge. It may be advantageous to address this limitation and employ LLMs for document retrieval in a Tamil language question-answering system. This research aims to harness the benefits of LLMs and identify a suitable document retrieval approach for a closed-domain, sentence-level question-answering system for Tamil. Addressing domain-specific information retrieval can be accomplished through the fine-tuning of LLMs or Retrieval-Augmented Generation (RAG) (Li et al., 2024; Mittal et al., 2024; Gupta et al., 2024).

In this study, the RAG method is chosen due to its ability to ensure accuracy by integrating retrieval and generative models. The main contributions of this work include proposing a multifaceted, context-augmented approach and employing the RAG process with the fine-tuned generative model "Tamil-Llama" for answer creation.

Related Work

The following work delves into the synergistic combination of RAG (Retrieval-Augmented Generation) and generative models. It provides an overview of existing literature on RAG-based information retrieval and the development of Language Models (LLMs) for the Tamil language. The development of LLMs in Tamil represents a significant advancement in natural language processing, particularly given the richness of the language and its cultural significance. Although LLMs are well-established in mainstream languages such as English, their development in ancient languages like Tamil is still in its nascent stages. Currently, Tamil LLMs (Shliazhko et al., 2024; Kakwani et al., 2020; Lin et al., 2021) are based on multilingual models, with the first model "GPT-2 Tamil" (Mahendiran, 2021) serving as a foundational starting point. This model, although limited in its capabilities due to being trained with fewer samples compared to other language models, laid the groundwork for further developments. Subsequent improvements to the model, such as fine-tuning with additional datasets (Radford et al., 2018; Caswell et al., 2020; Kumar et al., 2022), have significantly enhanced its performance. An advanced version of the Tamil LLM, known as "Tamil Llama," has been proposed (Balachandran et al., 2023), offering a more comprehensive landscape of the language model. This generative model is considered the most advanced among existing Tamil LLMs and is utilized as a key component in information retrieval for Question Answering systems (Le Scao et al., 2022).

The integration of RAG-based information retrieval has been particularly beneficial for enhancing LLMs, as it allows for effective information retrieval based on graphs. Notably, the WebQuestionsSP (WebQSP) open-domain Question Answering system, which relies on the Freebase dataset, has demonstrated the effectiveness of RAG by

achieving a 61% accuracy rate without any external knowledge. Furthermore, several closed-domain Question Answering systems have effectively utilized variants of RAG for document retrieval (Jiang et al., 2023; Rawat et al., 2020). This literature survey highlights the significant progress that has been made in generative models for the Tamil language. Leveraging the capabilities of these models and training them with specific closed-domain contexts can greatly benefit users by providing a robust closed-domain Question Answering system. It is evident that generative models can be effectively tuned with the RAG-based mechanism, and based on this consideration, a RAG generative model framework is proposed in this work (Xu et al., 2024).

Methods

Based on the findings from the literature survey, a novel approach called the Context Tamil Llama QA system attempts to find a solution for efficient information retrieval, focusing on the problems of question-specific answers and bridging the gap between context and knowledge. This method uses a hybridization strategy wherein the most relevant answer to the question is obtained through the classifier combined with the non-parametric external expertise from the RAG. This section details the components and the steps involved in document retrieval combining the generative model and retrieval process. Figure 1 presents the overall methodology of the proposed system.

Multifaceted Context Augmented Retrieval

The major component in this framework is the information retrieval engine that helps the QA system give the most relevant answers to the query.

