Multi-Agent Frameworks in Domain-Specific Question Answering Tasks

Stanford CS224N Custom Project

Maria Angelika-Nikita
Department of Computer Science
Stanford University
mariaang@stanford.edu

Spencer Louis Paul
Department of Computer Science
Stanford University
spaul2@stanford.edu

Ethan Duncan He-Li Hellman
Department of Computer Science
Stanford University
hellman1@stanford.edu

Abstract

Diversity in opinion, thought, and background is widely valued in decision-making processes, underscoring the belief that collective knowledge often surpasses that of the individual. This paper investigates whether this principle extends to language models, specifically examining whether a combination of different language models achieves superior performance in question-answering tasks compared to any single model. We explore this hypothesis across various dimensions, especially considering the vast differences in architecture and training corpora of today's models. Our aim is to determine whether the collective capabilities of homogeneous and heterogeneous model ensembles can exceed those of individual models, particularly across different domains of specific knowledge tasks. To this end, we utilize a diverse selection of models and an extensive dataset collection covering areas such as mathematical reasoning, (Hendrycks et al., 2021), general knowledge QA (Joshi et al., 2017), and jurisprudential analysis (Guha et al., 2023). We find that heterogeneous ensembles are most effective when attempting to improve performance on very challenging tasks where baseline accuracy is low. Additionally, we further corroborate the notion that performance scales with ensemble size.

1 Key Information to include

- Project Type: Custom
- Team contribution: All teammates contributed equally. Each of the 3 took ownership over one of the datasets and all contributed equally to writing the paper.
- Mentor: Nelson Liu

2 Introduction

The advent of Large Language Models (LLMs) has revolutionized the field of Natural Language Processing (NLP), offering unprecedented capabilities in understanding and generating human language. Among the various strategies employed to enhance the performance of LLMs, the integration of multi-agent systems has emerged as a promising avenue. "More Agents Is All You Need" by Li et al. (2024) underscores the significance of employing multiple debating agents or chains-of-thought pipelines to improve problem-solving abilities across a wide spectrum of NLP tasks. This research builds on the premise that augmenting the number of LLM agents can lead to substantial performance

enhancements, drawing from evidence that suggests a positive correlation between agent collaboration and improved task outcomes.

Recent works, including those by Zhu et al. (2024) and Alawwad et al. (2024), have further highlighted the evolving complexity and potential of LLMs in fields beyond traditional NLP tasks, suggesting that increasing agent numbers could offer a simple yet effective means to boost performance. Motivated by these insights, our research aims to explore the effectiveness of multi-agent frameworks across diverse tasks, including question-answering, arithmetic reasoning, and legal analysis. We hypothesize that ensembling multiple agents, coupled with consensus mechanisms based on voting, can lead to enhanced output congruency and overall performance improvements. By experimenting with diverse model ensembles—such as GPT-4, LLaMA-13B, and Gemini—we aim to leverage a broader knowledge base and achieve higher benchmark scores.

Our research builds upon the findings of Li et al. (2024) while expanding the scope of investigation. Specifically, we contribute novel insights into the following questions:

1. Does homogeneous ensembling on domain-specific tasks improve performance with a greater number of agents? We aim to add nuance to Li et al.'s research by expanding the range of models and datasets used, investigating the scenarios in which ensemble scaling is most effective.
2. How does the efficacy of diverse model ensembles compare to homogeneous ones? We hypothesize that leveraging diversity in the knowledge base of multiple models will yield greater accuracy compared to the homogeneous ensembles used in Li et al.'s study.

Through rigorous experimentation and analysis, we aim to contribute to the ongoing dialogue on the scalability and utility of agent ensembles in enhancing the capabilities of Large Language Models. Our findings will provide valuable insights into the optimal design and deployment of multi-agent systems, paving the way for further advancements in the field of NLP.

3 Related Work

Our research is inspired by advancements in the field of large language models (LLMs), particularly focusing on **Homogeneous Model Scaling** and **Heterogeneous LLM Ensembling**. We draw upon these methodologies, combining and extending them to explore the scalability and performance of heterogeneous LLM ensembles.

