Revisiting Query Variation Robustness of Transformer Models

Tim Hagen
University of Kassel
and hessian.AI

Harrisen Scells
University of Kassel
and hessian.AI

Martin Potthast
University of Kassel,
hessian.AI, and ScaDS.AI

Abstract

The most commonly used transformers for retrieval at present, BERT and T5, have been shown not to be robust to query variations such as typos or paraphrases. Although this is an important prerequisite for their practicality, this problem has hardly been investigated. More recent large language models (LLMs), including instruction-tuned LLMs, have not been analyzed yet, and only one study looks beyond typos. We close this gap by reproducing this study and extending it with a systematic analysis of more recent models, including Sentence-BERT, CharacterBERT, E5-Mistral, AnglE, and Ada v2. We further investigate if instruct-LLMs can be prompted for robustness. Our results are mixed in that the previously observed robustness issues for cross-encoders also apply to bi-encoders that use much larger LLMs, albeit to a lesser extent. While further LLM scaling may improve their embeddings, their cost-effective use for all but large deployments is limited. Training data that includes query variations allows LLMs to be fine-tuned for more robustness, but focusing on a single category of query variation may even degrade the effectiveness on others.[1]

1 Introduction

Despite their proficiency with natural language, transformer-based large language models (LLMs) trained for document ranking, like BERT or T5, are not robust to ill-formed queries, including queries with typos and queries that omit less important words (Penha et al., 2022; Sidiropoulos and Kanoulas, 2022; Zhuang et al., 2022). Zhuang et al. (2023) reason that these variations are hardly represented in the LLMs' training data. However, query variations are the norm in practice: about 70% of information-seeking queries to web search engines are keyword queries (White et al., 2015) instead of fully formed questions, and up to 26% of queries contain typos (Wang et al., 2003). However, due to their superior effectiveness, current information retrieval (IR) systems use these 'embedding models'[2] despite their lack of robustness.

To our knowledge, this phenomenon has hardly been investigated with respect to information retrieval (Section 2). Penha et al. (2022) contributed the most exhaustive study to date. They measure the changes in ranking effectiveness of various models for an originally intended, well-formed query from TREC DL'19 (Craswell et al., 2020) and ANTIQUE (Hashemi et al., 2020) compared to randomly generated but semantically equivalent variations. However, no study yet investigated the ranking robustness of LLM-based embedding models more recent than BERT (Devlin et al., 2019) and T5 (Raffel et al., 2020) with respect to query variations. Given the fast pace at which LLMs have been scaled, it is unclear whether scaling alone suffices to solve the problem.

In this paper, we investigate the robustness of recent large language models against purely syntactic, query variants which preserve semantics. We reproduce Penha et al.'s experiment on the robustness of cross encoders to different query variants from well-formed queries. In addition to comparing the ranking effectiveness of query variations to well-formed queries, we measure the anisotropy adjusted cosine similarity of the variations' embeddings compared to their respective original's to quantify the robustness of the embeddings, i.e., how much input variations affect the models' outputs. Doing so, we also extend Penha et al.'s study, to the state-of-the-art embedding models Sentence-BERT, CharacterBERT, E5-Mistral, AnglE, and Ada v2.

[1]Our code, results, and artifacts can be found at github.com/webis-de/EMNLP-24.

[2]Although there is a wide range of models and architectures, we focus on the representations they compute and therefore refer to them collectively as 'embedding models'.

Our reproduction results are consistent with previous work, showing that existing dense retrieval models are not robust to semantically identical ill-formed query variants. We also find LLM-based embedding models to suffer from the same problem, but to a lesser extent. Our experiments show that robustness to typos can be improved substantially using character-level transformers and typo aware pre-training, but this does not generalize to other query variations. We observe that focusing on a single category of query variation may even degrade effectiveness on other categories of variations. Moreover, some models were less robust to keyword queries than to typos, presumably because transformer models use stop words to aggregate the context (Clark et al., 2019; Ethayarajh, 2019).

2 Related Work

Though LLMs are effective at information retrieval and natural language processing tasks (Kocoń et al., 2023), they are not robust to language variations like typos or keywording (Zhuang and Zuccon, 2021; Penha et al., 2022; Zhuang et al., 2022; Sidiropoulos and Kanoulas, 2022). Previous research identified two key contributing factors to this lack of robustness: (1) LLMs and ranking models are mostly trained on clean data, rendering variations like typos and keywords out of domain (Zhuang et al., 2023), and (2) character-level information is lost during tokenization (Almagro et al., 2023; Zhuang and Zuccon, 2022). For example, 'weird' and its misspelling 'wierd' are tokenized by BERT as `weird` and `wi ##er ##d`.

Typo awareness Solutions to improve robustness of transformer-based models for information retrieval can be categorized into improving robustness using typo aware (pre-)training strategies (Tasawong et al., 2023; Zhuang et al., 2023, 2022), using datasets with input variations (Penha et al., 2022; Bailey et al., 2016), and developing models using character-level tokenization schemes (Zhuang and Zuccon, 2022; Almagro et al., 2023).

Zhuang et al. (2023) propose pre-training using a typo aware masked language modeling strategy to strengthen a model's robustness to typos. Previous studies improved typo robustness by fine-tuning models on datasets augmented with misspellings (Zhuang et al., 2022; Tasawong et al., 2023). Sidiropoulos and Kanoulas (2022, 2024) train dual encoders using a contrastive loss to learn closer representations for words and their typos while keeping other words' representations apart. Sidiropoulos and Kanoulas find that typos in less prevalent words degrade effectiveness more than typos in common words and identify BERT's WordPiece tokenizer as a limiting factor in this respect.

To address the tokenizer's shortcomings, Zhuang and Zuccon (2022) use CharacterBERT (Boukkouri et al., 2020), a BERT variant that creates character-level token representations instead of word pieces. They fine-tune it on a typo-augmented dataset, minimizing two loss components: the KL-divergence of the relevance label distributions of a query and its typo variant, and contrastive cross entropy loss to retain effectiveness. Tasawong et al. (2023) call these components 'robustness' and 'effectiveness' and additionally propose 'alignment' to learn similar representations for queries and their variants. CharacterBERT's character-level tokenizer is crucial for more typo robustness as it can better recognize small input variations. Almagro et al. (2023) propose the LExical-aware Attention module (LEA), which adds a learnable bias to the attention scores based on text similarity.

Beyond typos The research mentioned above only focuses on typos. Penha et al. (2022) take the robustness analysis further by creating a test collection covering three more classes of query variations that retain semantics. To our knowledge, the only other test collection for query variations beyond typos is UQV100 (Bailey et al., 2016). However, it does not guarantee identical semantics across query variations, nor does it specify the category of the variation. We make extensive use of Penha et al.'s collection to study the query robustness of a spectrum of transformer-based retrieval models, ranging from the cross-encoders used to supplement the collection, to typo aware embedding models, to embedding models based on LLMs.

Robustness issues are also not confined to IR. Ravichander et al. (2021), Sidiropoulos et al. (2022), and Qiang et al. (2024) observe a significant drop in effectiveness introduced by synonyms, paraphrasing, and errors induced by automatic speech recognition when providing a voice interface for language models. Zheng and Saparov (2023) analyze the accuracy degradation due to typos, synonyms, repetition, and 'shortcuts' (providing part of the answer together with the prompt) observed with prompted embedding models on natural language tasks. By adding perturbed examples in few-shot prompts, they improved robustness to all variations except typos.