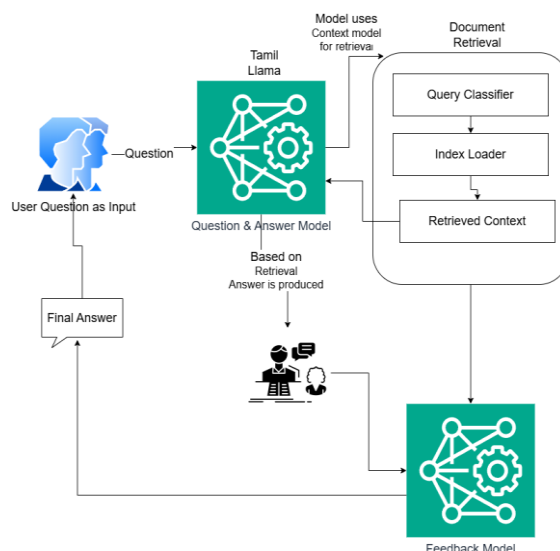


Fig. 1: Methodology overview of the RAG-based Tamil Llama Question and Answering System

This is only possible if the retrieval engine gets the most appropriate information. For performing this process, various classifiers are used, and the key steps that are part of this component are as follows:

1. The question from the user is received as input and passed to the ensemble classifier model
2. Based on the ensemble classification model, the questions are categorized and assigned to their corresponding domain or class. The class here refers to the type of the questions
3. Then the information retrieval is carried out based on the class the question is associated with
4. Next, the role of the retriever is to obtain the relevant information, and the retriever here is the bag-of-words retrieval that performs the retrieval process based on the word's presence in the query. The words in the question are ranked, and then, based on the ranking, the retrieval is done. Then the vector-based retrieval is also done, and the results of this combined retrieval are fed to the Tamil Llama for the generation of the responses to the query

Query Classifier

This component takes care of the separation of the information based on the question asked. Here, basically a grouping of the information happens. After the grouping is done, every group of information is assigned a label. The entire process of adding an index to the information based on the question is performed through several steps, and they are as follows:

1. The information is grouped based on the question and domain, and each group generated is assigned a label
2. Directory loading is carried out so that each domain's data is loaded separately with the assistance of the Llama index
3. The ingestion process that takes care of the loading of the essential data and the required transformations is carried out. The transformation here means the identification of the key entities of the question that, in turn, can aid in the relevant information extraction for the RAG. So, if we take a question like "நடிகர் விஜய் மனைவி யார்? (who is actor Vijay's wife?) the Llama model identifies the key entity as விஜய் (Vijay) and the other relevant token is மனைவி(wife)
4. The sentence splitter performs the process of dividing the content into chunks. Thus, here the identification of the indent of the question is carried out, which in turn can help in the classification of the questions. For example: for the same question, the question's indent is கணவனை அடையாளம் காணவும்
5. After the indent is identified, the indices are generated with the assistance of the word2Vec embeddings that vectorize the data for retrieval

6. The context of the question is then identified by the generative model and for the example question the contextual aspects are "தனிப்பட்ட உறவுகள்," "திருமண நிலை" மற்றும் "தற்போதைய மனைவி."
7. Thus, the vectors generated are labelled considering the context. Thus, the context-based RAG in turn helps in building separate information based on the domain, and thus this method is the ideal one for a closed-domain question and answering system

This user-provided set of questions is categorized with the assistance of different classifiers, and labelling is done for each domain. Various machine learning models can be used for this categorization, and some of the base models that are used for the query classification are logistic regression, naïve Bayes, and Support Vector Machine (SVM). In this method, an ensemble of the different algorithms is carried out to ensure good accuracy in the classification, and the information retriever can thus deliver the most important information.

Context Retrieval

Upon classifying the queries, the retrieval process is initiated using a strategy that combines syntax and semantics. Context retrieval is performed with a normalization retrieval score ranging from 0 to 1. Once the model comprehends the question's context, an appropriate response can be formulated. In the specific example under consideration, pertinent contextual elements encompass the relationship and status. The model then conducts an analysis for the filtering process; in this context, entities are ranked based on the query's intention. The top relations, reflecting contextual significance, are then chosen. In the given scenario, the model identifies the significant relations as "மக்கள்.நபர்.மனைவி_கள்." These retrieval components are subsequently converted into the RAG component, which oversees the answer generation. The fusion of context retrieval with the Tamil Llama ensures that the retrieved information is contextual and refers to the database. The RAG-based framework empowers the Tamil Llama model to access the specific domain relevant for obtaining the answer, enabling the model to prioritize crucial information closely associated with the queries (Yih et al., 2016).

