LLMForum-RAG: A Multilingual, Multi-domain Framework for Factual Reasoning via Weighted Retrieval and LLM Collaboration

Soham Chaudhuri[1], **Dipanjan Saha**[2], **Dipankar Das**[3]
[1]Dept. of Electrical Engineering, Jadavpur University, Kolkata, India
[2,3]Dept. of CSE, Jadavpur University, Kolkata, India
{ sohamchaudhuri.12.a.38, sahadipanjan6, dipankar.dipnil2005}@gmail.com

Abstract

LLMs have emerged as a transformative technology, enabling a wide range of tasks such as text generation, summarization, question answering, and more. The use of RAG with LLM is on the rise to provide deeper knowledge bases of various domains. In the present study, we propose a RAG framework that employs weighted Rocchio mechanism for retrieval and LLM collaborative forum with supervision for generation. Our framework is evaluated in two downstream tasks: a biomedical question answering (BioASQ-QA) and a multilingual claim verification (e.g. in English, Hindi, and Bengali) to showcase its adaptability across various domains and languages. The proposed retriever is capable to achieve substantial improvement over BM25 of $+8\%$ (BioASQ-QA), $+15\%$ (English), $+5\%$ (Hindi), and $+20\%$ (Bengali) for Recall@5. In veracity classification, our framework achieves an average answer correctness of 0.78 on BioASQ-QA while achieving F1-score of 0.59, 0.56, and 0.41 for English, Hindi and Bengali languages, respectively. These results demonstrate the effectiveness and robustness of our framework for retrieval and generation in multilingual and multi-domain settings.

1 Introduction

Advent of **L**arge **L**anguage **M**odels (LLMs) drastically changes the arena of **N**atural **L**anguage **G**eneration (NLG). Although the LLMs are trained on vast amount of data, their knowledge cut-off often restricts access to recent or emerging information. Training of these billion-parameter models regularly in order to cope up with up-to-date information is computationally heavy, expensive and financially prohibitive. Therefore, Lewis et al. (2021) proposed **R**etrieval **A**ugmented **G**eneration (RAG) to mitigate such problems. The framework combines information retrieval from external knowledge bases with generation, enabling models to access recent developments or domain-specific information without retraining.

Our RAG framework also incorporates **L**LM **C**ollaboration **F**orum (LCF), a **M**ulti-**A**gent **S**ystem (MAS) with a supervising agent. These agent systems are increasing in popularity day by day along with the developments of LLM requiring less human assistance. Lazaridou et al. (2017) explored the possibility of agents in the context of referential games (e.g., two agents, the source and the receiver identify targets by conversing among them). On the other hand, MAS-Zero from (Ke et al., 2025) was evaluated on graduate level QA, math, and reported an improvement of accuracy of 7.44% over baselines. Jimenez-Romero et al. (2025) and Lim et al. (2024) showcased the application of multi-agent systems in swarm intelligence and manufacturing systems too. In contrast, our framework consists of three agents collaborated on all the queries and retrieved contexts to generate their opinions. The context was divided into 3 parts with 10% overlap and distributed among them to reduce knowledge dilution. Finally, a supervising agent participates in their discussion and generates the final answer.

In RAG, very few works have been attempted in Indic Low-resource languages like Hindi and Bengali, even though they are the 5^{th} and 6^{th} most spoken languages in the world. The TraSe architecture from (Ipa et al., 2025) followed a translative approach where Bengali texts were translated to English in order to answer for a query, and the generated answer was translated back to Bengali. A Hindi-based text embedding was proposed specially for RAG applications in (Innovations, 2025), which showed quite promising results. On the other hand, RAG retrieval in English based on pseudo-relevance feedback have been explored thoroughly in (Wang et al., 2021) and (Wang et al., 2023).