3.1 Homogenous Model Scaling

The foundation of our approach stems from the scalability principle of homogeneous LLMs. Li et al. (2024) Li et al. (2024) demonstrated that LLM performance could be amplified by increasing the number of homogeneous agents. Their methodology, leveraging a consensus voting mechanism among models like GPT-3.5 and Llama, serves as a baseline for understanding model scaling benefits Li et al. (2024). Zhu et al. (2024) Zhu et al. (2024) and Alawwad et al. (2024) Alawwad et al. (2024) further reinforce this concept, showing how scaling can enhance capabilities in information retrieval and educational applications. These studies underline the potential yet highlight the challenges, such as computational demands and generalization issues, informing our approach's development towards overcoming these limitations.

3.2 Heterogeneous LLM Ensembling

Building on the concept of model scaling, we incorporate insights from heterogeneous LLM ensembling. Du et al. (2021) Du et al. (2022) introduced GLaM, utilizing a mixture-of-experts framework to scale models efficiently, indicating the importance of modular, expert-driven design for performance improvement Du et al. (2022). This method aligns with Shazeer et al.'s (2017) Shazeer et al. (2017) exploration of sparsely-gated MoE layers, suggesting a path towards large, yet efficient neural networks. Moreover, the Adaptive Mixtures of Local Experts (Jacobs et al., 1991) Jacobs et al. (1991) provide a paradigm for dynamic model blending based on input context, further informing our approach towards ensembling.

Distinctly, while Wan et al. (2024) Wan et al. (2024) and Jiang et al. Jiang et al. (2023) emphasize supervised learning frameworks for LLM fusion, we note the limitations imposed by extensive task-specific data requirements. Inspired by these frameworks, our research diverges by exploring unsupervised and semi-supervised methods, aiming to retain the ensembling benefits while reducing the dependency on large annotated datasets.

In synthesizing these concepts, our novel contribution lies in the fusion of heterogeneous LLMs scaled across different dimensions. We propose a unique framework that integrates the computational efficiency and modular adaptability of MoE with the robust scalability principles of homogeneous model scaling. This approach not only seeks to enhance performance but also to improve generalizability and efficiency, addressing the challenges highlighted by prior works.

4 Approach

In this study, we initially replicate the method presented in the paper by Li et al. (Li et al., 2024), focusing on the scaling effect of increasing the number of identical model agents on performance metrics. Notably, our research uses models GPT-3.5-Turbo, Llama2 13B, and Gemini Pro. The ensemble approach used is outlined in figure 1. It consists of two phases: sampling and voting.

Sampling. Let x represent a question in one of our dataset. The question is passed into a prompt constructor P to generate a query q for a specific model $\mathcal{M}$. $q = P(x, \mathcal{M})$. Prompts were model-dependent, since some allowed system prompting while others did not (see appendix for prompt template examples). In the homogeneous setting, we query model $\mathcal{M}$ N times to generate a response r where $r = \mathcal{M}(q)$. For each response, we apply a task specific parser to ensure consistency with the ground truth targets in each dataset during evaluation. We obtain a set of Responses $R = \{r_1, r_2, \cdots, r_N\}$. In the heterogeneous model setting, we query various models $\mathcal{M}_1, \mathcal{M}_2, \cdots, \mathcal{M}_J$ each $\frac{N}{J}$ times so that our total ensemble response size is N. For the purposes of observing scaling behavior, we increase N in steps of $i \cdot J \forall i \in \{1, 5, 10, 20\}$.

Voting Let the final answer be denoted as a. Since all of our tasks parsed and constrained to be discrete valued answered, we obtained a by taking the the most frequent answer in R. $a = argmax_{r_i \in R} V(r_i)$ where $V(r_i)$ returned the frequency count of r_i in the response set.

While our methodology followed the work done by Li. We decided to code our own pipeline for this procedure since Li's did not support handling of non-homogeneous model ensembles. Additionally, Li's repository did not include task construction for most of the datasets we evaluated. We wanted to leverage more modern libraries like LangChain to make agent interoperability flexible. Moreover, this allowed us to utilize more state-of-the-art models that are accessible via the LangChain API libraries. In today's world where there the most advanced models from different research institutes consistently compete to outperform one another, we ask the crucial question of whether or not they can work together to outperform themselves.

Baselines Since we were constrained by computational cost and experiment time, we sampled each of the datasets outlined in the Experiments section to create subsets of the tasks on the order of 100 questions. As a result, we decided to establish our own baselines for a fair comparison. Our research is not concerned with beating existing benchmarks on these datasets, but more so investigating the performance gap between homogeneous model ensembles and heterogeneous ones.