Experimentation and Results Analysis

Dataset

The context of Tamil Llama QA is assessed using the CHAI dataset, which contains 368 instances for Tamil. This dataset is organized with attributes such as context, questions, answer text, and the starting index of the answer. The questions cover a broad range of topics without being limited to a specific domain.

Table 1: Comparison of the conventional machine learning classifier as the ensemble classifier in categorizing the questions

Model	Training Accuracy	Testing Accuracy	Recall	Precision
Logistic Regression	0.86	0.82	0.82	0.82
Gaussian Naives	0.92	0.90	0.90	0.90
Ensembled	1.0	1.0	1.0	1.0
Transformer	1.0	0.62	0.62	0.61

The evaluation involves conducting experiments with 3-hop questions and assessing the performance of the Tamil Llama using various classifiers, both with and without the inclusion of RAG.

Results

A comparative analysis is carried out to identify the better classifier for the query classification. The classification uses only the base machine learning algorithm and boosted ensemble classifier, and then the transformer model is used for fine-tuning the questions related to books, people, and facts domains, and their performance is evaluated using accuracy, precision, and recall. Table 1 presents the performance of the classification model in the categorization of the questions, and as shown in Table 1, the performance of the ensemble models is better than that of the transformer model, with 100% accuracy with a small sample size. Thus, the classification approach highly contributes to the improvement of the performance of the QA system with the retrieval of relevant answers.

During the evaluation of the larger dataset like the MultiIndic PMT, the categorization of the questions contributed greatly to the evaluation of the workflow. The questions are categorized as high and low, denoting the association of the question to that particular class. So the major classifications are factoid high, factoid low, description high, and description low. These are questions on factual reasoning that are marked as high and low based on the frequency of the answer. Then the time taken for answering the 20 questions of each category is also evaluated to understand the response time of the proposed workflow.

From the findings presented in Tables 2 and 3, the adoption of classifier logic with question segregation has reduced the information retrieval time. Table 2 retrieval time does not exceed 60%, and due to this, the response time is also reduced, and the range between them is 1-10%.

The cosine similarity score is also used to measure the difference between the actual answer and the generated answer. Table 4 presents the comparison results of the cosine similarity value with and without the inclusion of the RAG.

Along with the evaluation metrics, the variation between the actual and predicted answers is also validated with human rating.

The comparison is made only with context-based retrieval without the inclusion of RAG and with the inclusion of RAG.

Table 2: Comparison of the information retrieval with and without RAG

Question Type	Retrieval time change %
Description, High	-38.21%
Description, low	-20.87%
Factoid, High	-59.97%
Factoid, low	-45.12%

Table 3: Comparison of the response time with and without RAG

Question Type	Retrieval time change %
Description, High	-4.21%
Description, low	-10.43%
Factoid, High	-14.13%
Factoid, low	-1.26%

Table 4: Comparison of cosine similarity

Question Type	Average Cosine without RAG	Average Cosine with RAG
Description, High	0.88	0.89
Description, Low	0.86	0.87
Factoid, High	0.85	0.85
Factoid, Low	0.85	0.86

As shown in Fig. 2, the proposed RAG and context-based retrieval have an average rating of 3-4 for the majority of questions, showcasing its ability to retrieve the relevant content.

In accordance with Table 5, the RAG model received high ratings when more questions were posed by humans. This indicates that human evaluators tend to assign higher scores to longer answers, as reducing the length of the answer could potentially diminish its context. In such cases, there is a need to enhance the corpus. Additionally, it was observed that certain instances where the RAG model was excluded also generated proficient answers for specific questions. This illustrates the model's capability to retrieve relevant answers and context without redundancy, showcasing its adeptness in handling diverse types of queries.

