# Adversarial BEIR: Benchmarking Information Retrieval Models Against Query Perturbations

Anonymous ACL submission

## Abstract

Information retrieval plays a crucial role in many applications, serving as the primary mechanism for accessing relevant data within large and complex datasets. This study investigates the robustness of retrievers against adversarial queries, employing 17 distinct query perturbation techniques across three granularity levels: character, word, and sentence. Our findings reveal that top-performing retrievers exhibit significant vulnerabilities to these adversarial queries, resulting in notable performance degradation. Additionally, we explore the capability of Large Language Models (LLMs) to generate adversarial queries autonomously, without human intervention. By prompting LLMs to create paraphrases of queries and subsequently annotating these using both automated and manual methods, we assess their effectiveness in this task. We introduce **Adversarial BEIR**, a comprehensive benchmark for measuring the robustness of retrievers to adversarial queries. By sharing our benchmark and detailed methods, we enable researchers to evaluate the robustness of their retrievers and create additional adversarial samples.

## 1 Introduction

In a world where we are surrounded by vast amounts of data, efficient retrieval is a key element of information systems such as RAG (Lewis et al., 2020). Pre-trained language models have proven their worth in the field of information retrieval in recent years (Xiao et al., 2024; Li et al., 2023c; Wang et al., 2024a,b). However, relatively small changes in the input can result in outputs that are not in line with expectations (Lin et al., 2025; Zhong et al., 2024). While newly introduced retrieval models achieve high performance, their robustness to adversarial query perturbations remains underexplored. Evaluating these models against perturbed queries is crucial to understanding their real-world reliability.

Deep neural networks (DNNs) have been shown to be vulnerable to adversarial examples (Goodfellow et al., 2014; Kurakin et al., 2016; Goswami et al., 2018). Robustness to adversarial samples has been widely studied in Natural Language Processing (NLP) field, with various works exploring attacks on textual inputs and methods for improving model resilience. Recent research has examined adversarial robustness of Large Language Models (Wei et al., 2023; Jones et al., 2023) and techniques to improve it (Agrawal et al., 2025).

Although some studies have evaluated retrieval models against adversarial queries (Penha et al., 2022; Li et al., 2023b), there is no standardized benchmark that allows users to assess the quality of their retrieval systems against adversarial examples. Moreover, there is no information on how models currently considered as state-of-the-art perform on adversarial queries. This motivates us to build a unified robustness evaluation benchmark for information retrieval models. Our contributions are as follows:

1. We introduce Adversarial BEIR, a benchmark comprising adversarial queries generated using 17 different construction methods.

2. We conduct a comprehensive robustness evaluation of state-of-the-art information retrieval models.

3. We explore the feasibility of leveraging LLMs to automatically generate adversarial queries and manually assess their quality.

We release the code, data, and reproduction scripts to facilitate the application of all methods used in this work[1] and plan to provide a streamlined framework for evaluating retrieval models with our benchmark observations.

[1] https://anonymous.4open.science/r/AdvBEIR-BBD3

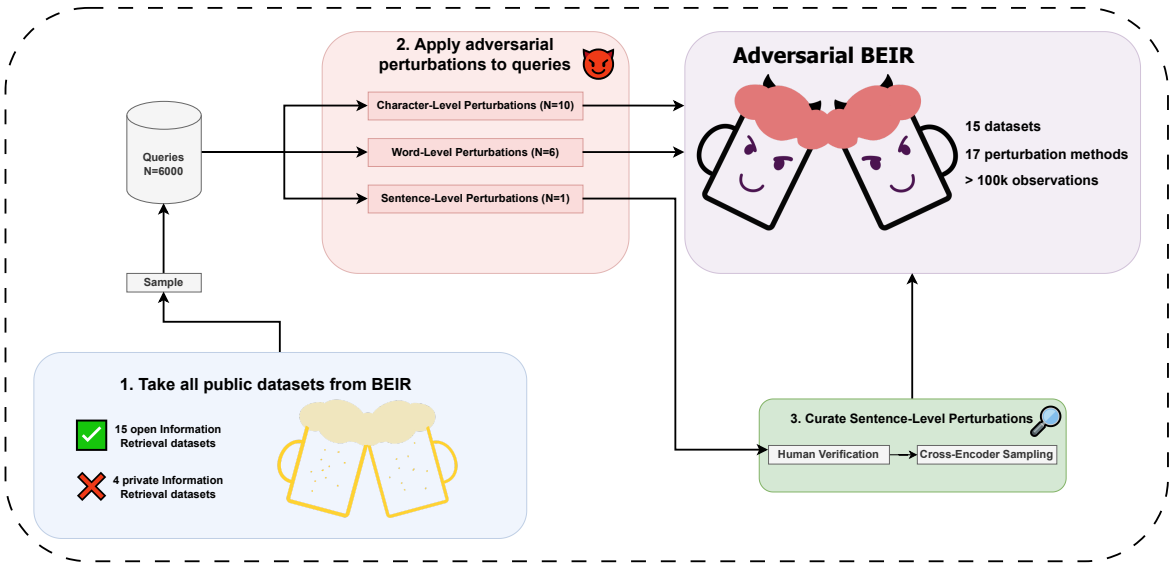


Figure 1: Our benchmark involves sampling 6000 observations from the public BEIR datasets. We apply 17 query perturbation methods across three levels of granularity. Sentence-level perturbations are verified post-generation and sampled based on Cross Encoders' supervision. In total, we generate over 100,000 test queries to evaluate the robustness of retrieval models.

## 2 Related work

**Adversarial attacks on textual inputs.** Even relatively minor perturbations, such as typos in textual input, can substantially affect model performance (Belinkov and Bisk, 2018; Rychalska et al., 2019). In the domain of information retrieval, neural-based models have historically struggled to retain performance when exposed to adversarial data (Wu et al., 2022; Zhuang and Zuccon, 2021). Applying semantic- and character-based perturbations to widely used passage ranking datasets causes a loss in retrieval quality, particularly affecting short queries (Campos et al., 2023). Perturbed data can not only serve as a foundation for evaluation, but can also enhance the training process, improving the overall robustness of a model (Rychalska et al., 2019; Tomonari et al., 2022).

**Adversarial benchmarks in NLP.** Previous research on the robustness of NLP systems often involved *ad hoc* input modifications created using evaluation toolkits such as TextAttack (Morris et al., 2020) or OpenAttack (Zeng et al., 2021) or other unsupervised methods like automated paraphrase generation (Campos et al., 2023). However, aside from the transient nature of such data, it has been shown that widely used textual perturbations can generate invalid samples (Zang et al., 2020), which undermines the validity of the research conducted. This emphasizes the need for consistent, reliable, and reusable adversarial datasets for robustness evaluation. Such motivations have driven research like ANLI dataset (Nie et al., 2020), where a human-and-model-in-the-loop approach is employed to generate difficult-to-assess samples. Similarly, AdvGLUE (Wang et al., 2021) builds upon the widely used Natural Language Understanding benchmark GLUE (Wang et al., 2018), providing an adversarial benchmark dataset that facilitates the systematic examination of perturbations' impact. However, this type of work is currently lacking in the context of information retrieval.

## 3 Dataset

### 3.1 Overview

We present the overview of Adversarial BEIR in Figure 1. Our benchmark dataset consists of 6000 observations covering various fields, such as medicine, finance or climate. It has been sampled using all publicly available datasets from the BEIR (Thakur et al., 2021) benchmark. We compare the original distribution of datasets in the initial scenario (collection of all datasets) and our benchmark version (after sampling) and provide corresponding descriptive statistics in Table 1. The sampling procedure is as follows:

1. Take all publicly available BEIR datasets.
2. Draw 300 initial samples from each dataset (or all if there are less than 300) to assure that

| Dataset | Percentage | | Query Length | | Document Length | |
|---|---|---|---|---|---|---|
| | Initial | Benchmark | Mean | Std | Mean | Std |
| MSMARCO | 13.08 | 9.52 | 6.08 | 2.53 | 64.65 | 25.58 |
| TREC-COVID | 0.09 | 0.83 | 11.58 | 3.63 | 171.78 | 158.21 |
| NFCorpus | 0.61 | 5.02 | 3.56 | 2.83 | 258.75 | 99.43 |
| NQ | 6.47 | 7.13 | 9.25 | 1.73 | 91.65 | 70.05 |
| HotpotQA | 13.88 | 9.8 | 19.79 | 10.73 | 51.35 | 37.01 |
| FiQA-2018 | 1.21 | 5.23 | 12.19 | 5.03 | 157.03 | 151.7 |
| ArguAna | 2.64 | 5.75 | 224.23 | 106.8 | 189.25 | 104.49 |
| Touche-2020 | 0.09 | 0.82 | 7.55 | 1.98 | 335.38 | 456.53 |
| CQADupstack | 24.64 | 13.68 | 9.78 | 4.29 | 177.79 | 208.74 |
| Quora | 18.74 | 11.55 | 10.84 | 4.49 | 13.03 | 7.07 |
| DBPedia | 0.75 | 5.07 | 5.77 | 3.07 | 54.59 | 27.69 |
| SCIDOCS | 1.87 | 5.47 | 9.84 | 3.67 | 188.43 | 137.87 |
| FEVER | 12.49 | 9.3 | 9.41 | 3.75 | 94.96 | 124.33 |
| Climate-FEVER | 2.88 | 5.83 | 22.82 | 10.72 | 94.96 | 124.33 |
| SciFact | 0.56 | 5 | 13.83 | 5.3 | 232.03 | 102.56 |

Table 1: Statistics of datasets used in our benchmark, including query and document length distributions and dataset proportions before and after the sampling process. The mean and standard deviation of the query length were calculated on the queries before sampling the benchmark.

sufficient amount of observations from each collection will be included in the benchmark.

3. Redraw from the remaining samples according to the scaling factor (size proportion of specific datasets after the initial sampling) to top up to the desired number of 6000 samples.

Number of samples from original BEIR has been matched to our capabilities of manual annotation, which was necessary for some perturbations.

### 3.2 Query Perturbation

Query perturbation is a technique used to modify a query by introducing intentional changes or distortions to its content, which involve altering characters, words, or whole sentences. Perturbed queries help us understand the behavior of models and assess the robustness of information retrieval systems in a wide range of real world scenarios.

To ensure that our benchmark effectively assesses search engine resilience, we developed 17 different methods for creating perturbed queries, drawing inspiration from previous research, as well as developing our own approaches to modifying queries. Information about perturbations contained in our benchmark is presented in Table 2. We focus on three levels of perturbations: character, word and sentence which modify individual characters, words or entire queries respectively.

Each method, apart from Automatic Paraphrase (P17), operates on its specific perturbation strength. We define perturbation strength as the fraction of characters or words (depending on perturbation level) that are affected by a specific modification. In some scenarios, such as OCR Error (P5) or Word Lemmatization (P14), there is a limited number of places where a perturbation can be applied. In such cases, perturbation strength stands for the percentage of positions that qualify for a perturbation and will be edited. To perform Context-Aware Perturbation (P11), we apply a methodology similar to CLARE (Li et al., 2021).