Our baselines are established based on the single-agent performance metrics for these models as reported in the original experiments. This allows us to measure the incremental gains from adding more agents of the same model type, and subsequently, more agents of different model types. Our evaluation metric used across all experiments is accuracy.

Figure 2 displays baseline results for our three tested models across our three datasets. Llama significantly under-preforms compared to GPT and Gemini accross all three tasks. We hypothesise that this is due to the much smaller model size of Llama compared to the other two. It is also worth noting that it is not always the case that accuracy scales with ensemble size. Specifically in the TriviaQA dataset we see that accuracy largely remained constant regardless of ensemble size. Similarly, we see marginal gains on the LegalBench dataset when increasing the ensemble size, though notably, the Llama model sees the greatest improvements.

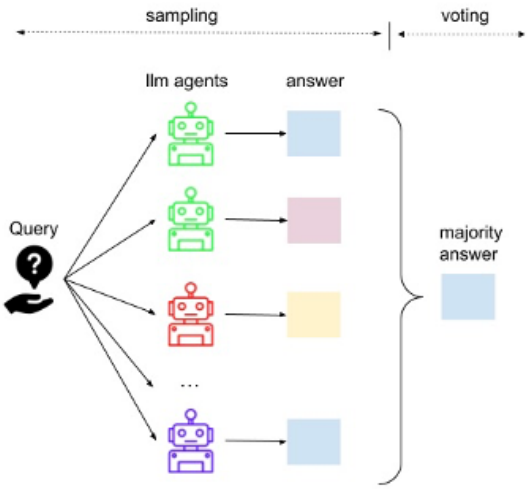

Figure 1: Depiction of Ensemble Method. Same colored LLM Agents represent the same model architecture

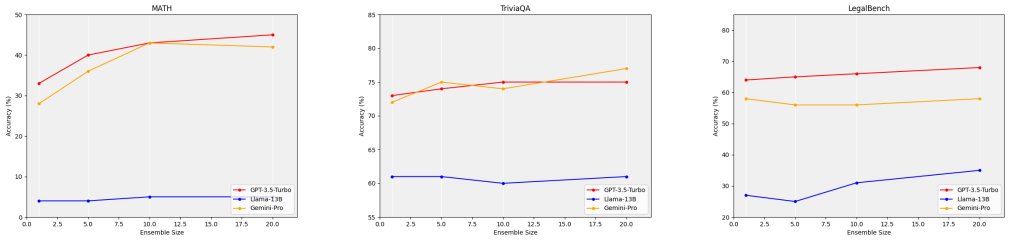

Figure 2: Baseline performance of varying ensemble sizes on MATH, Trivia QA, and Legal Bench for homogenous model ensembles

5 Experiments

5.1 Data

For arithmetic reasoning, we use the MATH dataset, which consists of 12,500 competition-level problems Hendrycks et al. (2021). For general QA, TriviaQA is chosen for its wide range of trivia questions and open-domain QA capabilities Joshi et al. (2017). It contains questions and associated evidence documents, assessing models on reading comprehension. In domain-specific QA, LegalBench will assess models on legal reasoning Guha et al. (2023). For all datasets, we sampled 100-200 QA pairs since we were constrained by the cost of making API calls and long inference time associated with multi-agent frameworks. Future work should consider scaling dataset size to investigate how the results generalize to a larger setting.

5.2 Evaluation method

For all of our datasets, we use accuracy as our evaluation metric. This is the standard used for all benchmarks. We also leave some opportunity for more qualitative, human-based evaluations where appropriate. For the MATH dataset, we directly compare our results with those reported in "More Agents Is All You Need," adopting their experimental settings and performance metrics to ensure an accurate and direct comparison. For the other datasets, TriviaQA and LegalBench, we use benchmarks and leaderboards from Papers with Code as our comparison points. This approach ensures that for general and domain-specific QA tasks, we are measuring our methodology against the most current and competitive standards in the field, providing a clear assessment of our multi-agent approach's effectiveness and innovation.

