Search Query Rephrasing Using GPT-3

Gordon Kamer
Harvard University

Abstract

Simple rules for finding matching pages from a query often fail to understand what the user is attempting to find. By contrast, large language models (LLMs) show deep semantic understanding. As a result, researchers have proposed using LLMs in the search context. A common approach is to use an embedding generated by the model and then to generate similarity scores with potential results' embeddings. Like other methods, it is difficult to implement and scale. Moreover, it does not take advantage of LLMs' potential to understand not only what the query is formally asking but also what the user is likely looking for. I propose a simpler method to leverage LLMs in search: allow the LLM to rephrase the query a few times; perform each search independently; and return a reranking of results. I find that in a small-scale user study of this method, the aggregated results are preferred to the original query by a small margin that is not statistically significant ($p=0.247$). A qualitative assessment of the rephrased queries shows that GPT-3 is successful at generating the intended rephrasings. More work is needed to improve and determine the efficacy of this approach, such as by evaluating the method on less powerful search engines.

1 Introduction

Large language models (LLMs) show deep language understanding [1], [2]. As a result, LLMs have been used to improve search [3]–[5]. Since they can understand queries, they might be able to better match the queries with search results than traditional methods. A standard approach relies on matching LLM embeddings of the query with embeddings of the possible search results [3]–[7]. This technique works well

but introduces other problems - for one, it may require the LLM to be fed paragraphs of text from every potential match in the search.[1] I propose a simpler technique: ask the language model to rephrase an original query a few times and return an aggregated ranking of results. Many search results should be invariant to non-semantic changes in the query, such as when a word is swapped with a synonym or when ancillary words are removed. The theory is that results that are robust to these changes should be ranked more highly. The language model may also improve the query by searching for useful documents that the user did not consider asking for. This technique can be easily integrated into existing search fields without significant changes, and it requires the language model to be fed only the query rather than the results as well. The system could be a supplement to other machine learning-based approaches, or it could be an easy way to improve otherwise unsophisticated search tools within websites.

I prototyped this approach and tested it on a variety of plausible Google search queries. I then conducted a small user study asking users to compare the original Google results with the re-ranked results. Users preferred the aggregated results slightly more frequently than the original results. This preference is not statistically significant. More work is needed to determine: rules to find queries that are most susceptible to improvement by rephrasing; optimal language model prompts; and optimal re-ranking algorithms. Future work might also seek to test the system on less advanced search fields than Google where there is more room for improvement.

Section (2) explains the aspects of large language models that are relevant to prior work and the rest of the paper, including a short discussion of their capabilities,

[1] That particular problem is faced by *Elicit*, an academic search engine, which feeds paper abstracts to GPT-3 in an online fashion.

how they function generally, and the details of how query embeddings are generated. Section (3) reviews previous work that has sought to apply language models to search. Section (4) describes how the language model rephrases queries in this system and describes the experiments, the results of which are presented in section (5). A discussion and analysis of the results are presented in section (6). Limitations and future work appear in section (7). The code for the entire project can be found on GitHub, at https://github.com/gkamer8/gpt-rephrasing.

2 Large Language Models

A language model seeks to output a probability distribution over text. Recent language models are typically large, overparameterized neural networks with Transformer architectures. First, I will describe briefly the considerations that have led to the dominance of Transformers and relate those qualities to our task of improving search queries. Second, I will give an overview of the Transformer architecture so that the later discussion of prior work and the methodological choices of this paper can be understood.

In order to achieve human level performance at language modeling, the model must have some understanding of the world and the meaning of the words it is manipulating. Consider the sentence, “He didn’t put the trophy into the suitcase because the [MASK] was too big”. The task is to predict what word the “[MASK]” replaced. To every human, it is obvious that the sentence should read, “...because the *trophy* was too big.” Moreover, if the words “too big” were replaced with “too small”, it is

obvious that the sentence should read, "...because the *suitcase* was too small". To arrive at that conclusion, a model might have to have some understanding of, among other things, what putting something into a suitcase means and how large a typical trophy is compared to a suitcase. These sentences are examples of Winograd schema problems.[2]

Many previous approaches have failed to model the kind of knowledge demonstrated by the previous example, such as LSTMs [10] and CNNs [11]. Transformer models, surprisingly to its creators[3], *are often successful* at modeling Winograd schema problems. This deep understanding improves with scale [13].