### 3.3 Creating sentence-level query perturbations automatically

Recent studies have explored methods for automatically expanding and paraphrasing queries. For instance, Alaofi et al. (2023) use a LLM to generate synthetic queries and verify their similarity to human-generated ones. Our Automatic Paraphrase perturbation is similar to query refinement methods studied in various works. Chan et al. (2024) enhance the model to rewrite, decompose, and clarify queries in the RAG scenario. Li et al. (2023b) refine queries with LLM to represent different demographic groups and iteratively verify new queries. In our work, besides measuring retrieval model robustness against adversarial queries, we aim to address three additional research questions:

R1 How accurate are modern Language Models in query paraphrasing?

| Symbol | Name | Level | Description |
|---|---|---|---|
| P1 | Capitalization | Character | Capitalizes characters. |
| P2 | Keyboard Character Insert | Character | Inserts adjacent keyboard characters next to chosen ones. |
| P3 | Keyboard Character Replace | Character | Replaces characters with adjacent keyboard characters. |
| P4 | Random Neighbor Character Swap | Character | Swaps characters with their neighboring characters in text. |
| P5 | OCR Error | Character | Simulates OCR errors by distorting characters. |
| P6 | Punctuation | Character | Inserts / deletes / replaces punctuation marks. |
| P7 | Random Character Delete | Character | Deletes a percentage of characters. |
| P8 | Random Character Insert | Character | Inserts random characters at a percentage of positions. |
| P9 | Random Character Replace | Character | Replaces a percentage of characters. |
| P10 | Mobile Phone Character Miss | Character | Replaces characters with their corresponding symbol. |
| P11 | Context-Aware Perturbation | Word | Inserts, merges or deletes semantically similar words. |
| P12 | Word Duplicate | Word | Duplicates words. |
| P13 | Words Join | Word | Joins adjacent words. |
| P14 | Word Lemmatization | Word | Converts words to their lemma. |
| P15 | Word Position Swap | Word | Swaps positions of selected word pairs. |
| P16 | Word Stemming | Word | Applies stemming to words. |
| P17 | Automatic Paraphrase | Sentence | Paraphrases query using a LLM or backtranslation. |

Table 2: List of perturbations applied to the queries from our benchmark dataset.

| Dataset | Method | Original Query | Perturbed Query |
|---|---|---|---|
| HotpotQA | P4 | Which system of parliament **was** modeled after **the** United Kingdom and is **also** used in Canada? | Which system of parliament **aws** modeled after **teh** United Kingdom and is **alos** used in Canada? |
| NQ | P7 | where do they get the hair for **a** hair transp**l**ant | where do they get the hair for hair transpant |
| Touche-2020 | P11 | Should more gun control laws be **enacted**? | Should more gun control laws be **passed**? |
| CQADupstack | P16 | Why do **large** IT projects tend to fail or have big cost/**schedule overruns**? | Why do **larg** IT project tend to fail or have big cost/**schedul overrun**? |
| DBPedia | P17 | **Give me all professional skateboarders from Sweden.** | **List of pro skateboarders in Sweden.** |

Table 3: Examples of observations from Adversarial BEIR. We perturb the query on three levels of granularity: character, word, and sentence. The manipulated elements have been highlighted in **orange**. In the last case (P17) we operate on the sentence level, therefore the whole input query is considered for the perturbation.

R2 What size of a model do we need to employ to the automatic query paraphrase generation to obtain high-quality outputs?

R3 How well is a 'LLM as a judge' setting aligned with a human annotator in the task of annotating the quality of automatic query paraphrase generation?

To address R1 and R2, we generated paraphrases of benchmark queries using two methods: paraphrasing with Qwen2.5 (Qwen et al., 2025) in three sizes (0.5B, 7B, and 32B parameters) and backtranslation. We evaluated several translation models for backtranslation, selecting the optimal one based on the highest BERTScore (Zhang et al., 2020), indicating better preservation of semantics in the paraphrase, and the largest Levenshtein distance, indicating the highest level of perturbation. Details are presented in Table 7 in the Appendix.

After generation, paraphrases were shuffled to remove model information. Annotators then labeled whether each paraphrase preserved the original query's semantics. Post-annotation, we used four LLMs: DeepSeek-R1-Distill-Llama-70B (DeepSeek-AI et al., 2025), Llama-3.3-70B-Instruct (Grattafiori et al., 2024), Qwen2.5-72B-Instruct (Qwen et al., 2025), and Command A (Cohere et al., 2025), for automatic annotation. Further in this subsection we refer to them as DeepSeek-R1, Llama-3.3, Qwen2.5, and Command A.

The annotation results are presented in Figure 2. According to human assessment, the smallest model (0.5B) shows clear limitations in paraphrase generation, with 39.7% of its outputs labeled invalid and 3.6% marked as exact or near duplicates. In contrast, the 32B model demonstrates strong per-

| Dataset | Perturbation Level | BM25 | UAE-Large | bge-large | gte-large | modernbert | e5-large-instruct | gte-Qwen2-7B-instruct |
|---|---|---|---|---|---|---|---|---|
| MSMARCO | Character | 12.8 (-9.5) | 26.3 (-15.7) | 27.2 (-15) | 29.1 (-13.8) | 27.1 (-13) | 31 (-9.3) | 40.1 (-4.3) |
| | Word | 18.9 (-3.4) | 36.9 (-5.1) | 37.9 (-4.3) | 39.2 (-3.7) | 37.1 (-3) | 37.5 (-2.8) | 42.8 (-1.6) |
| | Sentence | 12.4 (-9.8) | 29.9 (-12) | 30.3 (-11.9) | 31 (-11.8) | 29.6 (-10.3) | 30.3 (-9.7) | 33.2 (-11.2) |
| TREC-COVID | Character | 45.9 (-22.9) | 56.1 (-20.3) | 55.1 (-19.7) | 64.4 (-13) | 65.4 (-18.9) | 75.6 (-6.5) | 81.2 (-0.5) |
| | Word | 60.3 (-8.5) | 68.6 (-7.8) | 67.6 (-7.2) | 73.1 (-4.3) | 79.6 (-4.7) | 79.3 (-2.8) | 80.8 (-0.9) |
| | Sentence | 47.9 (-20.9) | 58.8 (-17.6) | 57.7 (-17.1) | 64.5 (-12.9) | 73.3 (-11) | 75.7 (-6.4) | 75 (-6.7) |
| NFCorpus | Character | 28.5 (-6) | 27.3 (-11.6) | 26.9 (-11.4) | 27.1 (-9.8) | 23.2 (-10.5) | 27.8 (-7.8) | 33.8 (-6.8) |
| | Word | 33.4 (-1.1) | 37.3 (-1.6) | 36.8 (-1.5) | 35.4 (-1.5) | 32.5 (-1.2) | 34.7 (-0.9) | 39.9 (-0.7) |
| | Sentence | 27.3 (-6.3) | 35.2 (-4) | 34.4 (-4.2) | 34.6 (-2.8) | 31.3 (-2.9) | 33 (-3) | 38.8 (-2.2) |
| NQ | Character | 22.5 (-9.1) | 40.1 (-16.8) | 39.2 (-16.2) | 41.1 (-15.1) | 44.9 (-16.7) | 51.4 (-11.4) | 59.5 (-5) |
| | Word | 27.1 (-4.5) | 50.2 (-6.7) | 50 (-5.4) | 51.2 (-5) | 58 (-3.6) | 59.7 (-3.1) | 62.6 (-1.9) |
| | Sentence | 29.9 (-2.1) | 53.4 (-4) | 52.7 (-3) | 50.9 (-5.6) | 56.2 (-5.5) | 56.6 (-6.8) | 60 (-5) |
| HotpotQA | Character | 45.5 (-14.7) | 56.9 (-16.4) | 57.1 (-17.3) | 56 (-10.3) | 52 (-13.5) | 56.9 (-11) | 67.4 (-4.2) |
| | Word | 53.2 (-7) | 66.3 (-7) | 67.5 (-6.9) | 62.2 (-4.1) | 62.6 (-2.9) | 64.2 (-3.7) | 69.4 (-2.2) |
| | Sentence | 49.1 (-10.9) | 53.4 (-4) | 67.8 (-6.6) | 62 (-4.4) | 60.8 (-4.7) | 61.9 (-6) | 68.5 (-3.2) |
| FiQA-2018 | Character | 18.9 (-5.7) | 34.8 (-10.5) | 34.4 (-11.4) | 49.1 (-14.3) | 30.2 (-9.5) | 39.6 (-6.2) | 59 (-3.2) |
| | Word | 22.4 (-2.2) | 41.4 (-3.9) | 41.9 (-3.9) | 57.8 (-5.6) | 37.7 (-2) | 43.5 (-2.3) | 59.9 (-2.3) |
| | Sentence | 18.7 (-6) | 39.2 (-6.3) | 38.6 (-7.5) | 49.9 (-13.6) | 35.5 (-4.5) | 38.4 (-7.7) | 52.4 (-10.2) |
| ArguAna | Character | 44.3 (-2.7) | 62.4 (-2) | 59.3 (-5.2) | 69.7 (-1.6) | 41 (-6.5) | 53.7 (-1.9) | 58.8 (-0.9) |
| | Word | 45.6 (-1.4) | 63.5 (-0.9) | 62.1 (-2.4) | 70.5 (-0.8) | 47.1 (-0.4) | 55.0 (-0.6) | 58.9 (-0.8) |
| | Sentence | 36.6 (-9.8) | 59.1 (-5.3) | 57.8 (-6.5) | 62.8 (-8.3) | 45.5 (-2.1) | 49.2 (-6.6) | 51.6 (-7.7) |
| Touche-2020 | Character | 24.9 (-9.8) | 16.9 (-8.1) | 16.1 (-8.7) | 16 (-6.5) | 23.6 (-6.4) | 21 (-3.9) | 30.3 (-2.4) |
| | Word | 30.7 (-4) | 22.2 (-2.8) | 22 (-2.8) | 20 (-2.5) | 28.1 (-1.9) | 23 (-1.9) | 30.8 (-1.9) |
| | Sentence | 16.7 (-18) | 19.9 (-5.1) | 18.7 (-6.1) | 16.6 (-5.9) | 21.9 (-8.1) | 16.7 (-8.2) | 23.2 (-9.5) |
| CQADupstack | Character | 22.6 (-6.2) | 31 (-11.3) | 32 (-11.6) | 32.7 (-11.2) | 31.6 (-12.2) | 35.3 (-7.7) | 40.4 (-4.7) |
| | Word | 26.5 (-2.3) | 39.2 (-3.1) | 40.2 (-3.4) | 40.2 (-3.7) | 40.8 (-3) | 40.9 (-2.1) | 43.1 (-2) |
| | Sentence | 19.7 (-8.9) | 34.6 (-7.8) | 35.7 (-8.3) | 35.3 (-8.3) | 35.7 (-8.1) | 34.3 (-9.2) | 36.5 (-9) |
| Quora | Character | 59.4 (-18.6) | 72.8 (-15.4) | 74.9 (-13.2) | 77.7 (-11.1) | 75.3 (-12.6) | 83.6 (-4.7) | 87.1 (-2.1) |
| | Word | 70.5 (-7.5) | 83.4 (-4.8) | 84.1 (-4) | 85.9 (-2.9) | 85.7 (-2.2) | 86.7 (-1.6) | 87.6 (-1.6) |
| | Sentence | 35 (-43.3) | 72.2 (-16) | 72.2 (-15.8) | 74.1 (-14.7) | 73.1 (-14.7) | 74 (-14.2) | 74.2 (-14.9) |
| DBPedia | Character | 16.9 (-13.5) | 27.2 (-16.1) | 25.5 (-16.6) | 28.1 (-16.2) | 22 (-17.9) | 25.8 (-11.1) | 43.4 (-7.1) |
| | Word | 26.9 (-3.5) | 38.4 (-4.9) | 36.7 (-5.4) | 39.3 (-5) | 35.7 (-4.2) | 34.7 (-2.2) | 48 (-2.5) |
| | Sentence | 22.1 (-7.8) | 39.8 (-3.4) | 37 (-5) | 37.5 (-6.5) | 33.7 (-5.8) | 31.3 (-5.2) | 45.1 (-5.4) |
| SCIDOCS | Character | 13.4 (-3.8) | 18.8 (-5) | 18.3 (-5.1) | 21.9 (-6.3) | 14.6 (-4.3) | 17 (-2.8) | 28 (-2.2) |
| | Word | 15.9 (-1.3) | 21.9 (-1.9) | 21.5 (-1.9) | 25.6 (-2.6) | 17.7 (-1.2) | 18.7 (-1.1) | 28.7 (-1.5) |
| | Sentence | 14.1 (-3.3) | 21 (-3.1) | 20.8 (-2.9) | 23 (-5.3) | 16.3 (-2.7) | 17 (-3) | 26.5 (-4) |
| FEVER | Character | 41.8 (-19.2) | 69.5 (-19) | 68.8 (-19) | 79.3 (-14.9) | 70.4 (-16.8) | 68.2 (-9.7) | 84.4 (-9.6) |
| | Word | 53.9 (-7.1) | 81.8 (-6.7) | 80.6 (-7.2) | 89.2 (-5) | 83 (-4.2) | 74.5 (-3.4) | 90.4 (-3.6) |
| | Sentence | 49.2 (-11.7) | 80.5 (-8.1) | 79.7 (-8.1) | 86.2 (-8) | 80.5 (-6.8) | 71.3 (-6.6) | 86.3 (-7.9) |
| Climate-FEVER | Character | 12.7 (-4.6) | 33.2 (-6.4) | 31.2 (-7.1) | 42.3 (-7.3) | 31.2 (-6.3) | 31.9 (+0.7) | 44.2 (-2.5) |
| | Word | 14.9 (-2.4) | 37.2 (-2.4) | 35.3 (-3) | 46.5 (-3.1) | 35.7 (-1.8) | 31.7 (+0.5) | 45.1 (-1.6) |
| | Sentence | 16.1 (-1.2) | 37.8 (-1.8) | 36.5 (-1.8) | 44.4 (-5.2) | 35.6 (-1.9) | 29.4 (-1.8) | 43 (-3.7) |
| SciFact | Character | 59.5 (-9.6) | 68.2 (-5.9) | 67.5 (-7.1) | 74.5 (-8) | 62.1 (-7.7) | 67.2 (-4.7) | 77.1 (-2.2) |
| | Word | 66.1 (-3) | 71.9 (-2.2) | 72.1 (-2.5) | 80.5 (-2) | 68.5 (-1.3) | 70 (-1.9) | 78.7 (-0.6) |
| | Sentence | 62.9 (-6) | 73 (-1.1) | 72.7 (-1.9) | 79.4 (-3) | 67.4 (-2.3) | 69.4 (-2.3) | 77.9 (-1.4) |