5.3 Experimental details

For all experiments, we set model temperature to 1 since we wanted non-deterministic outputs allowing our multi-agent framework to benefit from a diversity of responses. For our baselines, we ran the datasets on agent ensembles of the following sizes: 1, 5, 10, 20. With agent ensembles of more than one the final outputted answer was determined by simply selecting the answer output by the greatest number of agents. For duel model heterogeneous ensembles we evaluated agent ensembles of the following sizes: 4, 8, 16. We allocated $\frac{N}{2}$ agents of each model type. We decided to start at 4 since an ensemble of 2 would reduce down to a single model where the majority vote will always be the output of the first model in the ensemble. For the ensemble of all three models together we evaluated the following size ensembles: 3, 6, 9, 12, and 24.

For the TriviaQA dataset, we calculated the bleu score between all pairs of model outputs and selected the one with the highest total bleu score as the final output. Training time scales linearly with the number of agents in the experiment so there is not an exact number but it ranged roughly from thirty minutes to five hours per experiment. To that end, we utilized a thread pooling approach to speed up the execution of simultaneous queries. This was able to accelerate our experiment times anywhere from $2 - 10x$. However, due to the cost of such experiments, this was only able to speed up our experiment time by so much due to rate limiting issues.

5.4 Results

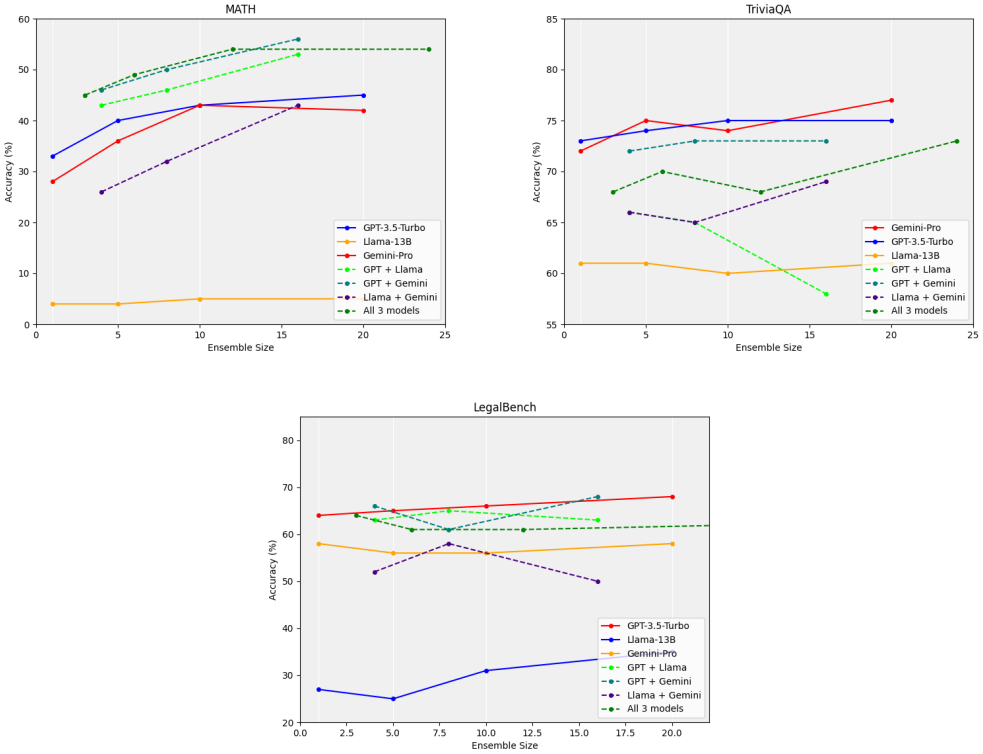

Figure 3: Comparison of heterogeneous ensembles (dashed) v baselines

Given the observations from the original paper by Li et al., we expected to see an overall increase in performance across all homogeneous model ensembles as we scaled the ensemble size. Indeed, this is generally what we observed. For example, on the MATH task, Llama's performance improved by 1% when scaling from 1 to 20 agents, while GPT 3.5 Turbo and Gemini saw gains of 12% and 14%, respectively. Interestingly, the effect of scaling varied slightly across different tasks. Llama benefited the most from scaling on the LegalBench task with a gain of 8%, whereas Gemini and GPT 3.5 Turbo had the greatest performance gains on the MATH task.

