Unsupervised Corpus Poisoning Attacks in Continuous Space for Dense Retrieval

Yongkang Li
y.li7@uva.nl
University of Amsterdam
Amsterdam, The Netherlands

Panagiotis Eustratiadis
p.efstratiadis@uva.nl
University of Amsterdam
Amsterdam, The Netherlands

Simon Lupart
s.c.lupart@uva.nl
University of Amsterdam
Amsterdam, The Netherlands

Evangelos Kanoulas
e.kanoulas@uva.nl
University of Amsterdam
Amsterdam, The Netherlands

Abstract

This paper concerns corpus poisoning attacks in dense information retrieval, where an adversary attempts to compromise the ranking performance of a search algorithm by injecting a small number of maliciously generated documents into the corpus. Our work addresses two limitations in the current literature. First, attacks that perform adversarial gradient-based word substitution search do so in the discrete lexical space, while retrieval itself happens in the continuous embedding space. We thus propose an optimization method that operates in the embedding space directly. Specifically, we train a perturbation model with the objective of maintaining the geometric distance between the original and adversarial document embeddings, while also maximizing the token-level dissimilarity between the original and adversarial documents. Second, it is common for related work to have a strong assumption that the adversary has prior knowledge about the queries. In this paper, we focus on a more challenging variant of the problem where the adversary assumes no prior knowledge about the query distribution (hence, unsupervised). Our core contribution is an adversarial corpus attack that is fast and effective. We present comprehensive experimental results on both in- and out-of-domain datasets, focusing on two related tasks: a top-1 attack and a corpus poisoning attack. We consider attacks under both a white-box and a black-box setting. Notably, our method can generate successful adversarial examples in under two minutes per target document; four times faster compared to the fastest gradient-based word substitution methods in the literature with the same hardware. Furthermore, our adversarial generation method generates text that is more likely to occur under the distribution of natural text (low perplexity), and is therefore more difficult to detect.

CCS Concepts

• **Information systems → Retrieval models and ranking**; • **Computing methodologies → Natural language processing**.

Keywords

Dense Retrieval, Adversarial Attack, Corpus Poisoning

1 Introduction

Dense retrieval [15] has become a widely used paradigm in information retrieval (IR), utilizing neural language models to encode queries and documents [2]. However, such neural models are susceptible to adversarial attacks [38], making adversarial robustness research an important topic in IR. Adversarial attacks in IR typically aim to compromise the ranking performance of a retrieval model (e.g., [43]). In this paper, we specifically focus on corpus poisoning attacks (e.g., [19, 37, 46]), where an adversary attacks by injecting maliciously generated documents into a corpus. The attacker aims to promote uninformative documents and maximize their visibility in the top-ranked results of arbitrary search rankings. We assume that existing documents in the corpus have already been encoded and indexed, and therefore we do not have edit access i.e., we may not *replace* one document with another. Instead, we can only generate new documents to add to the corpus, analogous to how search engines like Google and Bing continuously index newly added web documents, making them available for retrieval.

Contemporary poisoning methods aim to pollute the corpus with documents that not only achieve high rankings, but also are nonsensical to users. It is worth noting that under this threat model *imperceptibility is neither required nor feasible* [6]. The attack is considered successful when a user encounters the adversarial document positioned at the top, reads it, and perceives it as useless. Most previous studies have focused on gradient-based word substitution, e.g., models based on HotFlip [11, 36, 37, 43, 46]. Such methods first duplicate an existing document in the corpus, and then iteratively replace individual tokens with new ones, adversarially generated to maximize the retriever's error. However, this not only results in a significant time complexity ([47] report 2 hours of search time for 50 tokens using an NVIDIA A100 GPU), but also induces a misalignment of objectives, as each replacement of a single token occurs in the lexical space, while retrieval itself computes the representation of the entire document in the embedding space. Bridging this gap is challenging, as there are discontinuities in the adversarial generation process, e.g., decoding token embedding samples from the language distribution of the decoder [17].

Moreover, most attacks in dense retrieval commonly make the assumption of a target query at the time of attack, that is used to inform how documents are corrupted. To name a few, AGGD[37], IDEM[5], PRADA [43], PAT [21], MCARA [24], TARA [22], as well as the aforementioned corpus poisoning attacks. We claim that this

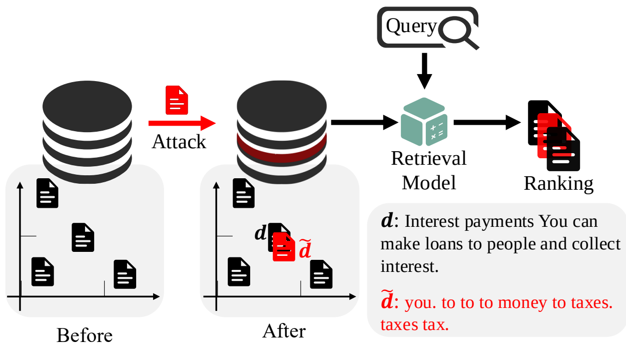


Figure 1: Illustration of our unsupervised corpus poisoning attack under our threat model. We attack a retriever's ranking performance by generating uninformative documents with high relevance scores. For example, encoding the original document d and its adversarial counterpart $\tilde{d}$ with SimLM produces similar embeddings, but $\tilde{d}$ is nonsensical.

is a strong assumption from a practical standpoint, as we cannot always rely on knowing the target queries in advance, as well as from a scientific standpoint, as it raises concerns of overfitting the attack models on specific queries. In this paper, we introduce a more realistic and challenging scenario, termed *unsupervised* corpus poisoning (Figure 1). In this context, "unsupervised" refers to the fact that there is no prior knowledge about the query distribution at the time of attack, and the attack method itself is only informed by the target document.

We address these two current limitations and propose a corpus poisoning attack that operates directly in the embedding, rather than lexical space. Our method consists of two main components: a reconstruction model and a perturbation model. The reconstruction model can recover a document from its contextual token-level embeddings. The perturbation model is trained to maintain the geometric distance between the original and adversarial document embeddings, while also maximizing the token-level dissimilarity between the original and adversarial documents. We examine two types of adversarial attacks that differ in their criteria for target document selection: one that corrupts the top-1 document of an arbitrary ranking (the query that produced the ranking is unknown), and one that corrupts the k most "central" documents in a corpus, with the hypothesis that these documents affect a lot of queries.

Our method is a direct improvement over state-of-the-art (SOTA) HotFlip-based methods. It performs up to par in white-box attack scenarios, but demonstrates stronger transferability properties in a black-box setting. Our attack is both fast and effective, generating successful adversarial examples at four times the speed of the fastest HotFlip-based approach [19]. Moreover, the incorporation of a reconstruction model ensures that our generated outputs closely mimic natural documents, resulting in significantly lower perplexity compared to HotFlip-based methods, making them more difficult to detect by perplexity-based filtering.

Finally, we briefly discuss that the computational efficiency of our method enables the possibility of adversarial training, by using the generated adversarial documents as negative samples. We do not present extensive experimentation in this direction, but enough to suggest to the reader that it is a promising direction for future work.

2 Related work

Dense Retrieval: Following the initial success of dense retrieval models with DPR [15], recent advancements include topic-aware sampling (TAS-B [12]), unsupervised training with intermediate pseudo queries (Contriever [14]), data augmentation under diversity constraints (DRAGON+ [20]), and pre-training with representation bottleneck (SimLM [41]). Our work investigates the vulnerability and robustness of these SOTA models.

Word Substitution Attacks: Our method belongs to the family of word substitution ranking attacks [42], similar to PRADA [43], MCARA [24] or TARA [22], as well as the family of corpus poisoning attacks such as Order-Disorder [21] and HotFlip [11, 19, 37, 46]. However, these methods assume prior knowledge about a query distribution that guides training, and perturbation search. Our work introduces the novel, *unsupervised* setting, and is uniquely positioned separately from these methods, as we make no assumptions about what kind of queries correspond to the attacked rankings/corpus. The only training signal for our attack method is the document itself. To contextualize our work within the existing literature, we mention that our work is in a similar direction as Zhong et al. [46], except (i) no queries are used during training, (ii) perturbation search happens in the embedding space, and (iii) our approach is significantly faster and generates adversarial documents with lower perplexity.

Embedding Space Perturbations: Our work makes use of a reconstruction model, for which we draw inspiration from Vec2Text-based approaches [28, 47]. We build upon the Vec2Text paradigm by learning instance-wise optimal perturbations, rather than being restricted to additive random noise. Furthermore, our motivation to work in the embedding space, and not on a token level, stems from recent work that leverages Transformer models to generate more powerful attacks, e.g., BERT-ATTACK [18].

Imperceptibility of Attacks: There is a line of adversarial IR work that corrupts documents with word substitution attacks [22–24, 43], aiming to maintain imperceptibility. This imperceptibility refers to preserving the original document's semantics while boosting its ranking during attacks. However, in agreement with [6], we argue that imperceptibility is neither necessary nor practical from the user's perspective. This is because an attack succeeds only when the user actively reads the top-ranked adversarial document and perceives it as nonsensical, We found this assumption to be compatible with the "realistic attack" angle of our work.