In the present study, we propose an enhanced **P**seudo-**R**elevance **F**eedback (PRF) mechanism in-

Proceedings of the 14th International Joint Conference on Natural Language Processing and the 4th Conference of the Asia-Pacific Chapter of the Association for Computational Linguistics, pages 1426–1431

spired from the Rocchio algorithm (Rocchio, 1971) by employing similarity scores between retrieved documents and the query as weights. We experimented on a biomedical dataset and a multilingual claim verification dataset for retrieval and generation. Without requiring labeled relevance judgments, our Rocchio-based PRF achieves $+13.9\%$ relative gains in Recall@3 and $+8.7\%$ in Recall@5 in experiments on the BioASQ-QA dataset, consistently outperforming BM25[1] (Robertson and Zaragoza, 2009). The improvements of $+15.1\%$ (English), $+5.5\%$ (Hindi), and $+20\%$ (Bengali) for Recall@5 have shown better performance over multilingual settings. Moreover, the LLM collaboration forum achieved an average answer correctness score of 0.78 on the BioASQ-QA dataset and obtained F1-scores of 0.59, 0.56, and 0.41 for English, Hindi and Bengali, respectively on the claim verification task.

2 Dataset Description

The framework requires evaluation in various aspects, from the performance of retriever to answer generation. In case of retriever's performance, BioASQ-QA[2] dataset (Krithara et al., 2023) and claim verification dataset (Das et al., 2024) were used. The BioASQ-QA dataset has 40221 text corpora, 4719 queries, and answers based on context. For each question, a gold standard context is labeled with IDs. On the other hand, the claim dataset has a claim-evidence pair and the veracity level. We extracted all the evidences with respect to a specific ID and prepared the evidence corpus for evaluation. These datasets were used for measuring performances of retrieval from the structured knowledge bases whereas the claim dataset had 2367 English, 1114 Hindi, and 2282 Bengali claims. For answer generation, LLM collaborative forum was tested on all datasets. These answers were then compared against all the gold standard answers.

3 Methodology

Our RAG framework, as shown in Figure 1 comprises several components that facilitate retrieval and generation across various domains. Overall, the framework contains two main components - (i) a Rocchio based accompanied by a PRF-enhanced retriever from a vector database, and (ii) a Multi-agent system with a Supervising agent for generation. We used chromaDB[3] for the vector database and an open source `meta-llama/Llama-3.1-8B-Instruct`[4] (Grattafiori et al., 2024) for all the agents. We used LangChain[5] community packages to initialize the agents and to stream the framework wherever required. For embedding generation, we used a sentence transformer Multilingual-E5-Large[6] (Wang et al., 2024) for converting queries and contexts into embedding vectors for the claim. Since, BioASQ-QA is based on a biomedical domain, we used BGE-small-bioasq[7] to identify more semantics related to the biomedical domain.

3.1 Query Reformation

The initial query was passed through an agent to reform it into more structured query. This structured query was further utilized to retrieve contexts from the vector database after converting it to an embedding vector. Different datasets were employed for different component evaluations but reformation was investigated across all the datasets. Each agent had a closed-loop feedback to keep the agent from hallucinating. This query reformation is necessary for a general use case since the context window of LLMs is limited, and this reformation also helps keep queries under 256 tokens by prompting.

3.2 LLM Control Block

A control block has been developed to keep the agents in the framework from hallucinating. A hallucinated generation from any agent can pollute the framework entirely. Thus the control block had two main components: a semantic analysis component and a *LLM-as-a-Judge* (Gu et al., 2025) component. The semantic analysis was done using the sentence transformers mentioned before for the respective dataset. The sentence transformers convert the text pairs to semantic embeddings and then cosine similarity is calculated and normalized to generalize the reference frame all over the framework.

For the *LLM-as-a-Judge*, we used the same `meta-llama/Llama-3.1-8B-Instruct` model to