This kind of understanding can be useful for search. Search engine users, even those who are computer literate, rarely use the advanced search options present in most systems. Their queries are also often only a few words [14]. Natural language queries thus ask a lot of the search engine. It must identify synonyms of the few keywords in the query to find all of the relevant matches. It must also learn certain dependencies between words in the query: for example, "phone not apple" should rank Samsung's website above Apple's. In this paper, we also explore the possibility that the search engine could understand what the user is intending to find rather than what the user actually searched for. For example, a user asking for programming help might want to ask for a code snippet rather than just an explanation of a problem.

This paper uses a Transformer model, GPT-3, to improve search queries. The Transformer architecture takes a sequence as input and outputs a sequence of equal

[2] The first cited example of this kind appeared in [8]. The Winograd Schema Challenge - in which computers are asked to disambiguate sentences of this nature - has been proposed as a replacement to the Turing test [9].

[3] The author recommends viewing [12] in the reader's spare time, in which a co-creator of the Transformer explains their paper, gives the "trophy"/"suitcase" Winograd example, and states their surprise at the model's success.

length. It usually suffices to pretend that the sequences are composed of words, though they are technically “tokens”.[4] Each token is projected into a d-dimensional vector, which is called an embedding. Ideally, the vectors are semantically meaningful (e.g., “king” - “man” + “woman” = “queen”, as in [16]). In a Transformer, each embedding is slowly altered, mostly independently[5], into a new sequence of embeddings. In the case of an autoregressive Transformer, the resulting embeddings are meant to represent the next word in the sequence; thus the outputs are the inputs shifted left, where the last embedding is a prediction for a word not yet seen in the text.[6] Finally, the vectors are then projected back into word probabilities, optionally using the transpose of the original embedding matrix. The embedding projections are learned by the model during training, end-to-end, using some optimization algorithm like Stochastic Gradient Descent. Other parameters that alter the embeddings and determine how they interact are similarly learned.

GPT, the name of a series of models developed by OpenAI [13], [17], [18] stands for “Generative Pre-trained Transformer”. The “generative” part specifies that the training was done using autoregressive masking, which prevents each word from seeing the words after it in the sequence. The model is therefore suitably trained to output continuations of text. For example, if we were to input “The sky is” to GPT-3, the final embedding associated with “is” might map the token “blue” to high probability, “red” to lower probability, and “firetruck” to almost zero probability. We could then sample that

[4] Tokens are often just words, though they are sometimes sub-words (like “un-” or “-able”), punctuation (like “!”), sequences of punctuation (like “---”), or special delimiters (like “beginning of sequence”). For more information, see [15].
[5] The critical stage at which the words interact is called the attention block. For more information, see [2].
[6] During training, it is useful to let the model make predictions about every word in the sentence in parallel. Each prediction (output embedding) depends only on the previous words in an autoregressive model, using masking [2].

probability distribution, likely returning “blue”, and feed “The sky is blue” into GPT-3. We could continue this process indefinitely, though GPT-3 allows a context window of only 2048 tokens. The technique used to sample generations from GPT-3 is critical toward producing realistic continuations.

Finally, a critical aspect of using language models effectively is the prompt. The words “The sky is” in the precious example are referred to as the model’s “context”. A suitable prompt injects context that makes the likeliest continuations something useful. For example, a user seeking to learn ingredients for meatballs might start their query with “The following is a list of ingredients for meatballs: 1. Beef, 2.”. The prompt used in this paper appears in section (4).

3 Prior Work in Search

The literature on search engines is almost as old as the internet itself. Some of the most important works focus on the graph structure of the web as a starting point for ranking websites, determining importance through random walks or other graph operations [19], [20]. Other works, which are the primary inspiration for this paper, combine prospective search results using rank aggregation methods [21]. Newer approaches have focused on using AI and machine learning [22].

A popular method for using language models in search is to use them to produce query embeddings, which are then matched with document embeddings via a cosine similarity score. Prior work applying this method includes a product retrieval algorithm [3], improvements to the Baidu search engine [6], and a search scheme based on BERT [7].