Error Analysis

The proposed system performs effectively with single-step retrieval tasks, in which the answers are directly obtained from specific information. However, its performance noticeably declines when addressing questions that require multiple steps to extract relevant content, especially with multi-answer questions.

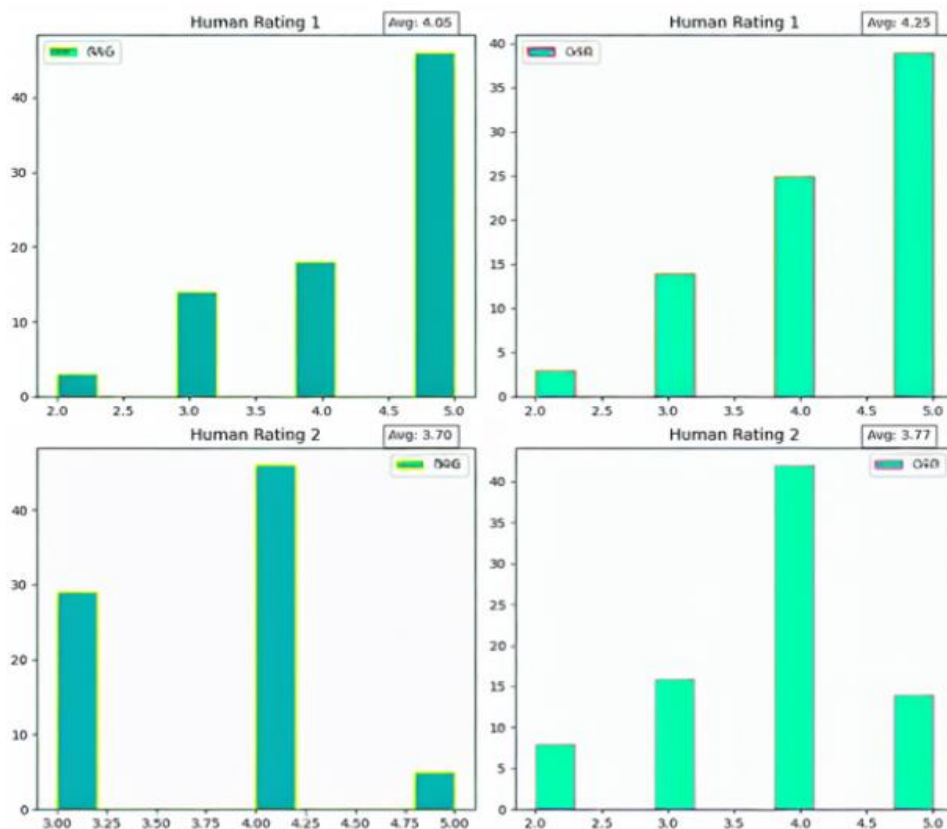


Fig. 2: Human evaluation of the proposed system

Table 5: Comparison of Human Rating with and without the inclusion of RAG

Human Rating 1 (with RAG)	
Answer size	Evaluation Score
5	2.0-2.25
15	2.85-3.25
18	3.75-4.20
60	4.75-5.0
Human Rating 1 (without RAG)	
4	2-2.5
13	2.75-3.25
25	3.75-4.15
38	4.75-5.0
Human Rating 2 (with RAG)	
28	3.00-3.15
60	4.00-4.25
7	4.95-5.00
Human Rating 2 (without RAG)	
8	2.0-2.25
15	2.95-3.25
55	3.75-4.0
14	4.75-5.0

For example, the question “Which film is directed and acted by Kamal Haasan?” is classified as a multi-answer question, as it can generate various responses related to Kamal Haasan's works, but may not necessarily be the answers to the specific question asked by the user. For the

CHAI dataset, which operates in a more structured and closed domain, the system attains an error rate of just 0.2%.