Table 4: NDCG@10 metric value on the Adversarial BEIR across domain datasets for selected models. The metric value for the perturbed version of the benchmark is displayed in black. The difference between the metric values before and after applying perturbations is shown in parentheses next to each value. This difference is highlighted in red if the metric value decreases after perturbation and in green if it increases.

formance, generating 87.2% valid outputs and only 0.2% duplicates. While the 7B model shows significant improvement over the 0.5B version, the difference between 7B and 32B is smaller (4.1%), suggesting diminishing returns at larger scales. Back-translation performs poorly, with 22.8% invalid outputs and 14.5% near-duplicates, often failing to introduce meaningful adversarial variation.

Humans were stricter than any LLM judge, labeling the highest number of paraphrases as invalid across all annotators. The annotation process is inherently difficult, as judgments about what constitutes a "duplicate" are subjective, interpretations of "very close to the original" vary across annotators (see Figure 8). DeepSeek's model illustrates this challenge, labeling many more examples as duplicates. Annotator agreement metrics are shown in Figure 3. Since DeepSeek-R1 marked significantly more duplicates, we also include agreement metrics only for observations it labeled as valid or invalid in Figure 4 in the Appendix.

Motivated by recent findings on bias in LLM-as-a-judge settings (Koo et al., 2024; Li et al., 2025), we examined if Qwen2.5 would favor outputs from models in its own family. However, this was not the case. Command A and LLaMA-3.3 labeled more generations from Qwen models as valid.

We found that none of the human-model pairs exhibited strong agreement. The agreement between humans and DeepSeek's model was fair, while other human-model pairs showed moderate agreement. Conversely, some model pairs (Qwen2.5-

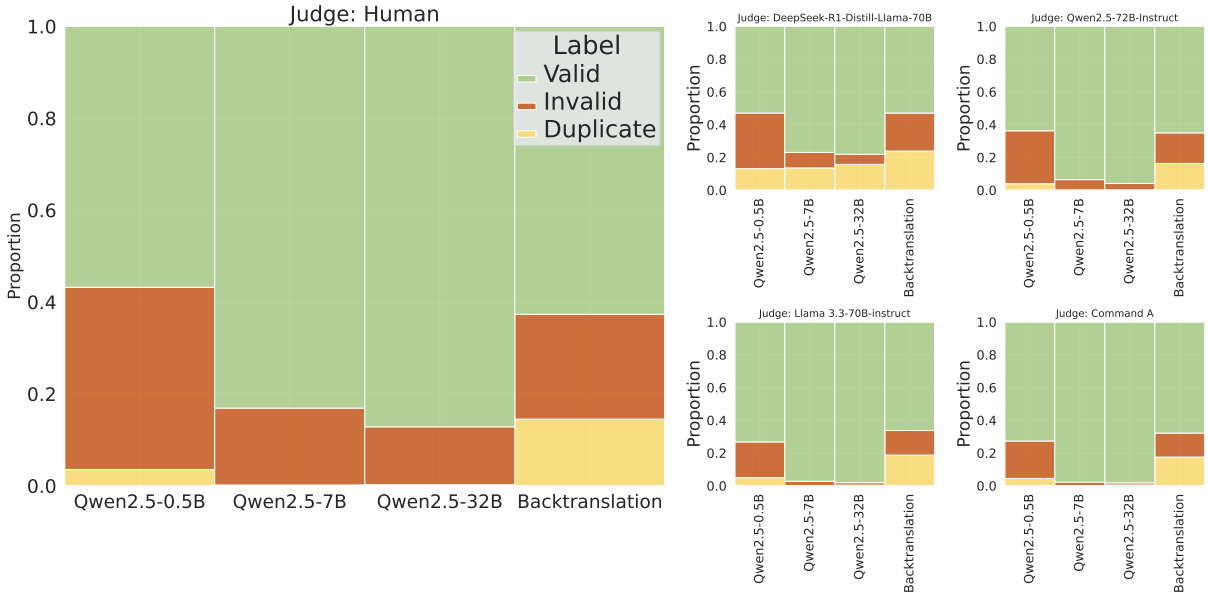


Figure 2: Results of the human and automatic annotation of the automatic query paraphrasing task. Valid stands for the correct paraphrase generated by a model, otherwise an example was labeled as Invalid. If the model created an exact or close duplicate, the Duplicate label was assigned.

Llama-3.3, Qwen2.5-Command A, Llama-3.3-Command A) demonstrated strong agreement. It shows that even though LLMs are often employed as judges for automatic annotation, their assessments may differ among themselves (as seen in the alignment between DeepSeek-R1 and other models). In addition, for the task of automatic labeling of paraphrases, most of them align with the human annotator only to a moderate degree.

### 3.4 Selecting paraphrases for the benchmark

To perform Automatic Paraphrase Perturbation (P17), we employed four models: three Large Language Models from the Qwen2.5 series (Qwen et al., 2025) and one translation-based model. Each generated paraphrase was human-annotated for validity. The observations were annotated by internal employees with expertise in applied linguistics. To identify the most challenging paraphrases among the valid ones, we introduced a selection method leveraging a group of Cross Encoder models. Cross Encoders, widely adopted in retrieval-based question answering and search applications (Wang et al., 2019; Nogueira and Cho, 2020), are effective at assessing textual similarity and relevance. For each valid paraphrase, we computed a score using each Cross Encoder and produced individual model-based rankings. These rankings were then transformed using their reciprocal values (i.e., $1/\text{rank}$), which places greater emphasis on higher-ranked paraphrases. We then aggregated the reciprocal ranks across all models, and selected the paraphrase with the lowest total score. The algorithmic formulation of this selection process is presented below. We share the list of the Cross Encoders used in this procedure in Table 9.

Let $Q$ be a query and $P = \{p_1, p_2, \ldots, p_n\}$ the set of valid paraphrases generated by models for query $Q$. Let $M = \{m_1, m_2, \ldots, m_k\}$ denote the set of Cross Encoders. Each model $m_j \in M$ assigns similarity scores and induces a ranking $r_j(p_i) \in \{1, \ldots, n\}$ for each paraphrase $p_i$. The reciprocal rank is:

$$s_j(p_i) = \frac{1}{r_j(p_i)} \tag{1}$$

The total score across models is:

$$S(p_i) = \sum_{j=1}^{k} \frac{1}{r_j(p_i)} \tag{2}$$

The selected paraphrase minimizes this total:

$$p^* = \arg\min_{p_i \in P} S(p_i) \tag{3}$$

This approach prioritizes valid paraphrases that are least similar to the original query, thereby ensuring the selection of the most challenging samples.

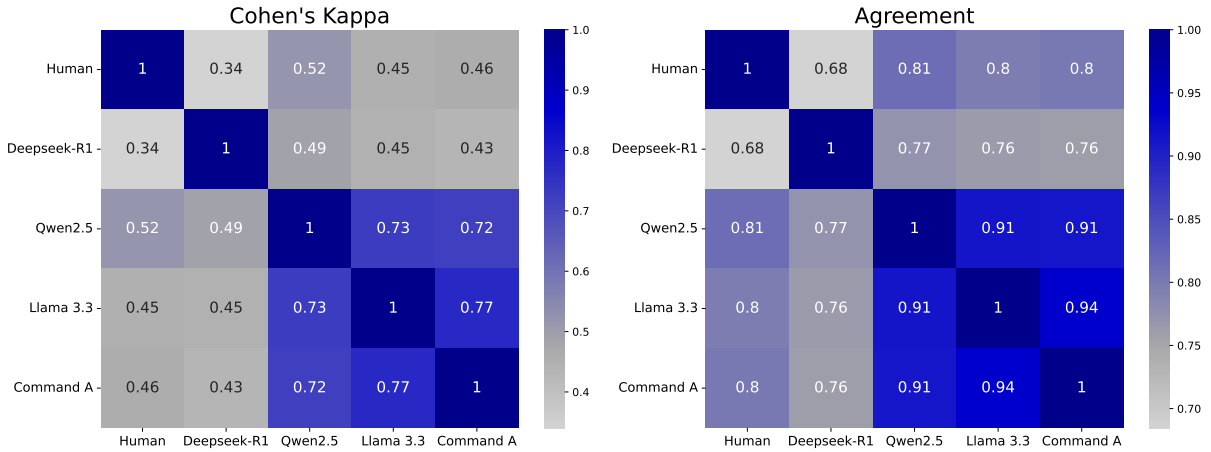


Figure 3: Inter-annotator agreement metrics for human-model and model-model pairs. Agreement is the percentage of observations which had the same label assigned by both annotators within a given pair.

## 4 Results

To determine an appropriate perturbation strength for each method, we generated a dataset consisting of 100 randomly sampled observations. Each method was then applied to this test set, and the observations were labeled based on whether the resulting perturbations remained comprehensible and preserved the semantics of the original query. A perturbation strength was considered suitable as the default for our experiments if at least 95% of the test set was classified as valid. Details regarding the perturbation strengths used for each method are provided in Table 12 in the Appendix D.

The models selected for the evaluation process were chosen based on their performance on the MTEB benchmark (Muennighoff et al., 2023), focusing on those currently ranked highest on the leaderboard. All evaluated models are listed in Table 8 in Appendix. We employed a version of ModernBERT (Warner et al., 2024), which was further trained using datasets and methodologies outlined in Nussbaum et al. (2025). We report the mean NDCG@10 value across all datasets and perturbation levels in Table 4.

### 4.1 General findings

Across all evaluated datasets, every level of perturbation causes a decline in model performance except two cases (e5-large-instruct evaluated with character and word-level perturbations on Climate-FEVER dataset). It appears that perturbations are less detrimental for very long queries, such as those found in the Arguana dataset, due to the amount of information contained in the text and the lower probability of losing valuable information. However, for shorter queries, where the median length variations between different datasets are minimal, this effect is less pronounced and does not significantly impact the model's robustness. We visualize this relationship in Figure 7 in the Appendix.

Furthermore, we note that model size plays a critical role in susceptibility to adversarial samples. For example, UAE-Large and bge-large (both based on ~330M parameter BERT-base models) exhibit average performance drops of 7.6% and 7.8%, respectively. Larger models such as e5-large-instruct (~560M parameters) and gte-Qwen2-7B-instruct (~7.6B parameters) demonstrate reduced vulnerability, with drops of 5% and 4.1%.

### 4.2 Retrievers are not robust to character-level perturbations

One of the most prominent observation from examining Table 4 is the significantly worse performance of all models on queries perturbed at the character-level in comparison to the word-level ones, hence the statement in the title of this section.

Since all perturbation strengths have been chosen using intelligibility preservation criteria (Appendix D), the results for both levels should be similar if models were to match human performance in capturing semantics of perturbed queries.