Recall that a Transformer produces an embedding for each token in its input. The question is then which embedding(s) should be used to generate the similarity score. The OpenAI GPT-3 embedding endpoint uses the embedding of the “end of sequence” token [23]. The “beginning of sequence” (<BOS>) and “end of sequence” (<EOS>) tokens are used during training to separate examples. The <EOS> token can see all of the prior input (the entire sequence) but is essentially untrained, since no relevant input follows it. Therefore, it can be used as a “free space” for fine-tuning the embeddings. During fine-tuning, the OpenAI team found naturally-occurring pairs of semantically similar text and trained the output embedding of <EOS> in a contrastive fashion. The contrastive learning objective seeks to minimize the difference in embeddings between semantically similar examples while maximizing the difference in embeddings between semantically dissimilar examples. The authors show that these embeddings can perform well when used for search on some popular benchmarks like MSMARCO [24].

Products using GPT-3’s search embedding endpoint have sprung up over the past year. Elicit.org performs searches on a database of academic articles. A preliminary search retrieves 1000 papers, and the results are re-ranked by the abstract’s similarity to the query embedding [25]. Another product, available at https://metaphor.systems/, takes a new approach entirely: it directly asks the language model for a link to a useful webpage, the hope being that the language model has memorized sufficiently many urls during training. The language model used in Metaphor is fine-tuned on urls.

4 Methods

This paper proposes using GPT-3 to rephrase queries. We start with a prompt, the continuation of which is then processed into the new queries. The prompt used appears in figure 1. The user's search query replaces "{user query}".

```
Please rephrase the following Google search queries 3 times each. The new
queries should be noticeably different from the originals but should retain their
same essential meanings. They should, among other things, replace certain
words with synonyms and jumble the order of the words. Sometimes extra words
or commands can be added so that the query is more likely to give useful results.

Example:

Query: "How to set up internet on Samsung smart tv"
Rephrasings:
1. "How to connect WiFi to Samsung smart TV"
2. "connect internet to smart tv, samsung step by step"
3. "how to set up wifi connection on samsung tv, easy"

Query: "What does NP-hard mean?"
Rephrasings:
1. "non-deterministic polynomial time meaning explained, simple"
2. "computer science np-hard significance"
3. "What is NP and NP-hard CS explained"

Query: "matrix multiplication pytorch"
Rephrasings:
1. "How to do matrix multiplication in Pytorch"
2. "Pytorch matrix multiply code example"
3. "matrix product pytorch tensor operations"

Query: "{user query}"
Rephrasings:
1. "
```

Figure 1: The prompt fed to GPT-3 to create query rephrasings

I find that the generated rephrasings are highly sensitive to variations in the prompt. First, it is critical that sufficiently many helpful examples are provided. Though more examples increase the computational cost of generations, I found that even one extra example significantly increases the quality of the continuations.

Rules for sampling from GPT-3 are also critical. The following experiments were done with max_tokens=256, temperature=1, presence_penalty=1.5,

frequency_penalty=2, and best_of=4.[7] The presence and frequency penalties force the model to use words that have not appeared already in the continuation, reducing the chance that the model repeats itself verbatim (a common occurrence without high penalties). The best_of parameter specifies that the model is meant to generate 4 versions of the continuation; the most plausible complete continuation (as determined by a function of the output word probabilities) is selected.

Once the rephrasings are processed, a request is made to the Google search API to retrieve ordered search results for each query. I use a Borda count to rerank the results. The original results are included in the scoring.

In order to test the utility of the results, a small user study was conducted. A convenience sample of 8 users, all college undergraduates, participated. They performed 10 tests altogether. Users were prompted with two Google search results pages, side-by-side, with 10 results each. The original query appeared at the top of the page. One side had results that corresponded to the aggregated rankings, and the other had results that corresponded to the original ranking. The sides were randomly shuffled for each instance of the test. Users continued the test by clicking one of three buttons: the first indicated that the left results were better; the second indicated that the results were “basically the same”; and the third indicated that the right results were better. The results were recorded only if the user completed the entire test.

Users were asked to read a page of instructions before starting. The instructions asked users to choose the set of results that they thought were “best”. The instructions also explained which qualities they might consider when determining the “best” set of results. In particular, they were asked to consider: how helpful the results would likely be

[7] Details on the sampling method can be found in [26].

to the person who asked the query; the quality of the resources present; and the variety of the results. Participants were asked to spend about a minute on each comparison. The full instructions are included in Appendix A.

Two tests were produced with 22 queries each (most respondents to the second test had not seen the first). The queries were chosen arbitrarily from a combination of the author's own search history and publicly available datasets of search queries. The publicly available queries were selected from [27] and [28]. While it might have been epistemologically more disciplined to choose the queries according to some rule, it must be stated that this search rephrasing technique is not expected to work well for some types of queries. This limitation is discussed in section (6). For example, queries of only one word (like "twitter") are unlikely to be helped through rephrasing. Moreover, some results depend on the geographical location of the person performing the search, such as "restaurants near me", or the time of the search, such as "mets red sox score". Results for those queries could not be replicated accurately for the experiment.