This performance metric, however, changes significantly in open-domain Question Answering (QA), where the scope of available information is wider and less organized, resulting in a higher error rate. This difference underscores the challenges presented in open-domain scenarios, where the system must navigate a more complex set of data to provide accurate responses to user queries. (Zhang et al., 2024)

Conclusion

This research introduces a groundbreaking method for context retrieval using the Tamil Llama model. The methodology includes a query classifier that categorizes questions, impacting response time and the accuracy of answers generated.

The key findings of this study are as follows:

- The incorporation of context augmentation significantly improved the Tamil Llama's ability to retrieve answers swiftly, resulting in reduced overall response time without compromising answer quality
- The cosine similarity between actual and predicted answers showed minimal variation

- Utilizing an ensemble classifier for query classification contributed to the accurate retrieval of answers

This study represents a pioneering effort using the Tamil Llama, and there is tremendous potential for further enhancement. Future optimizations to the model could include the auto-generation of class labels across multiple domains.

Acknowledgment

Thank you to the publisher for their support in the publication of this research article. We are grateful for the resources and platform provided by the publisher, which have enabled us to share our findings with a wider audience. We appreciate the efforts of the editorial team in reviewing and editing our work, and we are thankful for the opportunity to contribute to the field of research through this publication.

Funding Information

The authors have not received funds from any source.

Author's Contributions

Jeevit Davidson S. and M. Murali: Contributed to the design and implementation of the research, to the analysis of the results and to the writing of the manuscript.

Ethics

The authors have not received any financial support or funding to report.

References

- Alzubi, J. A., Jain, R., Singh, A., Parwekar, P., & Gupta, M. (2023). COBERT: COVID-19 Question Answering System Using BERT. *Arabian Journal for Science and Engineering*, 48(8), 11003–11013. <https://doi.org/10.1007/s13369-021-05810-5>
- Badugu, S., & Manivannan, R. (2020). A study on different closed domain question answering approaches. *International Journal of Speech Technology*, 23(2), 315–325. <https://doi.org/10.1007/s10772-020-09692-0>
- Balachandran, A. (2023). Tamil-llama: A new tamil language model based on llama 2. *ArXiv:2311.05845*. <https://doi.org/10.48550/arXiv.2311.05845>
- Le Scao, T., Fan, A., Akiki, C., Pavlick, E., Ilić, S., Hesslow, D., Castagné, R., Luccioni, A. S., Yvon, F., Gallé, M., Tow, J., Rush, A. M., Biderman, S., Webson, A., Ammanamanchi, P. S., Wang, T., Sagot, B., Muennighoff, N., Moral, A. V. del, ... et al. (2022). BLOOM: A176B-Parameter Open-Access Multilingual Language Model. *ArXiv*. <https://doi.org/10.48550/arXiv.2211.05100>
- Caswell, I., Breiner, T., van Esch, D., & Bapna, A. (2020). Language ID in the Wild: Unexpected Challenges on the Path to a Thousand-Language Web Text Corpus. *Proceedings of the 28th International Conference on Computational Linguistics*, 6588–6608. <https://doi.org/10.18653/v1/2020.coling-main.579>
- Gupta, A., Shirgaonkar, A., Balaguer, A., Benara, V., de Freitas Cunha, R. L., M. Estevão Filho, R. de, Hendry, T., Holstein, D., Marsman, J., & Mecklenburg, N. (2024). RAG vs Fine-tuning: Pipelines, Tradeoffs, and a Case Study on Agriculture. *ArXiv*. <https://doi.org/10.48550/arXiv.2401.08406>
- Jiang, J., Zhou, K., Dong, Z., Ye, K., Zhao, X., & Wen, J.-R. (2023). StructGPT: A General Framework for Large Language Model to Reason over Structured Data. *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, 9237–9251. <https://doi.org/10.18653/v1/2023.emnlp-main.574>
- Kakwani, D., Kunchukuttan, A., Golla, S., N.C., G., Bhattacharyya, A., Khapra, M. M., & Kumar, P. (2020). IndicNLP Suite: Monolingual Corpora, Evaluation Benchmarks and Pre-trained Multilingual Language Models for Indian Languages. *Findings of the Association for Computational Linguistics: EMNLP 2020*, 4948–4961. <https://doi.org/10.18653/v1/2020.findings-emnlp.445>
- Kumar, A., Shrotriya, H., Sahu, P., Mishra, A., Dabre, R., Puduppully, R., Kunchukuttan, A., Khapra, M. M., & Kumar, P. (2022). IndicNLG Benchmark: Multilingual Datasets for Diverse NLG Tasks in Indic Languages. *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, 5363–5394. <https://doi.org/10.18653/v1/2022.emnlp-main.360>
- Li, L., Zhou, J., Gao, Z., Hua, W., Fan, L., Yu, H., Hagen, L., Zhang, Y., Assimes, T. L., Hemphill, L., & Ma, S. (2024). A scoping review of using Large Language Models (LLMs). *ArXiv*. <https://doi.org/10.48550/arXiv.2405.03066>
- Lin, X. V., Mihaylov, T., Artetxe, M., Wang, T., Chen, S., Simig, D., Ott, M., Goyal, N., & Bhosale, S. (2021). Few-shot learning with multilingual language models. *Proceedings of the 2022 Conference on Empirical Methods in Natural Language*, 2119–2152. <https://doi.org/10.18653/v1/2022.emnlp-main.616>
- Mahendiran, A. (2021). abinayam/gpt-2-tamil (GPT2-Tamil). *Hugging Face Repository*.
- Mittal, A., Murthy, R., Kumar, V., & Bhat, R. (2024). Towards understanding and mitigating the hallucinations in NLP and Speech. *Proceedings of the 7th Joint International Conference on Data Science & Management of Data (11th ACM IKDD CODS and 29th COMAD)*, 489–492. <https://doi.org/10.1145/3632410.3633297>