3 Methodology

In this section, we first introduce the foundational concepts of dense bi-encoder models. Then, we describe our unsupervised corpus poisoning adversarial attack settings, which lead to two conditions for effective adversarial documents. Based on these conditions, we design an optimization process for generating adversarial content, utilizing both a reconstruction model and a perturbation model.

3.1 Preliminaries

The task of dense retrieval concerns scoring a collection of documents, i.e., corpus, $D = \{d_1, d_2, ..., d_{|D|}\}$ according to their relevance against a query, $q \in Q$. To do so, queries and documents are projected as vectors onto an embedding space by a neural language encoder, and relevance is defined as the dot product, cosine similarity, or L2 Euclidean distance of these vectors. We denote $E(\cdot)$ as the encoder, and its output is token-level embeddings for the entire document, $e_d = E(d) \in \mathbb{R}^{|d| \times \hbar}$, where $\hbar$ is the hidden dimension of the retrieval encoder, typically $\hbar = 768$. d is any document in D, and $|d|$ denotes the number of tokens in d, where $d = \left\{t_1, t_2, ..., t_{|d|}\right\}$ after tokenization. Each t_i is a token from the vocabulary V, and the size of V is $|V|$.

Note that we use the same encoder for queries and documents (i.e., weight-sharing), though different encoders could be used. Since we focus on a query-independent formulation, this design hyperparameter is not crucial. For scoring $\text{sim}(\cdot)$ during retrieval, we rely solely on the `[CLS]` token as the document embedding, while the specific scoring function depends on the retriever. However, for reconstruction, we utilize all token-level embeddings.

Problem Formulation: For unsupervised corpus poisoning attacks, we base ourselves on a white box attack setting where we know the document encoder $E(\cdot)$ and similarity function $\text{sim}(\cdot)$. We inject adversarially generated documents into the corpus, D, that satisfy two necessary conditions:

- **Embedding Similarity Condition**: The adversarial document $\tilde{d}$ should be as similar as possible to the target document d_i in the embedding space, ensuring a high ranking.
- **Semantic Irrelevance Condition**: The adversarial document $\tilde{d}$ should be irrelevant to the target document d_i from a human perspective.

A successful attack occurs only when the user actively reads the top-ranked adversarial document $\tilde{d}$ and perceives it as uninformative.

3.2 Reconstruction Model

Adversarial attacks in Natural Language Processing are challenging due to the discrete nature of words, where optimizing a single token with word substitution by its gradients is not consistent with optimizing the whole document. In contrast, the continuous nature of the embedding space allows for gradient-based optimization on the whole document. To bridge this gap, we thus develop a reconstruction model enabling us to perturb embeddings directly – where gradients of documents can be computed – rather than manipulating discrete tokens. More specifically, we train a reconstruction model that is able to recover original tokens from retrieval contextual token-level embeddings, such that $d \simeq R\left(E\left(d\right)\right)$, with $e_d = E(d) \in \mathbb{R}^{|d| \times \hbar}$. Figure 2 shows the details of training, where forward propagation formula is as follows:

$$P\left(d' \mid e_d; R\right) = R\left(e_d\right) = R\left(E\left(d\right)\right) \qquad (1)$$

where $P\left(d' \mid e_d; R\right) \in \mathbb{R}^{|d| \times |V|}$ represents the token-level predicted probabilities within the token space V, abbreviated as $P\left(d'\right)$. We can get each predicted token-ids of reconstructed document d' through an $\arg\max(\cdot)$ function: $d' = \arg\max_{p\left(t_i'\right) \forall t_i' \in V} P\left(d'\right)$.

In our design, the reconstruction model $R(\cdot)$ consists of a multi-layer transformer encoder structure combined with a multi-class classification layer as the last layer. The number of classes in the last layer should correspond to the number of tokens in $E(\cdot)$, which is 30,522 for BERT-based retrieval models. During the training, we

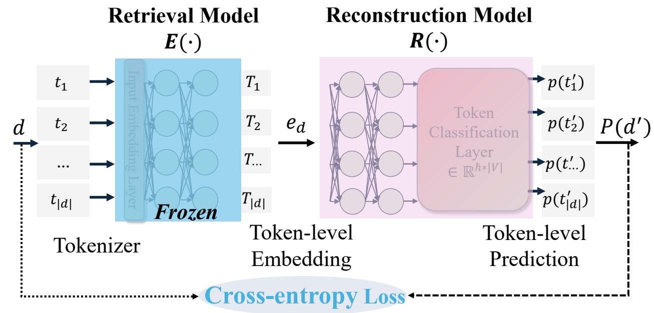


Figure 2: The training pipeline for the reconstruction model.

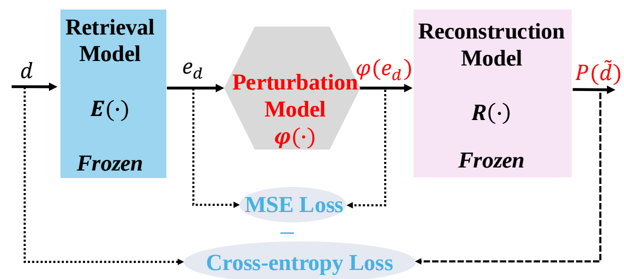


Figure 3: The pipeline of generating new adversarial documents. The perturbation model is trained using a combined loss to transform the embeddings of target documents into adversarial ones. Then, the trained reconstruction model recovers adversarial embeddings to adversarial documents.

freeze the retrieval model $E(\cdot)$ and train $R(\cdot)$ by **minimizing** a cross-entropy loss as reconstruction model (**RM**) loss:

$$L_{RM} = -\sum^{|D|} d \cdot \log\left(P\left(d' \mid e_d; R\right)\right) \qquad (2)$$

After training, we can encode a document at the token level using the retriever and then use the reconstruction model to recover the text from its contextual embeddings, which means $d' \simeq d$.

3.3 Adversarial Generation Optimization

As we mentioned above in the attack problem formulation part, a qualified adversarial document $\tilde{d}$ needs to satisfy both the embedding similarity condition and the semantic irrelevance condition. For the embedding similarity condition, given a target document, we can optimize the Euclidean distance of embeddings between the adversarial document $\tilde{d}$ and the target document d. And the semantic irrelevance condition can be achieved by optimizing the number of common tokens between the adversarial document and the target documents. The fewer common tokens the two documents share, the more different their semantic will be.

To achieve these two conditions simultaneously, we use a perturbation mode $\varphi(\cdot)$ and design an adversarial generation optimization process with two loss functions, which is shown in Figure 3. In this process, the input is a target document d, and the output is its adversarial document $\tilde{d}$.

To optimize the perturbation mode $\varphi(\cdot)$ for the embedding similarity condition, we **minimize** the mean square error (**MSE**) loss between two embeddings:

$$L_{MSE} = \mathbb{E}\left[(\varphi\left(e_d\right) - e_d)^2\right] \tag{3}$$

To implement the semantic irrelevance condition and maximize the token-level dissimilarity between target document d and its adversarial document $\tilde{d}$, we **maximize** the cross-entropy (**CE**) loss at token level as follows:

$$L_{CE} = d \cdot \log\left(P\left(\tilde{d} \mid e_d; \varphi\right)\right) \tag{4}$$

where $P\left(\tilde{d} \mid e_d; \varphi\right) \in \mathbb{R}^{|d| \times |V|}$ is predicted probabilities of each token in the adversarial document $\tilde{d}$, abbreviated as $P\left(\tilde{d}\right)$. The perturbation model $\varphi(\cdot)$ is optimized to **minimize** the following loss function:

$$L_{Attack} = L_{MSE} - L_{CE} \tag{5}$$

due to the differing scales of these two losses ($L_{CE} \in [0, \infty)$), we use a hyperparameter $\lambda = 5$ by default to clip the cross-entropy loss in our implementation, which allows more effective optimization, as shown in the following:

$$L_{Attack} = L_{MSE} - \min\left(L_{CE}, \lambda\right) \tag{6}$$

where λ controls the degree of semantic similarity at the token level. A larger λ value indicates a smaller token overlap between d and its adversarial document $\tilde{d}$.

In this paper, we use a three-layer perceptron to implement the perturbation model. Notably, although the perturbation model described above is document-specific, meaning a unique attack model is initialized for generating each adversarial document, its lightweight design with few parameters allows for fast and efficient training. Throughout the optimization process using the L_{Attack} loss, the model updates prediction probabilities $P\left(\tilde{d}\right)$ continuously and we can select tokens of the output adversarial document by $\tilde{d} = \arg\max_{p(\tilde{t}_i)\forall\tilde{t}_i \in V} P\left(\tilde{d}\right)$.

4 Experiments

In this section, we demonstrate the effectiveness and efficiency of the proposed method through extensive experiments conducted on two attack tasks and multiple datasets.

4.1 Experimental Setup

In this subsection, we outline the experimental setup, covering datasets, retrievers, evaluation metrics, and implementation details.