Query variation		Example	Valid variants	
Category	Transform. heuristic		TREC DL '19	ANTIQUE
Original		*what is durable medical equipment consist of*		
Misspelling	NeighbCharSwap	*what is durable **mdeical** equipment consist of*	43 (100.00%)	199 (99.50%)
	RandomCharSub	*what is durable **medycal** equipment consist of*	42 (97.67%)	197 (98.00%)
	QWERTYCharSub	*what is durable medical equipment **xonsist** of*	42 (97.67%)	182 (91.50%)
Naturality	RemoveStopWords	*~~**what is**~~ durable medical equipment consist ~~**of**~~*	37 (86.05%)	199 (99.50%)
	T5DescToTitle	*~~**what is**~~ durable medical equipment ~~**consist of**~~*	35 (81.40%)	136 (68.00%)
Ordering	RandomOrderSwap	***medical** is durable **what** equipment consist of*	43 (100.00%)	200 (100.00%)
Paraphrasing	BackTranslation	*what is **sustainable** medical equipment ~~**consist of**~~*	23 (53.49%)	93 (46.50%)
	T5QQP	*what is durable medical equipment ~~**consist of**~~*	26 (60.47%)	105 (52.50%)
	WordEmbedSynSwap	*what is durable **medicinal** equipment consist of*	27 (62.79%)	124 (62.00%)
	WordNetSynSwap	*what is **long lasting** medical equipment consist of*	16 (37.21%)	71 (35.50%)

Table 1: Examples of query variations when applying transformation heuristics to the query *'what is durable medical equipment consists of'*, and the number of valid (i.e., semantically identical) variations generated. Not all variations exemplified may be valid. This table is reproduced from Penha et al. (2022, Table 3).

3 Methods and Experimental Setup

This section describes the experimental setup consisting of the datasets with query variations, the embedding models used for the experiments and the experiments performed.

3.1 Query Variation Dataset

We use the query variation dataset by Penha et al.[3] They define four types of query variations that preserve semantics and suggest transformation heuristics to create them automatically: Misspelling, naturality (which refers to turning a fully formed query into a keyword query), word ordering, and paraphrasing. To create the dataset, the transformations were applied independently to each of the test queries in TREC DL'19 and each of the validation queries in ANTIQUE.[4] The resulting queries were manually filtered to keep only semantically identical query variations. Table 1 shows examples for each transformation prior to filtering.

3.2 Embedding Models

In addition to BERT and T5 used in the experiments of Penha et al., we include newer embedding models in our experiments: SBERT (Reimers and Gurevych, 2019) often serves as a new baseline for BERT-based embedding models. CharacterBERT-DR-ST (Zhuang and Zuccon, 2022), or CBERT for brevity, represents models specifically designed to be robust to typos. E5 (Wang et al., 2024) (the E5-mistral-7b-instruct variant), AnglE (Li and Li, 2023) (the UAE-Large-v1 variant), and Ada v2 (Greene et al., 2022) (OpenAI's embedding model text-embedding-ada-002), represent the state of the art in embedding models: E5-mistral-7b-instruct and UAE-Large-v1 were the leading models on the MTEB ranking list at the time of the experiment, and text-embedding-ada-002 is a leading commercial model.[5] The MTEB dataset was created specifically for the comparison of embedding models for various natural language processing and information retrieval tasks (Muennighoff et al., 2023). Further information on the models and their training can be found in the Appendix A.1.

3.3 Experimental Setup

The experiment reproduces the setup of Penha et al. (2022), which examines the impact of each query variation category on ranking effectiveness, to investigate the *ranking robustness* of a model. Furthermore, we introduce a second experiment to investigate the *embedding robustness* of a model by measuring how similar the embedding of a query variation is to that of the original query.

To assess ranking robustness, we evaluate an embedding model in the second step of a reranking retrieval pipeline, using it as a dual encoder, and computing the difference in nDCG@10 ('ΔnDCG@10') when ranking on the original query and its variants. Ideally, ΔnDCG@10 should be 0, as semantically identical queries should result in the same rankings. A positive value indicates that the model is more effective for the query variant than for the original query, a negative value indicates less effectiveness. For the initial retrieval in the first step of our pipeline, we use the official top 1000 test set of TREC DL'19, and the top 1000 documents returned by BM25 for ANTIQUE.

[3] github.com/Guzpenha/query_variation_generators/

[4] Penha et al. used ANTIQUE's validation queries instead of the test queries as stated in their paper (see Section 4.1).

[5] huggingface.co/spaces/mteb/leaderboard

The evaluation of ANTIQUE is zero-shot for all models. We have not fine-tuned the models for ANTIQUE and, to our knowledge, neither have the authors of the models.

Our experimental setup differs slightly from the main experiment of Penha et al. in that we always perform the initial retrieval with BM25 on the original query. We are only interested in the robustness of the dual encoders and not that of the entire retrieval pipeline. However, our results are comparable to their results, since Penha et al. have shown that the robustness of the re-ranker is similar to that of the entire pipeline.

To assess embedding robustness, we compute the embeddings of a query and a query variant and measure the anisotropy adjusted cosine similarity between them. We define the anisotropy adjusted cosine similarity as

$$\text{adjcossim}(v, v') = \frac{\text{cossim}(v, v') - \mu}{1 - \mu}, \quad (1)$$

where μ is the expected cosine similarity when embedding two randomly selected inputs (see Table 5 for the values we used). Embedding models often embed into localized subspaces instead of the entire embedding space (Mu and Viswanath, 2018; Ethayarajh, 2019), which makes it difficult to compare cosine similarities across models without renormalization by the adjusted cosine similarity (see Appendix A.2 for details). Note that embeddings with a cosine similarity of 1 also have an adjusted cosine similarity of 1, while the expected adjusted similarity of any two strings is 0. Since the embeddings are semantic representations and since Penha et al. has ensured that each variant is semantically identical to its original query, their similarity should ideally be 1.

4 Results and Discussion

First, we present the similarities and differences in our reproduction of Penha et al.'s experiments. Then, we compare these results with various other models from the literature to see how they generalize. Finally, we evaluate the impact that typo awareness can have by comparing the robustness of different architectures and fine-tuning CBERT and prompt-tuning E5 using a training set created from the query variations of Penha et al.'s dataset.

4.1 Reproduction

To reproduce the robustness results by Penha et al. for BERT and T5, we reran their code with slight modifications:[6] (1) We updated the versions in the `requirements.txt` since previous versions were not supported anymore, (2) we fixed minor runtime errors which presumably occurred due to the version updates, and (3) we resolved an error in the evaluation routine for BERT on ANTIQUE. As previously noted, Penha et al. did not evaluate on ANTIQUE's test set but on the validation split defined by `ir_datasets` (MacAvaney et al., 2021), `antique/train/split200-valid`, which is a split of ANTIQUE's official training set, since ANTIQUE officially does not have a validation set. However, Penha et al. fine-tune BERT on ANTIQUE's official training set (`antique/train`) instead of `ir_datasets`' training set (`antique/train/split200-train`), thus training on part of their test data. We also remap ANTIQUE's graded relevance labels to the range 0-3 as described by Hashemi et al. in its README.[7]

As we use the same test set, the same pre-trained T5 model and their code, we expect the results for T5 to be nearly identical to those reported by Penha et al., apart from what can reasonably be attributed to rounding differences during inference. For BERT on TREC DL'19 we expect similar results to the original paper, but discrepancies beyond rounding errors are expected due to the stochastic nature of fine-tuning. On ANTIQUE, we expect that our results for BERT are dramatically worse regarding each query variant with the largest expected drop for the original queries.

Table 2a presents our reproduction results. The results are generally as expected—including the large drop in BERT's effectiveness on ANTIQUE as compared to the original paper's results. However, we see four instances of our reproduction deviating stronger than explained by the reasoning above. We most notably observe large deviations in both paraphrasing variants that replace a word with its synonym (WordEmbedSynSwap and WordNetSynSwap), which yield better results on TREC DL'19 in our reproduction.

