Understanding Misunderstandings: Evaluating LLMs on Networking Questions

Mubashir Anwar
University of Illinois Urbana-Champaign, USA
manwar@illinois.edu

Matthew Caesar
University of Illinois Urbana-Champaign, USA
caesar@illinois.edu

ABSTRACT

Large Language Models (LLMs) have demonstrated impressive abilities in tackling tasks across numerous domains. The capabilities of LLMs could potentially be applied to various computer networking tasks, including network synthesis, management, debugging, security, and education. However, LLMs can be unreliable: they are prone to reasoning errors and may hallucinate incorrect information. Their effectiveness and limitations in computer networking tasks remain unclear. In this paper, we attempt to understand the capabilities and limitations of LLMs in network applications. We evaluate misunderstandings regarding networking related concepts across 3 LLMs over 500 questions. We assess the reliability, explainability, and stability of LLM responses to networking questions. Furthermore, we investigate errors made, analyzing their cause, detectability, effects, and potential mitigation strategies.

CCS CONCEPTS

• **Networks** → *Network management*;

KEYWORDS

Large Language Models, Computer Networking, Characterization Study

1 INTRODUCTION

Large Language Models (LLMs) have experienced rapid growth in recent years, showcasing remarkable capabilities in generating human-like text, answering questions, and making useful inferences across various tasks. These advancements have resulted in the widespread adoption of LLMs across numerous domains. Notably, ChatGPT, a chatbot based on Generative Pre-trained Transformer (GPT) models [6, 48], has emerged as the fastest-growing application of all time [29]. The success of LLMs has spurred interest and potential applications across numerous industry segments [7], with various academic domains demonstrating a variety of revolutionary potential use cases [47, 53, 55, 57].

The networking community is no exception. LLMs are already being considered for various applications within network engineering, including network configuration [45, 62], extracting protocol specifications [52], automating code generation for network management [43], and even networking tasks such as viewport prediction and adaptive bitrate streaming [67]. New use cases are likely to emerge in the future. LLMs have demonstrated exceptional performance in addressing a variety of challenges akin to those encountered in networking, including code generation [41], anomaly detection [55], root-cause analysis [11], and software verification [57]. While the integration of LLMs into networking applications appears inevitable, our present understanding of the potential and constraints of LLMs in addressing tasks related to computer networks is limited. Despite the impressive capabilities, LLMs are known to be far from perfect. They are prone to hallucinations [49], often contain inaccuracies in given facts [61], and frequently exhibit issues with reasoning [3]. To utilize the power of LLMs while avoiding such issues, understanding the capabilities and limitations of LLMs in network applications is crucial.

However, what does it mean for LLMs to be useful for networking applications? How do we evaluate their potential utility and harms? Perhaps we are interested in questions such as: do LLMs comprehend[1] networking jargon? Are they familiar with networking concepts? Can they reason through networking problems? In what topics do they excel, and where do they falter? And if they do make errors, how easily can these be detected? In this paper, we try to understand the capabilities and limitations of LLMs in network applications. We analyze the reliability, explainability, and stability of LLM responses to networking questions. Additionally, we analyze LLM misunderstandings, exploring their causes, detectability, effects, and viable mitigation strategies. We conduct a thorough examination of LLM answers to questions sourced from online courses and practice materials for network management certifications. We systematically analyze every incorrect response across multiple dimensions, utilizing a taxonomy we developed. Our analysis encompasses over 500 questions and involves three popular LLMs: GPT-3.5 [6], GPT-4 [48], and Claude 3 [1].

Some of our key findings are:

- Although LLMs can at times correctly answer even advanced networking questions, they frequently make very simple mistakes that humans are unlikely to make. Some of these mistakes can be detected by individuals with basic networking skills, but many incorrect answers are likely to go unnoticed.
- GPT-4 and GPT-3.5 struggle with certain topics, such as questions pertaining to IP addresses.
- Self-correction, which is prompting LLMs to analyze and refine their own answers, can improve the performance in some topics (e.g., handling IP addresses) but degrade it in others.
- Incorrect explanations by LLMs can cause serious misconceptions about networking concepts in the reader. More advanced LLMs can cause even more misconceptions because their hallucinations are more believable.
- We can mitigate some of the harms of LLMs through minimal human oversight, resulting in up to 15% increase in accuracy.

[1] LLMs lack human-like understanding and are often likened to “stochastic parrots” [4] as they are trained solely to predict the next word. Nonetheless, we employ anthropomorphizing terms for descriptive convenience.

- The model's confidence in its answers can be utilized to detect some errors, but this confidence is not well-calibrated across all types of errors and topics.
- Conceptual or informational recall errors represent the primary cause of inaccuracies across all LLMs. This highlights the potential for enhancing performance by pre-training LLMs on text specifically related to networking, laying a stronger foundational understanding of the domain.
- The responses provided by LLMs lack stability and can fluctuate with even minor variations in the prompt.

We have released our datasets and analysis [2]. We hope this study provides valuable insights to the industry regarding the feasibility of employing LLMs for network management, to equip researchers with practical guidance on leveraging LLMs for networking purposes, and to offer educators insights into the effective utilization of LLMs as educational aids.

2 RELATED WORK

In recent years, LLMs have shown impressive performance across various tasks, including answering medical queries [12, 34, 53], performing legal tasks [19, 47], solving problems in mathematics [16, 18, 66] and general sciences [2, 8], helping with code [41, 54], facilitating education [17, 63], performing anomaly detection [55], root-cause analysis [11], and software verification [57]. While there has been prior work to explore the usage of LLMs for network configuration [45, 62] and network operations [33, 44], to the best of our knowledge, there is currently no existing evaluation study focusing on general networking-related question and answers.

While most of the prior studies measure the performance of LLMs based on their accuracy (e.g., correct answers), some attempt to give a deeper analysis of the responses. Singhal *et al.* [53] employed manual analysis of LLM responses to medical queries to measure agreement with scientific and clinical consensus, likelihood and possible extent of harm, reading comprehension, and recall of relevant clinical knowledge. Wang *et al.* [61] used human analysis to assess the factual correctness of LLM responses across various datasets. Liang *et al.* [39] developed a taxonomy incorporating metrics such as fairness, bias, toxicity etc., to measure the capabilities and limitations of LLMs, employing different datasets for evaluating each metric. In our work, we categorize the performance of LLMs using metrics relevant to our target application—computer networking. This method of devising a taxonomy to understand performance is common in studies of software bugs [9, 10], and we employ it to analyze the errors made by LLMs in responding to networking questions. We use insights from studies involving misunderstandings in students on mathematical concepts [38, 46, 58, 60] and sciences [36, 56] to develop our methodology on understanding the cause of errors made by LLMs.

3 METHODOLOGY

To understand the capabilities and limitations of LLM in answering questions related to computer networking, we analyze answers from three different LLMs: *gpt-3.5-turbo-0125* (GPT-3.5), *gpt-4-1106-preview* (GPT-4), and *claude-3-opus-20240229* (Claude 3). We analyze the performance of these LLMs on practice questions for CCNA 200-301 exams [13] and online networking courses. Our questions set consists of 503 multiple-choice questions (MCQs)[3]. We categorize courses into three groups: basic [26, 59] (148 questions), advanced [24, 69] (123 questions), and Cisco certification-related [13, 14] courses[4] (232 questions). Additionally, we report results for questions from these courses that involve IP addresses (82 questions).

While most prior studies on evaluating LLMs across different applications focus solely on accuracy (i.e., percentage of correct answers), this metric alone offers only a shallow overview of the models' capabilities and limitations [5, 28, 30, 70]. To gain a comprehensive understanding of LLMs' potential, we perform a detailed examination by manually analyzing the explanations across various dimensions (Section 3.1). We provide explicit instructions to the LLMs, asking them to select the correct choice, provide explanations for their answers, cite relevant sources of information, and report their confidence level in each response. Subsequently, we analyze the answers, focusing on the incorrect answers to recognize the models' limitations and identify potential areas for improvement. Each answer was analyzed independently by at least two students with relevant knowledge and disagreements were resolved by a third student. We observed moderate agreement in the explainability category (Cohen's Kappa = 0.44) and substantial agreement in all other categories (Cohen's Kappa > 0.60). Additional details on the prompt, parameters, courses, and disagreement metrics are provided in the appendix.