The main limitations on the number of queries used for the experiment were the ability to find participants who were willing to complete a long test and the daily limit of the Google search API.

5 Results

Users preferred the aggregated results 35% of the time, preferred the original results 30% of the time, and rated the results as "basically the same" 35% of the time. The total number of comparisons made between the 10 users was 137. Conditioned on the user stating a preference between the two results, participants preferred the

aggregated rankings in 54% of matchups.

If we assumed that users prefer the aggregated results exactly 50% of the time, conditioned on having a preference, we would observe a preference for aggregation as high as 54% with probability 0.247 for n=137. This probability was determined using the cumulative mass function of a binomial distribution assuming that each comparison is independent. This result is not significant at the 0.05 level (p=0.247).

Aggregation was preferred six to one in “patellar tendinitis same as runners knee”, the query for which the aggregation was most popular (favored 6-1). The aggregation also performed well for the query “problems with Philips Hue smart light bulb” (favored 6-2). The aggregation performed worst for the query “roe vs. wade what happened” (favored 2-5). The aggregation also performed poorly on the query “Unifrance company id for NHK” (favoried 2-5). Every user rated the results as “Basically the same” for the query “Best times of year to visit Myrtle Beach”.

6 Analysis and Discussion

Overall, the system failed to provide significantly better results than the Google search. Participants often found that the results looked very similar (35% of the time). However, GPT-3 can still provide faithful, varied, and helpful rephrasings. Figure 2 shows some examples of the generated rephrasings. For “What is the STRONGEST muscle in the human body?”, GPT-3 gives variations that ask about “the greatest force” and “the most power”. For “ai for home improvement”, GPT-3 asks for “examples of using aI for home renovation” and in one rephrasing replaces “AI” with “artificial intelligence”.

Query	county of Saint Lucia national cricket team	ai for home improvement	What is the STRONGEST muscle in the human body?
Rephrasings	What is the Saint Lucia national cricket team's county? To which county does the Saint Lucia national cricket team belong? Where is the Saint Lucia national cricket team from (county) ?	How can I use AI for home improvement? What are some examples of using AI for home renovation? How might artificial intelligence be used in the future for home improvement projects?	What is the most powerful muscle in the human body? What muscle can exert the greatest force in humans? Which human muscle is capable of generating the most power?

Figure 2: Some examples of generated rephrasings

	Aggregation preferred 6-1	**Original preferred 5-2**
Query	patellar tendinitis same as runners knee	roe vs wade what happened
Rephrasings	Is patellar tendonitis the same thing as runner's knee? What is the difference between patellar tendinitis and runner's knee Comparing Runner's Knee with Patellar Tendinitis	What is the Roe v. Wade case? What was the outcome of Roe vs Wade Supreme Court Case: Roe v Wade 1973 decision

Figure 3: The rephrasings generated for the most and least helpful aggregation

For the rephrasings of the “patellar tendinitis” query - which generated the most popular aggregated ranking - it is interesting to note that one of the rephrasings spells tendinitis as “tendonitis” (see figure 3). Google reports “tendinitis”, the original version, as a misspelling of “tendonitis”, present in the one rephrasing; however, the Mayo Clinic prefers the original, supposedly incorrect, spelling [29]. GPT-3 uses both spellings in its rephrasings, though Google recognizes only one. The prompt never asks GPT-3 to use different spellings of a word, but it is obviously within the spirit of what the prompt is asking. This example shows how advanced LLMs can outperform the optimized logic of modern search engines.