Moreover, our heterogeneous ensembling results were not entirely as expected. Notably, ensembling substantially improves performance on some tasks but not others. For instance, the heterogeneous ensembling approach underperformed the baselines established by the homogeneous ensembles on the TriviaQA dataset, while clearly outperforming the baselines (set by the homogeneous Gemini and GPT 3.5 ensembles) on the MATH task. Interestingly, while the performance gains plateau for the homogeneous ensembles on the MATH task, we observe that the performance gains for heterogeneous ensembles show less of the same patterns of plateauing.

Intriguingly, we see that any inclusion of a Llama model tends to decrease overall performance. This is most evident on the TriviaQA task, where the performance of the ensemble model with all three models actually performs worse than the ensemble of GPT 3.5 Turbo and Gemini alone. However, on the MATH task, combining GPT 3.5 Turbo and Llama outperforms the homogeneous ensemble of GPT 3.5 models, despite Llama's lower individual performance compared to the other models. While the combination of GPT 3.5 Turbo and Llama is still worse than the combination of GPT 3.5 and Gemini and the ensemble of all three, these results are surprising given the substantially worse performance of the homogeneous ensemble of Llama models on the same task and the overall negative effect of including Llama elsewhere.

6 Analysis

As we consider the data, we observe a few trends that align with our expectations based on the experiments of Li et al. Namely, as we scale the size of homogeneous ensembles, we are able to elicit marginal performance gains. Additionally, we see evidence that ensembling and scaling heterogeneous models can boost performance even further. However, including an underperforming model in our heterogeneous ensembles has a mixed effect on ensemble performance.

We posit that the variance in performance is largely due to the difficulty and type of task. For the LegalBench and TriviaQA tasks, the baselines for the more performant models (GPT 3.5 and Gemini) are already high, with scores of roughly 70% and nearly 80% on each task respectively. In contrast, the baseline scores for these models on the MATH task are much lower, hovering at around 40%. Ensembling models that already perform well on a task might have diminishing returns, as there is less room for improvement, and the models are more likely to agree on the correct answer. However, for a difficult task like MATH, where individual model performance is lower, introducing more agents can substantially boost performance. This is because the MATH task involves complex problem-solving, where the correct answer is often unique, while there are numerous possible incorrect answers. When ensembling models for the MATH task, the probability of multiple models converging on the same correct answer is higher than the probability of them agreeing on a specific incorrect answer, given the vast space of potential incorrect responses. As a result, ensembling is more likely to amplify the signal of the correct answer while reducing the noise of incorrect answers. In contrast, for tasks like TriviaQA and LegalBench, where the answers are more constrained and the individual model performance is higher, the models are more likely to agree on both correct and incorrect answers, leading to diminishing returns from ensembling. This phenomenon can be thought of as a form of "wisdom of the crowd," where the diversity of responses helps to mitigate individual model errors and improve overall performance, particularly for complex tasks with a large space of possible answers.

Our analysis further reveals several intriguing patterns across all tasks and agents. First, as ensemble size increases, the percentage of instances where at least one agent provides the correct answer monotonically increases. Second, the percentage of instances where all agents disagree with one another remains consistently low, near zero. This is crucial because constant disagreement among agents would render the ensembling strategy ineffective, with final answers essentially determined at random by our voting mechanism.

Most notably, we observe that for both homogeneous and heterogeneous ensembles, the percentage of instances where at least one agent provides the correct answer, but the majority votes for the incorrect answer, grows as the ensemble size increases. For example, when ensembling all three models on the LegalBench task, this percentage rises from 20% to 37% as the ensemble size grows from 3 to 24 agents. This finding suggests that there may be a more sophisticated approach to distilling the correct answer from an ensemble, beyond simple majority voting.

7 Conclusion

Our study demonstrates the potential benefits of ensembling and scaling both homogeneous and heterogeneous models for improving performance on a variety of tasks. However, the effectiveness of ensembling varies depending on the task difficulty and the individual performance of the models being ensembled. Our analysis highlights the need for further research into intelligent methods for distilling correct answers from ensembles, beyond simple majority voting.

One key finding is the increasing presence of correct answers within ensembles, even when the majority vote favors an incorrect answer. This observation suggests that future research should explore more sophisticated voting strategies to capitalize on this phenomenon. For more difficult tasks, closing the gap between the presence of correct answers and the majority vote could lead to substantial performance gains. Potential approaches might include the use of confidence scoring or leveraging model ensembling itself as a means for answer distillation. Just as diversity of opinion is important for answer generation, it may also prove to be an effective strategy for answer distillation. We propose this as a promising direction for future research.