We also use our results to derive strategies to improve the performance of LLMs on networking questions. We employ four strategies: self-correction [42], one-shot prompting [6], majority voting [35], and fine-tuning, all of which have demonstrated performance improvements in prior work in other domains. **Self-correction** involves LLMs refining their responses based on feedback to their previous outputs. **One-shot prompting** involves providing a single example of the task (in this case, answering networking questions) within the prompt. **Majority voting** involves using the answer that most models agree upon. In our case, we perform simple majority voting by selecting the answer that two of the three LLMs agree on[5]. **Fine-tuning** is a process where a pre-trained language model is further trained on a specific dataset to adapt it to a particular task or domain. Finally, we compare the accuracy of our chosen LLMs with that of smaller, open-source models.

3.1 Taxonomy

We analyze the performance of LLMs along the following dimensions:

1. Accuracy: Accuracy represents the correctness of answers provided by LLMs. It enables the quantification of LLM capabilities on answering questions, providing a crucial baseline for more comprehensive evaluations. Moreover, accuracy facilitates comparative analyses among various LLMs across diverse topics, offering insights into their relative strengths and weaknesses.

[2] https://github.com/mudbri/LLM-Network-Eval

[3] Employing MCQs to assess the performance of LLMs is a common approach [15, 32, 40] for question-answering benchmarks.

[4] Cisco certification-related courses are categorized separately because they focus on questions directly relevant to network operators.

[5] If there is no agreement on the answer, we select the answer provided by the best-performing LLM, which, in our case, was Claude 3.

Method: We allocate equal weight to each question and employ a "right-minus-wrong" grading methodology. Here, the score for each question is determined by subtracting the number of chosen incorrect answers from the number of chosen correct answers. This approach enables partial credit allocation, contributing to a more nuanced evaluation process.

2. Detectability: This pertains to how easily errors in LLM outputs can be identified. Considering the propensity of LLMs to generate misleading information that appears credible [49], evaluating detectability is crucial. Undetected errors can result in misconceptions about networking concepts and hinder efficient network management, leading to issues such as misconfigurations, misunderstandings of network architecture, and inaccurate debugging.
Method: We employ two different methods to evaluate error detectability. Firstly, we assess whether individuals with basic networking knowledge (e.g., college students) could identify errors in LLM answers by examining the accompanying explanations. Secondly, we analyze the confidence levels of LLMs in their responses by studying the likelihood of each token prediction within the provided answer choices. GPT-4 and GPT-3.5[6] provide log probabilities of every output token, indicating the likelihood of each token occurring in the sequence given the context, serving as a measure of the model's confidence in its output.

3. Cause: For every incorrect answer, we analyze the underlying causes of the misunderstanding. It is important to understand the reasons behind the errors made by LLMs. This entails evaluating their grasp of technical terminology and concepts pertinent to networking, as well as their ability to reason through networking problems. Understanding the causes not only highlights the limitations of LLMs but also sheds light on areas for improvement. Furthermore, it offers users valuable perspectives on the applicability of LLMs to their specific scenarios and suggests potential strategies for error mitigation.
Method: Here, we look at 3 steps of answering a question. (i) Understanding the question, (ii) Finding relevant facts/concepts necessary, and (iii) Reasoning through the relevant facts (building relevant connections between facts and question goals) to answer the question. We note where in that logical process a mistake is made. Furthermore, we conduct a finer-grained analysis to elaborate on the causes of misunderstandings, introducing additional categories to describe misinterpretations of questions and reasoning errors.

4. Explainability: Explainability refers to the quality of explanations provided by LLMs in support of their answers. Measuring the explainability of these models is crucial for practical applications. Since LLMs are anticipated to work alongside humans, it is imperative that they not only execute networking-related tasks effectively but also offer clear explanations for their decisions, preferably with credible sources for their information.
Method: We manually check whether the explanation effectively communicates why the answer is correct. Additionally, we check each source cited in the explanation, assessing (i) the authenticity of the sources (e.g., functional links, legitimate books), (ii) the nature of the sources (e.g., book, RFC, article, documentation), and (iii) the relevance of the sources to the question at hand.

5. Effects: This is to measure the impact of wrong answers on the users. We want to know if the explanation given by the LLM could cause conceptual misunderstandings. Seemingly reliable misinformation poses a significant danger, as inaccurate information can persist despite correction [25, 51], can impede skill acquisition [20], and even replace previously correct knowledge [23]. Understanding the possible effects of errors would help users identify when to use LLMs and how much to rely on them. For topics that could cause critical harm, reliance on LLMs at this stage should be low.
Method: For every wrong answer, we assess whether the explanation has the potential to induce a conceptual misunderstanding and identify the specific technical concept that could be misconstrued by users as a result of a misconception (i.e., the subtopic underlying the misunderstanding).

6. Stability: Stability is the ability to adapt to variations in tasks. It is crucial in networking, as consistent decision-making is essential for ensuring the comprehensibility and stability of networks. It is important that LLMs maintain stability and avoid yielding vastly different decisions in response to minor changes in prompts.
Method: To assess this, we prompt the LLMs to answer the same questions but rearrange the order of choices. We report the differences in scores compared to the original answers. A stable model should have very few differences, as changing the order of choices ideally should not alter the provided answers at all.

4 RESULTS

In this section, we present results obtained by applying our methodology to the answers provided by LLMs to networking questions.

4.1 Accuracy

Figure 1a illustrates[7] the accuracy of LLMs across different categories of questions. Overall, Claude 3 and GPT-4 demonstrate high accuracy in answering networking questions (89.6% and 88.7% respectively), whereas GPT-3.5 shows much lower accuracy (76.0%). For basic networking courses, all three LLMs achieve over 90% accuracy. However, accuracy drops notably for advanced questions, likely due to less exposure to advanced concepts in their training data compared to basic ones. When examining questions in the dataset that contain IP addresses, both GPT-based models perform notably worse, suggesting that they struggle to handle or interpret IP addresses correctly. Our analysis revealed that while LLMs often identify correct concepts and information, they may make small mistakes in applying those concepts. We labeled answers as inferable if a student with basic understanding in the field could likely deduce the correct answer from the explanation. With answer inference, GPT-3.5 demonstrates marked improvement in Cisco-related and IP-related questions (Figure 1b). This highlights that human involvement can substantially enhance the effectiveness of LLMs.

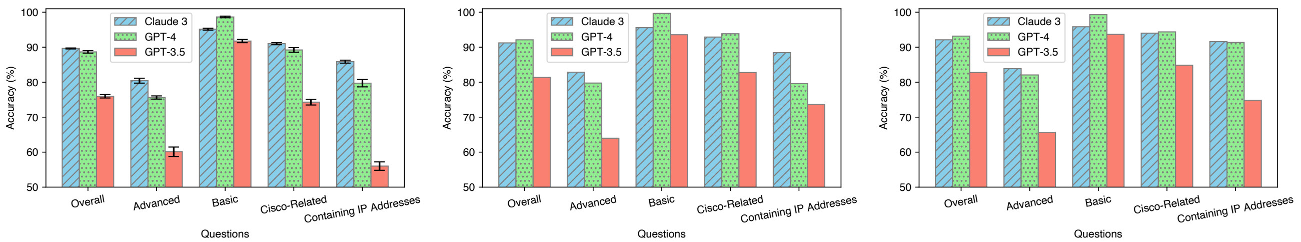
4.2 Detection

4.2.1 *Detection by humans.*

Figure 1c displays the accuracy of LLMs after filtering out answers identifiable as erroneous by individuals with a basic understanding of networking concepts. We notice that errors are usually more frequently detectable in GPT-4

[6] Claude 3 does not offer per-token probabilities.