4.1.1 Datasets. Since all retrievers in this paper are fine-tuned on the MS MARCO-Passage-Ranking dataset (**MS MARCO**) [3], we use its entire corpus to train the reconstruction model. And then the reconstruction models are tested on the corpus of Natural Questions(**NQ**) [16], another widely used dataset.

For adversarial attacks, we select **TREC DL 19** [9] and **TREC DL 20** [8] as in-domain target datasets as they share the same corpus with MS MARCO. Additionally, we select **NQ**, **Quora** [13], **FiQA** [27], and **Touché-2020** [4] from the **BEIR** [39] benchmark as out-of-domain datasets due to the diverse performance of retrieval models across these datasets. The statistics of these datasets are shown in Table 1.

4.1.2 Retrieval Models. We select **SimLM**[1] [41] as the primary target attack retriever as it represents one of the SOTA bi-encoder retrievers. We also select **Contriever** [14], **E5-base-v2** [40], **TAS-B** [12], **DRAGON+** [20], and **RetroMAE** [44] as alternative attack targets and train their reconstruction models, respectively. All the retrievers aforementioned are fine-tuned on the MSMARCO dataset, distinguishing them from their respective pre-trained models.

For black-box attacks, we select four retrieval models with different structures as targets: **DPR** [15] (bi-encoder), **SimLM re-ranker** (cross-encoder), **ColBERTv2** [33] (late interaction), and **RankL-LaMA** [26] (generative model). All retrieval models used in this paper are publicly available and frozen without additional training.

4.1.3 Baseline Methods. In this paper, we use three adversarial attack methods as baselines:

Random Noise: Following [28, 47], we add Gaussian noise on token-level embedding to replace our attack model.

$$\varphi\left(e_{d_i}\right) = e_{d_i} + \beta \cdot \epsilon,\ \epsilon \sim \mathcal{N}(0, 1) \tag{7}$$

where β is a hyperparameter controlling the injected noise amount. We select $\beta = 0.5$ by default following the search method in [28].

Random Token: We randomly replace tokens in the target document with arbitrary random tokens at a ratio of $p = 0.3$, where the ratio is searched in a similar way to Random Noise.

HotFlip-based: It is one of the most widely used gradient-based word substitution methods and serves as a foundational technique for many other approaches in generating adversarial documents [22, 24, 37, 43, 46]. In this paper, we refer to Zhong et al. [46] and use the codebase from Li et al. [19], as it is an accelerated version.

4.1.4 Evaluate Metrics. We use Normalized Discounted Cumulative Gain (specifically, **nDCG@10**) for retrieval performance.

To evaluate the reconstructed model, we use four widely used metrics: **Accuracy**, **Precision**, **Recall**, and **F1** for this token-level multi-class classification task. It is worth noting that we report the macro-averaged scores for Precision, Recall, and F1.

To evaluate the attack performance, we assess from two perspectives: Embedding Similarity and Semantic Irrelevance, which align with the two conditions for attack success.

For Embedding Similarity, our evaluation uses two metrics: Attack Success Rate (**ASR**) and **Top@k**, which are defined as follows:

- **ASR**: The attack success rate in this paper is proposed as the ratio of rankings for which the rank of relevant documents is affected by adversarial attacks, as illustrated in Figure 4. Its practical significance lies in representing the proportion of rankings in which the user encounters adversarial documents before obtaining all relevant documents.
- **Top@k**: It is one of the most commonly used in the literature [1, 19, 21, 23, 46], which shows the ratio of queries that have at least one adversarial document in its top-k retrieval result. We select Top@10 and Top@50 in this paper.

[1] https://huggingface.co/intfloat/simlm-base-msmarco-finetuned

Table 1: Statistics of datasets used in our work [39]. Avg. D/Q indicates the average number of relevant documents per query.

Datasets	Task	Domain	Title	Relevancy	#Corpus	Train	Test or Dev		Avg. Word Lengths	
						#Pairs	#Query	Avg.D/Q	Query	Document
MS MARCO	Passage-Retrieval	Misc.	×	Binary	8,841,823	532,761	6,980	1.1	5.96	55.98
TREC DL 19	Passage-Retrieval	Misc.	×	Binary	8,841,823	532,761	43	95.4	5.96	55.98
TREC DL 20	Passage-Retrieval	Misc.	×	Binary	8,841,823	532,761	54	66.8	5.96	55.98
NQ	Question Answering	Wikipedia	✓	Binary	2,681,468	132,803	3,452	1.2	9.16	78.88
Quora	Retrieval	Quora	×	Binary	522,931	—	10,000	1.6	9.53	11.44
FiQA	Question Answering	Finance	×	Binary	57,638	14,166	648	2.6	10.77	132.32
Touché-2020	Retrieval	Misc.	✓	3-level	382,545	—	49	19.0	6.55	292.37

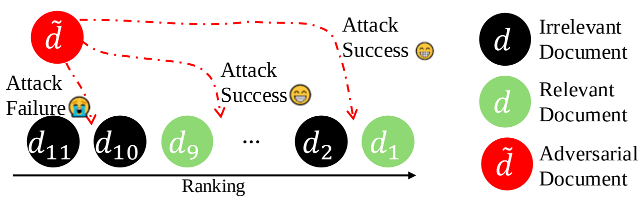


Figure 4: Definition of Attack Success Rate (ASR). For an arbitrary ranking, if there is an adversarial document $\tilde{d}$ that exceeds the relevant document with the lowest score d_9, this ranking is considered as attacked successfully.

For Semantic Irrelevance, we use: the BiLingual Evaluation Understudy (**BLEU**[2]) [30] score and questions (**Q1** and **Q2**), which are defined as follows:

- **BLEU**: it is a widely used word-level metric for evaluating the correspondence between a machine's output and that of a human.
- **Questions**: we design two question prompts (**Q1** and **Q2**) and then ask Large Language Model (LLM) and human expert to answer them. The details of these two questions are shown in Table 2. We report the ratio of "**NO**" response, the higher the better, as our adversarial documents would meet the semantic irrelevance condition. We select the latest GPT-4o-mini[3] [29] for LLM evalution.

In addition, we also use the **Perplexity**[4] metric, calculated by the LLaMA-3.2 1B model [10], to assess the fluency of the adversarial documents. A lower perplexity value indicates higher fluency in the adversarial document.

4.1.5 Implementation Details. To train the reconstruction model, we use a learning rate of 1e-5, train for 5 epochs, and set the batch size to 128×4 GPUs, with a maximum text length of 128. For all attack experiments, we set $\lambda = 5$, the learning rate is 5e-4. We run both our method and the HotFlip-based methods for 3000 epochs. The HotFlip-based adversarial document is initialized as a `[MASK]` token list of the same length as the target document. Our experiments are mainly implemented using Pytorch 2.1 on a Ubuntu server with Nvidia L4 ×24G × 8 GPUs, AMD EPYC 9554P CPU, and 384G memory. Our code is available at **https://github.com/liyongkang123/unsupervised_corpus_poisoning**.

[2]https://huggingface.co/spaces/evaluate-metric/bleu
[3]https://platform.openai.com/docs/models/gpt-4o-mini
[4]https://huggingface.co/spaces/evaluate-metric/perplexity

Table 2: Two simplified question prompts used for LLM and Human semantic evaluation. We report the ratio of "NO" in their responses.

Question	Prompt
Q1	You are an expert in relevance assessment. I will provide you with a query posed by a user, followed by a document. Your task is to determine **whether the document answers the user's question**. Please respond directly and solely with "Yes" or "No".
Q2	You are an expert in relevance assessment. I will provide you with two documents, and you need to assess **whether these two documents express the same information**. Please respond directly and only with "Yes" or "No".

Table 3: Performance of reconstruction models tested on the NQ corpus. We report the macro-averaged score for the Precision, Recall, and F1. All retrievers come from fine-tuned versions on the MSMARCO rather than raw pre-trained models.

Retrievers	Accuracy ↑	Precision ↑	Recall ↑	F1 ↑
SimLM	0.9914	0.9449	0.9501	0.9463
Contriever	0.9880	0.9425	0.9458	0.9430
E5-base-v2	0.9778	0.9245	0.9098	0.9135
TAS-B	0.9915	0.9443	0.9522	0.9471
DRAGON+	0.9899	0.9416	0.9414	0.9399
RetroMAE	0.9911	0.9373	0.9426	0.9386

4.2 Reconstruction Model Performance

For each retriever $E(\cdot)$, we fine-tune a corresponding reconstruction model $R(\cdot)$ on the MS MARCO corpus, initializing it from an uncased BERT large model. Table 3 shows the reconstruction capability of our method, tested on the NQ corpus (out-of-domain).

It can be observed that reconstruction models effectively learn the relationship between contextual token-level embeddings and texts, with an Accuracy of around 0.99. While the overall F1 score appears relatively low, it still averages above 0.93. This is due to the differences between the NQ corpus and MS MARCO, which pose challenges for the model in accurately predicting tokens with

Table 4: The Top-1 white-box attack performance for attacking SimLM on six datasets. We report not only on two attack success conditions—embedding similarity and semantic irrelevance—but also on the perplexity and time cost for generating each adversarial document (measured in seconds). All results are averaged over three runs with different random seeds.