4.2 Model Robustness

Table 2b presents the mean nDCG@10 scores achieved when re-ranking with each model on TREC DL'19 and ANTIQUE while applying a single query transformation (the query 'variant' in the table). Figure 1 shows the ΔnDCG@10 resulting from each transformation category. Like

[6] github.com/Guzpenha/query_variation_generators
[7] ciir.cs.umass.edu/downloads/Antique/readme.txt

		(a) Reproduction				(b) Generalization									
Query variation		**TREC DL '19**		**ANTIQUE**		**TREC DL '19**					**ANTIQUE**				
Category	Transform. heuristic	BERT	T5	BERT	T5	*SBERT*	*CBERT*	*E5*	*AnglE*	*Ada v2*	*SBERT*	*CBERT*	*E5*	*AnglE*	*Ada v2*
Original		.65 /.66	.70 /.70	.42 /**.29**	.33 /.33	.70	.64	.69	.70	.69	.25	.29	.41	.36	.37
Misspelling	NeighbCharSwap	.42*/.42*	.50*/.50*	.29*/**.19***	.25*/.25*	.52*	.59*	.66	.55*	.61*	.18*	.26*	.37*	.29*	.31*
	RandomCharSub	.33*/.34*	.40*/.39*	.28*/**.19***	.25*/.24*	.56*	.60	.60	.57	.58*	.20*	.26*	.37*	.30*	.31*
	QWERTYCharSub	.39*/.38*	.45*/.44*	.30*/**.18***	.27*/.26*	.60*	.56*	.62	.55*	.59*	.20*	.26*	.38*	.32*	.31*
Naturality	RemoveStopWords	.64 /.64	.69 /.70	.38*/**.26***	.32*/.32	.69	.62	.69	.68	.68	.22*	.24*	.36*	.32*	.28*
	T5DescToTitle	.54*/.55*	.57*/.59	.27*/**.25***	.24*/**.29***	.62	.58	.62	.61	.62	.20*	.22*	.31*	.29*	.23*
Ordering	RandomOrderSwap	.64 /.65	.70 /.70	.41*/**.28**	.33*/.33	.67	.58	.66	.65	.62*	.25	.27	.39*	.35	.34*
Paraphrasing	BackTranslation	.55*/.58	.61*/.61	.31*/**.26**	.26*/**.32**	.60	.57	.64	.63	.62	.25	.28	.40	.36	.36
	T5QQP	.64 /.64	.71 /.71	.39*/**.26**	.32*/.30	.67	.61	.65	.69	.68	.23	.25	.37*	.34	.33
	WordEmbedSynSwap	.47*/**.59**	.56*/**.65**	.33*/**.23***	.28*/.28*	.66	.59	.61	.66	.64	.23	.26	.40*	.36	.35*
	WordNetSynSwap	.45*/**.62**	.55*/**.71**	.32*/**.22***	.27*/.28*	.58	.62	.64	.64	.61	.19	.24	.34	.30*	.29*

* significant difference (Bonferroni corrected two-sided paired Student's T-Test at $p < 5\%$) to ranking on the original query

Table 2: (a) Reproduction results. Cells indicate [theirs]/[ours], where 'theirs' is the nDCG@10 reported by Penha et al. (2022) and 'ours' the score we achieved when repeating their experiment as described. Values in bold indicate large differences we discussed in Section 4.1. The models re-rank the top 100 passages initially retrieved by BM25. (b) nDCG@10 of embedding models on TREC DL '19 and ANTIQUE. The most effective model per variant and dataset is underlined. The models re-rank the top 1000 passages initially retrieved by BM25.

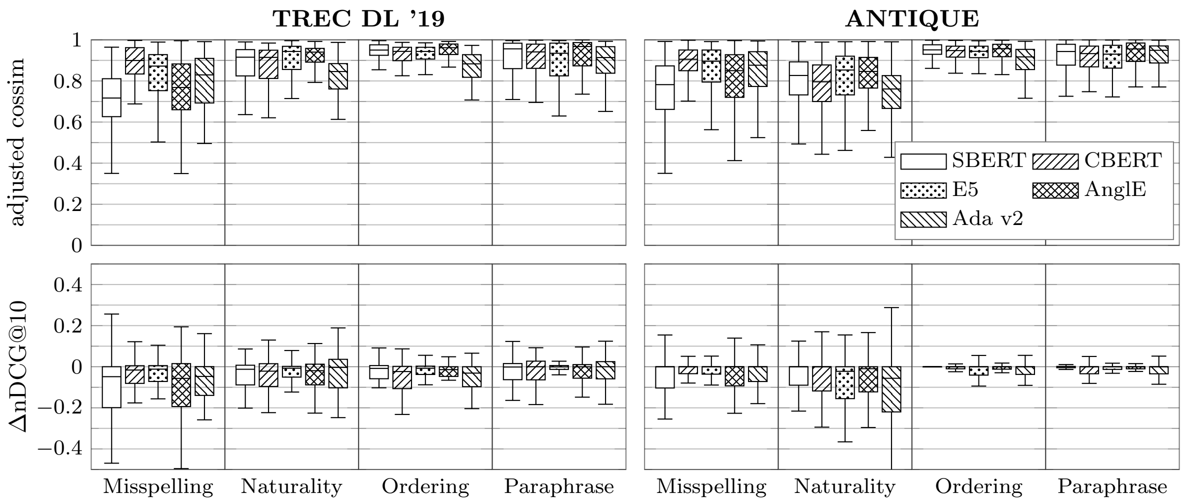

Figure 1: Robustness per model, variation category, and dataset. Embedding-robustness (top) is the adjusted cosine similarity of the original query's embedding to its variant's. Ranking-robustness (bottom) is the difference in effectiveness from ranking on the query variant. For clarity, 1981 outliers out of 18400 data points are not shown.

Penha et al., we observe that, while these transformations can improve effectiveness on some queries (positive ΔnDCG@10), the mean effectiveness is not improved statistically significantly. Table 2b shows that only effectiveness degradation is statistically significant. Similar to Penha et al., we can observe that the embedding models we tested are most robust to transformations from the 'ordering' category (i.e., median close to 0 and spread the least in Figure 1). We also observe that the misspelling category has a considerably smaller effect on ANTIQUE than on TREC DL'19. Penha et al. hypothesize that this occurs since queries in TREC DL'19 are shorter. However, concrete comparisons, e.g., about query length, cannot be made using nDCG across datasets due to differences in query, document, and relevance assessments distributions. The experiments on ANTIQUE also have a vastly larger sample size as ANTIQUE's training set contains more queries (Table 1).

Figure 1 presents the embedding robustness in terms of adjusted cosine similarity between each model's embedding of the original query and its variants. A similarity closer to 0 indicates unrelated semantic representations, while a similarity of 1 indicates semantic identity. Like with embedding robustness, we can observe similar trends on both datasets. The ordering category is the easiest, then paraphrasing. All models are least robust to transformations from the naturality and the misspelling category, except CBERT, which is considerably more embedding robust to the misspelling category than other models. AnglE is the most embedding robust model overall (median adjusted cosine similarity close to 1 and least spread) except for robustness to misspelling, which it is the second least robust to. The largest model we tested, E5, is generally similarly robust to the most robust model per category. This is especially interesting on the misspelling category, where its median embedding robustness is only slightly worse than CBERT's, but the spread is larger, i.e., E5 is similarly robust for the median query but in some worst cases the misspelling can have a larger negative effect on embedding robustness. Note, however, that in practice E5 and CBERT are similarly ranking robust while E5 is considerably more effective on misspellings despite missing a character-level tokenizer and specific fine-tuning for typo robustness. Thus, E5 presents an interesting proof of concept for the application of LLM-based embedding models to information retrieval. In short, we find that, though contemporary large embedding models can be a lot more ranking-robust to semantics retaining query transformations than BERT-based embedding models, they still are not robust. Further, their robustness (though not their effectiveness) can be closely matched by far more efficient BERT-based embedding models (CBERT on misspellings and AnglE on all other categories).

Finally, CBERT demonstrates that character tokenization paired with typo-aware training improves robustness to typos. However, comparing SBERT's and CBERT's ranking-robustness (Figure 1), shows that this robustness does not translate to other variation categories as CBERT consistently exhibits slightly worse robustness across all other variants. We leave comparing character level architectures and typo-aware training strategies for future work.