Most modern transformer-based models operate on neither characters nor words but on tokens. Due to the nature of the tokenization process we strongly believe that perturbation introduced on character-level leads to much more noise in the retriever's input tokens, which could be the main source of the observed performance degradation. Recent studies (Zhuang and Zuccon, 2022; Zhuang

et al., 2023) have reached similar conclusions, but they only propose five character-level perturbations for model evaluation and compare against models that are no longer state-of-the-art.

The issue with character-level perturbations is that they usually introduce additional tokens into the input (example in Appendix F), which likely were not encountered in such contexts during the training phase. In consequence, the final sentence embeddings derived from mean pooling are disrupted by these noisy tokens.

| Level | Mean No. Tokens | Mean Token Length | Mean Jaccard Index |
|---|---|---|---|
| No Perturb. | 26.19 | 4.30 | 1.00 |
| Character | 34.47 | 3.54 | 0.55 |
| Word | 27.38 | 4.15 | 0.80 |

Table 5: Tokenization statistics aggregated on the perturbation level. Mean token length is a number of characters per token and Mean Jaccard index is calculated on sets of unique tokens from original and perturbed query.

Looking at the aggregated tokenization statistics in Table 5 on the character level around 30% more tokens are introduced with smaller information density (almost 1 character less per token), and on average almost half of the tokens are different in perturbed query (0.55 Jaccard index). On the word-level, these statistics are much closer to the baseline ones calculated for original queries. This is a first strong indicator that even though character-level perturbations are applied at half the strength of word-level ones (Table 12), they introduce significantly more noise to the input.

| Level | Mean Char. Edit Dist. | Mean Token Edit Dist. |
|---|---|---|
| Character | 13.56 | 10.44 |
| Word | 5.28 | 10.87 |

Table 6: Comparison of mean Levenshtein edit distances between original and perturbed queries, aggregated based on the level of introduced perturbations.

Analysis of aggregated edit distances from Table 6 shows that although on average token-wise distances are similar for character- and word-level perturbations, the comparison of character-wise distances unveils a much more disruptive nature of character-level perturbations.

We have shown strong evidence, that such significant difference between character- and word-level perturbations in models performance (Table 4) stems from the tokenization process that is not robust to character-level perturbation. There is a lot of opportunity for further research like utilizing various spelling corrections methods (Hladek et al., 2020) to alleviate this performance degradation.

### 4.3 LLM-generated paraphrases pose a high challenge to retrievers

Methods such as CLARE (Li et al., 2021) have been widely used in the NLP domain to prepare adversarial samples. However, CLARE is a method operating on the token level. In subsection 3.3 we proved that Large Language Models can automatically generate valid, high quality sentence-level query paraphrases. Results from Table 4 show that sentence-level perturbations almost always cause a higher drop in performance than the word-level ones. We attribute this phenomenon to the enhanced capability of LLMs to modify text extensively while preserving core semantics. Unlike CLARE, which applies token-level perturbations in isolation—risking rapid semantic degradation—LLMs iteratively refine edits by maintaining contextual alignment with the original query. This enables LLMs to introduce more substantial syntactic changes without compromising meaning. To quantify this, we computed the Jaccard index between original and perturbed queries. For CLARE-based Context-Aware Perturbations, the average similarity was 0.83, reflecting limited lexical alteration. In contrast, LLM-generated paraphrases achieved an average Jaccard index of 0.37, indicating significant lexical variation yet effective semantic retention.

## 5 Conclusion

We present Adversarial BEIR, an information retrieval benchmark comprising query perturbations from 15 datasets, generated using 17 distinct methods across three granularity levels: character, word, and sentence. Our findings demonstrate that current state-of-the-art retrievers are not robust to adversarial queries, indicating that this challenge remains unresolved. Furthermore, through both human and automated evaluations of the automatic paraphrase generation method, we observe a lack of strong agreement among human-model and model-model pairs, highlighting the complexity of automatic annotation for this task. We believe that the data, code, and results we provide will serve as a valuable foundation for future research on the robustness of retrievers against query perturbations.

## 6 Limitations

We demonstrated that even top-performing retrievers face challenges with adversarial queries at three levels: character, word, and sentence. At the character level, excessive modifications like typos or deletions might overly distort the query's meaning. Despite manually calibrating the intensity of each perturbation, there might be individual examples of such characteristics in our evaluation dataset, which has more than 100,000 observations. Future work might involve deriving an automatic approach to detect such samples.

Additionally, this study concentrates on the evaluation of general-purpose text retrieval. Subsequent research could productively explore how our findings generalize to more specialized retrieval tasks, such as programming code retrieval, which are not tested in this work.

Moreover, certain models available on the MTEB leaderboard lack transparency regarding their training datasets. Consequently, this lack of clarity complicates comparative analyses of these models, particularly in evaluating their robustness against perturbed queries.

## References


Aryan Agrawal, Lisa Alazraki, Shahin Honarvar, Thomas Mensink, and Marek Rei. 2025. Enhancing LLM robustness to perturbed instructions: An empirical study. In *ICLR 2025 Workshop on Building Trust in Language Models and Applications*.

Marwah Alaofi, Luke Gallagher, Mark Sanderson, Falk Scholer, and Paul Thomas. 2023. Can generative llms create query variants for test collections? an exploratory study. In *Proceedings of the 46th International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR '23, page 1869–1873, New York, NY, USA. Association for Computing Machinery.

Yonatan Belinkov and Yonatan Bisk. 2018. Synthetic and natural noise both break neural machine translation. In *International Conference on Learning Representations*.

Daniel Campos, ChengXiang Zhai, and Alessandro Magnani. 2023. Noise-robust dense retrieval via contrastive alignment post training. *Preprint*, arXiv:2304.03401.

Chi-Min Chan, Chunpu Xu, Ruibin Yuan, Hongyin Luo, Wei Xue, Yike Guo, and Jie Fu. 2024. RQ-RAG: Learning to refine queries for retrieval augmented generation. In *First Conference on Language Modeling*.

Jianlyu Chen, Shitao Xiao, Peitian Zhang, Kun Luo, Defu Lian, and Zheng Liu. 2024. M3-embedding: Multi-linguality, multi-functionality, multi-granularity text embeddings through self-knowledge distillation. In *Findings of the Association for Computational Linguistics: ACL 2024*, pages 2318–2335, Bangkok, Thailand. Association for Computational Linguistics.

Team Cohere, :, Aakanksha, Arash Ahmadian, Marwan Ahmed, Jay Alammar, Milad Alizadeh, Yazeed Alnumay, Sophia Althammer, Arkady Arkhangorodsky, Viraat Aryabumi, Dennis Aumiller, Raphaël Avalos, Zahara Aviv, Sammie Bae, Saurabh Baji, Alexandre Barbet, Max Bartolo, Björn Bebensee, Neeral Beladia, Walter Beller-Morales, Alexandre Bérard, Andrew Berneshawi, Anna Bialas, Phil Blunsom, Matt Bobkin, Adi Bongale, Sam Braun, Maxime Brunet, Samuel Cahyawijaya, David Cairuz, Jon Ander Campos, Cassie Cao, Kris Cao, Roman Castagné, Julián Cendrero, Leila Chan Currie, Yash Chandak, Diane Chang, Giannis Chatziveroglou, Hongyu Chen, Claire Cheng, Alexis Chevalier, Justin T. Chiu, Eugene Cho, Eugene Choi, Eujeong Choi, Tim Chung, Volkan Cirik, Ana Cismaru, Pierre Clavier, Henry Conklin, Lucas Crawhall-Stein, Devon Crouse, Andres Felipe Cruz-Salinas, Ben Cyrus, Daniel D'souza, Hugo Dalla-Torre, John Dang, William Darling, Omar Darwiche Domingues, Saurabh Dash, Antoine Debugne, Théo Dehaze, Shaan Desai, Joan Devassy, Rishit Dholakia, Kyle Duffy, Ali Edalati, Ace Eldeib, Abdullah Elkady, Sarah Elsharkawy, Irem Ergün, Beyza Ermis, Marzieh Fadaee, Boyu Fan, Lucas Fayoux, Yannis Flet-Berliac, Nick Frosst, Matthias Gallé, Wojciech Galuba, Utsav Garg, Matthieu Geist, Mohammad Gheshlaghi Azar, Ellen Gilsenan-McMahon, Seraphina Goldfarb-Tarrant, Tomas Goldsack, Aidan Gomez, Victor Machado Gonzaga, Nithya Govindarajan, Manoj Govindassamy, Nathan Grinsztajn, Nikolas Gritsch, Patrick Gu, Shangmin Guo, Kilian Haefeli, Rod Hajjar, Tim Hawes, Jingyi He, Sebastian Hofstätter, Sungjin Hong, Sara Hooker, Tom Hosking, Stephanie Howe, Eric Hu, Renjie Huang, Hemant Jain, Ritika Jain, Nick Jakobi, Madeline Jenkins, JJ Jordan, Dhruti Joshi, Jason Jung, Trushant Kalyanpur, Siddhartha Rao Kamalakara, Julia Kedrzycki, Gokce Keskin, Edward Kim, Joon Kim, Wei-Yin Ko, Tom Kocmi, Michael Kozakov, Wojciech Kryściński, Arnav Kumar Jain, Komal Kumar Teru, Sander Land, Michael Lasby, Olivia Lasche, Justin Lee, Patrick Lewis, Jeffrey Li, Jonathan Li, Hangyu Lin, Acyr Locatelli, Kevin Luong, Raymond Ma, Lukáš Mach, Marina Machado, Joanne Magbitang, Brenda Malacara Lopez, Aryan Mann, Kelly Marchisio, Olivia Markham, Alexandre Matton, Alex McKinney, Dominic McLoughlin, Jozef Mokry, Adrien Morisot, Autumn Moulder, Harry Moynehan, Maximilian Mozes, Vivek Muppalla, Lidiya Murakhovska, Hemangani Nagarajan, Alekhya Nandula, Hisham Nasir, Shauna Nehra, Josh Netto-Rosen, Daniel Ohashi, James Owers-Bardsley, Jason Ozuzu, Dennis Padilla, Gloria Park, Sam Passaglia, Jeremy Pekmez, Laura Penstone, Aleksandra Piktus, Case Ploeg, Andrew Poulton, Youran

Qi, Shubha Raghvendra, Miguel Ramos, Ekagra Ranjan, Pierre Richemond, Cécile Robert-Michon, Aurélien Rodriguez, Sudip Roy, Sebastian Ruder, Laura Ruis, Louise Rust, Anubhav Sachan, Alejandro Salamanca, Kailash Karthik Saravanakumar, Isha Satyakam, Alice Schoenauer Sebag, Priyanka Sen, Sholeh Sepehri, Preethi Seshadri, Ye Shen, Tom Sherborne, Sylvie Shang Shi, Sanal Shivaprasad, Vladyslav Shmyhlo, Anirudh Shrinivason, Inna Shteinbuk, Amir Shukayev, Mathieu Simard, Ella Snyder, Ava Spataru, Victoria Spooner, Trisha Starostina, Florian Strub, Yixuan Su, Jimin Sun, Dwarak Talupuru, Eugene Tarassov, Elena Tommasone, Jennifer Tracey, Billy Trend, Evren Tumer, Ahmet Üstün, Bharat Venkitesh, David Venuto, Pat Verga, Maxime Voisin, Alex Wang, Donglu Wang, Shijian Wang, Edmond Wen, Naomi White, Jesse Willman, Marysia Winkels, Chen Xia, Jessica Xie, Minjie Xu, Bowen Yang, Tan Yi-Chern, Ivan Zhang, Zhenyu Zhao, and Zhoujie Zhao. 2025. Command a: An enterprise-ready large language model. *Preprint*, arXiv:2504.00698.