Datasets	Attack Methods	Embedding Similarly			Sematic Irrelevance			Perplexity↓	Time Cost (s)↓
		ASR↑	Top@10↑	Top@50↑	BLEU↓	LLM Q1↑	LLM Q2↑		
TREC DL 19 nDCG@10=0.650	Random Noise	0.735	0.023	0.046	0.037	**1.000**	**0.984**	3469.2	0.1
	Random Token	0.736	0.147	0.357	0.349	0.930	0.566	1638.2	**0.1**
	HotFlip-based	0.899	**0.682**	0.767	**0.020**	0.992	0.861	6032.6	512.1
	Ours	**0.984**	0.434	**0.791**	0.093	0.954	0.752	**188.9**	119.9
TREC DL 20 nDCG@10=0.639	Random Noise	0.679	0.012	0.062	0.027	**1.000**	**0.982**	3410.3	0.1
	Random Token	0.710	0.185	0.383	0.355	0.907	0.642	1793.4	**0.1**
	HotFlip-based	0.778	**0.617**	0.673	**0.019**	0.961	0.747	6390.9	449.8
	Ours	**0.957**	0.500	**0.753**	0.079	0.944	0.753	**166.0**	115.6
NQ nDCG@10=0.426	Random Noise	0.063	0.030	0.100	0.052	**1.000**	**0.993**	3001.5	**0.1**
	Random Token	0.177	0.217	0.4	0.348	0.940	0.597	2889.4	0.1
	HotFlip-based	**0.557**	**0.773**	**0.817**	**0.025**	0.983	0.830	8021.7	588.3
	Ours	0.417	0.490	0.700	0.099	0.940	0.793	**231.2**	127.5
Quora nDCG@10=0.865	Random Noise	0.003	0.027	0.047	**0.040**	**1.000**	**1.000**	4104.9	**0.1**
	Random Token	0.057	0.187	0.287	0.349	0.967	0.883	10265.8	0.1
	HotFlip-based	**0.140**	**0.527**	**0.593**	0.074	0.970	0.730	59468.5	136.5
	Ours	0.043	0.320	0.513	0.051	0.993	0.923	**461.7**	96.5
FiQA nDCG@10=0.224	Random Noise	0.483	0.047	0.173	0.074	**1.000**	**0.980**	2532.8	**0.1**
	Random Token	0.387	0.103	0.220	0.335	0.997	0.830	1664.4	0.1
	HotFlip-based	0.660	**0.580**	0.640	**0.013**	1.000	0.960	6072.2	770.3
	Ours	**0.760**	0.370	**0.643**	0.102	0.983	0.833	**159.0**	138.7
Touché-2020 nDCG@10=0.162	Random Noise	0.408	0.000	0.020	**0.038**	**1.000**	**0.986**	4495.5	0.1
	Random Token	0.639	0.101	0.245	0.353	0.939	0.748	7366.4	**0.1**
	HotFlip-based	0.666	**0.449**	0.544	0.051	1.000	0.735	14037.5	231.0
	Ours	**0.986**	0.422	**0.633**	0.086	0.959	0.803	**522.2**	102.4

extremely low frequency. Consequently, this discrepancy results in a lower macro-averaged score.

4.3 Top-1 Attack

In this section, we introduce the Top-1 attack, which simulates real-world scenarios by attacking a ranking. We select SimLM, as the target model to attack. We randomly sampled up to 100 test queries from each of the 6 datasets, using SimLM to retrieve documents. The top-ranked document d was input into our perturbation method and three baseline methods, which generated the adversarial document $\tilde{d}$ and inserted into the corpus. This adversarial document $\tilde{d}$ is expected to be retrieved at a very high rank in that ranking. It must be noted that the top-1 attack here targets a ranking, which in real-world scenarios does not necessarily originate directly from a specific query or multiple queries.

4.3.1 White-box Attack Performance. We first conduct experiments in a white-box setting, where the target retriever is known—in our case, SimLM. The results of the attack are presented in Table 4, where we have the following observations:

• Under the three metrics of Embedding Similarity—ASR, Top@10, and Top@50—Ours and the HotFlip-based method perform comparably across the six datasets. The Random Noise and Random Token methods exhibit the poorest performance due to their inherent randomness and lack of control.

• Under the Semantic Irrelevance metric, Random Noise performs the best because adding small-scale random noise makes the embedding represent a document entirely unrelated to the target document, maximizing semantic irrelevance. We can also observe that the Random Token method performs poorly because, after replacing 30% of the tokens, 70% remain unchanged, resulting in a relatively high BLEU score (around 0.35). Moreover, we observe that HotFlip achieves lower LLM Q2 scores when attacking Quora. Upon closer inspection, this is due to HotFlip's tendency to select words consistent with the original document during gradient computation when the target document is short. Overall, Ours and the HotFlip-based method still perform comparably for semantic irrelevance.

• For the perplexity metric, our method achieves significantly lower scores than the other three methods, indicating that our adversarial documents are more difficult to detect using perplexity-based filters. Notably, even with only 30% of tokens altered, the Random Token method produces remarkably high perplexity scores on the LLama-3.2 1B model. The low perplexity of our method is likely attributed to the reconstruction model, where each token is predicted based on its contextual embedding derived from the entire adversarial document during reconstruction.

• For the time cost, Random Noise, and Random Token require negligible processing time, while HotFlip is four times slower than our method on average. Notably, we use the fastest accelerated HotFlip implementation [19] here, whereas Zhong et al. [46] require over 2 hours per document [47].

Considering all the results in Table 4, our method matches the SOTA HotFlip-based model in attack effectiveness, while producing adversarial documents with significantly lower perplexity and four times higher efficiency.

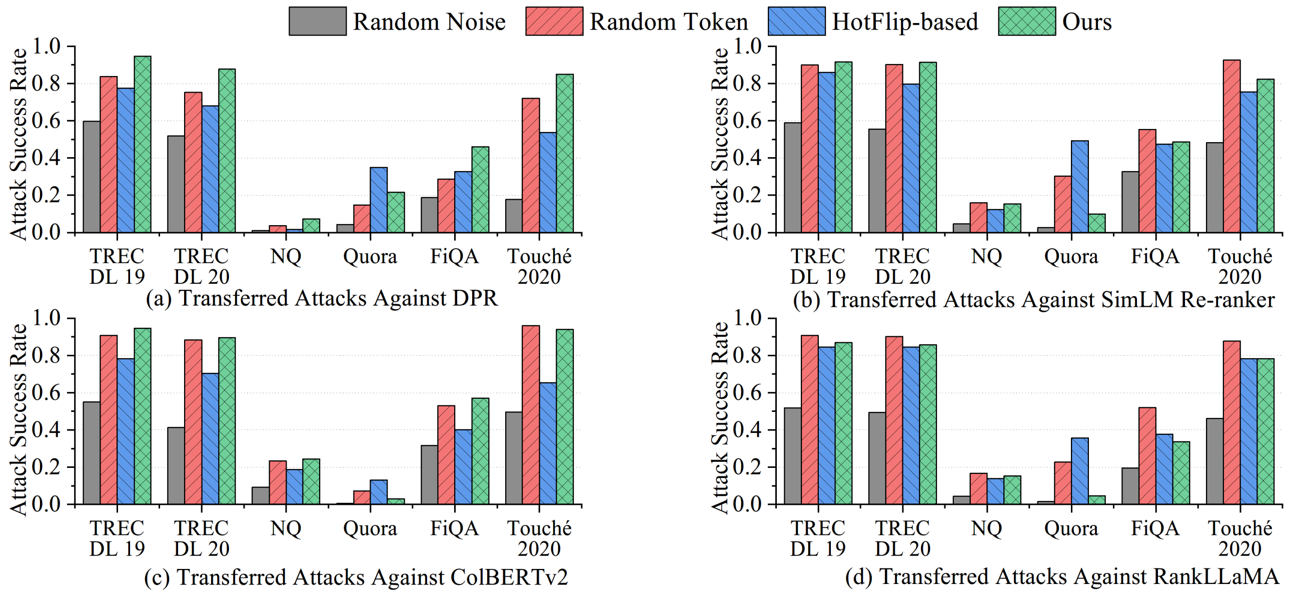


Figure 5: The Top-1 black-box attack results by transferring adversarial documents from SimLM to other retrieval models.

4.3.2 Black-box Attack Performance. To evaluate the effectiveness of our method in a transfer-based black-box attack setting, where the target retrieval model is unknown, we use the same sampled queries and their generated adversarial documents from the previous white-box attack section. Then we test the attack success rate based on the retrieval ranking from four retrieval models: DPR, SimLM re-ranker, ColBERTv2, and RankLLaMA.

It is worth noting that, in the literature [22–24, 43], black-box attacks typically involve distilling a surrogate model first, then using white-box methods to attack that surrogate model. However, distilling LLMs like LLaMA is too computationally intensive in our experiments. Therefore, to ensure fairness across all target models and to reduce the computational workload, we refrain from training surrogate models. Instead, we follow the approach in [19] and directly use the adversarial documents generated by various attack methods during attacks on SimLM and apply them to the target models.