4.3 Robustness Across Architectures

Figure 2 presents the ranking robustness of various models on TREC DL'19 and ANTIQUE. Specifically, we compare the BERT-based dual encoder SBERT with the BERT-based cross-encoder monoBERT (denoted by 'BERT'), the most robust ranking model from the original work, monoT5 ('T5'), and the most robust model from our experiments, E5. Models with the V subscript evaluate the entire ranking pipeline's robustness when re-ranking BM25's top 1000 passages. That is, the query *variant* is used for both initial retrieval and re-ranking, whereas models denoted by O, use the *original* query for initial retrieval.

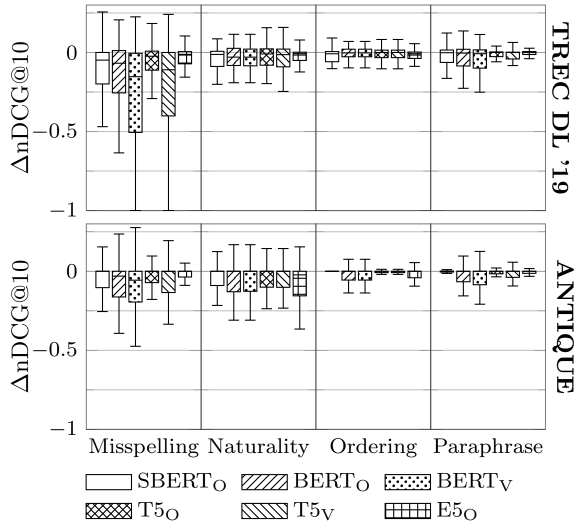

Figure 2: Ranking-robustness of cross-encoders compared to the most robust model from Table 2 (b). O denotes retrieval on the original query and re-ranking using the variant. V denotes retrieval and re-ranking using the query variant. 1731 outliers hidden for clarity.

Interestingly, neither architecture, cross-encoder or dual encoder, appears more robust than the other. Both architectures further display similar trends: all models and the pipelines based on these are least robust to typos, then paraphrasing and naturality, and all are most robust to ordering. For the pipelines' robustness, we observe that, while BM25 does not meaningfully impact the robustness to naturality and ordering, since it itself filters stop words and is a bag-of-word model such that ordering does not matter, it heavily degrades robustness in response to typos, especially on TREC DL'19. We hypothesize two causes to explain the robustness of BERT_O and T5_O over BERT_V and T5_V respectively: (1) the number of initially retrieved documents using the original query influences robustness of the output rankings of the re-ranker, and (2) errors propagate in the pipeline and if initial retrieval on the typo-induced query does not contain relevant documents, the pipeline's effectiveness as whole worsens regardless of the re-ranker. Note that (1) describes an advantage for BERT_O and T5_O over BERT_V and T5_V respectively, while (2) gives a disadvantage of BERT_V and T5_V over BERT_O

Category	Transform. heuristic	Valid variants
Misspelling	NeighbCharSwap	502K (99.83%)
	RandomCharSub	492K (97.92%)
	QWERTYCharSub	503K (99.98%)
Naturality	RemoveStopWords	448K (89.16%)
Ordering	RandomOrderSwap	502K (99.73%)

Table 3: The number of query variations generated for TREC DL '19's training set.

and $T5_O$. In practice, these findings show that, to evaluate only the re-ranking stage and mitigate (1), the number of initially retrieved documents should be sufficiently large (we chose 1000 but believe values between 100 and 1000 are sufficient). We leave investigating (2) for future research.

4.4 Training Robust Models

Lastly, we investigated if CBERT could be fine-tuned or E5 be prompted differently to produce more robust embeddings. To generate a training set, we applied the misspelling, ordering and RemoveStopWords transformations from Penha et al. to every query in the TREC DL'19 training set and applied automatic labeling: variants identical to the original query or variants that are empty strings are invalid and valid otherwise. Table 3 shows statistics of the new dataset. For training, we fine-tuned CBERT on the same loss objective it was trained with but using the new dataset. Since Mistral-7B-instruct, the model used by E5, is designed to follow instructions, we performed prompt-tuning (Lester et al., 2021). That is, we froze all the model's parameters and learned a prefix for the input embeddings. This fixed prefix replaces the instruction. E5 was trained using the same loss objective as CBERT and, for efficiency, we trained it 4bit-quantized. Appendix A.3 further presents an experiment on manually prompting E5 to be more typo-robust.

Table 4 and Figure 3 present the robustness and effectiveness of the tuned models when re-ranking the top 1000 passages returned by BM25, respectively. If the fine-tuning using noisy input behaved similar to Zheng and Saparov's few-shot prompting experiments with noisy examples, we would expect robustness and mean effectiveness on all variations except typos to improve. Table 4 shows that this is not the case: there are only negligible differences in mean effectiveness between CBERT's and E5's tuned and untuned variant.[8] Both models still

Query variation		ANTIQUE			
Category	Transform. heuristic	CBERT	$\text{CBERT}_{\text{Tuned}}$	E5	E5_{Tuned}
Original		.29	.27	**.41**	**.41**
Misspelling	NeighbCharSwap	.26*	.23*	.37*	**.38***
	RandomCharSub	.26*	.24*	**.37***	**.37***
	QWERTYCharSub	.26*	.25	**.38***	**.38***
Naturality	RemoveStopWords	.24*	.24*	**.36***	**.36***
	T5DescToTitle	.22*	.23*	.31*	**.33***
Ordering	RandomOrderSwap	.27	.27	**.39***	**.39***
Paraphrasing	BackTranslation	.28	.27	**.40**	.39
	T5QQP	.25	.25	**.37***	**.37**
	WordEmbedSynSwap	.26	.24*	**.40***	**.40***
	WordNetSynSwap	.24	.22	.34	**.35**

* significant difference (Bonferroni corrected two-sided paired Student's T-Test at $p < 5\%$) to ranking on the original query

Table 4: nDCG@10 of CBERT and E5 on ANTIQUE before and after fine-tuning on our training set. The models re-rank the top 1000 passages initially retrieved by BM25. The most effective model per variant is highlighted bold. See Table 8 in the appendix for details.

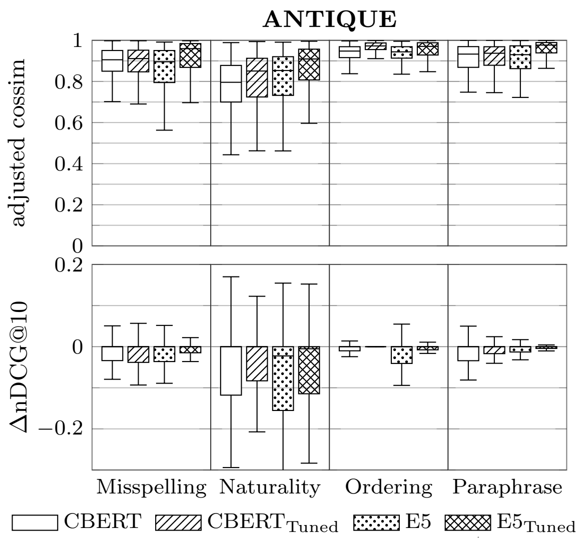

Figure 3: Robustness of CBERT and E5 before and after fine-tuning on our training set. Each model re-ranks the top 1000 passages retrieved by BM25.

exhibit statistically significant effectiveness degradation due to variations, yet Figure 3 highlights considerable improvements across both models in embedding robustness and ranking robustness in terms of the median and spread. The only exceptions are (1) CBERT's ranking robustness to misspellings, which can be expected since CBERT was previously fine-tuned specifically for this case, and (2) both models' ranking robustness to naturality which may be explained by Ethayarajh's finding that transformer based language models use stop

[8]This occurs irrespective the initial retriever (Appendix A.4).

words to aggregate their contexts, or Clark et al. who observe some of BERT's attention heads to have learned syntactic structures around stop words (e.g., prepositions attending to their objects).

Altogether, we find that our experiment shows promise in the feasibility of robustness to naturally occurring language phenomena; either by fine-tuning BERT-based models or solely by (soft-) prompting large language models differently. The models' problems with keyword queries indicate that embedding models may have to be trained (or even be pre-trained) on incomplete sentences more.