DeepSeek-AI, Daya Guo, Dejian Yang, Haowei Zhang, Junxiao Song, Ruoyu Zhang, Runxin Xu, Qihao Zhu, Shirong Ma, Peiyi Wang, Xiao Bi, Xiaokang Zhang, Xingkai Yu, Yu Wu, Z. F. Wu, Zhibin Gou, Zhihong Shao, Zhuoshu Li, Ziyi Gao, Aixin Liu, Bing Xue, Bingxuan Wang, Bochao Wu, Bei Feng, Chengda Lu, Chenggang Zhao, Chengqi Deng, Chenyu Zhang, Chong Ruan, Damai Dai, Deli Chen, Dongjie Ji, Erhang Li, Fangyun Lin, Fucong Dai, Fuli Luo, Guangbo Hao, Guanting Chen, Guowei Li, H. Zhang, Han Bao, Hanwei Xu, Haocheng Wang, Honghui Ding, Huajian Xin, Huazuo Gao, Hui Qu, Hui Li, Jianzhong Guo, Jiashi Li, Jiawei Wang, Jingchang Chen, Jingyang Yuan, Junjie Qiu, Junlong Li, J. L. Cai, Jiaqi Ni, Jian Liang, Jin Chen, Kai Dong, Kai Hu, Kaige Gao, Kang Guan, Kexin Huang, Kuai Yu, Lean Wang, Lecong Zhang, Liang Zhao, Litong Wang, Liyue Zhang, Lei Xu, Leyi Xia, Mingchuan Zhang, Minghua Zhang, Minghui Tang, Meng Li, Miaojun Wang, Mingming Li, Ning Tian, Panpan Huang, Peng Zhang, Qiancheng Wang, Qinyu Chen, Qiushi Du, Ruiqi Ge, Ruisong Zhang, Ruizhe Pan, Runji Wang, R. J. Chen, R. L. Jin, Ruyi Chen, Shanghao Lu, Shangyan Zhou, Shanhuang Chen, Shengfeng Ye, Shiyu Wang, Shuiping Yu, Shunfeng Zhou, Shuting Pan, S. S. Li, Shuang Zhou, Shaoqing Wu, Shengfeng Ye, Tao Yun, Tian Pei, Tianyu Sun, T. Wang, Wangding Zeng, Wanjia Zhao, Wen Liu, Wenfeng Liang, Wenjun Gao, Wenqin Yu, Wentao Zhang, W. L. Xiao, Wei An, Xiaodong Liu, Xiaohan Wang, Xiaokang Chen, Xiaotao Nie, Xin Cheng, Xin Liu, Xin Xie, Xingchao Liu, Xinyu Yang, Xinyuan Li, Xuecheng Su, Xuheng Lin, X. Q. Li, Xiangyue Jin, Xiaojin Shen, Xiaosha Chen, Xiaowen Sun, Xiaoxiang Wang, Xinnan Song, Xinyi Zhou, Xianzu Wang, Xinxia Shan, Y. K. Li, Y. Q. Wang, Y. X. Wei, Yang Zhang, Yanhong Xu, Yao Li, Yao Zhao, Yaofeng Sun, Yaohui Wang, Yi Yu, Yichao Zhang, Yifan Shi, Yiliang Xiong, Ying He, Yishi Piao, Yisong Wang, Yixuan Tan, Yiyang Ma, Yiyuan Liu, Yongqiang Guo, Yuan Ou, Yuduan Wang, Yue Gong, Yuheng Zou, Yujia He, Yunfan Xiong, Yuxiang Luo, Yuxiang You, Yuxuan Liu, Yuyang Zhou, Y. X. Zhu, Yanhong Xu, Yanping Huang, Yaohui Li, Yi Zheng, Yuchen Zhu, Yunxian Ma, Ying Tang, Yukun Zha, Yuting Yan, Z. Z. Ren, Zehui Ren, Zhangli Sha, Zhe Fu, Zhean Xu, Zhenda Xie, Zhengyan Zhang, Zhewen Hao, Zhicheng Ma, Zhigang Yan, Zhiyu Wu, Zihui Gu, Zijia Zhu, Zijun Liu, Zilin Li, Ziwei Xie, Ziyang Song, Zizheng Pan, Zhen Huang, Zhipeng Xu, Zhongyu Zhang, and Zhen Zhang. 2025. Deepseek-r1: Incentivizing reasoning capability in llms via reinforcement learning. *Preprint*, arXiv:2501.12948.

Ian J. Goodfellow, Jonathon Shlens, and Christian Szegedy. 2014. Explaining and harnessing adversarial examples. *CoRR*, abs/1412.6572.

Gaurav Goswami, Nalini Ratha, Akshay Agarwal, Richa Singh, and Mayank Vatsa. 2018. Unravelling robustness of deep learning based face recognition against adversarial attacks. AAAI'18/IAAI'18/EAAI'18. AAAI Press.

Aaron Grattafiori, Abhimanyu Dubey, Abhinav Jauhri, Abhinav Pandey, Abhishek Kadian, Ahmad Al-Dahle, Aiesha Letman, Akhil Mathur, Alan Schelten, Alex Vaughan, Amy Yang, Angela Fan, Anirudh Goyal, Anthony Hartshorn, Aobo Yang, Archi Mitra, Archie Sravankumar, Artem Korenev, Arthur Hinsvark, Arun Rao, Aston Zhang, Aurelien Rodriguez, Austen Gregerson, Ava Spataru, Baptiste Roziere, Bethany Biron, Binh Tang, Bobbie Chern, Charlotte Caucheteux, Chaya Nayak, Chloe Bi, Chris Marra, Chris McConnell, Christian Keller, Christophe Touret, Chunyang Wu, Corinne Wong, Cristian Canton Ferrer, Cyrus Nikolaidis, Damien Allonsius, Daniel Song, Danielle Pintz, Danny Livshits, Danny Wyatt, David Esiobu, Dhruv Choudhary, Dhruv Mahajan, Diego Garcia-Olano, Diego Perino, Dieuwke Hupkes, Egor Lakomkin, Ehab AlBadawy, Elina Lobanova, Emily Dinan, Eric Michael Smith, Filip Radenovic, Francisco Guzmán, Frank Zhang, Gabriel Synnaeve, Gabrielle Lee, Georgia Lewis Anderson, Govind Thattai, Graeme Nail, Gregoire Mialon, Guan Pang, Guillem Cucurell, Hailey Nguyen, Hannah Korevaar, Hu Xu, Hugo Touvron, Iliyan Zarov, Imanol Arrieta Ibarra, Isabel Kloumann, Ishan Misra, Ivan Evtimov, Jack Zhang, Jade Copet, Jaewon Lee, Jan Geffert, Jana Vranes, Jason Park, Jay Mahadeokar, Jeet Shah, Jelmer van der Linde, Jennifer Billock, Jenny Hong, Jenya Lee, Jeremy Fu, Jianfeng Chi, Jianyu Huang, Jiawen Liu, Jie Wang, Jiecao Yu, Joanna Bitton, Joe Spisak, Jongsoo Park, Joseph Rocca, Joshua Johnstun, Joshua Saxe, Junteng Jia, Kalyan Vasuden Alwala, Karthik Prasad, Kartikeya Upasani, Kate Plawiak, Ke Li, Kenneth Heafield, Kevin Stone, Khalid El-Arini, Krithika Iyer, Kshitiz Malik, Kuenley Chiu, Kunal Bhalla, Kushal Lakhotia, Lauren Rantala-Yeary, Laurens van der Maaten, Lawrence Chen, Liang Tan, Liz Jenkins, Louis Martin, Lovish Madaan, Lubo Malo, Lukas Blecher, Lukas Landzaat, Luke de Oliveira, Madeline Muzzi, Mahesh Pasupuleti, Mannat Singh, Manohar Paluri, Marcin Kardas, Maria Tsimpoukelli, Mathew Oldham, Mathieu Rita, Maya Pavlova, Melanie Kambadur, Mike Lewis, Min Si, Mitesh Kumar Singh,

Mona Hassan, Naman Goyal, Narjes Torabi, Nikolay Bashlykov, Nikolay Bogoychev, Niladri Chatterji, Ning Zhang, Olivier Duchenne, Onur Çelebi, Patrick Alrassy, Pengchuan Zhang, Pengwei Li, Petar Vasic, Peter Weng, Prajjwal Bhargava, Pratik Dubal, Praveen Krishnan, Punit Singh Koura, Puxin Xu, Qing He, Qingxiao Dong, Ragavan Srinivasan, Raj Ganapathy, Ramon Calderer, Ricardo Silveira Cabral, Robert Stojnic, Roberta Raileanu, Rohan Maheswari, Rohit Girdhar, Rohit Patel, Romain Sauvestre, Ronnie Polidoro, Roshan Sumbaly, Ross Taylor, Ruan Silva, Rui Hou, Rui Wang, Saghar Hosseini, Sahana Chennabasappa, Sanjay Singh, Sean Bell, Seohyun Sonia Kim, Sergey Edunov, Shaoliang Nie, Sharan Narang, Sharath Raparthy, Sheng Shen, Shengye Wan, Shruti Bhosale, Shun Zhang, Simon Vandenhende, Soumya Batra, Spencer Whitman, Sten Sootla, Stephane Collot, Suchin Gururangan, Sydney Borodinsky, Tamar Herman, Tara Fowler, Tarek Sheasha, Thomas Georgiou, Thomas Scialom, Tobias Speckbacher, Todor Mihaylov, Tong Xiao, Ujjwal Karn, Vedanuj Goswami, Vibhor Gupta, Vignesh Ramanathan, Viktor Kerkez, Vincent Gonguet, Virginie Do, Vish Vogeti, Vítor Albiero, Vladan Petrovic, Weiwei Chu, Wenhan Xiong, Wenyin Fu, Whitney Meers, Xavier Martinet, Xiaodong Wang, Xiaofang Wang, Xiaoqing Ellen Tan, Xide Xia, Xinfeng Xie, Xuchao Jia, Xuewei Wang, Yaelle Goldschlag, Yashesh Gaur, Yasmine Babaei, Yi Wen, Yiwen Song, Yuchen Zhang, Yue Li, Yuning Mao, Zacharie Delpierre Coudert, Zheng Yan, Zhengxing Chen, Zoe Papakipos, Aaditya Singh, Aayushi Srivastava, Abha Jain, Adam Kelsey, Adam Shajnfeld, Adithya Gangidi, Adolfo Victoria, Ahuva Goldstand, Ajay Menon, Ajay Sharma, Alex Boesenberg, Alexei Baevski, Allie Feinstein, Amanda Kallet, Amit Sangani, Amos Teo, Anam Yunus, Andrei Lupu, Andres Alvarado, Andrew Caples, Andrew Gu, Andrew Ho, Andrew Poulton, Andrew Ryan, Ankit Ramchandani, Annie Dong, Annie Franco, Anuj Goyal, Aparajita Saraf, Arkabandhu Chowdhury, Ashley Gabriel, Ashwin Bharambe, Assaf Eisenman, Azadeh Yazdan, Beau James, Ben Maurer, Benjamin Leonhardi, Bernie Huang, Beth Loyd, Beto De Paola, Bhargavi Paranjape, Bing Liu, Bo Wu, Boyu Ni, Braden Hancock, Bram Wasti, Brandon Spence, Brani Stojkovic, Brian Gamido, Britt Montalvo, Carl Parker, Carly Burton, Catalina Mejia, Ce Liu, Changhan Wang, Changkyu Kim, Chao Zhou, Chester Hu, Ching-Hsiang Chu, Chris Cai, Chris Tindal, Christoph Feichtenhofer, Cynthia Gao, Damon Civin, Dana Beaty, Daniel Kreymer, Daniel Li, David Adkins, David Xu, Davide Testuggine, Delia David, Devi Parikh, Diana Liskovich, Didem Foss, Dingkang Wang, Duc Le, Dustin Holland, Edward Dowling, Eissa Jamil, Elaine Montgomery, Eleonora Presani, Emily Hahn, Emily Wood, Eric-Tuan Le, Erik Brinkman, Esteban Arcaute, Evan Dunbar, Evan Smothers, Fei Sun, Felix Kreuk, Feng Tian, Filippos Kokkinos, Firat Ozgenel, Francesco Caggioni, Frank Kanayet, Frank Seide, Gabriela Medina Florez, Gabriella Schwarz, Gada Badeer, Georgia Swee, Gil Halpern, Grant Herman, Grigory Sizov, Guangyi, Zhang, Guna Lakshminarayanan, Hakan Inan, Hamid Shojanazeri, Han Zou, Hannah Wang, Hanwen Zha, Haroun Habeeb, Harrison Rudolph, Helen Suk, Henry Aspegren, Hunter Goldman, Hongyuan Zhan, Ibrahim Damlaj, Igor Molybog, Igor Tufanov, Ilias Leontiadis, Irina-Elena Veliche, Itai Gat, Jake Weissman, James Geboski, James Kohli, Janice Lam, Japhet Asher, Jean-Baptiste Gaya, Jeff Marcus, Jeff Tang, Jennifer Chan, Jenny Zhen, Jeremy Reizenstein, Jeremy Teboul, Jessica Zhong, Jian Jin, Jingyi Yang, Joe Cummings, Jon Carvill, Jon Shepard, Jonathan McPhie, Jonathan Torres, Josh Ginsburg, Junjie Wang, Kai Wu, Kam Hou U, Karan Saxena, Kartikay Khandelwal, Katayoun Zand, Kathy Matosich, Kaushik Veeraraghavan, Kelly Michelena, Keqian Li, Kiran Jagadeesh, Kun Huang, Kunal Chawla, Kyle Huang, Lailin Chen, Lakshya Garg, Lavender A, Leandro Silva, Lee Bell, Lei Zhang, Liangpeng Guo, Licheng Yu, Liron Moshkovich, Luca Wehrstedt, Madian Khabsa, Manav Avalani, Manish Bhatt, Martynas Mankus, Matan Hasson, Matthew Lennie, Matthias Reso, Maxim Groshev, Maxim Naumov, Maya Lathi, Meghan Keneally, Miao Liu, Michael L. Seltzer, Michal Valko, Michelle Restrepo, Mihir Patel, Mik Vyatskov, Mikayel Samvelyan, Mike Clark, Mike Macey, Mike Wang, Miquel Jubert Hermoso, Mo Metanat, Mohammad Rastegari, Munish Bansal, Nandhini Santhanam, Natascha Parks, Natasha White, Navyata Bawa, Nayan Singhal, Nick Egebo, Nicolas Usunier, Nikhil Mehta, Nikolay Pavlovich Laptev, Ning Dong, Norman Cheng, Oleg Chernoguz, Olivia Hart, Omkar Salpekar, Ozlem Kalinli, Parkin Kent, Parth Parekh, Paul Saab, Pavan Balaji, Pedro Rittner, Philip Bontrager, Pierre Roux, Piotr Dollar, Polina Zvyagina, Prashant Ratanchandani, Pritish Yuvraj, Qian Liang, Rachad Alao, Rachel Rodriguez, Rafi Ayub, Raghotham Murthy, Raghu Nayani, Rahul Mitra, Rangaprabhu Parthasarathy, Raymond Li, Rebekkah Hogan, Robin Battey, Rocky Wang, Russ Howes, Ruty Rinott, Sachin Mehta, Sachin Siby, Sai Jayesh Bondu, Samyak Datta, Sara Chugh, Sara Hunt, Sargun Dhillon, Sasha Sidorov, Satadru Pan, Saurabh Mahajan, Saurabh Verma, Seiji Yamamoto, Sharadh Ramaswamy, Shaun Lindsay, Shaun Lindsay, Sheng Feng, Shenghao Lin, Shengxin Cindy Zha, Shishir Patil, Shiva Shankar, Shuqiang Zhang, Shuqiang Zhang, Sinong Wang, Sneha Agarwal, Soji Sajuyigbe, Soumith Chintala, Stephanie Max, Stephen Chen, Steve Kehoe, Steve Satterfield, Sudarshan Govindaprasad, Sumit Gupta, Summer Deng, Sungmin Cho, Sunny Virk, Suraj Subramanian, Sy Choudhury, Sydney Goldman, Tal Remez, Tamar Glaser, Tamara Best, Thilo Koehler, Thomas Robinson, Tianhe Li, Tianjun Zhang, Tim Matthews, Timothy Chou, Tzook Shaked, Varun Vontimitta, Victoria Ajayi, Victoria Montanez, Vijai Mohan, Vinay Satish Kumar, Vishal Mangla, Vlad Ionescu, Vlad Poenaru, Vlad Tiberiu Mihailescu, Vladimir Ivanov, Wei Li, Wenchen Wang, Wenwen Jiang, Wes Bouaziz, Will Constable, Xiaocheng Tang, Xiaojian Wu, Xiaolan Wang, Xilun Wu, Xinbo Gao, Yaniv Kleinman, Yanjun Chen, Ye Hu, Ye Jia, Ye Qi, Yenda Li, Yilin Zhang, Ying Zhang, Yossi Adi, Youngjin Nam, Yu, Wang, Yu Zhao, Yuchen Hao, Yundi Qian, Yunlu Li, Yuzi He, Zach Rait, Zachary