The results, averaged over three runs, are shown in Figure 5. We can observe that, when transferred to four black-box models, the attack success rate of our method surpasses that of the HotFlip-based method on most datasets, indicating better transferability. We speculate that this is because our adversarial documents have lower perplexity (and potentially higher fluency), comparable to normal texts, making them more effective at deceiving other retrieval models.

We also find that the Random Token method has the highest attack success rate. However, as shown in Table 4, the BLEU scores of the adversarial documents generated by Random Token are very high, which does not satisfy the condition of semantic irrelevance. Therefore, considering both the ASR and BLEU metrics, the Random Token attack is not an effective method.

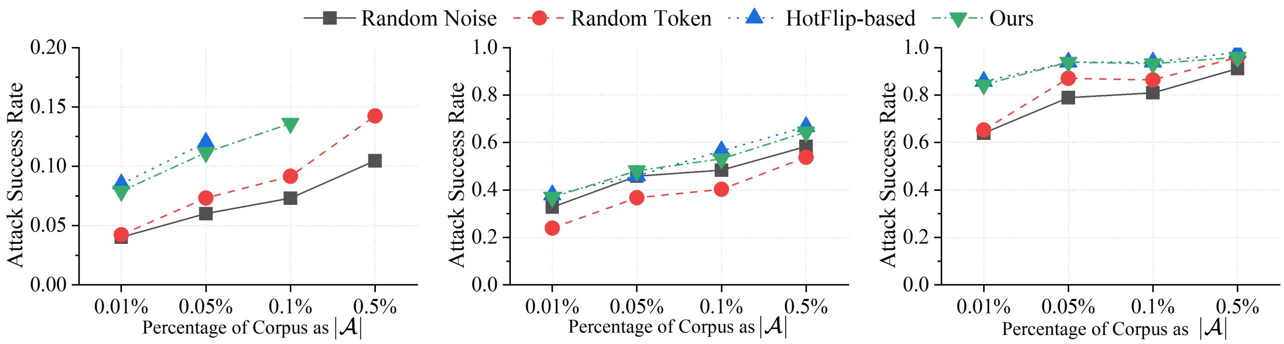
Another interesting finding is that, when we compare the four target models, we observe that the success rate of all attack methods transferred to the DPR model is almost always lower than that of SimLM re-ranker and ColBERTv2. This indicates that the robustness of the DPR model is higher than that of SimLM re-ranker and ColBERTv2. Similarly, RankLLaMA also demonstrates high robustness, as its success rate after being attacked is relatively low.

Table 5: Human and LLM evaluation on semantic irrelevance.

Attack Methods	Question 1		Question 2	
	LLM	Human	LLM	Human
Random Noise	1.000	1.000	0.967	0.989
Random Token	1.000	0.900	0.767	0.922
HotFlip-based	1.000	0.867	0.800	0.853
Ours	1.000	0.900	0.830	0.930

4.3.3 Human evaluation Vs. LLM evaluation. In this paper, we primarily use LLMs to evaluate semantic irrelevance from a human perspective, as LLMs are not only much cheaper and easier to test on a large scale but are also highly effective at making judgments [7, 32]. However, human evaluation is still necessary, as we are unclear whether humans and LLMs align consistently on this specific issue. Therefore, from all the documents generated by white-box attacks in Table 4, we randomly select 30 adversarial documents from each attack method and invite three experts in the information retrieval field to evaluate Question 1 and Question 2. At the same time, we also use LLMs to evaluate these documents.

The experimental results are shown in Table 5. We can observe that for Question 1, LLMs exhibit stricter judgment, consistently answering “No” when uncertain, whereas humans are more lenient. For Question 2, LLMs are relatively more permissive, particularly when keywords are highly repetitive, often interpreting them as semantically similar (resulting in the lowest score for Random Token). Overall, Random Token and HotFlip perform poorly on both questions, while our method effectively preserves semantic irrelevance.



(a) Corpus Poisoning on NQ (b) Corpus Poisoning on FiQA (c) Corpus Poisoning on Touché-2020

Figure 6: Corpus poisoning attack results on three datasets. Some data points are not included in NQ due to computational complexity. The number of injected adversarial documents $|\mathcal{A}|$ is determined by multiple percentages of corpus size.

4.4 Corpus Poisoning Attack

The Top-1 attack validated the effectiveness of our method when attacking an arbitrary ranking. However, there are situations where no target ranking exists. For these cases, we refer to it as a corpus poisoning attack task. Additionally, unlike the definition of corpus poisoning in Zhong et al. [46], we follow the approach of Li et al. [19], which requires that attacks be conducted without prior knowledge of the queries. This is because, in real-world scenarios, an attacker can obtain a sample of the corpus distribution through manual inputs and by observing the ranking results, while acquiring real queries presents greater challenges.

In this experiment, we attack SimLM on NQ, FiQA, and Touché-2020 due to their varying corpus sizes and apply attacks using all four attack methods. We do k-means clustering on the corpus, clustering them into $|\mathcal{A}|$ categories. All documents in each cluster can be considered a ranking, sorted by their distance to the cluster centroid. We attack the document closest to the centroid in each cluster (top-1 document of each ranking), and obtain adversarial documents $\mathcal{A} = \left\{\tilde{d}_1, \ldots, \tilde{d}_{|\mathcal{A}|}\right\}$. We use all test queries of these two datasets to evaluate the performances. Due to the varying sizes of different corpus, we used the proportion of the corpus as the number of clustering clusters $|\mathcal{A}|$ to ensure fairness.

The corpus poisoning attack results are shown in Figure 6, where we have the following findings:

• The attack success rate generally increases with the number of clusters. Moreover, the performance of Ours and HotFlip-based methods is comparable, both outperforming the Random Noise and Random Token methods. This trend is consistent with the white-box Top-1 attack results in Table 4.

• All methods show a higher attack success rate on Touché-2020 compared to their performance on FiQA and NQ. We speculate that this may be because Touché-2020 has a higher average number of relevant documents per query, making it more susceptible to attacks. Specifically, in Touché-2020, even with only 0.01% of adversarial documents, a high attack success rate can be achieved.

• On the NQ dataset, our method significantly outperforms random noise, which might indicate that as the corpus size increases, the random noise approach becomes less effective.

In summary, our method and the HotFlip-based method are comparable in terms of attack effectiveness. However, considering attack efficiency and perplexity, the method proposed in this paper still holds an advantage.

5 Additional Discussion

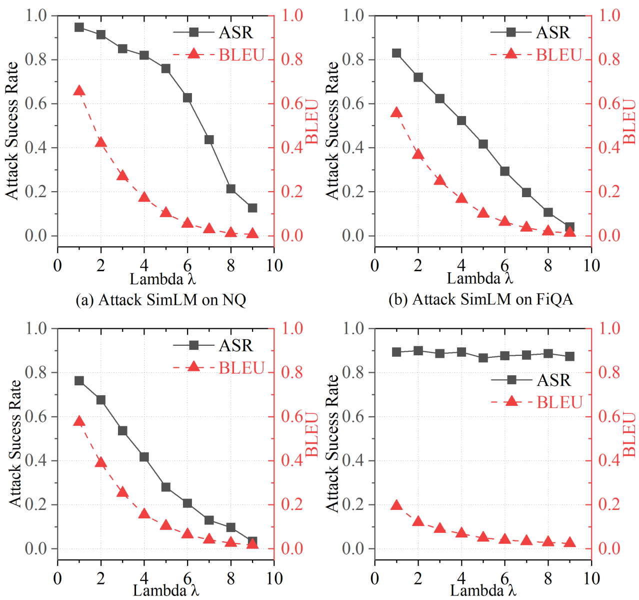
In this section, we conduct additional analysis to provide a comprehensive evaluation of the performance of our approach.

5.1 Hyper-parameter Study

In Equation 6, the loss function L_{Attack} has two components: minimizing L_{MSE} ensures that the output embedding has the minimum Euclidean distance to the input, while maximizing L_{CE} encourages that the output embedding results in different tokens after reconstruction. These two losses compete during training, making optimization challenging without effective regulation. We tested different weights for the losses and multi-task learning methods (e.g., MGDA [34]), but the results were unsatisfactory. Ultimately, we found that truncating L_{CE} stabilized the model's output.

To demonstrate the effect of lambda λ on different datasets and retrievers, we selected the NQ and FiQA datasets and conducted experiments using lambda λ values from 1 to 9 on SimLM, TAS-B, and Contriever. The experimental results are shown in Figure 7.

By comparing all the subfigures in Figure 7, we can observe that the BLEU score decreases monotonically with the increase of lambda, indicating that we can adjust the degree of semantic dissimilarity by controlling lambda. By comparing Figure 7 (a) and (b), we can observe that for attacking the same retriever on different datasets, $\lambda = 5$ is a suitable value to achieve both a high attack success rate and a relatively low BLEU score. By comparing Figure 7(a), (c), and (d), we observe that $\lambda = 5$ exhibits strong robustness across attacks on various retrievers, highlighting its high generalizability. Furthermore, in Figure 7(d), we observe that the attack on Contriever always achieves a very high success rate, indicating that the Contriever model is highly vulnerable and susceptible to attacks. This conclusion aligns with the experimental findings in [19, 46].