As the number of language models continues to grow, with more companies developing unique and high-performing models, we see a constant improvement in individual model performance. However, there should be an increasing focus on studying how these models can be leveraged in conjunction with one another, rather than simply replacing one another. By exploring the potential of ensemble methods and developing innovative techniques for answer distillation, we can harness the collective intelligence of multiple models to solve increasingly complex problems. Our findings contribute to the growing body of work on ensemble learning in NLP and provide valuable insights for the design and application of ensemble models across different domains. As the field of natural language processing continues to evolve, leveraging the strengths of diverse models through collaboration may prove increasingly valuable to driving further advancements and unlocking the full potential of language models.

References

Hessa Abdulrahman Alawwad, Areej Alhothali, Usman Naseem, Ali Alkhathlan, and Amani Jamal. 2024. Enhancing textbook question answering task with large language models and retrieval augmented generation.

Nan Du, Yanping Huang, Andrew M. Dai, Simon Tong, Dmitry Lepikhin, Yuanzhong Xu, Maxim Krikun, Yanqi Zhou, Adams Wei Yu, Orhan Firat, Barret Zoph, Liam Fedus, Maarten Bosma, Zongwei Zhou, Tao Wang, Yu Emma Wang, Kellie Webster, Marie Pellat, Kevin Robinson, Kathleen Meier-Hellstern, Toju Duke, Lucas Dixon, Kun Zhang, Quoc V Le, Yonghui Wu, Zhifeng Chen, and Claire Cui. 2022. Glam: Efficient scaling of language models with mixture-of-experts.

Neel Guha, Julian Nyarko, Daniel E. Ho, Christopher Ré, Adam Chilton, Aditya Narayana, Alex Chohlas-Wood, Austin Peters, Brandon Waldon, Daniel N. Rockmore, Diego Zambrano, Dmitry Talisman, Enam Hoque, Faiz Surani, Frank Fagan, Galit Sarfaty, Gregory M. Dickinson, Haggai Porat, Jason Hegland, Jessica Wu, Joe Nudell, Joel Niklaus, John Nay, Jonathan H. Choi, Kevin Tobia, Margaret Hagan, Megan Ma, Michael Livermore, Nikon Rasumov-Rahe, Nils Holzenberger, Noam Kolt, Peter Henderson, Sean Rehaag, Sharad Goel, Shang Gao, Spencer Williams, Sunny Gandhi, Tom Zur, Varun Iyer, and Zehua Li. 2023. Legalbench: A collaboratively built benchmark for measuring legal reasoning in large language models.

Dan Hendrycks, Collin Burns, Saurav Kadavath, Akul Arora, Steven Basart, Eric Tang, Dawn Song, and Jacob Steinhardt. 2021. Measuring mathematical problem solving with the math dataset. *NeurIPS*.

Robert A. Jacobs, Michael I. Jordan, Steven J. Nowlan, and Geoffrey E. Hinton. 1991. Adaptive mixtures of local experts. *Neural Computation*, 3(1):79–87.

Dongfu Jiang, Xiang Ren, and Bill Yuchen Lin. 2023. Llm-blender: Ensembling large language models with pairwise ranking and generative fusion.

Mandar Joshi, Eunsol Choi, Daniel Weld, and Luke Zettlemoyer. 2017. triviaqa: A Large Scale Distantly Supervised Challenge Dataset for Reading Comprehension. *arXiv e-prints*, page arXiv:1705.03551.

Junyou Li, Qin Zhang, Yangbin Yu, Qiang Fu, and Deheng Ye. 2024. More agents is all you need.

Noam Shazeer, Azalia Mirhoseini, Krzysztof Maziarz, Andy Davis, Quoc Le, Geoffrey Hinton, and Jeff Dean. 2017. Outrageously large neural networks: The sparsely-gated mixture-of-experts layer.

Fanqi Wan, Xinting Huang, Deng Cai, Xiaojun Quan, Wei Bi, and Shuming Shi. 2024. Knowledge fusion of large language models.

Yutao Zhu, Huaying Yuan, Shuting Wang, Jiongnan Liu, Wenhan Liu, Chenlong Deng, Haonan Chen, Zhicheng Dou, and Ji-Rong Wen. 2024. Large language models for information retrieval: A survey.