DeVito, Zef Rosnbrick, Zhaoduo Wen, Zhenyu Yang, Zhiwei Zhao, and Zhiyu Ma. 2024. The llama 3 herd of models. *Preprint*, arXiv:2407.21783.

Daniel Hladek, Ján Staš, and Matus Pleva. 2020. Survey of automatic spelling correction.

Erik Jones, Anca Dragan, Aditi Raghunathan, and Jacob Steinhardt. 2023. Automatically auditing large language models via discrete optimization. In *International Conference on Machine Learning*, pages 15307–15329. PMLR.

Ryan Koo, Minhwa Lee, Vipul Raheja, Jong Inn Park, Zae Myung Kim, and Dongyeop Kang. 2024. Benchmarking cognitive biases in large language models as evaluators. In *ACL (Findings)*, pages 517–545.

Sneha Kudugunta, Isaac Caswell, Biao Zhang, Xavier Garcia, Derrick Xin, Aditya Kusupati, Romi Stella, Ankur Bapna, and Orhan Firat. 2023. Madlad-400: a multilingual and document-level large audited dataset. In *Proceedings of the 37th International Conference on Neural Information Processing Systems*, NIPS '23, Red Hook, NY, USA. Curran Associates Inc.

Alexey Kurakin, Ian J. Goodfellow, and Samy Bengio. 2016. Adversarial examples in the physical world. *CoRR*, abs/1607.02533.

Patrick Lewis, Ethan Perez, Aleksandra Piktus, Fabio Petroni, Vladimir Karpukhin, Naman Goyal, Heinrich Küttler, Mike Lewis, Wen-tau Yih, Tim Rocktäschel, Sebastian Riedel, and Douwe Kiela. 2020. Retrieval-augmented generation for knowledge-intensive nlp tasks. In *Proceedings of the 34th International Conference on Neural Information Processing Systems*, NIPS '20, Red Hook, NY, USA. Curran Associates Inc.

Chaofan Li, Zheng Liu, Shitao Xiao, and Yingxia Shao. 2023a. Making large language models a better foundation for dense retrieval. *CoRR*, abs/2312.15503.

Dawei Li, Renliang Sun, Yue Huang, Ming Zhong, Bohan Jiang, Jiawei Han, Xiangliang Zhang, Wei Wang, and Huan Liu. 2025. Preference leakage: A contamination problem in llm-as-a-judge. *Preprint*, arXiv:2502.01534.

Dianqi Li, Yizhe Zhang, Hao Peng, Liqun Chen, Chris Brockett, Ming-Ting Sun, and Bill Dolan. 2021. Contextualized perturbation for textual adversarial attack. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 5053–5069, Online. Association for Computational Linguistics.

Xianming Li and Jing Li. 2024. AoE: Angle-optimized embeddings for semantic textual similarity. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1825–1839, Bangkok, Thailand. Association for Computational Linguistics.

Xiaopeng Li, Lixin Su, Pengyue Jia, Xiangyu Zhao, Suqi Cheng, Junfeng Wang, and Dawei Yin. 2023b. Agent4ranking: Semantic robust ranking via personalized query rewriting using multi-agent llm. *Preprint*, arXiv:2312.15450.

Zehan Li, Xin Zhang, Yanzhao Zhang, Dingkun Long, Pengjun Xie, and Meishan Zhang. 2023c. Towards general text embeddings with multi-stage contrastive learning. *Preprint*, arXiv:2308.03281.

Leon Lin, Hannah Brown, Kenji Kawaguchi, and Michael Shieh. 2025. Single character perturbations break llm alignment. *Proceedings of the AAAI Conference on Artificial Intelligence*, 39:27473–27481.

John Morris, Eli Lifland, Jin Yong Yoo, Jake Grigsby, Di Jin, and Yanjun Qi. 2020. TextAttack: A framework for adversarial attacks, data augmentation, and adversarial training in NLP. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, pages 119–126, Online. Association for Computational Linguistics.

Niklas Muennighoff, Nouamane Tazi, Loic 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*, pages 2014–2037, Dubrovnik, Croatia. Association for Computational Linguistics.

Yixin Nie, Adina Williams, Emily Dinan, Mohit Bansal, Jason Weston, and Douwe Kiela. 2020. Adversarial NLI: A new benchmark for natural language understanding. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 4885–4901, Online. Association for Computational Linguistics.

Rodrigo Nogueira and Kyunghyun Cho. 2020. Passage re-ranking with bert. *Preprint*, arXiv:1901.04085.

Zach Nussbaum, John Xavier Morris, Andriy Mulyar, and Brandon Duderstadt. 2025. Nomic embed: Training a reproducible long context text embedder. *Transactions on Machine Learning Research*. Reproducibility Certification.

Gustavo Penha, Arthur Câmara, and Claudia Hauff. 2022. Evaluating the robustness of retrieval pipelines with query variation generators. In *Advances in Information Retrieval*, pages 397–412, Cham. Springer International Publishing.

Qwen, :, An Yang, Baosong Yang, Beichen Zhang, Binyuan Hui, Bo Zheng, Bowen Yu, Chengyuan Li, Dayiheng Liu, Fei Huang, Haoran Wei, Huan Lin, Jian Yang, Jianhong Tu, Jianwei Zhang, Jianxin Yang, Jiaxi Yang, Jingren Zhou, Junyang Lin, Kai Dang, Keming Lu, Keqin Bao, Kexin Yang, Le Yu, Mei Li, Mingfeng Xue, Pei Zhang, Qin Zhu, Rui Men, Runji Lin, Tianhao Li, Tianyi Tang, Tingyu Xia, Xingzhang Ren, Xuancheng Ren, Yang Fan, Yang Su, Yichang Zhang, Yu Wan, Yuqiong Liu, Zeyu Cui, Zhenru Zhang, and Zihan Qiu. 2025. Qwen2.5 technical report. *Preprint*, arXiv:2412.15115.

Barbara Rychalska, Dominika Basaj, Alicja Gosiewska, and Przemysław Biecek. 2019. Models in the wild: On corruption robustness of neural nlp systems. In *Neural Information Processing*, pages 235–247, Cham. Springer International Publishing.

Nandan Thakur, Nils Reimers, Andreas Rücklé, Abhishek Srivastava, and Iryna Gurevych. 2021. BEIR: A heterogeneous benchmark for zero-shot evaluation of information retrieval models. In *Thirty-fifth Conference on Neural Information Processing Systems Datasets and Benchmarks Track (Round 2)*.

Hikaru Tomonari, Masaaki Nishino, and Akihiro Yamamoto. 2022. Robustness evaluation of text classification models using mathematical optimization and its application to adversarial training. In *Findings of the Association for Computational Linguistics: AACL-IJCNLP 2022*, pages 327–333, Online only. Association for Computational Linguistics.

Alex Wang, Amanpreet Singh, Julian Michael, Felix Hill, Omer Levy, and Samuel Bowman. 2018. GLUE: A multi-task benchmark and analysis platform for natural language understanding. In *Proceedings of the 2018 EMNLP Workshop BlackboxNLP: Analyzing and Interpreting Neural Networks for NLP*, pages 353–355, Brussels, Belgium. Association for Computational Linguistics.

Boxin Wang, Chejian Xu, Shuohang Wang, Zhe Gan, Yu Cheng, Jianfeng Gao, Ahmed Hassan Awadallah, and Bo Li. 2021. Adversarial glue: A multi-task benchmark for robustness evaluation of language models. In *Advances in Neural Information Processing Systems*.

Liang Wang, Nan Yang, Xiaolong Huang, Binxing Jiao, Linjun Yang, Daxin Jiang, Rangan Majumder, and Furu Wei. 2024a. Text embeddings by weakly-supervised contrastive pre-training. *Preprint*, arXiv:2212.03533.