(a) Attack SimLM on NQ (b) Attack SimLM on FiQA (c) Attack TAS-B on NQ (d) Attack Contriever on NQ

Figure 7: Hyper-parameter study about the trade-off of λ.

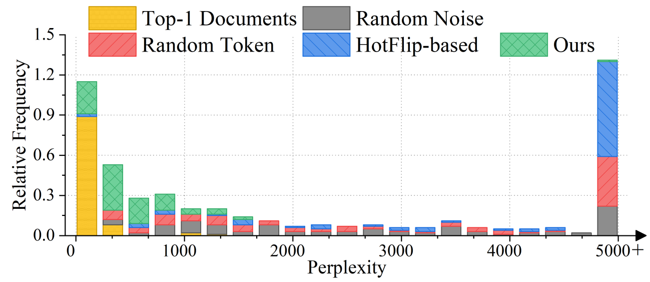


Figure 8: Stacked histogram showing the perplexity distribution of adversarial documents across all methods, with a maximum perplexity capped at 5000 for visualization clarity.

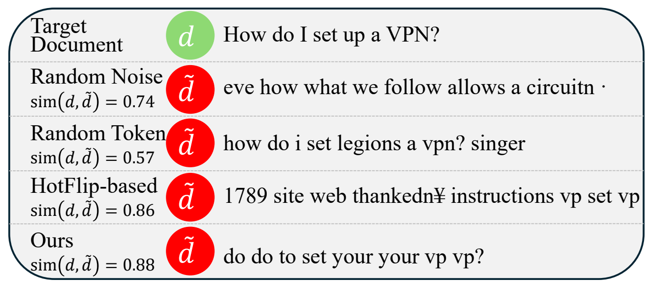


Figure 9: Case study from the Quora dataset (retrieval of similar questions). Target document ID: 182134.

5.2 Adversarial Training

Adversarial training is a vital way to improve the robustness of retrieval models [25, 31, 35, 45]. However, generating a large number of high-quality adversarial samples or hard negative samples has always been a challenging task. Given the efficiency of our method, we can create a substantial amount of adversarial samples offline for training, thereby enhancing the robustness of the model. We offer a preliminary experiment: We select 7000 positive query-document pairs from the MS MARCO training set and generate one adversarial document for each positive document with our method. We then use the adversarially generated documents as negative samples, and fine-tune SimLM using a standard DPR training setting with a contrastive loss and in-batch negative samples [15]. We observe that the attack success rate of the Top-1 attack decreases across all datasets after adversarial training, with an average relative reduction of 7.9%. This finding suggests that the model's resilience against our attacks has been enhanced, leading to improved robustness. Additionally, the fine-tuned SimLM model achieves a minor 0.002 increase in the average nDCG@10 across six datasets.

5.3 Perplexity Detection & Case Study

Figure 8 illustrates the perplexity of the top-1 documents and adversarial documents generated by four different methods during the attack on Quora, as described in Section 4.3. It can be observed that perplexity-based filtering [36] struggles to differentiate our method from the top-1 documents, whereas it easily distinguishes other methods, such as HotFlip.

Figure 9 presents a case study of attacking the Quora dataset. It can be observed that although the documents generated by HotFlip-based methods exhibit high similarity, they are extremely disorganized, resulting in a high perplexity. In contrast, the random token method significantly reduces similarity after replacing a few tokens. However, our method achieves the best overall performance.

6 Conclusions

In this paper, we propose an unsupervised corpus poisoning task under a realistic attack setting. Adversarial documents are generated using our reconstruction and perturbation models, trained with the dual objective of maximizing the token-level dissimilarity while maintaining high embedding similarity. Our attack is fast, transferable, and shows that SOTA dense retrieval models are vulnerable. Our experiments include two scenarios: one where the top-1 document in a ranking is targeted, and another that targets a small percentage of a corpus, based on clustering. Furthermore, our attack is hard to detect with perplexity metrics, as the adversarial examples generated, although nonsensical, follow the distribution of natural text more so than previous methods. By leveraging the efficiency of our method, we enable adversarial training by generating adversarial documents on a larger scale, with preliminary results showing reduced attack success without harming retrieval performance.

Acknowledgments

This work was partially supported by the China Scholarship Council (202308440220), the LESSEN project (NWA.1389.20.183) of the research program NWA ORC 2020/21 which is financed by the Dutch Research Council (NWO), and the PACINO project (215742) which is financed by the Swiss National Science Foundation (SNSF).

References

[1] Anonymous. 2024. GASLITEing the Retrieval: Poisoning Knowledge DBs to Mislead Embedding-based Search. In *Submitted to The Thirteenth International Conference on Learning Representations.* https://openreview.net/forum?id=LBd87fWerd under review.

[2] Leif Azzopardi, Charles L. A. Clarke, Paul B. Kantor, Bhaskar Mitra, Johanne R. Trippas, Zhaochun Ren, Mohammad Aliannejadi, Negar Arabzadeh, Raman Chandrasekar, Maarten de Rijke, Panagiotis Eustratiadis, William R. Hersh, Jin Huang, Evangelos Kanoulas, Jasmin Kareem, Yongkang Li, Simon Lupart, Kidist Amde Mekonnen, Adam Roegiest, Ian Soboroff, Fabrizio Silvestri, Suzan Verberne, David Vos, Eugene Yang, and Yuyue Zhao. 2024. Report on the Search Futures Workshop at ECIR 2024. *SIGIR Forum* 58, 1 (2024), 1–41. https://doi.org/10.1145/3687273.3687288

[3] Payal Bajaj, Daniel Campos, Nick Craswell, Li Deng, Jianfeng Gao, Xiaodong Liu, Rangan Majumder, Andrew McNamara, Bhaskar Mitra, Tri Nguyen, et al. 2016. Ms marco: A human generated machine reading comprehension dataset. *arXiv preprint arXiv:1611.09268* (2016).

[4] Alexander Bondarenko, Maik Fröbe, Meriem Beloucif, Lukas Gienapp, Yamen Ajjour, Alexander Panchenko, Chris Biemann, Benno Stein, Henning Wachsmuth, Martin Potthast, and Matthias Hagen. 2020. Overview of Touché 2020: Argument Retrieval. In *Working Notes of CLEF 2020 - Conference and Labs of the Evaluation Forum, Thessaloniki, Greece, September 22-25, 2020 (CEUR Workshop Proceedings, Vol. 2696)*, Linda Cappellato, Carsten Eickhoff, Nicola Ferro, and Aurélie Névéol (Eds.). CEUR-WS.org. https://ceur-ws.org/Vol-2696/paper_261.pdf

[5] Xuanang Chen, Ben He, Zheng Ye, Le Sun, and Yingfei Sun. 2023. Towards Imperceptible Document Manipulations against Neural Ranking Models. In *Findings of the Association for Computational Linguistics: ACL 2023, Toronto, Canada, July 9-14, 2023*, Anna Rogers, Jordan L. Boyd-Graber, and Naoaki Okazaki (Eds.). Association for Computational Linguistics, 6648–6664. https://doi.org/10.18653/V1/2023.FINDINGS-ACL.416

[6] Yangyi Chen, Hongcheng Gao, Ganqu Cui, Fanchao Qi, Longtao Huang, Zhiyuan Liu, and Maosong Sun. 2022. Why Should Adversarial Perturbations be Imperceptible? Rethink the Research Paradigm in Adversarial NLP. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, Yoav Goldberg, Zornitsa Kozareva, and Yue Zhang (Eds.). Association for Computational Linguistics, Abu Dhabi, United Arab Emirates, 11222–11237. https://doi.org/10.18653/v1/2022.emnlp-main.771

[7] David Cheng-Han Chiang and Hung-yi Lee. 2023. Can Large Language Models Be an Alternative to Human Evaluations?. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), ACL 2023, Toronto, Canada, July 9-14, 2023*, Anna Rogers, Jordan L. Boyd-Graber, and Naoaki Okazaki (Eds.). Association for Computational Linguistics, 15607–15631. https://doi.org/10.18653/V1/2023.ACL-LONG.870

[8] Nick Craswell, Bhaskar Mitra, Emine Yilmaz, and Daniel Campos. 2020. Overview of the TREC 2020 Deep Learning Track. In *Proceedings of the Twenty-Ninth Text REtrieval Conference, TREC 2020, Virtual Event [Gaithersburg, Maryland, USA], November 16-20, 2020 (NIST Special Publication, Vol. 1266)*, Ellen M. Voorhees and Angela Ellis (Eds.). National Institute of Standards and Technology (NIST). https://trec.nist.gov/pubs/trec29/papers/OVERVIEW.DL.pdf