4.5 Summary

Reproducing Penha et al. (2022)'s work, we found a fault in their experimental setup but, ultimately, reaffirmed all their key findings. We generalized their experiments to a wider range of model architectures by investigating the robustness of SBERT, CharacterBERT-DR-ST ('CBERT'), E5 Mistral ('E5'), AnglE, and Ada v2 regarding typos, naturality, ordering and paraphrasing. Our results show that all these embedding models, like the transformer models tested by Penha et al., are most robust to paraphrasing and ordering, except for CBERT, which is slightly more robust to typos than to ordering. While not robust, the largest model we tested, E5, is the most effective and most robust model overall, but not always the most robust. We could further improve CBERT's and E5's robustness via fine-tuning and prompt-tuning respectively. Yet, this robustness did not entail improved effectiveness.

Though E5 is too large for efficient ad-hoc retrieval, its improved robustness through prompt-tuning promises interesting opportunities for instruct-LLMs in IR research beyond the context of this paper: For example, allowing users to specify instructions containing what aspects are most important in their query or designers of retrieval systems specifying additional preprocessing steps, e.g., 'Fix typos and retrieve the most relevant passages'. Other contemporary work began investigating these ideas: Zhuang et al. (2024) explore prompting LLMs for retrieval and Weller et al. (2024a,b) let users add an instruction to concretize the information need expressed in the query.

5 Conclusion

We investigated the robustness of transformer-based retrieval models to query variations using a test collection with syntactic query variations that keep semantics. We reproduced Penha et al.'s baseline results and extended them by assessing the robustness of (1) much larger models and (2) a model designed to be typo-robust.

While we were not able to completely replicate the results by Penha et al., we obtained similar results and reaffirmed their conclusions. We show that, while the typo-aware CharacterBERT model was the most robust to typos, this did not lead to robustness to other types of query variations. Finally, we observed that the largest model we tested, while dramatically less efficient than other models, was generally more robust or competitive with all the other tested models regarding all query variations. However, none of the models were robust to the query variations and all were the least robust to typos or keyword queries. Our results indicate that focusing on typo robustness alone is not enough and highlight the need for datasets like Penha et al.'s such that typos and keyword queries no longer are out-of-distribution for IR models.

Ethical Considerations

We do not see any particular ethical ramifications of our work.

Third Party Artifacts Beyond the third party artifacts previously mentioned and cited in the paper, we used the following frameworks: HuggingFace Transformers (Wolf et al., 2020), PyTorch Lightning (Falcon and The PyTorch Lightning team, 2019), NumPy (Harris et al., 2020), pandas (The pandas development team), PEFT (Mangrulkar et al., 2022), PyTorch (Ansel et al., 2024), and pytrec_eval (Van Gysel and de Rijke, 2018).

Limitations

The scope of our experiments was constrained by the number of models per model architecture category that could be sensibly evaluated and by the availability of suitable datasets. Nevertheless, our experiments align with the scope of similar studies and focus on evaluating representative models within each category. Further, note that the sample size for variations from Penha et al.'s TREC DL'19 dataset is limited, particularly for the WordNet-SynSwap transformation, which produced only 16 valid instances.

Due to memory constraints, both inference and training on E5 Mistral were performed using 4-bit quantization. Precomputing all document representations for the TREC DL'19 training split would have been time-prohibitive, such that we prompt-tuned E5 Mistral on approximately half the data that CharacterBERT-DR-ST was fine-tuned on. Despite this, the observed improvements across both models suggest the main conclusions drawn from the experiment remain valid.

Finally, this study evaluates the robustness of the re-ranking stage and not the entire retrieval pipeline. As such, the observed effectiveness results may vary if the initial retrieval stage lacks robustness (see Section 4.3).

References

Mario Almagro, Emilio Almazán, Diego Ortego, and David Jiménez. 2023. LEA: Improving Sentence Similarity Robustness to Typos Using Lexical Attention Bias. In *Proceedings of the 29th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, pages 36–46, Long Beach CA USA. ACM.

Jason Ansel, Edward Yang, Horace He, Natalia Gimelshein, Animesh Jain, Michael Voznesensky, Bin Bao, Peter Bell, David Berard, Evgeni Burovski, Geeta Chauhan, Anjali Chourdia, Will Constable, Alban Desmaison, Zachary DeVito, Elias Ellison, Will Feng, Jiong Gong, Michael Gschwind, Brian Hirsh, Sherlock Huang, Kshiteej Kalambarkar, Laurent Kirsch, Michael Lazos, Mario Lezcano, Yanbo Liang, Jason Liang, Yinghai Lu, CK Luk, Bert Maher, Yunjie Pan, Christian Puhrsch, Matthias Reso, Mark Saroufim, Marcos Yukio Siraichi, Helen Suk, Michael Suo, Phil Tillet, Eikan Wang, Xiaodong Wang, William Wen, Shunting Zhang, Xu Zhao, Keren Zhou, Richard Zou, Ajit Mathews, Gregory Chanan, Peng Wu, and Soumith Chintala. 2024. PyTorch 2: Faster Machine Learning Through Dynamic Python Bytecode Transformation and Graph Compilation. In *29th ACM International Conference on Architectural Support for Programming Languages and Operating Systems, Volume 2 (ASPLOS '24)*. ACM.

Peter Bailey, Alistair Moffat, Falk Scholer, and Paul Thomas. 2016. UQV100: A Test Collection with Query Variability. In *Proceedings of the 39th International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 725–728, Pisa Italy. ACM.

Hicham El Boukkouri, Olivier Ferret, Thomas Lavergne, Hiroshi Noji, Pierre Zweigenbaum, and Junichi Tsujii. 2020. CharacterBERT: Reconciling ELMo and BERT for Word-Level Open-Vocabulary Representations From Characters. In *Proceedings of the 28th International Conference on Computational Linguistics, COLING 2020, Barcelona, Spain (Online), December 8-13, 2020*, pages 6903–6915. International Committee on Computational Linguistics.

Kevin Clark, Urvashi Khandelwal, Omer Levy, and Christopher D. Manning. 2019. What Does BERT Look At? An Analysis of BERT's Attention. In *Proceedings of the 2019 ACL Workshop BlackboxNLP: Analyzing and Interpreting Neural Networks for NLP, BlackboxNLP@ACL 2019, Florence, Italy, August 1, 2019*, pages 276–286. Association for Computational Linguistics.

Nick Craswell, Bhaskar Mitra, Emine Yilmaz, Daniel Campos, and Ellen M. Voorhees. 2020. Overview of the TREC 2019 deep learning track. In *Text REtrieval Conference (TREC)*.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, NAACL-HLT 2019, Minneapolis, MN, USA, June 2-7, 2019, Volume 1 (Long and Short Papers)*, pages 4171–4186. Association for Computational Linguistics.

Kawin Ethayarajh. 2019. How Contextual are Contextualized Word Representations? Comparing the Geometry of BERT, ELMo, and GPT-2 Embeddings. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 55–65, Hong Kong, China. Association for Computational Linguistics.

William Falcon and The PyTorch Lightning team. 2019. PyTorch Lightning.

Ryan Greene, Ted Sanders, Lilian Weng, and Arvind Neelakantan. 2022. New and improved embedding model. https://openai.com/blog/new-and-improved-embedding-model.

Charles R. Harris, K. Jarrod Millman, Stéfan J van der Walt, Ralf Gommers, Pauli Virtanen, David Cournapeau, Eric Wieser, Julian Taylor, Sebastian Berg, Nathaniel J. Smith, Robert Kern, Matti Picus, Stephan Hoyer, Marten H. van Kerkwijk, Matthew Brett, Allan Haldane, Jaime Fernández del Réo, Mark Wiebe, Pearu Peterson, Pierre Gérard-Marchant, Kevin Sheppard, Tyler Reddy, Warren Weckesser, Hameer Abbasi, Christoph Gohlke, and Travis E. Oliphant. 2020. Array programming with NumPy. *Nature*, 585:357?362.