[1] `https://pypi.org/project/rank-bm25/`

[2] `https://huggingface.co/datasets/rag-datasets/rag-mini-bioasq`

[3] `https://www.trychroma.com/`

[4] `https://huggingface.co/meta-llama/Llama-3.1-8B-Instruct`

[5] `https://www.langchain.com/`

[6] `https://huggingface.co/intfloat/multilingual-e5-large`

[7] `https://huggingface.co/juanpablomesa/bge-small-bioasq`

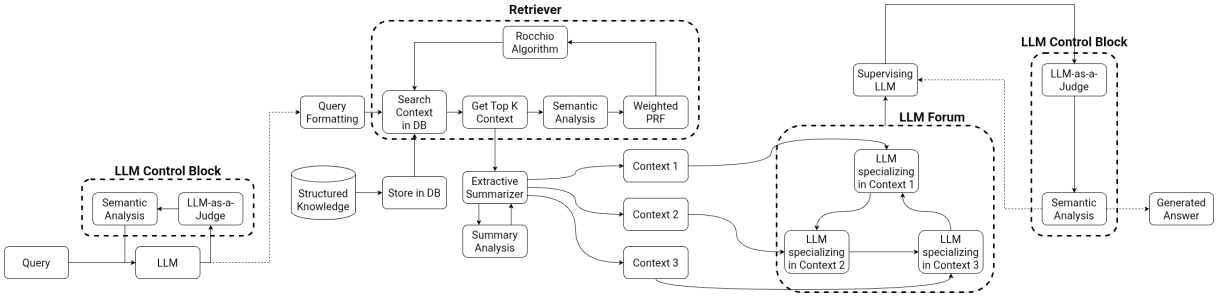

Figure 1: Overview of the multi-agent LLM framework with a control block, Rocchio retriever, and a collaboration forum of context-specialized agents. It improves query understanding, context selection, and answer generation through iterative summarization and supervised feedback for achieving better factual accuracy.

get LLM score for groundedness and answer relevancy. Considering averages over semantic analysis of question-answer, context-answer, and LLM scores to get the evaluation score of control block, we kept the threshold to be 0.5. If it is less than threshold, the agent had to generate again till it crosses the threshold.

3.3 Context Retrieval

Here, ChromaDB's query endpoint retrieves top_k context documents using K-NN and cosine similarity on embeddings. We then applied a weighted PRF based Rocchio algorithm for refinement. PRF marked the retrieved documents to be relevant (positive) or irrelevant (negative) to the query. We retrieved top_50 documents from ChromaDB's query endpoint and re-ranked them based on their cosine similarity with the query. The top_k documents were selected as relevant, and the remaining ones as irrelevant. Here, we experimented with 4 different k values $3, 5, 10$, and 20. For the relevant and irrelevant documents, we calculated the weighted centroids. Weights were considered to be their similarity scores with the query. The more similar a relevant document is, the more influential it should be close to the centroid's position in the embedding space. Since cosine similarity is based on similarity, we used $(1 - cosine\_score)$ for the irrelevant documents as influence on the irrelevant centroid should be inverse to their similarity scores. α, β, and γ were the weights that determine the influence of the initial query, centroids of relevant and irrelevant documents.

$$\vec{q'} = \alpha\vec{q} + \beta\left(\frac{\sum_{i=1}^{|D^+|} s_i^+ \vec{d_i^+}}{\sum_{i=1}^{|D^+|} s_i^+}\right) - \gamma\left(\frac{\sum_{j=1}^{|D^-|}(1 - s_j^-)\vec{d_j^-}}{\sum_{j=1}^{|D^-|}(1 - s_j^-)}\right)$$

α, β, and γ are the weights for the original query, positive examples, and negative examples, respectively. s_i^+ denotes the similarity score of the i^{th} positive example, and s_j^- denotes the similarity score of the j^{th} negative example. The values of α, β, and δ weights were all taken to be 1.0. The Rocchio algorithm was applied for a maximum of 1 epoch. In future ablation studies, we plan to experiment more with these hyperparameters. We used a summarizer component controlled by the average of ROUGE-1, ROUGE-2, and ROUGE-L scores (Lin, 2004) to summarize the context where concatenated context tokens became more than LLM tokens.

3.4 LLM Collaboration Forum

The **LLM C**ollaboration **F**orum (LCF) employs three agents. Retrieved or summarized context was split into three overlapping chunks (10% overlap) (Bhat et al., 2025) and assigned to each agent. Agents generate answers sequentially, using their context chunk and preceding agents' outputs as supplementary input. They are encouraged to build upon or refute others' views, ensuring close adherence to their context.

Refutations allow agents to act as pseudo *LLM-as-a-Judge*, reducing hallucination. Chunking prevents knowledge dilution by avoiding context overload. A supervising agent, within a closed feedback loop, generates the final answer based on the full discussion and original query. The process runs for two rounds to encourage mutual understanding.