[9] Nick Craswell, Bhaskar Mitra, Emine Yilmaz, Daniel Campos, and Ellen M. Voorhees. 2020. Overview of the TREC 2019 deep learning track. *CoRR* abs/2003.07820 (2020). arXiv:2003.07820 https://arxiv.org/abs/2003.07820

[10] Abhimanyu Dubey, Abhinav Jauhri, Abhinav Pandey, Abhishek Kadian, and et al. 2024. The Llama 3 Herd of Models. *CoRR* abs/2407.21783 (2024). https://doi.org/10.48550/ARXIV.2407.21783 arXiv:2407.21783

[11] Javid Ebrahimi, Anyi Rao, Daniel Lowd, and Dejing Dou. 2018. HotFlip: White-Box Adversarial Examples for Text Classification. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics, ACL 2018, Melbourne, Australia, July 15-20, 2018, Volume 2: Short Papers*, Iryna Gurevych and Yusuke Miyao (Eds.). Association for Computational Linguistics, 31–36. https://doi.org/10.18653/V1/P18-2006

[12] Sebastian Hofstätter, Sheng-Chieh Lin, Jheng-Hong Yang, Jimmy Lin, and Allan Hanbury. 2021. Efficiently Teaching an Effective Dense Retriever with Balanced Topic Aware Sampling. In *SIGIR '21: The 44th International ACM SIGIR Conference on Research and Development in Information Retrieval, Virtual Event, Canada, July 11-15, 2021.* ACM, 113–122. https://doi.org/10.1145/3404835.3462891

[13] Shankar Iyer, Nikhil Dandekar, and Kornél Csernai. 2017. First Quora Dataset Release: Question Pairs. https://quoradata.quora.com/First-Quora-Dataset-Release-Question-Pairs

[14] Gautier Izacard, Mathilde Caron, Lucas Hosseini, Sebastian Riedel, Piotr Bojanowski, Armand Joulin, and Edouard Grave. 2022. Unsupervised Dense Information Retrieval with Contrastive Learning. *Trans. Mach. Learn. Res.* 2022 (2022). https://openreview.net/forum?id=jKN1pXi7b0

[15] Vladimir Karpukhin, Barlas Oguz, Sewon Min, Patrick S. H. Lewis, Ledell Wu, Sergey Edunov, Danqi Chen, and Wen-tau Yih. 2020. Dense Passage Retrieval for Open-Domain Question Answering. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing, EMNLP 2020.* Association for Computational Linguistics, 6769–6781. https://doi.org/10.18653/V1/2020.EMNLP-MAIN.550

[16] Tom Kwiatkowski, Jennimaria Palomaki, Olivia Redfield, Michael Collins, Ankur P. Parikh, Chris Alberti, Danielle Epstein, Illia Polosukhin, Jacob Devlin, Kenton Lee, Kristina Toutanova, Llion Jones, Matthew Kelcey, Ming-Wei Chang, Andrew M. Dai, Jakob Uszkoreit, Quoc Le, and Slav Petrov. 2019. Natural Questions: a Benchmark for Question Answering Research. *Trans. Assoc. Comput. Linguistics* 7 (2019), 452–466. https://doi.org/10.1162/TACL_A_00276

[17] Haoran Li, Mingshi Xu, and Yangqiu Song. 2023. Sentence Embedding Leaks More Information than You Expect: Generative Embedding Inversion Attack to Recover the Whole Sentence. In *Findings of the Association for Computational Linguistics: ACL 2023, Toronto, Canada, July 9-14, 2023*, Anna Rogers, Jordan L. Boyd-Graber, and Naoaki Okazaki (Eds.). Association for Computational Linguistics, 14022–14040. https://doi.org/10.18653/V1/2023.FINDINGS-ACL.881

[18] Linyang Li, Ruotian Ma, Qipeng Guo, Xiangyang Xue, and Xipeng Qiu. 2020. BERT-ATTACK: Adversarial Attack Against BERT Using BERT. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, Bonnie Webber, Trevor Cohn, Yulan He, and Yang Liu (Eds.). Association for Computational Linguistics, Online, 6193–6202. https://doi.org/10.18653/v1/2020.emnlp-main.500

[19] Yongkang Li, Panagiotis Eustratiadis, and Evangelos Kanoulas. 2025. Reproducing HotFlip for Corpus Poisoning Attacks in Dense Retrieval. In *Advances in Information Retrieval - 47th European Conference on Information Retrieval, ECIR 2025, Lucca, Italy, April 6-10, 2025, Proceedings, Part IV (Lecture Notes in Computer Science, Vol. 15575).* Springer, 95–111. https://doi.org/10.1007/978-3-031-88717-8_8

[20] Sheng-Chieh Lin, Akari Asai, Minghan Li, Barlas Oguz, Jimmy Lin, Yashar Mehdad, Wen-tau Yih, and Xilun Chen. 2023. How to Train Your Dragon: Diverse Augmentation Towards Generalizable Dense Retrieval. In *Findings of the Association for Computational Linguistics: EMNLP 2023, Singapore, December 6-10, 2023.* Association for Computational Linguistics, 6385–6400. https://doi.org/10.18653/V1/2023.FINDINGS-EMNLP.423

[21] Jiawei Liu, Yangyang Kang, Di Tang, Kaisong Song, Changlong Sun, Xiaofeng Wang, Wei Lu, and Xiaozhong Liu. 2022. Order-Disorder: Imitation Adversarial Attacks for Black-box Neural Ranking Models. In *Proceedings of the 2022 ACM SIGSAC Conference on Computer and Communications Security* (Los Angeles, CA, USA) *(CCS '22).* Association for Computing Machinery, New York, NY, USA, 2025–2039. https://doi.org/10.1145/3548606.3560683

[22] Yu-An Liu, Ruqing Zhang, Jiafeng Guo, Maarten de Rijke, Wei Chen, Yixing Fan, and Xueqi Cheng. 2023. Topic-oriented Adversarial Attacks against Black-box Neural Ranking Models. In *Proceedings of the 46th International ACM SIGIR Conference on Research and Development in Information Retrieval, SIGIR 2023, Taipei, Taiwan, July 23-27, 2023*, Hsin-Hsi Chen, Wei-Jou (Edward) Duh, Hen-Hsen Huang, Makoto P. Kato, Josiane Mothe, and Barbara Poblete (Eds.). ACM, 1700–1709. https://doi.org/10.1145/3539618.3591777

[23] Yu-An Liu, Ruqing Zhang, Jiafeng Guo, Maarten de Rijke, Yixing Fan, and Xueqi Cheng. 2024. Multi-granular Adversarial Attacks against Black-box Neural Ranking Models. In *Proceedings of the 47th International ACM SIGIR Conference on Research and Development in Information Retrieval, SIGIR 2024, Washington DC, USA, July 14-18, 2024*, Grace Hui Yang, Hongning Wang, Sam Han, Claudia Hauff, Guido Zuccon, and Yi Zhang (Eds.). ACM, 1391–1400. https://doi.org/10.1145/3626772.3657704

[24] Yu-An Liu, Ruqing Zhang, Jiafeng Guo, Maarten de Rijke, Wei Chen, Yixing Fan, and Xueqi Cheng. 2023. Black-box Adversarial Attacks against Dense Retrieval Models: A Multi-view Contrastive Learning Method. In *Proceedings of the 32nd ACM International Conference on Information and Knowledge Management* (<conf-loc>, <city>Birmingham</city>, <country>United Kingdom</country>, </conf-loc>) *(CIKM '23).* Association for Computing Machinery, New York, NY, USA, 1647–1656. https://doi.org/10.1145/3583780.3614793

[25] Simon Lupart and Stéphane Clinchant. 2023. A Study on FGSM Adversarial Training for Neural Retrieval. In *Advances in Information Retrieval - 45th European Conference on Information Retrieval, ECIR 2023, Dublin, Ireland, April 2-6, 2023, Proceedings, Part II (Lecture Notes in Computer Science, Vol. 13981)*, Jaap Kamps, Lorraine Goeuriot, Fabio Crestani, Maria Maistro, Hideo Joho, Brian Davis, Cathal Gurrin, Udo Kruschwitz, and Annalina Caputo (Eds.). Springer, 484–492. https://doi.org/10.1007/978-3-031-28238-6_39

[26] Xueguang Ma, Liang Wang, Nan Yang, Furu Wei, and Jimmy Lin. 2024. Fine-Tuning LLaMA for Multi-Stage Text Retrieval. In *Proceedings of the 47th International ACM SIGIR Conference on Research and Development in Information Retrieval, SIGIR 2024, Washington DC, USA, July 14-18, 2024*, Grace Hui Yang, Hongning Wang, Sam Han, Claudia Hauff, Guido Zuccon, and Yi Zhang (Eds.). ACM, 2421–2425. https://doi.org/10.1145/3626772.3657951

[27] Macedo Maia, Siegfried Handschuh, André Freitas, Brian Davis, Ross McDermott, Manel Zarrouk, and Alexandra Balahur. 2018. WWW'18 Open Challenge: Financial Opinion Mining and Question Answering. In *Companion of the The Web Conference 2018 on The Web Conference 2018, WWW 2018, Lyon , France, April 23-27, 2018*, Pierre-Antoine Champin, Fabien Gandon, Mounia Lalmas, and Panagiotis G. Ipeirotis (Eds.). ACM, 1941–1942. https://doi.org/10.1145/3184558.3192301