Query: roe vs wade what happened	
Original Results	Aggregated Results
www.plannedparenthoodaction.org Roe v. Wade Overturned: Supreme Court Gave States the Right to ... The Supreme Court's Roe v. Wade ruling on January 22 ...	www.oyez.org Roe v. Wade | Oyez A case in which the Court struck down several Texas laws that criminalized abortion, holding that laws that impose an undue burden on a woman's right to ...
www.britannica.com Roe v. Wade | Summary, Origins, & Influence | Britannica Nov 8, 2022 ... Roe v. Wade, legal case in which the U.S. Supreme Court on January 22, 1973, ruled (7–2) that unduly restrictive state regulation of ...	supreme.justia.com Roe v. Wade :: 410 U.S. 113 (1973) :: Justia US Supreme Court Center Roe v. Wade: A person may choose to have an abortion until a fetus becomes viable, based on the right to privacy contained in the Due Process Clause of the ...
www.bbc.com Roe v Wade: What is US Supreme Court ruling on abortion? - BBC ... Jun 24, 2022 ... How has Roe v Wade been overturned? ... The Supreme Court has ruled in favour of Mississippi's ban on abortions after 15 weeks. In doing so, it ...	www.britannica.com Roe v. Wade | Summary, Origins, & Influence | Britannica Nov 8, 2022 ... Roe v. Wade, legal case in which the U.S. Supreme Court on January 22, 1973, ruled (7–2) that unduly restrictive state regulation of ...

Figure 4: This first three results to the query for which the original performed best

For “roe vs. wade what happened”, the query rephrasings add the words “case”, “Supreme Court”, “decision”, and the year, 1973. Figure 4 shows the first three results in the original and aggregated results (see Appendix B for more examples of results generated by aggregation). The aggregated results, consistent with the rephrasings, provide sources that focus more clearly on the legal aspects of the case. The original results include a Planned Parenthood link first, but the aggregated results show Oyez.org, a legal database of Supreme Court cases. Oyez.org organizes the facts of the case, the question, and the conclusion - resources more helpful to a law student than a layman. While the aggregated results provide more authoritative sources, users

favored results that focused more on the political implications of the case. In this example, the rephrasing method guessed, incorrectly, that users would find the authoritative legal-oriented sources more helpful than political-oriented ones.

7 Limitations and Future Work

The primary limitation of the system is that it is not expected to work well on all queries. For example, a user that searches “twitter” is just looking for a single link that Google ranks first. Additionally, the language model has a cut-off date for training. Therefore, it has difficulty reasoning about events that happened too recently. If the model were trained before a new president took office, for example, it might replace the word “the president” with the wrong person. In practice some of these issues can be addressed by applying aggregation only if the query has certain characteristics. Rank aggregation might only take place when the search engine detects a long query such as a question posed in natural language. Another heuristic might be to use aggregation only when given a unique query; results for these queries are likely under-optimized, since the search engine has no feedback to determine the most relevant results.

Successful rephrasing of queries is also dependent on the size of the model. This paper uses GPT-3, one of the largest models available. In order to reduce inference costs, practitioners might opt for smaller models that are less powerful. Smaller models might require new prompts.

The prompt for generating rephrasings could be further optimized in the future based on a particular language model’s characteristics. Some prior work investigates

prompt optimization, which can be phrased as a reinforcement learning problem [30]–[32].

Future work might also focus on providing additional benchmarks for performance. In particular, the system might be more effective when used on less powerful search engines, like Reddit. The effectiveness of this approach should also be evaluated against document/query embedding searches, mentioned in the review of prior work.

8 Conclusion

Search results might be improved by rephrasing the query and returning an aggregation of links. A system based on LLM rephrasing of queries could serve as a drop-in replacement to existing machine learning methods or as a supplement to more advanced systems. A small-scale user study of the rephrasing approach finds a modest, statistically insignificant improvement in results' quality. Future work could seek to evaluate this approach on less powerful search engines, refine the prompt to generate better rephrasings, try different aggregation methods, or determine rules for finding queries susceptible to improvement by rephrasing.

References

[1] J. Wei *et al.*, "Emergent Abilities of Large Language Models." arXiv, Jun. 15, 2022. Accessed: Oct. 12, 2022. [Online]. Available: http://arxiv.org/abs/2206.07682

[2] A. Vaswani *et al.*, "Attention Is All You Need," Jun. 2017, doi: 10.48550/arXiv.1706.03762.

[3] S. Y. Kim, H. Park, K. Shin, and K.-M. Kim, "Ask Me What You Need: Product Retrieval using Knowledge from GPT-3." arXiv, Jul. 06, 2022. Accessed: Oct. 12, 2022. [Online]. Available: http://arxiv.org/abs/2207.02516

[4] A. Jaech and M. Ostendorf, "Personalized Language Model for Query Auto-Completion." arXiv, Apr. 25, 2018. Accessed: Oct. 12, 2022. [Online]. Available: http://arxiv.org/abs/1804.09661