4 Results and Discussion

We experimented with four values of top_k for retrieval and used the top-5 retrieved documents for generation, as recall@5 was satisfactory across datasets and smaller context windows reduced knowledge dilution. For the retrieval task, we evaluated on recall, **M**ean **R**eciprocal **R**ank (MRR) (Craswell, 2009), **M**ean **A**verage

	BioASQ			Claim Verification								
				English			Hindi			Bengali		
Metrics	C_1	C_2	C_3	C_1	C_2	C_3	C_1	C_2	C_3	C_1	C_2	C_3
Recall@K_1	0.36	0.40	0.41 (+**13.9%**)	0.69	0.80	0.80 (+**15.9%**)	0.88	0.93	0.93 (+**5.7%**)	0.71	0.87	0.87 (+**22.5%**)
MRR	0.70	0.76	0.77 (+**10%**)	0.63	0.72	0.73 (+**15.9%**)	0.84	0.88	0.89 (+**5.9%**)	0.64	0.79	0.80 (+**25%**)
MAP	0.56	0.65	0.65 (+**16%**)	0.63	0.72	0.73 (+**15.9%**)	0.84	0.88	0.89 (+**5.9%**)	0.64	0.79	0.80 (+**25%**)
nDCG@K_1	0.71	0.78	0.78 (+**9.8%**)	0.65	0.74	0.75 (+**15.4%**)	0.85	0.90	0.90 (+**5.9%**)	0.66	0.81	0.82 (+**24.2%**)
HitRate@K_1	0.77	0.83	0.83 (+**7%**)	0.69	0.80	0.80 (+**15.9%**)	0.88	0.93	0.93 (+**5.7%**)	0.71	0.87	0.87 (+**22.5%**)
Recall@K_2	0.46	0.50	0.50 (+**8.7%**)	0.73	0.84	0.84 (+**15.1%**)	0.91	0.95	0.96 (+**5.5%**)	0.75	0.90	0.90 (+**20%**)
MRR	0.71	0.75	0.77 (+**8.4%**)	0.64	0.72	0.74 (+**15.6%**)	0.84	0.88	0.89 (+**5.9%**)	0.65	0.79	0.80 (+**23.1%**)
MAP	0.54	0.60	0.61 (+**13%**)	0.64	0.72	0.74 (+**15.6%**)	0.84	0.88	0.89 (+**5.9%**)	0.65	0.79	0.80 (+**23.1%**)
nDCG@K_2	0.73	0.77	0.78 (+**6.8%**)	0.66	0.75	0.76 (+**15.2%**)	0.86	0.90	0.91 (+**5.8%**)	0.68	0.82	0.83 (+**22.1%**)
HitRate@K_2	0.82	0.85	0.86 (+**4.8%**)	0.73	0.84	0.84 (+**15.1%**)	0.91	0.95	0.96 (+**5.5%**)	0.75	0.90	0.90 (+**20.%**)
Recall@K_3	0.58	0.63	0.63 (+**8.6%**)	0.78	0.87	0.88 (+**12.8%**)	0.94	0.98	0.98 (+**4.3%**)	0.80	0.94	0.94 (+**17.5%**)
MRR	0.71	0.75	0.76 (+**7%**)	0.65	0.72	0.74 (+**13.8%**)	0.85	0.88	0.89 (+**4.7%**)	0.66	0.79	0.81 (+**22.7%**)
MAP	0.51	0.56	0.58 (+**13.7%**)	0.65	0.72	0.74 (+**13.8%**)	0.85	0.88	0.89 (+**4.7%**)	0.66	0.79	0.81 (+**22.7%**)
nDCG@K_3	0.73	0.76	0.78 (+**6.8%**)	0.68	0.76	0.77 (+**13.2%**)	0.87	0.90	0.91 (+**4.6%**)	0.69	0.82	0.84 (+**21.7%**)
HitRate@K_3	0.86	0.88	0.89 (+**3.5%**)	0.78	0.87	0.88 (+**12.8%**)	0.94	0.98	0.98 (+**4.3%**)	0.80	0.94	0.94 (+**17.5%**)
Recall@K_4	0.69	0.72	0.74 (+**7.2%**)	0.83	0.91	0.91 (+**9.6%**)	0.96	0.99	0.99 (+**3.1%**)	0.84	0.96	0.96 (+**14.3%**)
MRR	0.72	0.75	0.77 (+**6.9%**)	0.65	0.73	0.74 (+**13.8%**)	0.85	0.89	0.89 (+**4.7%**)	0.66	0.80	0.81 (+**22.7%**)
MAP	0.52	0.56	0.58 (+**11.5%**)	0.65	0.73	0.74 (+**13.8%**)	0.85	0.89	0.89 (+**4.7%**)	0.66	0.80	0.81 (+**22.7%**)
nDCG@K_4	0.74	0.76	0.77 (+**4.2%**)	0.69	0.77	0.78 (+**13%**)	0.87	0.92	0.92 (+**5.8%**)	0.70	0.84	0.84 (+**20%**)
HitRate@K_4	0.90	0.90	0.91 (+**1.1%**)	0.83	0.91	0.91 (+**9.6%**)	0.96	0.99	0.99 (+**3.1%**)	0.84	0.96	0.96 (+**14.3%**)