[7] We present error bars to show the standard variation in accuracy across 5 runs and use results from a representative run for further analysis in the remainder of the paper.



(a) **Accuracy** (b) **Answers inferred from explanations** (c) **Excluding detectable errors**

Figure 1: Accuracy of LLMs across different categories of questions

and GPT-3.5 as compared to Claude 3. This is primarily because Claude 3 tends to make conceptual errors and provides convincing explanations, making error detection more challenging. Overall, the accuracy of all LLMs across all question types improves with human filtering of answers, indicating that human intervention can mitigate the potential harm caused by incorrect LLM answers.

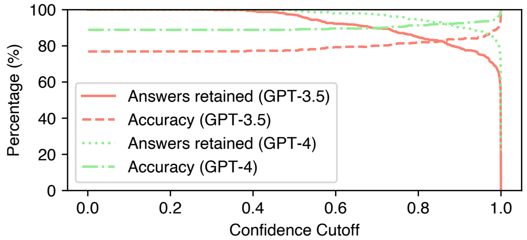


Figure 2: Percentage of answers retained and accuracy when discarding answers with model confidence below the cutoff

	GPT-3.5 (correct)	GPT-3.5 (incorrect)	GPT-4 (correct)	GPT-4 (incorrect)
Overall	0.960	0.845	0.988	0.889
Advanced	0.902	0.809	0.978	0.889
Basic	0.983	0.865	0.993	0.893
Cisco-Related	0.961	0.878	0.988	0.889
Containing IPs	0.913	0.886	0.985	0.887

Table 1: Average log probabilities, GPT's estimation of confidence of tokens, for correct and incorrect answers across different types of questions. Higher value means higher confidence

4.2.2 Machine Confidence. Another potential method to detect errors is to use the models' confidence[8] in their own answers. Table 1 illustrates that the model confidence is on average higher on correct answers than it is for incorrect answers. This finding suggests the possibility of establishing a confidence cutoff, below which answers are considered untrustworthy. As depicted in Figure 2, raising the cutoff increases model accuracy but reduces the number of answers deemed trustworthy. Selecting an appropriate cutoff, based on LLM performance and task accuracy requirements, can minimize undetected errors by LLMs. Furthermore, we also found that errors identifiable by humans show similar model confidence levels to those that are not, suggesting that humans can detect errors that using model confidence cutoffs might miss.

	GPT-3.5	GPT-4
Overall	0.169	0.086
Advanced	0.250	0.192
Basic	0.072	0.015
Cisco-Related	0.196	0.084
Containing IPs	0.349	0.167

Table 2: Expected Calibration Error (ECE) across different types of questions (lower is better)

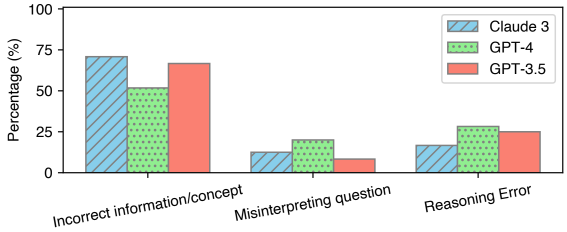


Figure 3: Causes of misunderstandings in incorrect answers

We also use the expected calibration error (ECE) [27], a metric commonly used to assess if the model's confidence in a prediction accurately reflects the probability of it being correct. This involves binning the LLM predictions by their confidence levels and measuring the average accuracy within each bin, weighted by the fraction of samples in each bin. The results in Table 2 show that overall, GPT-4 is better calibrated than GPT-3.5. However, for advanced courses and questions related to IP addresses, both GPT-3.5 and GPT-4 are poorly calibrated. This indicates that model confidence may not be a reliable indicator of accuracy for these types of questions.

4.3 Cause

Figure 3 shows the root causes of misunderstandings across the three LLMs. We observe that the majority of errors stem from either a failure to identify the relevant information/concepts or an incorrect recall of pertinent information/concepts, both categorized under "Incorrect information/concept". Moreover, LLMs demonstrate familiarity with most networking terminology, making only a few mistakes in interpreting questions. This suggests that training on more relevant networking data could enhance performance, as LLMs are likely to exhibit improved recall of pertinent information and concepts with relevant additional data. Further analysis of the causes of misunderstandings revealed that many of the reasoning errors made by LLMs are simple ones unlikely to be made by humans, such as selecting the incorrect choice despite providing a completely correct explanation. Moreover, a significant majority of errors made by GPT-based models in questions containing IP addresses stem

[8] In this section, we use the log probabilities provided by the LLMs for individual tokens of the answers as a measure of confidence. Additionally, we instructed the LLMs to verbalize their confidence level for each answer. However, this verbalized confidence was nearly always static, making it ineffective for error detection.

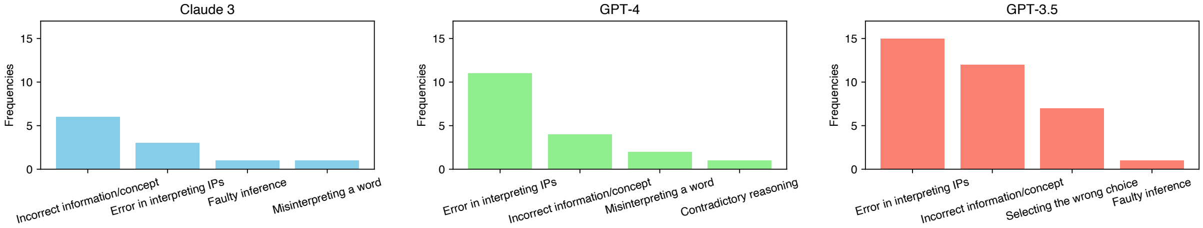


Figure 4: Top four most common errors, for each LLM, when answering questions involving IP addresses.

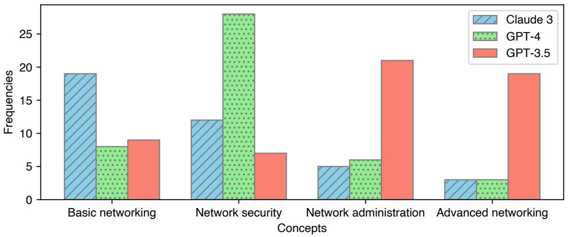


Figure 5: Frequency of misconceptions across different networking topics.

from misinterpretation of IPs (Figure 4). For instance, GPT-3.5 provided the following explanation for one of its incorrect answers, erroneously counting the number of "1s" in octets: *"The subnet mask 255.192.0.0 in binary is 11111111.11000000.00000000.00000000. This mask has 18 bits set to 1 (16 bits from the first two octets and 2 bits from the third octet)"*. These type of errors account for the relatively lower performance of GPT-based models on IP-related questions. These findings highlight the importance of targeted model training and tokenization on relevant networking data to improve LLM performance and show the need for careful evaluation of model capabilities, particularly for tasks involving IP addresses.

4.4 Effects

	GPT-3.5	GPT-4	Claude 3
Overall	56/141 (39.7%)	45/80 (56.3%)	39/71 (54.9%)
Advanced	31/64 (48.4)%	33/49 (67.4%)	25/39 (64.1%)
Basic	4/15 (26.7%)	1/3 (33.3%)	1/9 (11.1%)
Cisco-Related	21/62 (33.9%)	11/28 (39.3%)	13/23 (56.5%)
Containing IPs	10/36 (27.8%)	9/18 (50.0%)	8/12 (66.7%)

Table 3: Fraction of errors likely to cause misconceptions across different types of questions

Table 3 displays the proportion of errors potentially leading to misconceptions. Despite GPT-3.5 exhibiting lower overall performance, its explanations are relatively less prone to causing misconceptions compared to Claude 3 and GPT-4. This is primarily because GPT-3.5 generates less persuasive explanations for incorrect concepts and frequently commits easily identifiable errors. Additionally, the likelihood of causing misconceptions increases for advanced questions. This is often due to instances where LLMs fail to recall correct information or concepts, resulting in the generation of plausible yet erroneous hallucinations that may lead to misconceptions in readers. Figure 5 shows the topics of misconceptions for each LLM. Claude 3 tends to induce numerous misconceptions regarding basic networking concepts, whereas GPT-4 triggers significant misconceptions related to network security concepts. GPT-3.5 tends to elicit numerous misconceptions concerning topics in network administration and advanced networking. These findings highlight the need for careful selection of LLMs based on their tendency to cause misconceptions, especially for advanced and specialized topics.