[5] N. Sousa, N. Oliveira, and I. Praça, “Machine Reading at Scale: A Search Engine for Scientific and Academic Research,” *Systems*, vol. 10, no. 2, Art. no. 2, Apr. 2022, doi: 10.3390/systems10020043.
[6] Y. Liu *et al.*, “Pre-trained Language Model for Web-scale Retrieval in Baidu Search,” in *Proceedings of the 27th ACM SIGKDD Conference on Knowledge Discovery & Data Mining*, New York, NY, USA, Aug. 2021, pp. 3365–3375. doi: 10.1145/3447548.3467149.
[7] M. Patel, “TinySearch -- Semantics based Search Engine using Bert Embeddings.” arXiv, Aug. 07, 2019. doi: 10.48550/arXiv.1908.02451.
[8] T. Winograd, “Understanding natural language,” *Cognit. Psychol.*, vol. 3, no. 1, pp. 1–191, Jan. 1972, doi: 10.1016/0010-0285(72)90002-3.
[9] “Can Winograd Schemas Replace Turing Test for Defining Human-Level AI? - IEEE Spectrum.” https://spectrum.ieee.org/winograd-schemas-replace-turing-test-for-defining-humanlevel-artificial-intelligence (accessed Nov. 27, 2022).
[10] S. Hochreiter and J. Schmidhuber, “Long Short-Term Memory,” *Neural Comput.*, vol. 9, no. 8, pp. 1735–1780, Nov. 1997, doi: 10.1162/neco.1997.9.8.1735.
[11] W. Wang and J. Gang, “Application of Convolutional Neural Network in Natural Language Processing,” in *2018 International Conference on Information Systems and Computer Aided Education (ICISCAE)*, Jul. 2018, pp. 64–70. doi: 10.1109/ICISCAE.2018.8666928.
[12] *Attention is all you need; Attentional Neural Network Models | Łukasz Kaiser | Masterclass*, (Oct. 04, 2017). Accessed: Nov. 27, 2022. [Online Video]. Available: https://www.youtube.com/watch?v=rBCqOTEfxvg
[13] T. B. Brown *et al.*, “Language Models are Few-Shot Learners.” arXiv, Jul. 22, 2020. doi: 10.48550/arXiv.2005.14165.
[14] J. Wells, M. Truran, and J. Goulding, “Search habits of the computer literate,” in *Proceedings of the 18th conference on Hypertext and hypermedia - HT ’07*, Manchester, UK, 2007, p. 37. doi: 10.1145/1286240.1286251.
[15] J. J. Webster and C. Kit, “Tokenization as the Initial Phase in NLP,” in *COLING 1992 Volume 4: The 14th International Conference on Computational Linguistics*, 1992. Accessed: Nov. 27, 2022. [Online]. Available: https://aclanthology.org/C92-4173
[16] E. Vylomova, L. Rimell, T. Cohn, and T. Baldwin, “Take and Took, Gaggle and Goose, Book and Read: Evaluating the Utility of Vector Differences for Lexical Relation Learning.” arXiv, Aug. 13, 2016. Accessed: Dec. 01, 2022. [Online]. Available: http://arxiv.org/abs/1509.01692
[17] A. Radford, K. Narasimhan, T. Salimans, and I. Sutskever, “Improving Language Understanding by Generative Pre-Training,” p. 12.
[18] A. Radford, J. Wu, R. Child, D. Luan, D. Amodei, and I. Sutskever, “Language Models are Unsupervised Multitask Learners,” p. 24.
[19] L. Page, S. Brin, R. Motwani, and T. Winograd, “The PageRank Citation Ranking: Bringing Order to the Web.,” Nov. 11, 1999. http://ilpubs.stanford.edu:8090/422/ (accessed Dec. 03, 2022).
[20] J. Kleinberg, “Authoritative sources in a hyperlinked environment | Journal of the ACM,” Sep. 1999. https://dl.acm.org/doi/abs/10.1145/324133.324140 (accessed Dec. 03, 2022).
[21] C. Dwork, R. Kumar, M. Naor, and D. Sivakumar, “Rank aggregation methods for the Web,” in *Proceedings of the 10th international conference on World Wide Web*, New York, NY, USA, Apr. 2001, pp. 613–622. doi: 10.1145/371920.372165.
[22] K. M. K. Teja, “Optimizing the relevancy of Predictions using Machine Learning and NLP of Search Query,” 2014. https://www.semanticscholar.org/paper/Optimizing-the-relevancy-of-Predictions-using-and-Teja/ea635666bed527f89d372982db547a4c2e1f1c7f (accessed Dec. 03, 2022).
[23] A. Neelakantan *et al.*, “Text and Code Embeddings by Contrastive Pre-Training.” arXiv,