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

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

Zhiguo Wang, Patrick Ng, Xiaofei Ma, Ramesh Nallapati, and Bing Xiang. 2019. Multi-passage BERT: A globally normalized BERT model for open-domain question answering. 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 5878–5882, Hong Kong, China. Association for Computational Linguistics.

Benjamin Warner, Antoine Chaffin, Benjamin Clavié, Orion Weller, Oskar Hallström, Said Taghadouini, Alexis Gallagher, Raja Biswas, Faisal Ladhak, Tom Aarsen, Nathan Cooper, Griffin Adams, Jeremy Howard, and Iacopo Poli. 2024. Smarter, better, faster, longer: A modern bidirectional encoder for fast, memory efficient, and long context finetuning and inference. *Preprint*, arXiv:2412.13663.

Alexander Wei, Nika Haghtalab, and Jacob Steinhardt. 2023. Jailbroken: How does llm safety training fail? In *Advances in Neural Information Processing Systems*, volume 36, pages 80079–80110. Curran Associates, Inc.

Chen Wu, Ruqing Zhang, Jiafeng Guo, Yixing Fan, and Xueqi Cheng. 2022. Are neural ranking models robust? *ACM Trans. Inf. Syst.*, 41(2).

Shitao Xiao, Zheng Liu, Peitian Zhang, Niklas Muennighoff, Defu Lian, and Jian-Yun Nie. 2024. C-pack: Packed resources for general chinese embeddings. SIGIR '24, page 641–649, New York, NY, USA. Association for Computing Machinery.

Haoran Xu, Kenton Murray, Philipp Koehn, Hieu Hoang, Akiko Eriguchi, and Huda Khayrallah. 2025. X-ALMA: Plug & play modules and adaptive rejection for quality translation at scale. In *The Thirteenth International Conference on Learning Representations*.

Yuan Zang, Fanchao Qi, Chenghao Yang, Zhiyuan Liu, Meng Zhang, Qun Liu, and Maosong Sun. 2020. Word-level textual adversarial attacking as combinatorial optimization. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 6066–6080, Online. Association for Computational Linguistics.

Guoyang Zeng, Fanchao Qi, Qianrui Zhou, Tingji Zhang, Zixian Ma, Bairu Hou, Yuan Zang, Zhiyuan Liu, and Maosong Sun. 2021. OpenAttack: An open-source textual adversarial attack toolkit. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing: System Demonstrations*, pages 363–371, Online. Association for Computational Linguistics.

Tianyi Zhang, Varsha Kishore, Felix Wu, Kilian Q. Weinberger, and Yoav Artzi. 2020. Bertscore: Evaluating text generation with bert. In *International Conference on Learning Representations*.

Xin Zhang, Yanzhao Zhang, Dingkun Long, Wen Xie, Ziqi Dai, Jialong Tang, Huan Lin, Baosong Yang, Pengjun Xie, Fei Huang, Meishan Zhang, Wenjie Li, and Min Zhang. 2024. mGTE: Generalized long-context text representation and reranking models for multilingual text retrieval. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing: Industry Track*, pages 1393–1412, Miami, Florida, US. Association for Computational Linguistics.

Ming Zhong, Zhizhi Wu, and Nanako Honda. 2024. Deep learning based dense retrieval: A comparative study. *Preprint*, arXiv:2410.20315.

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 *Proceedings of the Annual International ACM SIGIR Conference on Research and Development in Information Retrieval in the Asia Pacific Region*, SIGIR-AP '23, page 212–222, New York, NY, USA. Association for Computing Machinery.

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*, pages 2836–2842, Online and Punta Cana, Dominican Republic. Association for Computational Linguistics.

Shengyao Zhuang and Guido Zuccon. 2022. Characterbert and self-teaching for improving the robustness of dense retrievers on queries with typos. In *Proceedings of the 45th International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR '22, page 1444–1454, New York, NY, USA. Association for Computing Machinery.

## A Backtranslation model selection

We employed two metrics to select the optimal model for backtranslation: BERTScore and mean Levenshtein distance. BERTScore was used to assess semantic similarity to the original query, while mean Levenshtein distance measured the degree of perturbation. As BERTScore values were comparable across all configurations, model selection was primarily based on the highest mean Levenshtein distance. We evaluated X-ALMA (Xu et al., 2025) and MADLAD-400 (Kudugunta et al., 2023) models. Detailed evaluation results are provided in Table 7.

| Model | Middle Language | Mean Levenshtein | Mean BERTscore |
|---|---|---|---|
| X-ALMA-13B | German | 40.676 | 0.852 |
| X-ALMA-13B | Korean | 65.166 | 0.853 |
| MADLAD-400-3B | German | 26.698 | 0.851 |
| MADLAD-400-3B | Korean | 46.248 | 0.852 |
| MADLAD-400-3B | Polish | 30.568 | 0.851 |
| MADLAD-400-10B | German | 27.582 | 0.851 |
| MADLAD-400-10B | Korean | 44.874 | 0.852 |
| MADLAD-400-10B | Polish | 30.399 | 0.851 |

Table 7: Mean Levenshtein distance and BERTscore are reported for all configurations considered during the model selection process for backtranslation. Given that all options recorded nearly identical BERTscore values, the model with the highest mean Levenshtein distance was chosen.

## B Models, Prompts and Sampling Parameters

We provide here the dense retrievers used in our evaluation, Cross Encoders utilized for paraphrase selection, prompts, and sampling parameters used for (i) automatic paraphrase generation and (ii) LLM-as-a-judge evaluation. List of evaluated dense retrievers is presented in Table 8. Cross Encoders used to select paraphrases for the benchmark are presented in Table 9. Figures 8 and 9 show the exact prompts used. Table 10 summarizes the sampling parameters, we used sampling parameters suggested by the authors of models.

| Model | Reference | Source |
|---|---|---|
| UAE-Large | Li and Li (2024) | Model card |
| bge-large | Xiao et al. (2024) | Model card |
| gte large | Li et al. (2023c) | Model card |
| modernbert | Warner et al. (2024) and Nussbaum et al. (2025) | Model card |
| e5-large-instruct | Wang et al. (2024c) | Model card |
| gte-Qwen2-7B-instruct | Li et al. (2023c) | Model card |

Table 8: List of dense retrievers used in the evaluation process.

| Model | Reference | Source |
|---|---|---|
| gte-multilingual-base | Zhang et al. (2024) | Model card |
| gte-reranker-modernbert | Li et al. (2023c) | Model card |
| bge-reranker-v2-m3 | Chen et al. (2024) | Model card |
| jina-reranker-v2 | Press release | Model card |
| bge-reranker-v2.5-gemma2 | Li et al. (2023a) | Model card |

Table 9: List of Cross Encoders used in the process of selecting paraphrases for the benchmark.

| Parameter | Paraphraser | Judge |
|---|---|---|
| Temperature | 0.9 | 0.6 |
| Top-p | 0.9 | 0.9 |

Table 10: Sampling parameters for paraphrase generation ("Paraphraser") and evaluation ("Judge").

## C Licenses of the datasets

We share the licenses of the evaluation datasets in Table 11.

## D Selecting perturbation strength for all methods

To ensure the intelligibility of each perturbed query, we established an appropriate perturbation strength for each method. We began by setting initial perturbation strength levels based on our intuition. We then randomly sampled 100 queries from the benchmark and applied all perturbation methods to this

| Dataset | License | Information |
|---|---|---|
| MSMARCO | CC BY 4.0 | Link |
| TREC-COVID | No license | Link |
| NFCorpus | Non-commercial | Link |
| NQ | CC BY-SA 3.0 | Link |
| HotpotQA | CC BY-SA 4.0 | Link |
| FiQA-2018 | No license | Link |
| ArguAna | CC BY 4.0 | Link |
| Touche-2020 | CC BY 4.0 | Link |
| CQADupstack | CC BY-SA 3.0 | Link |
| Quora | Quora Terms of Use | Link |
| DBPedia | CC BY-SA 3.0 | Link |
| SCIDOCS | CC BY 4.0 | Link |
| Fever | CC BY-SA 3.0 | Link |
| Climate-Fever | No license | Link |
| SciFact (claims) | CC BY 4.0 | Link |

Table 11: Licenses and source information of the datasets evaluated in our work.

test set. The resulting queries were manually annotated to assess semantic preservation and intelligibility. If more than 95% of the samples at a given perturbation strength were judged valid, that strength was selected for subsequent experiments. If the test sample did not meet these requirements, we performed another round of annotation, adjusting the perturbation strength for particular methods as needed. The selected perturbation strengths for all methods are summarized in Table 12.

| Symbol | Name | Strength |
|---|---|---|
| P1 | Capitalization | 50% |
| P2 | Keyboard Character Insert | 5% |
| P3 | Keyboard Character Replace | 5% |
| P4 | Random Neighbor Character Swap | 5% |
| P5 | OCR Error | 5% |
| P6 | Punctuation | 40% |
| P7 | Random Character Delete | 5% |
| P8 | Random Character Insert | 5% |
| P9 | Random Character Replace | 5% |
| P10 | Mobile Phone Character Miss | 5% |
| P11 | Context-Aware Perturbation | 10% |
| P12 | Word Duplicate | 10% |
| P13 | Words Join | 30%. |
| P14 | Word Lemmatization | 100% |
| P15 | Word Position Swap | 10% |
| P16 | Word Stemming | 100% |

Table 12: Selected perturbation strength for each perturbation method applied in our work. There is no perturbation strength for sentence-level perturbations, since the whole input query is perturbed during this process.

## E Annotation evaluation

In this section, we present additional results from the evaluation of manual and automatic paraphrase annotation. Figure 4 shows Cohen's kappa and percentage agreement between annotators for observations, which were not annotated as exact or near duplicates by the DeepSeek-R1-Distill-Llama model.

Figure 5 shows the average summed reverse rank (see Equation 2) for all models used in the Automatic Paraphrase (P17) method. Larger versions of Qwen achieve higher scores on average. Backtranslation outperforms Qwen models with 0.5B and 7B parameters in terms of reverse rank. However, the scores were computed only for valid paraphrases, and backtranslation generates a higher proportion of invalid outputs compared to Qwen 7B. Thus, while backtranslation produces higher-quality paraphrases when successful, it tends to output more invalid samples than medium- and large-sized LLMs like Qwen.

## F Example of disrupted tokenization

We present here an example of how the tokenization process can be disrupted by word level perturbations. Looking at the results of using ModernBERT's tokenizer on the word `"expecting"` and its various perturbed forms:

- `"expecting"` - original (unperturbed) word, tokens: `["expect", "ing"]`,
- `"expecying"` - character-level perturbed (P3) tokens: `["ex", "pe", "cy", "ing"]`,
- `"epxecting"` - character-level perturbed (P4) tokens: `["ep", "x", "ect", "ing"]`,
- `"awaiting"` - word-level perturbed (P11) tokens: `["aw", "ait", "ing"]`,
- `"expect"` - word-level perturbed (P11) tokens: `["expect"]`,

we can observe that making one change on the character level can lead to significant changes and results with more noisy input tokens, whereas word-level change will usually retain some of the original tokens or the core information (even tough perturbations on both levels can increase the number of tokens).