4.5 Explainability

	GPT-3.5	GPT-4	Claude 3
Incorrect answers	141	80	71
Sources provided	59	196	155
Sources working	35	130	105
Sources relevant	23	99	89

Table 4: Details about sources provided in incorrect answers by LLMs

To assess the explainability of LLMs in answering networking-related questions, we analyzed whether the answers included sensible explanations for the selected choices. We found that GPT-4 provided good explanations 91.3% of the times, Claude 3 did so 81.7% of the time, and GPT-3.5 only 73.1% of the times. While this data is only for incorrect answers, it reveals significant differences in explainability across LLMs.

We also examine the sources each LLM cited to support their answers (Table 4). All LLMs cited reliable sources, with Claude 3 and GPT-4 even citing many research papers when answering questions related to more advanced concepts. GPT-3.5 offered few sources, many of which were either non-functional or irrelevant. In contrast, both Claude 3 and GPT-4 provided multiple sources per answer, with nearly two-thirds being functional and almost half relevant. GPT-4 and GPT-3.5 predominantly provided website links, while Claude 3 frequently referenced book chapters, which may be less convenient. These findings suggest that GPT-4 might be a more preferable choice for tasks requiring high explainability, such as LLM-based network management with human involvement or education.

4.6 Stability

We assessed the stability of the LLMs by evaluating their performance on the same questions with reordered answer choices. A stable model should not be affected by minor changes to the task description, such as different choice orders for the same MCQ. However, we observed that this simple change resulted in significant differences in the outputs of all LLMs. GPT-3.5, GPT-4, and Claude 3 showed differences in 14.5%, 9.0%, and 9.5% of answers, respectively[9]. This indicates that LLMs are not very stable. In the context of networking operations where consistency is required for a lot of tasks (e.g., configurations, querying network state, reacting to changes), the instability of LLMs renders them unreliable.

[9] We observed similar results even when we minimized the randomness in LLM output by setting the temperature parameter to its minimum value, suggesting that reducing the randomness of the model does not make it much more stable.

4.7 Improvement Strategies

	GPT-3.5	GPT-4	Claude 3
Overall	-0.7%	-0.3%	-6.3%
Advanced	-4.2%	+0.1%	-8.4%
Basic	-1.4%	-2.0%	-5.4%
Cisco-Related	+1.6%	+0.6%	-5.7%
Containing IPs	+12.0%	+4.9%	-6.5%

Table 5: Change in accuracy through self-correction

4.7.1 Self-correction. Table 5 shows the change in accuracy through self-correction. Overall, performance deteriorates for all LLMs. This is consistent with findings in [31]. Notably, Claude 3's performance declines across all question categories, whereas GPT-4 and GPT-3.5 show improvement for questions containing IP addresses. This suggests that while self-correction may generally reduce performance, it can enhance accuracy for certain question types.

	GPT-3.5	GPT-4	Claude 3
Overall	+0.6%	+1.1%	+0.8%
Advanced	+0.7%	+1.9%	+1.9%
Basic	+1.9%	+0.7%	+1.6%
Cisco-Related	-0.4%	+1.0%	-0.3%
Containing IPs	-2.9%	+1.1%	+0.1%

Table 6: Change in accuracy through one-shot prompting

4.7.2 One-Shot Prompting. Table 6 shows slight improvements in the overall performance of all three LLMs through one-shot prompting. The results suggest that while the improvement is not substantial, one-shot prompting can still provide a beneficial edge in certain scenarios. These findings indicate that further exploration of prompting strategies [64, 65] could potentially yield more significant performance gains for networking applications.

4.7.3 Majority Voting. Compared to the best-performing LLM, majority voting increased the overall performance by 0.4% but decreased the performance in every individual category. While majority voting might be a viable technique with a different set of LLMs and tasks, our results do not conclusively support its usage.

4.7.4 Fine-tuning. We fine-tuned GPT-3.5[10] on 1,155 networking question and answers that resembled our target dataset. However, this did not result in any noticeable improvements (see Figure 10 in the appendix). As [68] highlights, most knowledge and concepts in LLMs are learned during pre-training, suggesting fine-tuning may have limited impact on addressing deeper conceptual and reasoning issues. Further research is needed to assess the efficacy of fine-tuning for networking.

4.8 Open Source LLMs

We also compared the performance of our chosen LLMs with smaller models by evaluating three popular open-source models: Llama3.1 (*Meta-Llama-3.1-8B-Instruct*), Gemma2 (*gemma-2-9b-it*), and Mistral (*Mistral-7B-Instruct-v0.2*) in their default settings. While these models each have fewer than 10 billion parameters compared to GPT-3.5's 175 billion, Gemma2's performance is notably close to that of GPT-3.5. Despite this, GPT-3.5 still outperforms the smaller models in all categories (Figure 6).

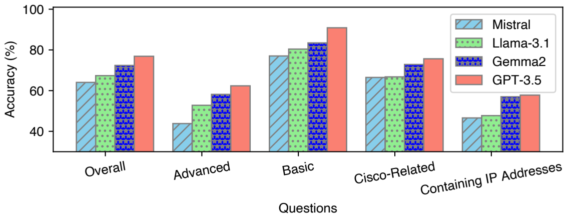


Figure 6: Comparison of the performance of open-source LLMs with GPT-3.5.

5 LIMITATIONS AND FUTURE WORK

In this section, we discuss some limitations of our study. Firstly, we had to manually analyze misunderstandings based on our taxonomy. While this is a common approach when dealing with unstructured data, such as in software engineering bug studies, our taxonomy might miss some useful dimensions. In the future, we aim to explore methods that enable automated categorization of misunderstandings. Secondly, there is a risk that LLMs may have encountered the answers of our test questions in their training datasets, leading to memorization. While this concern exists for most evaluation studies on LLMs, we minimize the impact of memorization by focusing on analyzing the errors. Lastly, we evaluated the models only on multiple-choice questions (MCQs), which might be simpler than real-world tasks. Although MCQs represent a controlled environment, our goal was to analyze the best-case scenario for LLMs. If LLMs struggle with certain topics even in MCQs, they are unlikely to perform well in real-world tasks related to those topics.

In the future, we aim to evaluate LLM performance on open-ended tasks and compare their results against those of experts to observe differences in approach and outcomes. We also intend to leverage multimodal LLMs to assess their understanding of networking-related diagrams (e.g., topologies) and explore Retrieval-Augmented Generation [37] to determine whether it enhances the ability to interpret network-specific text (e.g., logs, RFCs), thereby evaluating the real-world effectiveness of LLMs on networking tasks.

6 CONCLUSION

In this study, we conducted an in-depth analysis of the performance of three different LLMs on over 500 questions related to computer networking. We found that while some LLMs (GPT-4 and Claude 3) achieved above 88% accuracy on multiple-choice questions, they frequently made simple errors, which can lead to misconceptions about networking concepts. Furthermore, LLM outputs were not stable and could change with minor fluctuations in the input text. Our analysis revealed that GPT-4 and GPT-3.5 notably struggled with handling IP addresses. However, many of these errors are easily detectable, which can help minimize their impact. Even when errors occur, LLMs often provide relevant information that can be used to infer the correct answer. Finally, we discuss four strategies for improving LLM performance in networking tasks. While these strategies can improve performance on some topics, they may degrade it on others. Applied with caution, they can be effective.