Table 1: C_1 denotes the baseline BM25, C_2 is PRF-Rocchio, and C_3 is PRF-Weighted-Rocchio. K_x refers to the different top-k values used in the experiments, specifically 3, 5, 10, and 20. Tests were done on 500 query samples. C_3 of Bengali shows the best results with an improvement of 20% over the baselines.

Precision (MAP) (Beitzel et al., 2009), **N**ormalized **D**iscounted **C**umulative **G**ain (nDCG) (Wang et al., 2013) and HitRate. Table 1 shows our framework outperformed BM25. For our baseline, we used BM25. It is a strong baseline for retrieval tasks based on token overlap, term frequency and inverse document frequency of tokens (Sammut and Webb, 2010). We engaged Llama-3.1-8B-Instruct LLM model for all the agents initialized through the langchain community packages. Since the claim dataset involves binary classification of claim veracity, we used accuracy and F1-score to evaluate the LLM Forum with supervisor agent on English, Hindi, and Bengali subsets. For the BioASQ-QA factoid question-answering dataset, we evaluated using answer relevance, faithfulness, and correctness against gold answers using `facebook/bart-large-mnli`[8] (Lewis et al., 2019).

Metrics	English	Hindi	Bengali
Accuracy	60	58	47
Macro Avg. F1	59	56	41

Table 2: Accuracy and F1-score based on veracity level. All the values are given in %.

Llama-3.1-8B-instruct is not extensively trained on Bengali; therefore, the model struggled in Bengali. We have used Intel(R) Xeon(R) CPU @ 2.20GHz for all experiments.

[8] https://huggingface.co/facebook/bart-large-mnli

	Faithfulness	Relevance	Correctness
Mean	57	81	78
Std	13	7	12
Min	34	55	20
Max	93	95	96

Table 3: BioASQ-QA dataset was measured based on faithfulness, relevance, and answer correctness (all the values are given in %). Statistical parameters like mean, standard deviation, minimum and maximum scores were observed.

5 Conclusion and Future Work

Table 1 shows our framework consistently outperformed BM25, which is more prominent for top-3 and top-5 retrievals. Table 2 and Table 3 show the performance of the LCF. These results demonstrate the effectiveness and robustness of our framework for retrieval and generation.

Our RAG framework is neither domain-specific nor language-specific. It requires prior initialization of the proper language-specific embedding sentence transformer and text preprocessors. Our future plan includes detailed ablation studies on the various hyper-parameters and thresholds used in the framework. All the parameters mentioned in this study were selected manually after conducting a small number of experiments with a very small sample of the data sets.

Limitations

Despite promising results, our method also has some limitations. Our proposed method is a dense retrieval technique, which is is heavily based on semantic embeddings. Semantic embeddings can capture more semantic similarity than sparse retrievals, which are based on token overlap but requires good sentence transformers. BioASQ-QA dataset being in the biomedical domain requires sentence transformers, which are primarily trained in this domain, to identify better semantic information for retrieval. If we use the encoders from **S**tate-**o**f-**t**he-**A**rt (SOTA) LLMs, it would provide us with a very good embedding space but it would increase the framework's latency. So, there has to be a trade-off between latency and embedding quality. We used multilingual-e5-large for claim as it has been trained on over 100+ languages and bge-small-bioasq which is finetuned for the biomedical domain.

ChromaDB's native query retriever endpoint uses K-NN algorithm for vector similarity but if the number of documents in the corpus increase, it uses an approximate nearest neighbor algorithm known as **H**ierarchical **N**avigable **S**mall **W**orld graphs (HNSW) (Malkov and Yashunin, 2018). HNSWs reduces latency, but they also hamper accuracy to some extent. An increase in the number of documents may lead to a decrease in results from the vector store.