Jan. 24, 2022. Accessed: Nov. 28, 2022. [Online]. Available: http://arxiv.org/abs/2201.10005
[24] P. Bajaj *et al.*, “MS MARCO: A Human Generated MAchine Reading COmprehension Dataset.” arXiv, Oct. 31, 2018. Accessed: Dec. 01, 2022. [Online]. Available: http://arxiv.org/abs/1611.09268
[25] “FAQ | Elicit.” https://elicit.org/faq (accessed Dec. 01, 2022).
[26] “OpenAI API.” https://beta.openai.com (accessed Dec. 01, 2022).
[27] “Results - Dataset of Search Queries and Intents.” https://app.surgehq.ai/datasets/search-queries-and-intents (accessed Dec. 01, 2022).
[28] “Most Searched Thing on Google: Top Google Searches in 2022 (US & Worldwide),” *Semrush Blog*. https://www.semrush.com/blog/most-searched-keywords-google (accessed Dec. 01, 2022).
[29] “Patellar tendinitis - Symptoms and causes,” *Mayo Clinic*. https://www.mayoclinic.org/diseases-conditions/patellar-tendinitis/symptoms-causes/syc-20376113 (accessed Dec. 01, 2022).
[30] Z. Dai *et al.*, “Promptagator: Few-shot Dense Retrieval From 8 Examples.” arXiv, Sep. 23, 2022. doi: 10.48550/arXiv.2209.11755.
[31] B. Lester, R. Al-Rfou, and N. Constant, “The Power of Scale for Parameter-Efficient Prompt Tuning.” arXiv, Sep. 02, 2021. doi: 10.48550/arXiv.2104.08691.
[32] X. Liu *et al.*, “P-Tuning v2: Prompt Tuning Can Be Comparable to Fine-tuning Universally Across Scales and Tasks.” arXiv, Mar. 20, 2022. doi: 10.48550/arXiv.2110.07602.

Appendix A

Instructions

The test shows you two groups of Google results responding to a search query, written at the top of the page. You are asked to judge whether the search results presented on the right or left are best. If you are unable to markedly distinguish between the two, you should click “Basically the same”. At the end of the test, please send the results string to the test administrator.

The two sets of results will likely be quite similar. Your judgements should be based not only on the links that are presented but also the order in which they appear.

We recommend that you follow some of the links presented for each query, on each side. You should aim to spend around one minute on each comparison.

Some criteria you may want to consider when deciding whether the right or left results are better include:

- How helpful the results would be to the person who wrote the search query
    - This may or may not exactly align with the precise meaning of the search query
- The quality of the resources present in each set of results
- How repetitive the results are (where varied might be better)
- The quality of the very first result, or first couple of results

Please do not leave the page during the test, shut off your computer, or refresh the page. These actions will force you to start the test again.

If you have any questions before or during the test, please contact the test administrator.

Appendix B

The following tables show the first three results for each query that appears in the paper. The original results appear on the left, and the aggregated results appear on the right. To see all ten results for each of the 44 queries, see the GitHub page: https://github.com/gkamer8/gpt-rephrasing.

Query: patellar tendinitis same as runners knee	
Original Results	Aggregated Results
www.sports-health.com Jumper's Knee vs. Runner's Knee | Sports-health In contrast to patellofemoral pain (runner's knee), the knee pain from patellar tendinitis often decreases with time during activity as the tendon “warms up ...	www.sports-health.com Jumper's Knee vs. Runner's Knee | Sports-health In contrast to patellofemoral pain (runner's knee), the knee pain from patellar tendinitis often decreases with time during activity as the tendon “warms up ...
www.hopkinsmedicine.org Patellofemoral Pain Syndrome (Runner's Knee) | Johns Hopkins ... Runner's knee means that you have dull pain around the front of the knee (patella). This is where the knee connects with the lower end of the thighbone ..	medi-dyne.com The Difference Between Runner's Knee and Jumper's Knee Pain Runner's knee is a more commonly used name for a condition known as patellofemoral pain syndrome. This particular condition occurs as a result of femur and ...
orthoinfo.aaos.org Patellofemoral Pain Syndrome - OrthoInfo - AAOS Patellofemoral pain syndrome (PFPS) is a broad term used to describe pain in the ... It is sometimes called "runner's knee" or "jumper's knee" because it is ...	www.hopkinsmedicine.org Patellofemoral Pain Syndrome (Runner's Knee) | Johns Hopkins ... Runner's knee means that you have dull pain around the front of the knee (patella). ... some sports injuries affect women more often or in different ways.