[10] Support for fine-tuning *gpt-4-1106-preview* and *claude-3-opus-20240229* was not available at the time of writing.

REFERENCES

[1] Anthropic. 2024. Introducing the next generation of Claude. (Mar 2024). https://www.anthropic.com/news/claude-3-family

[2] Daman Arora, Himanshu Gaurav Singh, and Mausam. 2023. Have LLMs Advanced Enough? A Challenging Problem Solving Benchmark For Large Language Models. (2023). arXiv:cs.CL/2305.15074 https://arxiv.org/abs/2305.15074

[3] Yejin Bang, Samuel Cahyawijaya, Nayeon Lee, Wenliang Dai, Dan Su, Bryan Wilie, Holy Lovenia, Ziwei Ji, Tiezheng Yu, Willy Chung, Quyet V. Do, Yan Xu, and Pascale Fung. 2023. A Multitask, Multilingual, Multimodal Evaluation of ChatGPT on Reasoning, Hallucination, and Interactivity. (2023). arXiv:cs.CL/2302.04023 https://arxiv.org/abs/2302.04023

[4] Emily M Bender, Timnit Gebru, Angelina McMillan-Major, and Shmargaret Shmitchell. 2021. On the dangers of stochastic parrots: Can language models be too big?. In *Proceedings of the 2021 ACM conference on fairness, accountability, and transparency*. Association for Computing Machinery, Canada, 610–623.

[5] Marcel Binz and Eric Schulz. 2023. Using cognitive psychology to understand GPT-3. *Proceedings of the National Academy of Sciences* 120, 6 (2023), e2218523120.

[6] Tom B. Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, Sandhini Agarwal, Ariel Herbert-Voss, Gretchen Krueger, Tom Henighan, Rewon Child, Aditya Ramesh, Daniel M. Ziegler, Jeffrey Wu, Clemens Winter, Christopher Hesse, Mark Chen, Eric Sigler, Mateusz Litwin, Scott Gray, Benjamin Chess, Jack Clark, Christopher Berner, Sam McCandlish, Alec Radford, Ilya Sutskever, and Dario Amodei. 2020. Language Models are Few-Shot Learners. (2020). arXiv:cs.CL/2005.14165 https://arxiv.org/abs/2005.14165

[7] David Burch. 2023. Survey: Massive retooling around large language models underway. (Apr 2023). https://arize.com/blog/survey-massive-retooling-around-large-language-models-underway/

[8] Cayque Monteiro Castro Nascimento and André Silva Pimentel. 2023. Do Large Language Models Understand Chemistry? A Conversation with ChatGPT. *Journal of Chemical Information and Modeling* 63, 6 (2023), 1649–1655. https://doi.org/10.1021/acs.jcim.3c00285

[9] Gemma Catolino, Fabio Palomba, Andy Zaidman, and Filomena Ferrucci. 2019. Not all bugs are the same: Understanding, characterizing, and classifying bug types. *Journal of Systems and Software* 152 (2019), 165–181. https://doi.org/10.1016/j.jss.2019.03.002

[10] KS May Chan, Judith Bishop, Johan Steyn, Luciano Baresi, and Sam Guinea. 2009. A fault taxonomy for web service composition. In *Service-Oriented Computing-ICSOC 2007 Workshops: ICSOC 2007, International Workshops, Vienna, Austria, September 17, 2007, Revised Selected Papers 5.* Springer, Vienna, 363–375.

[11] Yinfang Chen, Huaibing Xie, Minghua Ma, Yu Kang, Xin Gao, Liu Shi, Yunjie Cao, Xuedong Gao, Hao Fan, Ming Wen, Jun Zeng, Supriyo Ghosh, Xuchao Zhang, Chaoyun Zhang, Qingwei Lin, Saravan Rajmohan, Dongmei Zhang, and Tianyin Xu. 2023. Automatic Root Cause Analysis via Large Language Models for Cloud Incidents. (2023). arXiv:cs.SE/2305.15778 https://arxiv.org/abs/2305.15778

[12] Joseph Chervenak, Harry Lieman, Miranda Blanco-Breindel, and Sangita Jindal. 2023. The promise and peril of using a large language model to obtain clinical information: ChatGPT performs strongly as a fertility counseling tool with limitations. *Fertility and Sterility* 120, 3, Part 2 (2023), 575–583. https://doi.org/10.1016/j.fertnstert.2023.05.151

[13] Cisco. 2024. Cisco Certified Network Associate. (Apr 2024). https://www.cisco.com/c/en/us/training-events/training-certifications/exams/current-list/ccna-200-301.html

[14] Cisco. 2024. Network Security. (May 2024). https://www.coursera.org/learn/network-security

[15] Peter Clark, Isaac Cowhey, Oren Etzioni, Tushar Khot, Ashish Sabharwal, Carissa Schoenick, and Oyvind Tafjord. 2018. Think you have Solved Question Answering? Try ARC, the AI2 Reasoning Challenge. (2018). arXiv:cs.AI/1803.05457 https://arxiv.org/abs/1803.05457

[16] Katherine M. Collins, Albert Q. Jiang, Simon Frieder, Lionel Wong, Miri Zilka, Umang Bhatt, Thomas Lukasiewicz, Yuhuai Wu, Joshua B. Tenenbaum, William Hart, Timothy Gowers, Wenda Li, Adrian Weller, and Mateja Jamnik. 2023. Evaluating Language Models for Mathematics through Interactions. (2023). arXiv:cs.LG/2306.01694 https://arxiv.org/abs/2306.01694

[17] Wei Dai, Jionghao Lin, Hua Jin, Tongguang Li, Yi-Shan Tsai, Dragan Gašević, and Guanliang Chen. 2023. Can Large Language Models Provide Feedback to Students? A Case Study on ChatGPT. In *2023 IEEE International Conference on Advanced Learning Technologies (ICALT)*. IEEE, Institute of Electrical and Electronics Engineers, Orem, 323–325. https://doi.org/10.1109/ICALT58122.2023.00100

[18] Xuan-Quy Dao and Ngoc-Bich Le. 2023. Investigating the Effectiveness of ChatGPT in Mathematical Reasoning and Problem Solving: Evidence from the Vietnamese National High School Graduation Examination. (2023). arXiv:cs.CL/2306.06331 https://arxiv.org/abs/2306.06331

[19] Aniket Deroy, Kripabandhu Ghosh, and Saptarshi Ghosh. 2023. How Ready are Pre-trained Abstractive Models and LLMs for Legal Case Judgement Summarization? (2023). arXiv:cs.CL/2306.01248 https://arxiv.org/abs/2306.01248

[20] Maeve G Donnelly and Amanda M Karsten. 2017. Effects of programmed teaching errors on acquisition and durability of self-care skills. *Journal of Applied Behavior Analysis* 50, 3 (2017), 511–528.

[21] examsdigest. 2024. Study, Learn, and Pass the CCNA 200-301. (May 2024). https://examsdigest.com/courses/cisco-ccna-200-301/

[22] ExamTopics. 2024. Cisco 200-301 Exam Actual Questions. (Sept 2024). https://www.examtopics.com/exams/cisco/200-301/view/1/

[23] Lisa K Fazio, Sarah J Barber, Suparna Rajaram, Peter A Ornstein, and Elizabeth J Marsh. 2013. Creating illusions of knowledge: learning errors that contradict prior knowledge. *Journal of Experimental Psychology: General* 142, 1 (2013), 1.

[24] Nick Feamster. 2024. Coursera - Online Courses and Credentials From Top Educators. Join for Free. (May 2024). https://www.coursera.org/learn/sdn/

[25] Zayba Ghazali-Mohammed. 2015. *Processing Inaccurate Information: Theoretical and Applied Perspectives from Cognitive Science and the Educational Sciences.* The MIT Press, Cambridge, MA.