[28] John X. Morris, Volodymyr Kuleshov, Vitaly Shmatikov, and Alexander M. Rush. 2023. Text Embeddings Reveal (Almost) As Much As Text. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing, EMNLP 2023, Singapore, December 6-10, 2023*, Houda Bouamor, Juan Pino, and Kalika Bali (Eds.). Association for Computational Linguistics, 12448–12460. https://doi.org/10.18653/V1/2023.EMNLP-MAIN.765

[29] OpenAI. 2023. GPT-4 Technical Report. *CoRR* abs/2303.08774 (2023). https://doi.org/10.48550/ARXIV.2303.08774 arXiv:2303.08774

[30] Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. 2002. Bleu: a Method for Automatic Evaluation of Machine Translation. In *Proceedings of the 40th Annual Meeting of the Association for Computational Linguistics, July 6-12, 2002, Philadelphia, PA, USA*. ACL, 311–318. https://doi.org/10.3115/1073083.1073135

[31] Dae Hoon Park and Yi Chang. 2019. Adversarial Sampling and Training for Semi-Supervised Information Retrieval. In *The World Wide Web Conference, WWW 2019, San Francisco, CA, USA, May 13-17, 2019*, Ling Liu, Ryen W. White, Amin Mantrach, Fabrizio Silvestri, Julian J. McAuley, Ricardo Baeza-Yates, and Leila Zia (Eds.). ACM, 1443–1453. https://doi.org/10.1145/3308558.3313416

[32] Hossein A. Rahmani, Emine Yilmaz, Nick Craswell, Bhaskar Mitra, Paul Thomas, Charles L. A. Clarke, Mohammad Aliannejadi, Clemencia Siro, and Guglielmo Faggioli. 2024. LLMJudge: LLMs for Relevance Judgments. In *Proceedings of The First Workshop on Large Language Models for Evaluation in Information Retrieval (LLM4Eval 2024) co-located with 10th International Conference on Online Publishing (SIGIR 2024), Washington D.C., USA, July 18, 2024 (CEUR Workshop Proceedings, Vol. 3752)*, Clemencia Siro, Mohammad Aliannejadi, Hossein A. Rahmani, Nick Craswell, Charles L. A. Clarke, Guglielmo Faggioli, Bhaskar Mitra, Paul Thomas, and Emine Yilmaz (Eds.). CEUR-WS.org, 1–3. https://ceur-ws.org/Vol-3752/paper8.pdf

[33] Keshav Santhanam, Omar Khattab, Jon Saad-Falcon, Christopher Potts, and Matei Zaharia. 2022. ColBERTv2: Effective and Efficient Retrieval via Lightweight Late Interaction. In *Proceedings of the 2022 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, NAACL 2022, Seattle, WA, United States, July 10-15, 2022*, Marine Carpuat, Marie-Catherine de Marneffe, and Iván Vladimir Meza Ruíz (Eds.). Association for Computational Linguistics, 3715–3734. https://doi.org/10.18653/V1/2022.NAACL-MAIN.272

[34] Ozan Sener and Vladlen Koltun. 2018. Multi-Task Learning as Multi-Objective Optimization. In *Advances in Neural Information Processing Systems 31: Annual Conference on Neural Information Processing Systems 2018, NeurIPS 2018, December 3-8, 2018, Montréal, Canada*, Samy Bengio, Hanna M. Wallach, Hugo Larochelle, Kristen Grauman, Nicolò Cesa-Bianchi, and Roman Garnett (Eds.). 525–536. https://proceedings.neurips.cc/paper/2018/hash/432aca3a1e345e339f35a30c8f65edce-Abstract.html

[35] Georgios Sidiropoulos and Evangelos Kanoulas. 2022. Analysing the Robustness of Dual Encoders for Dense Retrieval Against Misspellings. In *Proceedings of the 45th International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR '22)*. Association for Computing Machinery, New York, NY, USA, 2132–2136. https://doi.org/10.1145/3477495.3531818

[36] Congzheng Song, Alexander M. Rush, and Vitaly Shmatikov. 2020. Adversarial Semantic Collisions. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing, EMNLP 2020, Online, November 16-20, 2020*. Association for Computational Linguistics, 4198–4210. https://doi.org/10.18653/V1/2020.EMNLP-MAIN.344

[37] Jinyan Su, John X. Morris, Preslav Nakov, and Claire Cardie. 2024. Corpus Poisoning via Approximate Greedy Gradient Descent. *CoRR* abs/2406.05087 (2024). https://doi.org/10.48550/ARXIV.2406.05087 arXiv:2406.05087

[38] Christian Szegedy, Wojciech Zaremba, Ilya Sutskever, Joan Bruna, Dumitru Erhan, Ian J. Goodfellow, and Rob Fergus. 2014. Intriguing properties of neural networks. In *2nd International Conference on Learning Representations, ICLR 2014, Banff, AB, Canada, April 14-16, 2014, Conference Track Proceedings*, Yoshua Bengio and Yann LeCun (Eds.). http://arxiv.org/abs/1312.6199

[39] 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)*. https://openreview.net/forum?id=wCu6T5xFjeJ

[40] Liang Wang, Nan Yang, Xiaolong Huang, Binxing Jiao, Linjun Yang, Daxin Jiang, Rangan Majumder, and Furu Wei. 2022. Text Embeddings by Weakly-Supervised Contrastive Pre-training. *CoRR* abs/2212.03533 (2022). https://doi.org/10.48550/ARXIV.2212.03533 arXiv:2212.03533

[41] Liang Wang, Nan Yang, Xiaolong Huang, Binxing Jiao, Linjun Yang, Daxin Jiang, Rangan Majumder, and Furu Wei. 2023. SimLM: Pre-training with Representation Bottleneck for Dense Passage Retrieval. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), ACL 2023, Toronto, Canada, July 9-14, 2023*, Anna Rogers, Jordan L. Boyd-Graber, and Naoaki Okazaki (Eds.). Association for Computational Linguistics, 2244–2258. https://doi.org/10.18653/V1/2023.ACL-LONG.125

[42] Chen Wu, Ruqing Zhang, Jiafeng Guo, Wei Chen, Yixing Fan, Maarten de Rijke, and Xueqi Cheng. 2022. Certified Robustness to Word Substitution Ranking Attack for Neural Ranking Models. In *Proceedings of the 31st ACM International Conference on Information & Knowledge Management (CIKM '22)*. ACM. https://doi.org/10.1145/3511808.3557256

[43] Chen Wu, Ruqing Zhang, Jiafeng Guo, Maarten De Rijke, Yixing Fan, and Xueqi Cheng. 2023. PRADA: Practical Black-box Adversarial Attacks against Neural Ranking Models. *ACM Trans. Inf. Syst.* 41, 4, Article 89 (apr 2023), 27 pages. https://doi.org/10.1145/3576923

[44] Shitao Xiao, Zheng Liu, Yingxia Shao, and Zhao Cao. 2022. RetroMAE: Pre-Training Retrieval-oriented Language Models Via Masked Auto-Encoder. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing, EMNLP 2022, Abu Dhabi, United Arab Emirates, December 7-11, 2022*, Yoav Goldberg, Zornitsa Kozareva, and Yue Zhang (Eds.). Association for Computational Linguistics, 538–548. https://doi.org/10.18653/V1/2022.EMNLP-MAIN.35

[45] Hang Zhang, Yeyun Gong, Yelong Shen, Jiancheng Lv, Nan Duan, and Weizhu Chen. 2022. Adversarial Retriever-Ranker for Dense Text Retrieval. In *The Tenth International Conference on Learning Representations, ICLR 2022, Virtual Event, April 25-29, 2022*. OpenReview.net. https://openreview.net/forum?id=MR7XubKUFB

[46] Zexuan Zhong, Ziqing Huang, Alexander Wettig, and Danqi Chen. 2023. Poisoning Retrieval Corpora by Injecting Adversarial Passages. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing, EMNLP 2023, Singapore, December 6-10, 2023*, Houda Bouamor, Juan Pino, and Kalika Bali (Eds.). Association for Computational Linguistics, 13764–13775. https://doi.org/10.18653/V1/2023.EMNLP-MAIN.849

[47] Shengyao Zhuang, Bevan Koopman, Xiaoran Chu, and Guido Zuccon. 2024. Understanding and Mitigating the Threat of Vec2Text to Dense Retrieval Systems. In *Proceedings of the 2024 Annual International ACM SIGIR Conference on Research and Development in Information Retrieval in the Asia Pacific Region, SIGIR-AP 2024, Tokyo, Japan, December 9-12, 2024*, Tetsuya Sakai, Emi Ishita, Hiroaki Ohshima, Faegheh Hasibi, Jiaxin Mao, and Joemon M. Jose (Eds.). ACM, 259–268. https://doi.org/10.1145/3673791.3698414