In this study, we used the `meta-llama/Llama-3.1-8B-Instruct` model as it is an open-source and small model not specifically trained on low resource languages like Bengali and Hindi. Our future experiments will include larger, better models like `Llama-3.1-70B` model, GPT-4o (OpenAI et al., 2024) or Deepseek:R1 model (DeepSeek-AI et al., 2025).

Some samples of the BioASQ-QA dataset contain single-word answers. `Llama-3.1-8B-Instruct` was used with a fixed token completion size of 256, leading to over-generation as shown below:

Query:
Is there any algorithm for enhancer identification from chromatin state? \n

Generated Answer:
Chromatin state analysis can be leveraged for enhancer identification using various machine learning algorithms... \n

Gold Standard Answer:
Yes \n

Ethical Considerations

This study focuses on improving the retrieval and generation of factoid-based question answering. Datasets used are publicly available. Since LLMs can generate biased or incorrect information, special care has been taken to meticulously judge the answers based on relevance and correctness to the gold standard. No personally identifiable information was used in any part of the framework.

Since the biomedical domain is sensitive, we advise that the use of our framework must be done under strict human evaluation.

Acknowledgment

This work was supported by the Defence Research and Development Organisation (DRDO), New Delhi, under the project "Claim Detection and Verification using Deep NLP: an Indian perspective".

References

Steven M. Beitzel, Eric C. Jensen, and Ophir Frieder. 2009. *MAP*, pages 1691–1692. Springer US, Boston, MA.

Sinchana Ramakanth Bhat, Max Rudat, Jannis Spiekermann, and Nicolas Flores-Herr. 2025. Rethinking chunk size for long-document retrieval: A multi-dataset analysis. *Preprint*, arXiv:2505.21700.

Nick Craswell. 2009. *Mean Reciprocal Rank*, pages 1703–1703. Springer US, Boston, MA.

Shankha Shubhra Das, Pritam Pal, and Dipankar Das. 2024. Unveiling the truth: A deep dive into claim identification methods. In *Proceedings of the 38th Pacific Asia Conference on Language, Information and Computation*, pages 635–644, Tokyo, Japan. Tokyo University of Foreign Studies.

DeepSeek-AI, Daya Guo, Dejian Yang, Haowei Zhang, Junxiao Song, Ruoyu Zhang, Runxin Xu, Qihao Zhu, Shirong Ma, Peiyi Wang, Xiao Bi, Xiaokang Zhang, Xingkai Yu, Yu Wu, Z. F. Wu, Zhibin Gou, Zhihong Shao, Zhuoshu Li, Ziyi Gao, and 181 others. 2025. Deepseek-r1: Incentivizing reasoning capability in llms via reinforcement learning. *Preprint*, arXiv:2501.12948. Licensed under MIT.

Aaron Grattafiori, Abhimanyu Dubey, Abhinav Jauhri, Abhinav Pandey, Abhishek Kadian, Ahmad Al-Dahle, Aiesha Letman, Akhil Mathur, Alan Schelten, Alex Vaughan, Amy Yang, Angela Fan, Anirudh Goyal, Anthony Hartshorn, Aobo Yang, Archi Mitra, Archie Sravankumar, Artem Korenev, Arthur Hinsvark, and 542 others. 2024. The llama 3 herd of models. *Preprint*, arXiv:2407.21783. Licensed under llama3.1.

Jiawei Gu, Xuhui Jiang, Zhichao Shi, Hexiang Tan, Xuehao Zhai, Chengjin Xu, Wei Li, Yinghan Shen, Shengjie Ma, Honghao Liu, Saizhuo Wang, Kun Zhang, Yuanzhuo Wang, Wen Gao, Lionel Ni, and Jian Guo. 2025. A survey on llm-as-a-judge. *Preprint*, arXiv:2411.15594.

DeepMost Innovations. 2025. Hindi sentence embeddings model. https://huggingface.co/DeepMostInnovations/hindi-embedding-foundational-model.

Atia Shahnaz Ipa, Mohammad Abu Tareq Rony, and Mohammad Shariful Islam. 2025. Empowering low-resource languages: TraSe architecture for enhanced retrieval-augmented generation in Bangla. In *Proceedings of the 1st Workshop on Language Models for Underserved Communities (LM4UC 2025)*, pages 8–15, Albuquerque, New Mexico. Association for Computational Linguistics.