[26] Google. 2024. The Bits and Bytes of Computer Networking. (May 2024). https://www.coursera.org/learn/computer-networking

[27] Chuan Guo, Geoff Pleiss, Yu Sun, and Kilian Q. Weinberger. 2017. On Calibration of Modern Neural Networks. (2017). arXiv:cs.LG/1706.04599 https://arxiv.org/abs/1706.04599

[28] Thilo Hagendorff, Ishita Dasgupta, Marcel Binz, Stephanie C. Y. Chan, Andrew Lampinen, Jane X. Wang, Zeynep Akata, and Eric Schulz. 2024. Machine Psychology. (2024). arXiv:cs.CL/2303.13988 https://arxiv.org/abs/2303.13988

[29] Krystal Hu. 2023. ChatGPT sets record for fastest-growing user base - analyst note. (Feb 2023). https://www.reuters.com/technology/chatgpt-sets-record-fastest-growing-user-base-analyst-note-2023-02-01/

[30] Jie Huang and Kevin Chen-Chuan Chang. 2023. Towards Reasoning in Large Language Models: A Survey. (2023). arXiv:cs.CL/2212.10403 https://arxiv.org/abs/2212.10403

[31] Jie Huang, Xinyun Chen, Swaroop Mishra, Huaixiu Steven Zheng, Adams Wei Yu, Xinying Song, and Denny Zhou. 2024. Large Language Models Cannot Self-Correct Reasoning Yet. (2024). arXiv:cs.CL/2310.01798 https://arxiv.org/abs/2310.01798

[32] Yuzhen Huang, Yuzhuo Bai, Zhihao Zhu, Junlei Zhang, Jinghan Zhang, Tangjun Su, Junteng Liu, Chuancheng Lv, Yikai Zhang, Jiayi Lei, Yao Fu, Maosong Sun, and Junxian He. 2023. C-Eval: A Multi-Level Multi-Discipline Chinese Evaluation Suite for Foundation Models. (2023). arXiv:cs.CL/2305.08322 https://arxiv.org/abs/2305.08322

[33] Yudong Huang, Hongyang Du, Xinyuan Zhang, Dusit Niyato, Jiawen Kang, Zehui Xiong, Shuo Wang, and Tao Huang. 2023. Large Language Models for Networking: Applications, Enabling Techniques, and Challenges. (2023). arXiv:cs.NI/2311.17474 https://arxiv.org/abs/2311.17474

[34] Israt Jahan, Md Tahmid Rahman Laskar, Chun Peng, and Jimmy Huang. 2023. Evaluation of ChatGPT on Biomedical Tasks: A Zero-Shot Comparison with Fine-Tuned Generative Transformers. (2023). arXiv:cs.CL/2306.04504 https://arxiv.org/abs/2306.04504

[35] Dongfu Jiang, Xiang Ren, and Bill Yuchen Lin. 2023. LLM-Blender: Ensembling Large Language Models with Pairwise Ranking and Generative Fusion. (2023). arXiv:cs.CL/2306.02561 https://arxiv.org/abs/2306.02561

[36] Cazembe Kennedy, Aubrey Lawson, Yvon Feaster, and Eileen Kraemer. 2020. Misconception-Based Peer Feedback: A Pedagogical Technique for Reducing Misconceptions. In *Proceedings of the 2020 ACM Conference on Innovation and Technology in Computer Science Education (ITiCSE '20).* Association for Computing Machinery, New York, NY, USA, 166–172. https://doi.org/10.1145/3341525.3387392

[37] 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. (2021). arXiv:cs.CL/2005.11401 https://arxiv.org/abs/2005.11401

[38] Xiaobao Li. 2010. *Cognitive analysis of students' errors and misconceptions in variables, equations, and functions.* Ph.D. Dissertation. Texas A & M University.

[39] Percy Liang, Rishi Bommasani, Tony Lee, Dimitris Tsipras, Dilara Soylu, Michihiro Yasunaga, Yian Zhang, Deepak Narayanan, Yuhuai Wu, Ananya Kumar, Benjamin Newman, Binhang Yuan, Bobby Yan, Ce Zhang, Christian Cosgrove, Christopher D. Manning, Christopher Ré, Diana Acosta-Navas, Drew A. Hudson, Eric Zelikman, Esin Durmus, Faisal Ladhak, Frieda Rong, Hongyu Ren, Huaxiu Yao, Jue Wang, Keshav Santhanam, Laurel Orr, Lucia Zheng, Mert Yuksekgonul, Mirac Suzgun, Nathan Kim, Neel Guha, Niladri Chatterji, Omar Khattab, Peter Henderson, Qian Huang, Ryan Chi, Sang Michael Xie, Shibani Santurkar, Surya Ganguli, Tatsunori Hashimoto, Thomas Icard, Tianyi Zhang, Vishrav Chaudhary, William Wang, Xuechen Li, Yifan Mai, Yuhui Zhang, and Yuta Koreeda. 2023. Holistic Evaluation of Language Models. (2023). arXiv:cs.CL/2211.09110 https://arxiv.org/abs/2211.09110

[40] Chuang Liu, Renren Jin, Yuqi Ren, Linhao Yu, Tianyu Dong, Xiaohan Peng, Shuting Zhang, Jianxiang Peng, Peiyi Zhang, Qingqing Lyu, Xiaowen Su, Qun Liu, and Deyi Xiong. 2023. M3KE: A Massive Multi-Level Multi-Subject Knowledge Evaluation Benchmark for Chinese Large Language Models. (2023). arXiv:cs.CL/2305.10263 https://arxiv.org/abs/2305.10263

[41] Jiawei Liu, Chunqiu Steven Xia, Yuyao Wang, and Lingming Zhang. 2023. Is Your Code Generated by ChatGPT Really Correct? Rigorous Evaluation of Large Language Models for Code Generation. (2023). arXiv:cs.SE/2305.01210 https://arxiv.org/abs/2305.01210

[42] Aman Madaan, Niket Tandon, Prakhar Gupta, Skyler Hallinan, Luyu Gao, Sarah Wiegreffe, Uri Alon, Nouha Dziri, Shrimai Prabhumoye, Yiming Yang, Shashank Gupta, Bodhisattwa Prasad Majumder, Katherine Hermann, Sean Welleck, Amir Yazdanbakhsh, and Peter Clark. 2023. Self-Refine: Iterative Refinement with Self-Feedback. (2023). arXiv:cs.CL/2303.17651 https://arxiv.org/abs/2303.17651

[43] Sathiya Kumaran Mani, Yajie Zhou, Kevin Hsieh, Santiago Segarra, Ranveer Chandra, and Srikanth Kandula. 2023. Enhancing Network Management Using Code Generated by Large Language Models. (2023). arXiv:cs.NI/2308.06261 https://arxiv.org/abs/2308.06261

[44] Yukai Miao, Yu Bai, Li Chen, Dan Li, Haifeng Sun, Xizheng Wang, Ziqiu Luo, Yanyu Ren, Dapeng Sun, Xiuting Xu, Qi Zhang, Chao Xiang, and Xinchi Li. 2023. An Empirical Study of NetOps Capability of Pre-Trained Large Language Models. (2023). arXiv:cs.CL/2309.05557 https://arxiv.org/abs/2309.05557

[45] Rajdeep Mondal, Alan Tang, Ryan Beckett, Todd Millstein, and George Varghese. 2023. What do LLMs need to Synthesize Correct Router Configurations? (2023). arXiv:cs.NI/2307.04945 https://arxiv.org/abs/2307.04945

[46] Nitsa Movshovitz-Hadar, Orit Zaslavsky, and Shlomo Inbar. 1987. An empirical classification model for errors in high school mathematics. *Journal for research in mathematics Education* 18, 1 (1987), 3–14.