Helia Hashemi, Mohammad Aliannejadi, Hamed Zamani, and W. Bruce Croft. 2020. ANTIQUE: A Non-Factoid Question Answering Benchmark. In *Advances in Information Retrieval - 42nd European Conference on IR Research, ECIR 2020, Lisbon, Portugal, April 14-17, 2020, Proceedings, Part II*, volume 12036 of *Lecture Notes in Computer Science*, pages 166–173. Springer.

Albert Q. Jiang, Alexandre Sablayrolles, Arthur Mensch, Chris Bamford, Devendra Singh Chaplot, Diego de Las Casas, Florian Bressand, Gianna Lengyel, Guillaume Lample, Lucile Saulnier, Lélio Renard Lavaud, Marie-Anne Lachaux, Pierre Stock, Teven Le Scao, Thibaut Lavril, Thomas Wang, Timothée Lacroix, and William El Sayed. 2023. Mistral 7b. *CoRR*, abs/2310.06825.

Jan Kocoń, Igor Cichecki, Oliwier Kaszyca, Mateusz Kochanek, Dominika Szydło, Joanna Baran, Julita Bielaniewicz, Marcin Gruza, Arkadiusz Janz, Kamil Kanclerz, Anna Kocoń, Bartłomiej Koptyra, Wiktoria Mieleszczenko-Kowszewicz, Piotr Miłkowski, Marcin Oleksy, Maciej Piasecki, Łukasz Radliński, Konrad Wojtasik, Stanisław Woźniak, and Przemysław Kazienko. 2023. ChatGPT: Jack of all trades, master of none. *Information Fusion*, 99:101861.

Brian Lester, Rami Al-Rfou, and Noah Constant. 2021. The Power of Scale for Parameter-Efficient Prompt Tuning. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing, EMNLP 2021, Virtual Event / Punta Cana, Dominican Republic, 7-11 November, 2021*, pages 3045–3059. Association for Computational Linguistics.

Xianming Li and Jing Li. 2023. AnglE-optimized Text Embeddings. *Preprint*, arxiv:2309.12871.

Sean MacAvaney, Andrew Yates, Sergey Feldman, Doug Downey, Arman Cohan, and Nazli Goharian. 2021. Simplified data wrangling with ir_datasets. In *SIGIR*.

Sourab Mangrulkar, Sylvain Gugger, Lysandre Debut, Younes Belkada, Sayak Paul, and Benjamin Bossan. 2022. PEFT: State-of-the-art Parameter-Efficient Fine-Tuning methods. https://github.com/huggingface/peft.

Jiaqi Mu and Pramod Viswanath. 2018. All-but-the-Top: Simple and Effective Postprocessing for Word Representations. In *6th International Conference on Learning Representations, ICLR 2018, Vancouver, BC, Canada, April 30 - May 3, 2018, Conference Track Proceedings*. OpenReview.net.

Niklas Muennighoff, Nouamane Tazi, Loïc Magne, and Nils Reimers. 2023. MTEB: Massive Text Embedding Benchmark. In *Proceedings of the 17th Conference of the European Chapter of the Association for Computational Linguistics, EACL 2023, Dubrovnik, Croatia, May 2-6, 2023*, pages 2006–2029. Association for Computational Linguistics.

Tri Nguyen, Mir Rosenberg, Xia Song, Jianfeng Gao, Saurabh Tiwary, Rangan Majumder, and Li Deng. 2016. MS MARCO: A human generated machine reading comprehension dataset. In *Proceedings of the Workshop on Cognitive Computation: Integrating neural and symbolic approaches 2016 co-located with the 30th Annual Conference on Neural Information Processing Systems (NIPS 2016), Barcelona, Spain, December 9, 2016*, volume 1773 of *CEUR Workshop Proceedings*. CEUR-WS.org.

Gustavo Penha, Arthur Câmara, and Claudia Hauff. 2022. Evaluating the Robustness of Retrieval Pipelines with Query Variation Generators. In *Advances in Information Retrieval*, Lecture Notes in Computer Science, pages 397–412, Cham. Springer International Publishing.

Yao Qiang, Subhrangshu Nandi, Ninareh Mehrabi, Greg Ver Steeg, Anoop Kumar, Anna Rumshisky, and Aram Galstyan. 2024. Prompt Perturbation Consistency Learning for Robust Language Models. In *Findings of the Association for Computational Linguistics: EACL 2024, St. Julian's, Malta, March 17-22, 2024*, pages 1357–1370. Association for Computational Linguistics.

Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, and Peter J. Liu. 2020. Exploring the Limits of Transfer Learning with a Unified Text-to-Text Transformer. *J. Mach. Learn. Res.*, 21:140:1–140:67.

Abhilasha Ravichander, Siddharth Dalmia, Maria Ryskina, Florian Metze, Eduard Hovy, and Alan W. Black. 2021. NoiseQA: Challenge Set Evaluation for User-Centric Question Answering. In *Proceedings of the 16th Conference of the European Chapter of the Association for Computational Linguistics: Main Volume, EACL 2021, Online, April 19 - 23, 2021*, pages 2976–2992. Association for Computational Linguistics.

Nils Reimers and Iryna Gurevych. 2019. Sentence-BERT: Sentence Embeddings using Siamese BERT-Networks. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing, EMNLP-IJCNLP 2019, Hong Kong, China, November 3-7, 2019*, pages 3980–3990. Association for Computational Linguistics.

Victor Sanh, Lysandre Debut, Julien Chaumond, and Thomas Wolf. 2019. Distilbert, a distilled version of BERT: smaller, faster, cheaper and lighter. *CoRR*, abs/1910.01108.

Georgios Sidiropoulos and Evangelos Kanoulas. 2022. Analysing the Robustness of Dual Encoders for Dense Retrieval Against Misspellings. *Proceedings of the 45th International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 2132–2136.

Georgios Sidiropoulos and Evangelos Kanoulas. 2024. Improving the Robustness of Dense Retrievers Against Typos via Multi-Positive Contrastive Learning. In *Advances in Information Retrieval - 46th European Conference on Information Retrieval, ECIR 2024, Glasgow, UK, March 24-28, 2024, Proceedings, Part III*, volume 14610 of *Lecture Notes in Computer Science*, pages 297–305. Springer.

Georgios Sidiropoulos, Svitlana Vakulenko, and Evangelos Kanoulas. 2022. On the Impact of Speech Recognition Errors in Passage Retrieval for Spoken

Question Answering. In *Proceedings of the 31st ACM International Conference on Information & Knowledge Management, Atlanta, GA, USA, October 17-21, 2022*, pages 4485–4489. ACM.

Panuthep Tasawong, Wuttikorn Ponwitayarat, Peerat Limkonchotiwat, Can Udomcharoenchaikit, Ekapol Chuangsuwanich, and Sarana Nutanong. 2023. Typo-Robust Representation Learning for Dense Retrieval. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers), ACL 2023, Toronto, Canada, July 9-14, 2023*, pages 1106–1115. Association for Computational Linguistics.

The pandas development team. pandas-dev/pandas: Pandas.

Christophe Van Gysel and Maarten de Rijke. 2018. Pytrec_eval: An Extremely Fast Python Interface to trec_eval. In *SIGIR*. ACM.

Liang Wang, Nan Yang, Xiaolong Huang, Linjun Yang, Rangan Majumder, and Furu Wei. 2024. Improving text embeddings with large language models. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), ACL 2024, Bangkok, Thailand, August 11-16, 2024*, pages 11897–11916. Association for Computational Linguistics.

Peiling Wang, Michael W. Berry, and Yiheng Yang. 2003. Mining longitudinal web queries: Trends and patterns. *Journal of the American Society for Information Science and Technology*, 54(8):743–758.

Orion Weller, Benjamin Chang, Sean MacAvaney, Kyle Lo, Arman Cohan, Benjamin Van Durme, Dawn Lawrie, and Luca Soldaini. 2024a. FollowIR: Evaluating and Teaching Information Retrieval Models to Follow Instructions. *Preprint*, arxiv:2403.15246.