Cristian Jimenez-Romero, Alper Yegenoglu, and Christian Blum. 2025. Multi-agent systems powered by large language models: Applications in swarm intelligence. *Preprint*, arXiv:2503.03800.

Zixuan Ke, Austin Xu, Yifei Ming, Xuan-Phi Nguyen, Caiming Xiong, and Shafiq Joty. 2025. Mas-zero: Designing multi-agent systems with zero supervision. *Preprint*, arXiv:2505.14996.

Anastasia Krithara, Anastasios Nentidis, Konstantinos Bougiatiotis, and Georgios Paliouras. 2023. Bioasq-qa: A manually curated corpus for biomedical question answering. *Scientific Data*, 10:170. Licensed under cc-by-2.5.

Angeliki Lazaridou, Alexander Peysakhovich, and Marco Baroni. 2017. Multi-agent cooperation and the emergence of (natural) language. *Preprint*, arXiv:1612.07182.

Mike Lewis, Yinhan Liu, Naman Goyal, Marjan Ghazvininejad, Abdelrahman Mohamed, Omer Levy, Ves Stoyanov, and Luke Zettlemoyer. 2019. Bart: Denoising sequence-to-sequence pre-training for natural language generation, translation, and comprehension. *Preprint*, arXiv:1910.13461.

Patrick Lewis, Ethan Perez, Aleksandra Piktus, Fabio Petroni, Vladimir Karpukhin, Naman Goyal, Heinrich Küttler, Mike Lewis, Wen tau Yih, Tim Rocktäschel, Sebastian Riedel, and Douwe Kiela. 2021. Retrieval-augmented generation for knowledge-intensive nlp tasks. *Preprint*, arXiv:2005.11401.

Jonghan Lim, Birgit Vogel-Heuser, and Ilya Kovalenko. 2024. Large language model-enabled multi-agent manufacturing systems. *Preprint*, arXiv:2406.01893.

Chin-Yew Lin. 2004. ROUGE: A package for automatic evaluation of summaries. In *Text Summarization Branches Out*, pages 74–81, Barcelona, Spain. Association for Computational Linguistics.

Yu. A. Malkov and D. A. Yashunin. 2018. Efficient and robust approximate nearest neighbor search using hierarchical navigable small world graphs. *Preprint*, arXiv:1603.09320.

OpenAI, Josh Achiam, Steven Adler, Sandhini Agarwal, Lama Ahmad, Ilge Akkaya, Florencia Leoni Aleman, Diogo Almeida, Janko Altenschmidt, Sam Altman, Shyamal Anadkat, Red Avila, Igor Babuschkin, Suchir Balaji, Valerie Balcom, Paul Baltescu, Haiming Bao, Mohammad Bavarian, Jeff Belgum, and 262 others. 2024. Gpt-4 technical report. *Preprint*, arXiv:2303.08774.

Stephen Robertson and Hugo Zaragoza. 2009. The probabilistic relevance framework: Bm25 and beyond. *Found. Trends Inf. Retr.*, 3(4):333–389.

J. J. Rocchio. 1971. Relevance feedback in information retrieval. In G. Salton, editor, *The Smart retrieval system - experiments in automatic document processing*, pages 313–323. Englewood Cliffs, NJ: Prentice-Hall.

Claude Sammut and Geoffrey I. Webb, editors. 2010. *TF–IDF*, pages 986–987. Springer US, Boston, MA.

Liang Wang, Nan Yang, Xiaolong Huang, Linjun Yang, Rangan Majumder, and Furu Wei. 2024. Multilingual e5 text embeddings: A technical report. *arXiv preprint arXiv:2402.05672*. Licensed under MIT.

Xiao Wang, Craig Macdonald, Nicola Tonellotto, and Iadh Ounis. 2021. Pseudo-relevance feedback for multiple representation dense retrieval. In *Proceedings of the 2021 ACM SIGIR International Conference on Theory of Information Retrieval*, ICTIR '21, page 297–306. ACM.

Xiao Wang, Craig MacDonald, Nicola Tonellotto, and Iadh Ounis. 2023. Colbert-prf: Semantic pseudo-relevance feedback for dense passage and document retrieval. *ACM Trans. Web*, 17(1).

Yining Wang, Liwei Wang, Yuanzhi Li, Di He, Tie-Yan Liu, and Wei Chen. 2013. A theoretical analysis of ndcg type ranking measures. *Preprint*, arXiv:1304.6480.