Query: county of Saint Lucia national cricket team

Original Results	Aggregated Results
en.wikipedia.org Saint Lucia - Wikipedia Saint Lucia is an island country in the West Indies in the eastern Caribbean Sea on the ... The Windward Islands cricket team includes players from Saint Lucia and ...	en.wikipedia.org Saint Lucia - Wikipedia Saint Lucia is an island country in the West Indies in the eastern Caribbean Sea on the ... The Windward Islands cricket team includes players from Saint Lucia and ...
bastlucia.com Executive – Bankers Association of St. Lucia Team Categories: Executive ... He recently served as 2nd Vice President of the St. Lucia National Cricket Association having previously served as Deputy ...	en.wikipedia.org Tim David - Wikipedia Timothy Hays David (born 16 March 1996), better known as Tim David, is an Australian cricketer. He played for the Singapore national cricket team and ...
en.wikipedia.org Tim David - Wikipedia Timothy Hays David (born 16 March 1996), better known as Tim David, is an Australian cricketer. He played for the Singapore national cricket team and ...	bastlucia.com Executive – Bankers Association of St. Lucia Team Categories: Executive ... He recently served as 2nd Vice President of the St. Lucia National Cricket Association having previously served as Deputy ...

Query: ai for home improvement	
Original Results	Aggregated Results
fortune.com Homebound, Skipp: How A.I. makes home renovation jobs easier ... Mar 1, 2022 ... Skipp relies on A.I. to compile, analyze, and recommend home renovation options. Today the company is focused on kitchens, which happen to ...	fortune.com Homebound, Skipp: How A.I. makes home renovation jobs easier ... Mar 1, 2022 ... Skipp relies on A.I. to compile, analyze, and recommend home renovation options. Today the company is focused on kitchens, which happen to ...
mapzot.com AI for Home Improvement Site Selection Mapzot is Fast and Easy to use. AccuSite by Mapzot is the next generation Home Improvement Site Selection Software that combines Artificial Intelligence and ...	mapzot.com AI for Home Improvement Site Selection Mapzot is Fast and Easy to use. AccuSite by Mapzot is the next generation Home Improvement Site Selection Software that combines Artificial Intelligence and ...
www.facebook.com A.I. Home Improvements - Home | Facebook A.I. Home Improvements, Safety Beach. 264 likes. Professional Home Improvement Services! Interior and Exterior Solutions to satisfy your needs.	constructible.trimble.com 10 Examples of Artificial Intelligence in Construction Apr 6, 2022 ... Keep reading to understand how AI is used in construction and the 10 main benefits of

	using AI in construction. What is Artificial Intelligence ...

Query: What is the STRONGEST muscle in the human body?	
Original Results	Aggregated Results
www.loc.gov What is the strongest muscle in the human body? | Library of Congress Nov 19, 2019 ... The strongest muscle based on its weight is the masseter. With all muscles of the jaw working together it can close the teeth with a force as ...	www.loc.gov What is the strongest muscle in the human body? | Library of Congress Nov 19, 2019 ... There is absolute strength (maximum force), dynamic strength (repeated motions), elastic strength (exert force quickly), and strength ...
www.scientificamerican.com Fact or Fiction?: The Tongue Is the Strongest Muscle in the Body ... Aug 15, 2014 ... Fact or Fiction?: The Tongue Is the Strongest Muscle in the Body ... Is this agile appendage as brawny as people believe? ... It can bend, it can ...	www.livescience.com What's the Strongest Muscle in the Human Body? | Live Science Sep 29, 2010 ... If the title goes to the muscle that can exert the most force, the victor would be the soleus, or the calf muscle, according to Gray's ...
www.livescience.com What's the Strongest Muscle in the Human Body? | Live Science Sep 29, 2010 ... If the title goes to the muscle that can exert the most force, the victor would be the soleus, or the calf muscle, according to Gray's Anatomy, ...	www.scientificamerican.com Fact or Fiction?: The Tongue Is the Strongest Muscle in the Body ... Aug 15, 2014 ... It can bend, it can twist, it can suck, it can cup. The tongue is an essential, often playful part of human anatomy.