## G Visualizing individual examples

The main annotation results shared in the article show that the human was more rigorous during the labeling process and classified the most cases as Invalid among all annotators. While manually reviewing the annotation results, we noticed that the models sometimes labeled paraphrases that differed significantly in semantics from the original query as valid ones. To visualize this phenomenon,

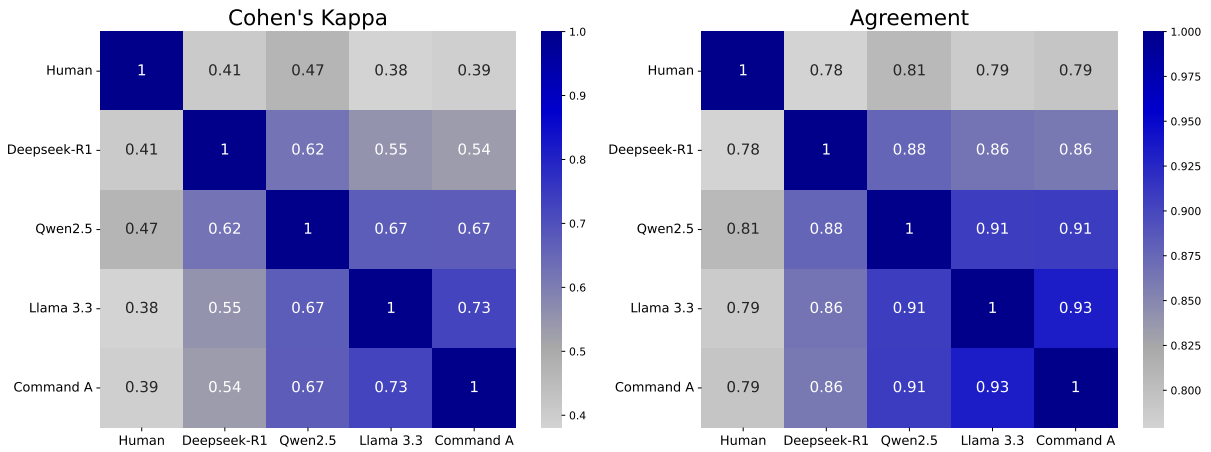


Figure 4: Inter-annotator agreement metrics for human-model and model-model pairs (without observations labeled as 'duplicate' by the DeepSeek-R1-Distill-Llama model). Agreement is the percentage of observations which had the same label assigned by both annotators within a given pair.

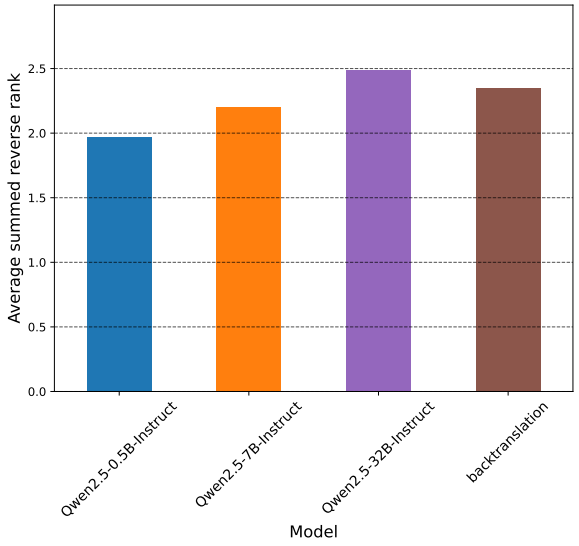


Figure 5: Average summed reverse rank based on reranker scores for Qwen models and the backtranslation model. The calculation includes only valid paraphrases generated for the benchmark dataset.

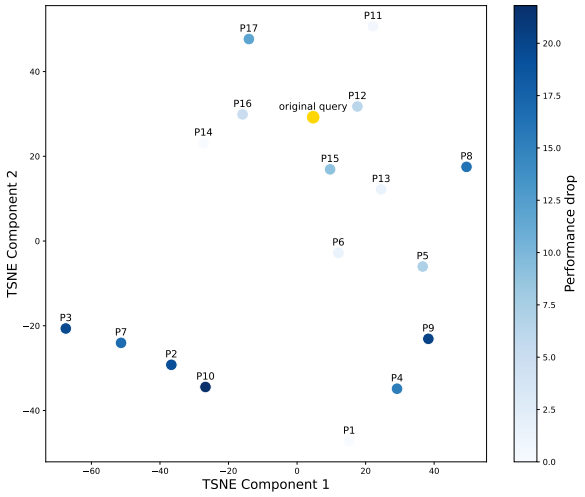


Figure 6: A t-SNE projection of bge large embeddings calculated for the query "how many eggs do rouen ducks lay a year" from the MSMARCO dataset. Points on the plot are colored based on the mean drop in NDCG@10 metric value recorded for specific perturbation method applied to our benchmark dataset. Projection of original query has been drawn in gold color.

we present the labels assigned by human annotator and Llama 3.3 for specific observations in Table 13.

In Figure 6, we visualize a t-SNE projection of the embeddings generated by the bge large model for the query 'how many eggs do rouen ducks lay a year' and its perturbations.

## H Influence of query length

Models are generally expected to exhibit greater resilience to perturbations when processing lengthy queries compared to shorter ones. In the case of short queries, which typically consist of key terms, perturbations are more likely to affect the most informative tokens, thereby significantly altering the semantic content. Conversely, longer queries provide a more extensive context, allowing the semantic meaning to be distributed across a larger number of tokens. As a result, perturbations in longer queries are less likely to substantially impact the overall meaning, as the model can still infer the intended semantics from the unperturbed portions of the query. In Figure 7, we present the relationship between the median query length and the change in performance for models across various benchmark datasets. Longer queries, such as

| Dataset Name | Query | Paraphrase | Human | Model |
|---|---|---|---|---|
| MSMARCO | why is housekeeping so important | housekeeping is essential for maintaining cleanliness, organization, and cleanliness throughout the house. | Invalid | Valid |
| NQ | who said i have just begun to fight | What did someone say they had started fighting? | Invalid | Valid |
| MSMARCO | what is a shoul | define shoulder | Invalid | Valid |
| SCIDOCS | Combining concept hierarchies and statistical topic models | Merging Concept Hierarchies with Statistical Topic Models | Valid | Duplicate |
| CQADupstack English | "Big black eyes" vs. "big and black eyes" | "difference between big black eyes and big and black eyes" | Valid | Invalid |
| CQADupstack Tex | What is a good way to show changes between two versions | What is a good way to show the changes between the two versions? | Duplicate | Valid |
| HotpotQA | Which documentary film was released first Tar Creek or Volcanic Sprint? | What documentary film has been released earlier than Tar Creek or Volcanic Sprint? | Invalid | Valid |

Table 13: Comparison between human and automatic annotation of Llama 3.3 on selected examples. We found that human is more restrictive than the LLM, which sometimes classifies a perturbation as correct despite significant changes in semantics. The term 'Human' refers to labels assigned by human annotators, while 'Model' represents labels generated by the model. The labels 'Valid,' 'Invalid,' and 'Duplicate' are highlighted with green, red, and yellow colors respectively.

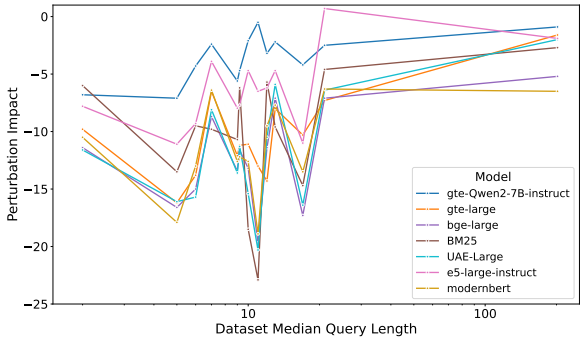


Figure 7: Perturbation impact for character-level perturbations depending on median query length in benchmark datasets. Perturbation impact is the mean difference in performance after and before applying perturbations.

those in the Arguana dataset, can help mitigate the performance drop caused by perturbations. However, for other datasets where queries are relatively short and the differences in query lengths among these datasets are minimal, this relationship is not observable.

## I Evaluation details

We conducted experiments on 15 diverse datasets and observed a significant performance drop in all examined models with each run. Given the substantial sample size, number of models and created perturbation methods, we report metrics based on a single evaluation round.

You are a linguistic evaluator. Your task is to assess whether a given paraphrase accurately conveys the same meaning as the original sentence.

**Instructions:**

1. The paraphrase must retain the same semantic meaning as the original.
2. It must include exactly the same amount of information as the original, without adding or omitting knowledge.
3. It must not imply much additional context or introduce many new interpretations.

**Do not penalize the paraphrase in the following situations:**

- **Acronyms and Expansions:** When the paraphrase substitutes between full names and acronyms.
  - **Example:**
    - Original: 'Organisation for Economic Co-operation and Development'
    - Paraphrase: 'What does OECD stand for?'
- **Synonyms or Equivalent Phrases:** When words or phrases are substituted with synonyms that preserve meaning.
  - **Example:**
    - Original: 'How to cook pasta quickly?'
    - Paraphrase: 'How to make pasta fast?'
- **Reordering or Structural Variations:** When words are reordered or the sentence is restructured, retaining the same intent.
  - **Example:**
    - Original: 'What is the capital of France?'
    - Paraphrase: 'France's capital is what?'
- **Implicit vs. Explicit Questions:** When phrasing switches between implicit and explicit forms without changing intent.
  - **Example:**
    - Original: 'What is Einstein known for?'
    - Paraphrase: 'What scientific contributions is Einstein famous for?'
- **Conversion Between Formats (e.g., Questions and Statements):** When a question is transformed into a statement or vice versa.
  - **Example:**
    - Original: 'Explain photosynthesis.'
    - Paraphrase: 'What is photosynthesis?'
- **Variations in Focus or Emphasis:** When emphasis shifts between parts of the sentence without altering meaning.
  - **Example:**
    - Original: 'Who discovered gravity?'
    - Paraphrase: 'Gravity was discovered by whom?'
- **Variations in Granularity:** When slight changes in specificity occur, but context implies equivalence.
  - **Example:**
    - Original: 'How many planets are in the solar system?'
    - Paraphrase: 'How many planets orbit the Sun?'
- **Simplifications That Retain Meaning:** When language is condensed or simplified while keeping the same intent.
  - **Example:**
    - Original: 'Steps to create a new email account on Gmail?'
    - Paraphrase: 'How to set up Gmail?'
- **Alternative Representations of Numerical Information:** When numerical formats or ranges are switched.
  - **Example:**
    - Original: 'What happened in the 20th century?'
    - Paraphrase: 'What events occurred between 1901 and 2000?'
- **Contextual Inferences with Unambiguous Terms:** When shorter or implicit expressions are used, remaining clear in context.
  - **Example:**
    - Original: 'What is the full form of NATO?'
    - Paraphrase: 'What does NATO stand for?'
- **Differences in Question Type (Why/How):** When closely related question types are switched but lead to the same answer.
  - **Example:**
    - Original: 'Why is the sky blue?'
    - Paraphrase: 'How does the sky appear blue?'

If the paraphrase changed the style of original sentence to the search query, e.g., 'What is the capital of France?' to 'Search for capital of France', then this kind transformation is acceptable.

For each pair of sentences, return a Python dictionary object with:

- `label`: 0 if the paraphrase is accurate, 1 otherwise.
- If the paraphrase is exactly the same or very close to the original sentence, the label should be set to "duplicate".

**Three examples:**

**Example 1:** Original sentence: 'The cat is sitting on the mat.'
Paraphrase: 'The mat has a cat sitting on it.'
Output: `{"label": 0}`

**Example 2:** Original sentence: 'The cat is sitting on the mat.'
Paraphrase: 'The cat is on the mat and it looks hungry. The dog wants to go to the pet store.'
Output: `{"label": 1}`

**Example 3:** Original sentence: 'The cat is sitting on the mat.'
Paraphrase: 'The cats are sitting on the mat.'
Output: `{"label": "duplicate"}`

Figure 8: Instructions given to human and automatic evaluator during the paraphrase annotation task. The information about desired output format and the linguistic evaluator role was given only to the automatic evaluator.

**Base prompt:**

You are a helpful, well educated assistant whose role is to generate a paraphrase of the supplied text. Output only the paraphrase, without any additional text. Do not insert additional knowledge. Keep the style and length of the text. Make sure to alter the original text. {*TASK_TYPE*}

**Task types:**

- **Question:** Your output should be in the form of a question.
- **Search Query:** Your output should be in the form of a short search query.
- **Argument:** Your output should be in the form of an argument.
- **Article Title:** Your output should be in the form of an article title.
- **Claim:** Your output should be in the form of a claim.

**Dataset - task type mapping:**

- **Question:** HotpotQA, Touche-2020, Quora
- **Search Query:** MSMARCO, TREC-COVID, NFCorpus, NQ, FIQA-2018, CQADupstack, DBPedia
- **Argument:** ArguAna
- **Article Title:** SCIDOCS
- **Claim:** FEVER, Climate-FEVER, SciFact

Figure 9: Prompt instructions used for paraphrase generation by the language model. The final prompt is constructed by combining a general base prompt with an instruction specific to the task type (e.g., question, search query). In this figure, the set of possible task types is listed directly below the base prompt, while the mapping from each dataset to its corresponding task type is provided at the bottom of the figure.