Orion Weller, Benjamin Van Durme, Dawn Lawrie, Ashwin Paranjape, Yuhao Zhang, and Jack Hessel. 2024b. Promptriever: Instruction-Trained Retrievers Can Be Prompted Like Language Models. *Preprint*, arXiv:2409.11136.

Ryen W. White, Matthew Richardson, and Wen-tau Yih. 2015. Questions vs. Queries in Informational Search Tasks. In *Proceedings of the 24th International Conference on World Wide Web*, pages 135–136, Florence Italy. ACM.

Thomas Wolf, Lysandre Debut, Victor Sanh, Julien Chaumond, Clement Delangue, Anthony Moi, Pierric Cistac, Tim Rault, Remi Louf, Morgan Funtowicz, Joe Davison, Sam Shleifer, Patrick von Platen, Clara Ma, Yacine Jernite, Julien Plu, Canwen Xu, Teven Le Scao, Sylvain Gugger, Mariama Drame, Quentin Lhoest, and Alexander Rush. 2020. Transformers: State-of-the-Art Natural Language Processing. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, pages 38–45, Online. Association for Computational Linguistics.

Hongyi Zheng and Abulhair Saparov. 2023. Noisy Exemplars Make Large Language Models More Robust: A Domain-Agnostic Behavioral Analysis. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 4560–4568, Singapore. Association for Computational Linguistics.

Shengyao Zhuang, Xueguang Ma, Bevan Koopman, Jimmy Lin, and Guido Zuccon. 2024. PromptReps: Prompting Large Language Models to Generate Dense and Sparse Representations for Zero-Shot Document Retrieval. *Preprint*, arxiv:2404.18424.

Shengyao Zhuang, Xinyu Mao, and Guido Zuccon. 2022. Robustness of Neural Rankers to Typos: A Comparative Study. In *Proceedings of the 26th Australasian Document Computing Symposium*, pages 1–6, Adelaide SA Australia. ACM.

Shengyao Zhuang, Linjun Shou, Jian Pei, Ming Gong, Houxing Ren, Guido Zuccon, and Daxin Jiang. 2023. Typos-aware Bottlenecked Pre-Training for Robust Dense Retrieval. In *Annual International ACM SIGIR Conference on Research and Development in Information Retrieval in the Asia Pacific Region, SIGIR-AP 2023, Beijing, China, November 26-28, 2023*, pages 212–222. ACM.

Shengyao Zhuang and Guido Zuccon. 2021. Dealing with Typos for BERT-based Passage Retrieval and Ranking. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing, EMNLP 2021, Virtual Event / Punta Cana, Dominican Republic, 7-11 November, 2021*, pages 2836–2842. Association for Computational Linguistics.

Shengyao Zhuang and Guido Zuccon. 2022. Character-BERT and Self-Teaching for Improving the Robustness of Dense Retrievers on Queries with Typos. In *SIGIR '22: The 45th International ACM SIGIR Conference on Research and Development in Information Retrieval, Madrid, Spain, July 11 - 15, 2022*, pages 1444–1454. ACM.

A Appendix

A.1 Model Descriptions

Table 6 presents the number of parameters and embedding dimensionality of each model we tested.

SBERT (Reimers and Gurevych, 2019) SBERT uses BERT's mean-pooled final hidden representation as a sentence embedding. We use the `msmarco-distilbert-cos-v5` checkpoint, which fine-tuned DistilBERT (Sanh et al., 2019) on MS MARCO Passage (Nguyen et al., 2016).

CBERT (Zhuang and Zuccon, 2022) CBERT replaces BERT's WordPiece Tokenizer with character level tokenization and is trained on a typo-induced version of MS MARCO Passage (Nguyen et al., 2016) using a proposed 'Self-Teaching' objective: The model's output on the original queries is used as a target for the output on the typoed queries. To achieve this, the loss has two components: 1) using KL-divergence to learn similar relevance-score distributions for original and typoed queries and 2) using supervised contrastive cross-entropy-loss to learn an effective ranker on the original queries.

E5 (Wang et al., 2024) To fine-tune Mistral-7B (Jiang et al., 2023) for the generation of embeddings, Wang et al. prompt GPT-4 to generate text retrieval tasks together with synthetic training data and fine-tune the official Mistral-7B checkpoint on this data using contrastive cross entropy loss. While the generated training data contains training samples in 93 different languages, the authors point out that, due to Mistral-7B's pre-training on predominantly English texts and since over 40% of training samples of the synthetic dataset are in English, $\text{E5}_{\text{mistral-7b}}$, which we call E5 for brevity, is not a multilingual model. E5 is by far the largest model we tested and represents the class of state-of-the-art LLM-based embedding models.

AnglE (Li and Li, 2023) Training objectives for embedding models often aim to learn a cosine similarity of 1 for two similar inputs and 0 otherwise. Since the cosine function is quite flat around these values, the gradient vanishes close to the targets, and it gets harder to improve the model further. To mitigate this Li and Li propose the 'Angle Objective' which interprets the d-dimensional real-valued embedding vectors as $\frac{d}{2}$-dimensional complex-valued vectors to compute the angular distance between two vectors while avoiding vanishing gradients. For our experiments, we use the `UAE-Large-V1` variant, which is a fine-tuned version of $\text{BERT}_{\text{LARGE}}$ (Devlin et al., 2019).

Ada v2 (Greene et al., 2022) OpenAI's text embedding model `text-embedding-ada-002`, which we call Ada v2 for ease of reading, was OpenAI's latest text embedding model at the time of our experiments. Although the training regime and model architecture are undisclosed, we include Ada v2 since it represents an interesting category: the state-of-the-art commercial embedding models.

Model	Expected cosine similarity	
	TREC DL'19	ANTIQUE
SBERT	0.036	0.050
CBERT	0.731	0.732
$\text{CBERT}_{\text{Tuned}}$	0.737	0.741
E5	0.555	0.568
E5_{Tuned}	0.243	0.243
AnglE	0.377	0.373
Ada v2	0.654	0.688

Table 5: The expected cosine similarity for each of the embedding models we tested.

Model	# Params	Embed. Dim.
SBERT	66M	768
CBERT	104M	768
E5	7110M	4096
AnglE	335M	1024
Ada v2	N/A	1536

Table 6: Number of parameters and embedding dimensionality of the tested models.

A.2 Anisotropy in Embedding Models

Table 5 presents the expected cosine similarity if two random queries were embedded using each of the embedding models we tested. Intuitively, one would expect a similarity of 0 for unrelated embeddings. However, as Table 5 shows and Mu and Viswanath (2018); Ethayarajh (2019) observed for static and contextualized embeddings respectively, this does not hold since embedding models' outputs often are not uniformly distributed around the origin (the embeddings are 'anisotropic') but directionally localized (Ethayarajh, 2019). Thus, cosine similarity is difficult to compare across embedding models: As seen in Figure 4, a cosine similarity of 0.73 is quite high for SBERT but indicates unrelated semantics for CBERT.

Ethayarajh (2019) observed similar anisotropy of embeddings with LLMs and most notably found that embeddings generated by GPT2 for any two randomly chosen words have a near perfect expected cosine similarity. To compare embedding similarity across models, Ethayarajh subtract μ, the expected cosine similarity given two random

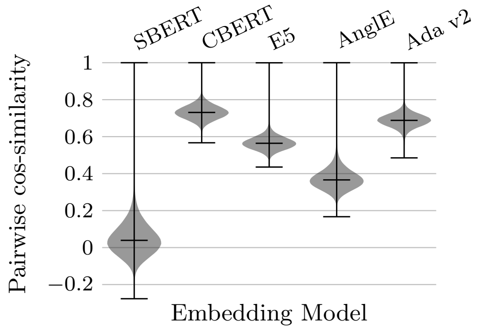

Figure 4: Cosine similarity between the embeddings of each query and every query variation in the dataset.

words. To keep these values within the 0-1 range, we additionally normalize cosine similarity for *anisotropy adjusted cosine similarity* (as defined in Equation (1)), where we calculate μ as the mean cosine similarity of any two queries in the variation dataset. Our values for μ are given in Table 5.