[47] John J. Nay, David Karamardian, Sarah B. Lawsky, Wenting Tao, Meghana Bhat, Raghav Jain, Aaron Travis Lee, Jonathan H. Choi, and Jungo Kasai. 2023. Large Language Models as Tax Attorneys: A Case Study in Legal Capabilities Emergence. (2023). arXiv:cs.CL/2306.07075 https://arxiv.org/abs/2306.07075

[48] OpenAI. 2024. GPT-4 Technical Report. (2024). arXiv:cs.CL/2303.08774 https://arxiv.org/abs/2303.08774

[49] Vipula Rawte, Amit Sheth, and Amitava Das. 2023. A Survey of Hallucination in Large Foundation Models. (2023). arXiv:cs.AI/2309.05922 https://arxiv.org/abs/2309.05922

[50] Sanfoundry. 2024. Computer Networks Questions and Answers. (Sept 2024). https://www.sanfoundry.com/computer-networks-mcqs-basics/

[51] Colleen M. Seifert. 2014. The Continued Influence Effect: The Persistence of Misinformation in Memory and Reasoning Following Correction. In *Processing Inaccurate Information: Theoretical and Applied Perspectives from Cognitive Science and the Educational Sciences*. The MIT Press, Cambridge, MA. https://doi.org/10.7551/mitpress/9737.003.0006 arXiv:https://direct.mit.edu/book/chapter-pdf/2267378/9780262325646_cac.pdf

[52] Prakhar Sharma and Vinod Yegneswaran. 2023. PROSPER: Extracting Protocol Specifications Using Large Language Models. In *Proceedings of the 22nd ACM Workshop on Hot Topics in Networks*. ACM, Cambridge, MA, 41–47.

[53] Karan Singhal, Shekoofeh Azizi, Tao Tu, S. Sara Mahdavi, Jason Wei, Hyung Won Chung, Nathan Scales, Ajay Tanwani, Heather Cole-Lewis, Stephen Pfohl, Perry Payne, Martin Seneviratne, Paul Gamble, Chris Kelly, Nathaneal Scharli, Aakanksha Chowdhery, Philip Mansfield, Blaise Aguera y Arcas, Dale Webster, Greg S. Corrado, Yossi Matias, Katherine Chou, Juraj Gottweis, Nenad Tomasev, Yun Liu, Alvin Rajkomar, Joelle Barral, Christopher Semturs, Alan Karthikesalingam, and Vivek Natarajan. 2022. Large Language Models Encode Clinical Knowledge. (2022). arXiv:cs.CL/2212.13138 https://arxiv.org/abs/2212.13138

[54] Giriprasad Sridhara, Ranjani H. G., and Sourav Mazumdar. 2023. ChatGPT: A Study on its Utility for Ubiquitous Software Engineering Tasks. (2023). arXiv:cs.SE/2305.16837 https://arxiv.org/abs/2305.16837

[55] Jing Su, Chufeng Jiang, Xin Jin, Yuxin Qiao, Tingsong Xiao, Hongda Ma, Rong Wei, Zhi Jing, Jiajun Xu, and Junhong Lin. 2024. Large Language Models for Forecasting and Anomaly Detection: A Systematic Literature Review. (2024). arXiv:cs.LG/2402.10350 https://arxiv.org/abs/2402.10350

[56] Kimberly Tanner and Deborah Allen. 2005. Approaches to biology teaching and learning: understanding the wrong answers—teaching toward conceptual change. *Cell biology education* 4, 2 (2005), 112–117.

[57] Norbert Tihanyi, Ridhi Jain, Yiannis Charalambous, Mohamed Amine Ferrag, Youcheng Sun, and Lucas C. Cordeiro. 2024. A New Era in Software Security: Towards Self-Healing Software via Large Language Models and Formal Verification. (2024). arXiv:cs.SE/2305.14752 https://arxiv.org/abs/2305.14752

[58] Ali Türkdoğan and Adnan Baki. 2013. Classification of middle school students' mistakes: mistake types. *Ankara University Journal of Faculty of Educational Sciences (JFES)* 46, 1 (2013), 67–88.

[59] Kevin Vaccaro. 2024. Computer Networking. (May 2024). https://www.coursera.org/learn/illinois-tech-computer-networking

[60] AKHN Veloo, Hariharan N Krishnasamy, and Wan Shahida Wan Abdullah. 2015. Types of student errors in mathematical symbols, graphs and problem-solving. *Asian Social Science* 11, 15 (2015), 324–334.

[61] Cunxiang Wang, Sirui Cheng, Qipeng Guo, Yuanhao Yue, Bowen Ding, Zhikun Xu, Yidong Wang, Xiangkun Hu, Zheng Zhang, and Yue Zhang. 2023. Evaluating Open-QA Evaluation. (2023). arXiv:cs.CL/2305.12421 https://arxiv.org/abs/2305.12421

[62] Changjie Wang, Mariano Scazzariello, Alireza Farshin, Simone Ferlin, Dejan Kostić, and Marco Chiesa. 2024. NetConfEval: Can LLMs Facilitate Network Configuration? *Proc. ACM Netw.* 2, CoNEXT2, Article 7 (jun 2024), 25 pages. https://doi.org/10.1145/3656296

[63] Rose E. Wang and Dorottya Demszky. 2023. Is ChatGPT a Good Teacher Coach? Measuring Zero-Shot Performance For Scoring and Providing Actionable Insights on Classroom Instruction. (2023). arXiv:cs.CL/2306.03090 https://arxiv.org/abs/2306.03090

[64] Xuezhi Wang, Jason Wei, Dale Schuurmans, Quoc Le, Ed Chi, Sharan Narang, Aakanksha Chowdhery, and Denny Zhou. 2023. Self-Consistency Improves Chain of Thought Reasoning in Language Models. (2023). arXiv:cs.CL/2203.11171 https://arxiv.org/abs/2203.11171

[65] Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, Brian Ichter, Fei Xia, Ed Chi, Quoc Le, and Denny Zhou. 2023. Chain-of-Thought Prompting Elicits Reasoning in Large Language Models. (2023). arXiv:cs.CL/2201.11903 https://arxiv.org/abs/2201.11903

[66] Tianwen Wei, Jian Luan, Wei Liu, Shuang Dong, and Bin Wang. 2023. CMATH: Can Your Language Model Pass Chinese Elementary School Math Test? (2023). arXiv:cs.CL/2306.16636 https://arxiv.org/abs/2306.16636

[67] Duo Wu, Xianda Wang, Yaqi Qiao, Zhi Wang, Junchen Jiang, Shuguang Cui, and Fangxin Wang. 2024. NetLLM: Adapting Large Language Models for Networking. In *Proceedings of the ACM SIGCOMM 2024 Conference (ACM SIGCOMM '24)*. ACM, Sydney, 661–678. https://doi.org/10.1145/3651890.3672268

[68] Chunting Zhou, Pengfei Liu, Puxin Xu, Srini Iyer, Jiao Sun, Yuning Mao, Xuezhe Ma, Avia Efrat, Ping Yu, Lili Yu, Susan Zhang, Gargi Ghosh, Mike Lewis, Luke Zettlemoyer, and Omer Levy. 2023. LIMA: Less Is More for Alignment. (2023). arXiv:cs.CL/2305.11206 https://arxiv.org/abs/2305.11206

[69] Xiaobo Zhou. 2024. TCP/IP and Advanced Topics. (May 2024). https://www.coursera.org/learn/tcp-ip-advanced

[70] Yan Zhuang, Qi Liu, Yuting Ning, Weizhe Huang, Zachary A. Pardos, Patrick C. Kyllonen, Jiyun Zu, Qingyang Mao, Rui Lv, Zhenya Huang, Guanhao Zhao, Zheng Zhang, Shijin Wang, and Enhong Chen. 2024. From Static Benchmarks to Adaptive Testing: Psychometrics in AI Evaluation. (2024). arXiv:cs.CL/2306.10512 https://arxiv.org/abs/2306.10512

A PROMPTS AND SETTINGS

A.1 Prompt

We used the prompt given in Figure 7 for evaluating LLMs.