- Radford, A., Narasimhan, K., Salimans, T., & Sutskever, I. (2018). *Improving language understanding by generative pre-training*.
- Rawat, B. P. S., Weng, W.-H., Min, S. Y., Raghavan, P., & Szolovits, P. (2020). Entity-Enriched Neural Models for Clinical Question Answering. *Proceedings of the 19th SIGBioMed Workshop on Biomedical Language Processing*, 112–122. <https://doi.org/10.18653/v1/2020.bionlp-1.12>
- Shliazhko, O., Fenogenova, A., Tikhonova, M., Kozlova, A., Mikhailov, V., & Shavrina, T. (2024). mGPT: Few-Shot Learners Go Multilingual. *Transactions of the Association for Computational Linguistics*, 12, 58–79. https://doi.org/10.1162/tac1_a_00633
- Xu, Z., Cruz, M. J., Guevara, M., Wang, T., Deshpande, M., Wang, X., & Li, Z. (2024). Retrieval-Augmented Generation with Knowledge Graphs for Customer Service Question Answering. *Proceedings of the 47th International ACM SIGIR Conference on Research and Development in Information Retrieval*, 2905–2909. <https://doi.org/10.1145/3626772.3661370>
- Yih, W., Richardson, M., Meek, C., Chang, M.-W., & Suh, J. (2016). The Value of Semantic Parse Labeling for Knowledge Base Question Answering. *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, 201–206. <https://doi.org/10.18653/v1/p16-2033>
- Zhang, Z., Fang, M., & Chen, L. (2024). RetrievalQA: Assessing Adaptive Retrieval-Augmented Generation for Short-form Open-Domain Question Answering. *Findings of the Association for Computational Linguistics ACL 2024*, 6963–6975. <https://doi.org/10.18653/v1/2024.findings-acl.415>