Note that anisotropy may not result in poor ranking robustness or worse effectiveness – E5 maps embeddings to a similarly small range as CBERT and Ada v2 but its ranking-robustness and effectiveness is the highest among all tested models. Anisotropy is, however, suboptimal, as Mu and Viswanath (2018) discovered that making embeddings more isotropic by simply subtracting a common mean vector improved effectiveness on general natural language processing tasks. We could not observe any improvements when applying their algorithm (Mu and Viswanath, 2018, Algorithm 1) to retrieval, likely because they study static embeddings and all models we tested are contextualized.

A.3 Promptable Embedding Models

Mistral's instructable nature teases a novel concept: promptable embedding models. Embeddings generated for retrieval should be asymmetric, i.e., the same text should not be mapped to the same embedding when embedded as a query and as a document. Otherwise, when the most relevant documents are retrieved by similarity, the query would rank the highest though it does not fulfill the information need. To mitigate this, Li and Li (2023) and Wang et al. (2024) recommend prompting the queries' generation for AnglE and E5 respectively.

Since Mistral, like other LLMs, shows unprecedented zero-shot effectiveness in general natural language tasks, we investigated if we could generate robust embeddings solely by prompting the Mistral-based embedding model E5 differently. Wang et al. (2024) recommend prompts of the form

Instruction	nDCG@10		
	Orig.	Typo	Δ
`Given a web search query, retrieve relevant passages that answer the query`	.71	.66	.050
`Given a web search query, fix typos and retrieve relevant passages that answer the query`	.70	.65	.056
`Synthesize the ideal query to express the given informationneed and retrieve relevant passages for it`	.72	.66	.067
`Do what you want`	.55	.44	.112

Table 7: A selection of the instructions we used to instruct E5's query embedding generation. The first instruction is the one recommended by Wang et al. (2024).

`Instruct: {instuction}\nQuery: {query}`, where `{instruction}` and `{query}` are replaced with the instruction and query respectively and `\n` marks a line break. To assess the instruction's impact on ranking effectiveness and robustness, we evaluated different instructions on the typo-induced dataset by Zhuang and Zuccon (2022). We focus on typos since E5 is least robust to these on TREC DL '19, and we avoid using the Penha et al. (2022) dataset as not to fit a prompt on a test-set.

A subset of the instructions we tested is shown in Table 7 and every instruction and their effectiveness on the original queries and typo-induced queries are plotted in Figure 5. The figure shows that the original instruction by the authors is already quite robust, and more effective rankings on the original queries are more robust to typos as well (instructions further to the right on the x-axis are brighter, i.e., closer to the ideal line). This shows that the zero-shot effectiveness observed with state-of-the-art LLMs may not translate to promptable embedding models based on these LLMs but choosing the right instruction can improve effectiveness and robustness. We have not found an instruction that improves ranking robustness considerably beyond what the author's instruction achieves, yet we can not rule out that such a prompt may exist. On the contrary, we observe the positive trend that more effective prompts improve robustness.

A.4 Different Initial Retrieval

To investigate effects induced by the initial retrieval, we further evaluated CBERT, E5 and their tuned variants with ColBERT v2 for initial retrieval. Both effectiveness and robustness behave similarly however, irrespective the initial retrieval model, as seen in Figure 6 and Table 8. This probably stems from the large number of initially retrieved passages.

ANTIQUE

Query variation		BM25 + ...				ColBERT v2 + ...			
Category	Transform. heuristic	CBERT	$\text{CBERT}_{\text{Tuned}}$	E5	E5_{Tuned}	CBERT	$\text{CBERT}_{\text{Tuned}}$	E5	E5_{Tuned}
Original		0.29	0.27	0.41	0.41	0.28	0.26	0.44	0.43
Misspelling	NeighbCharSwap	0.26*	0.23*	0.37*	0.38*	0.24*	0.22*	0.40*	0.40*
	RandomCharSub	0.26*	0.24*	0.37*	0.37*	0.25*	0.23*	0.39*	0.40*
	QWERTYCharSub	0.26*	0.25	0.38*	0.38*	0.25*	0.24	0.40*	0.40*
Naturality	RemoveStopWords	0.24*	0.24*	0.36*	0.36*	0.24*	0.23*	0.37*	0.39*
	T5DescToTitle	0.22*	0.23*	0.31*	0.33*	0.23*	0.22*	0.33*	0.35*
Ordering	RandomOrderSwap	0.27	0.27	0.39*	0.39*	0.26	0.25	0.41*	0.41*
Paraphrasing	BackTranslation	0.28	0.27	0.40	0.39	0.28	0.25	0.44	0.42
	T5QQP	0.25	0.25	0.37*	0.37	0.24	0.24	0.40*	0.39
	WordEmbedSynSwap	0.26	0.24*	0.40*	0.40*	0.25*	0.22*	0.42*	0.42*
	WordNetSynSwap	0.24	0.22	0.34	0.35*	0.23	0.20	0.36	0.38

* significant difference (Bonferroni corrected two-sided paired Student's T-Test at $p < 5\%$) to ranking on the original query

Table 8: Mean nDCG@10 on ANTIQUE of CBERT and E5 before and after fine-tuning on our training set. The models re-rank the top 1000 passages initially retrieved by BM25 and ColBERT v2.

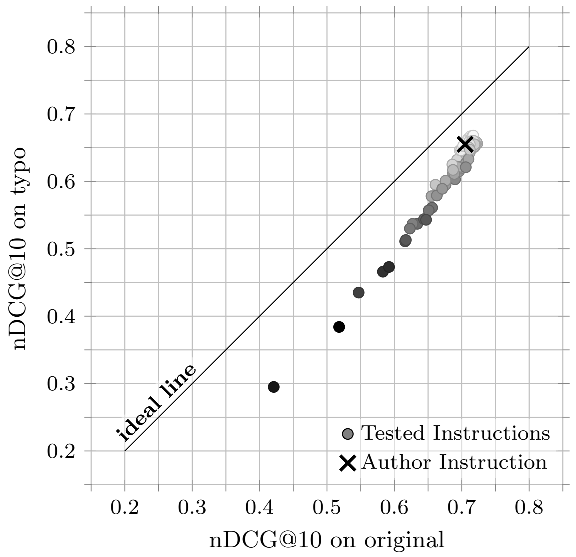

Figure 5: Prompt tuning results. Each point marks a prompt's effectiveness in face and absence of typos. The cross marks the prompt used by E5's authors. Points further from the ideal line (same effectiveness regardless of typos) are darker.

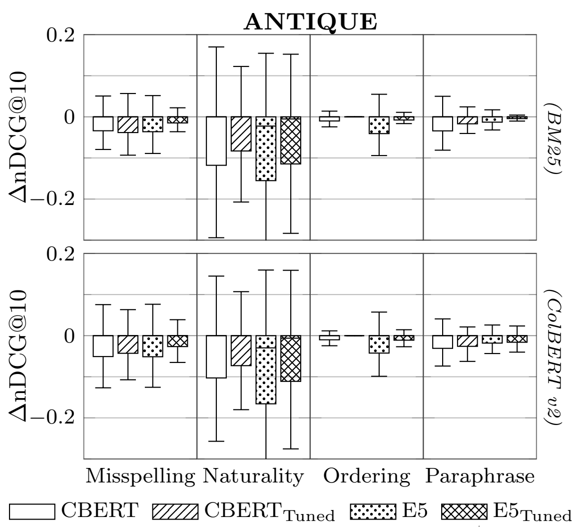

Figure 6: Ranking-robustness of CBERT and E5 before and after fine-tuning on our training set. Each model re-ranks the top 1000 passages retrieved by BM25 (top) and ColBERT v2 (bottom).