> As a virtual assistant specializing in computer networking, your task is to analyze and respond to multiple-choice questions from a course related to *[course-name]*. For each question, you are to identify the correct option(s) from the given choices. Your response should include:
>
> - The selected option(s) using their corresponding letter(s) in lowercase (e.g., a, b, c, d) or combination thereof if multiple selections are correct. Separate multiple selections with commas (e.g., a, c).
> - A concise explanation with reasons for your answer.
> - An estimated confidence level in your answer, expressed as a decimal between 0 and 1.
> - List of reputable sources of any fact, concept, or information used when answering the question. You can include multiple sources here.
>
> Format your response in JSON with keys for "answer", "explanation", "confidence", and "sources". Ensure the information provided is accurate, up-to-date as of your last training data, and clearly articulated for educational purposes.

Figure 7: Prompt for evaluating networking questions

We added the course name in the prompt for each course that we evaluated on (e.g., "Software Defined Networking", "Cisco CCNA examinations"). We requested the LLM to output in JSON format to simplify extracting the data. For one-shot prompting, we added the line *"When answering user questions follow this example:"* followed by an example question and its corresponding correct answer.

For self-correction, we used the prompt shown in Figure 8 followed by the original question and the answer given by the LLM.

> You will be given a multiple-choice question and answer related to computer networking. Assume that this answer could be either correct or incorrect. Review the answer carefully and report any serious problems you find. Make sure to evaluate the original answer. Please provide your responses in the following format:
> - Evaluation of original answer: (Detailed evaluation)
> - Correct Answer: (choices). The selected option(s) should only use their corresponding letter(s) in lowercase (e.g., a, b, c, d) or combination thereof if multiple selections are correct. Separate multiple selections with commas (e.g., a, c)

Figure 8: Prompt for self-correcting answers

A.2 Settings

We used the following settings to run LLMs:

- **temperature:** 0.5
- **max_tokens:** 500
- **logprobs:** True
- **seed:** 123
- **stop_sequences**=["}"]

Note that *logprobs* and *seed* were only set for GPT-4 and GPT-3.5, since Claude does not support these options. The *seed* parameter was set to ensure the output is as deterministic as possible, aiding in reproducibility. Enabling the *logprobs* parameter allows the LLM to return the confidence value for each generated token. The *stop_sequences* parameter was set in Claude 3 to ensure it stops generating text once the JSON output is complete. The *temperature* was set to 0.5 to balance creativity with staying on context. This value was determined through manual testing of different temperatures. *max_tokens* was set to 500 to prevent excessively long responses.

For fine-tuning GPT-3.5, we used the following settings:

- **Epochs:** 3
- **Batch size:** 2 for more than 1000 examples, 1 otherwise
- **Learning Rate Multiplier:** 2

B QUESTIONS

We used practice questions from six sources:

1. Cisco CCNA 200-301 Practice Tests by examsdigest [21] - categorized under Cisco-related course (197 questions)
2. Network Security [14] from Cisco's Cybersecurity Operations Fundamentals Specialization on Coursera - categorized under Cisco-related course (35 questions)
3. Software Defined Networking [24] (University of Chicago) from Coursera - categorized under advanced course (96 questions)
4. TCP/IP and Advanced Topics [69] (University of Colorado) from Coursera - categorized under advanced course (27 questions)
5. The Bits and Bytes of Computer Networking [26] from Google's IT Support Professional Certificate on Coursera - categorized under basic course (106 questions)
6. Computer Networking [59] (Illinois Tech) from Coursera - categorized under basic course (42 questions)

We excluded questions from these courses that had errors in the answers or questions, were overly specific to the course material (e.g., questions referencing a specific lecture), or referenced visual context (e.g. topology diagrams). For fine-tuning, we used practice questions for Cisco CCNA 200-301 [22] and general computer networking questions [50].

C ADDITIONAL RESULTS

We provide supplementary results that expand on our findings here. Figure 9 illustrates the types of sources cited by each LLM. Table 7 shows the overlap and discrepancies in correct and incorrect answers between the original questions and the reordered questions. Table 8 shows brief description of all categories that were analyzed by multiple students along with the corresponding Kappa score. Sources were analyzed by a single person. Table 9 illustrates the accuracy improvements achieved through majority voting compared to the original accuracy. It is important to note that the gains observed with GPT-3.5 are primarily attributable to the majority voting process favoring answers provided by GPT-4 and Claude-3.

	✓	X
Reordered ✓	317	28
Reordered X	45	113

(a) **GPT-3.5**

	✓	X
Reordered ✓	400	22
Reordered X	23	58

(b) **GPT-4**

	✓	X
Reordered ✓	406	22
Reordered X	26	49

(c) **Claude 3**

Table 7: Comparison of correct (✓) and incorrect (X) answers between original questions and reordered questions.

	Description	**Cohen's Kappa**
Misunderstanding (General)	General cause of misunderstanding by the LLM (wrong facts/concept, misinterpreting question, incorrect reasoning)	0.8137
Misunderstanding (Reasons)	Specific reasons for misunderstanding (e.g., misinterpreting a word, faulty inference, incorrect choice selected)	0.6751
Inferable	Can the correct answer be inferred from the explanation?	0.6342
Explainability	Was the answer justified with relevant explanations?	0.4409
Conceptual Error	Can the explanation cause a conceptual error in the reader?	0.6106
Detection	Can the reader detect that the given answer is incorrect by reading the explanation?	0.6360

Table 8: Descriptions of categories analyzed by multiple reviewers along with the Cohen's Kappa score

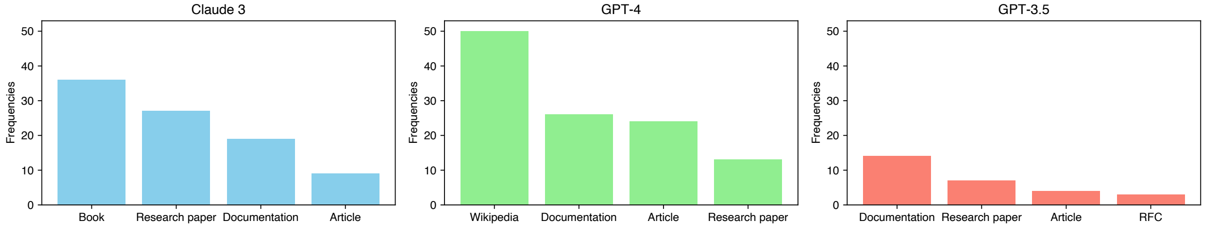


Figure 9: Top four most common types of sources, for each LLM, referenced in answers

	GPT-3.5	GPT-4	Claude 3
Overall	+13.2%	+1.2%	+0.4%
Advanced	+18.3%	+4.5%	-0.4%
Basic	+7.1%	-1.0%	+3.0%
Cisco-Related	+13.3%	+0.8%	-0.8%
Containing IPs	+25.7%	+4.0%	-2.4%

Table 9: Change in accuracy from majority voting compared to the original accuracy of the LLMs.

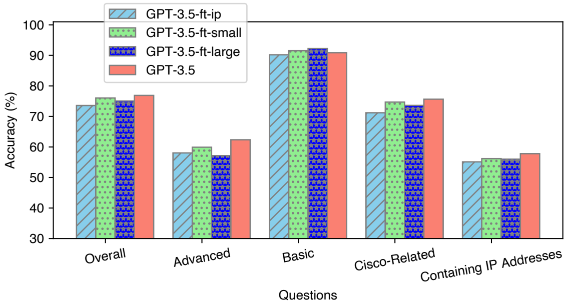


Figure 10: Comparison of the performance of fine-tuned LLMs with GPT-3.5.

Figure 10 shows the performance of base GPT-3.5 compared with fine-tuned versions of GPT-3.5. *GPT-3.5-ft-large* is the fine-tuned model trained on 1155 examples. *GPT-3.5-ft-small* is the fine-tuned model trained on 102 examples. *GPT-3.5-ft-ip* is the fine-tuned model trained on 25 examples of questions related to IP addresses. Notably, fine-tuning did not yield significant improvements, even for questions involving IP addresses.