

Neural Machine Unranking

Jingrui Hou, Axel Finke, and Georgina Cosma

Abstract—We tackle the problem of machine unlearning within neural information retrieval, termed Neural Machine UnRanking (NuMuR) for short. Many of the mainstream task- or model-agnostic approaches for machine unlearning were designed for classification tasks. First, we demonstrate that these methods perform poorly on NuMuR tasks due to the unique challenges posed by neural information retrieval. Then, we develop a methodology for NuMuR named Contrastive and Consistent Loss (CoCoL), which effectively balances the objectives of data forgetting and model performance retention. Experimental results demonstrate that CoCoL facilitates more effective and controllable data removal than existing techniques.

Index Terms—machine unlearning, neural ranking, information retrieval.

I. INTRODUCTION

Machine unlearning is the process of selectively removing specific data points from a trained machine-learning model [1, 2]. This task has gained significant attention in recent years as it addresses concerns regarding data privacy and model adaptability [3, 1, 4, 2].

In this work, we focus on neural ranking models nowadays used for *information retrieval (IR)*, i.e., on *neural IR*. In this context, machine unlearning may be needed for two main goals:

- addressing data-privacy concerns*, e.g., for deleting data of a user who has exercised their ‘right to be forgotten’ [1, 5];
- selectively deleting (e.g. outdated) information* [6, 7]. For instance, an IR system querying “What are the EU member states?” might need to exclude “UK” from its results post-2020 [8], illustrating a practical application of machine unlearning in IR systems.

It is therefore important to design methods for machine unlearning that can effectively deal with neural IR. Prominent existing model- and task-agnostic unlearning methods like *Amnesiac Unlearning* [9, 10] or *Negative Gradient Removal* [11, 12] (NegGrad) could be employed. However, these have been primarily designed for classification scenarios where it is typically possible to unlearn a class by deliberately damaging the model accuracy on the samples within that class; and Figure 1 illustrates that such unlearning strategies perform poorly in neural IR in the sense that reducing the performance of these models on the ‘forget set’ (i.e. on the data to be removed) incurs a severe performance loss on the ‘retain set’ (i.e. on the remaining data) and on test sets. We conjecture that this is due to strong dependencies in neural IR models, where removing individual data points disrupts learned patterns [13, 11, 14]. Another model-agnostic unlearning method is the teacher–student framework [15, 16]

which was likewise originally designed for classification tasks. However, as we discuss in detail in Section II-C, a naïve application of this approach to neural IR fails because the relevance scores generated by neural ranking models cannot easily be normalized.

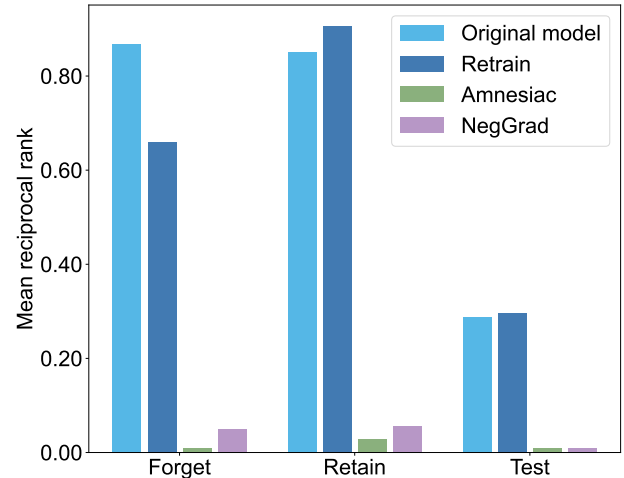


Fig. 1: Breakdown (in the sense of performance degradation on retain and tests sets) of classical machine unlearning baselines in neural information retrieval. The retrieval model and dataset are Contextualized Late Interaction over BERT (ColBERT) [17] and MircoSoft MACHine Reading COMprehension (MS MARCO) [18].

An additional challenge is that unlearning solutions which perform well on Goal a. may not be suitable for Goal b. and vice versa. To see this, note that the ideal (though typically prohibitively costly) solution for Goal a. would be to retrain the model from scratch on the retain set. However, even if such a ‘retrained’ model was available, there is no guarantee that its performance on the forget set would be low enough to satisfy Goal b.. Put differently, (re)training without the forget set does not achieve *controllable forgetting*, i.e. the ability to regulate the degree of performance loss on the forget set whilst ensuring minimal loss in retention performance and inference capability.

In this work, we introduce machine unlearning methodology for both Goals a. and b.. Our methodology is loosely based on the teacher–student framework from Chundawat et al. [15] and Kurmanji et al. [16] but tailored to the challenges of neural IR. Specifically, our contributions are as follows.

- 1) We formalise *Neural Machine UnRanking (NuMuR)* – the task of unlearning queries or documents within neural IR. We also provide two datasets to benchmark NuMuR.
- 2) We propose *Contrastive and Consistent Loss (CoCoL)* – a machine unlearning method specifically designed for the

J. Hou and G. Cosma are with the Department of Computer Science, School of Science, Loughborough University; A. Finke is with the Department of Mathematical Sciences, School of Science, Loughborough University.

Email:{J. Hou, A. Finke and G. Cosma}@lboro.ac.uk

NuMuR task.

- 3) We demonstrate that CoCoL improves upon baseline methods in experimental validations. Specifically, CoCoL achieves controllable forgetting, enabling variable scales of data removal without markedly degrading the model’s performance across both retain and test sets.

II. BACKGROUND AND PROBLEM DEFINITION

In this section, we provide, to our knowledge, the first formalisation of the task of machine unlearning within neural IR. We also explain why a naïve application of the teacher–student framework does not work in this context.

A. Machine unlearning

Let $\mathcal{W} \subseteq \mathbb{R}^d$ the *parameter space* and let \mathcal{S} be the universe of possible datasets. Let $M: \mathcal{S} \rightarrow \mathcal{W}$ be a *learning algorithm* which maps a *training set* $S \in \mathcal{S}$ to a *model* $w \in \mathcal{W}$. Learning algorithms may be random but we do not make this explicit in the notation. The *trained* model is then:

$$\mathcal{M}_{\text{train}} = M(S) := \arg \min_{w \in \mathcal{W}} L_S(w),$$

where $L_S(w)$ is some suitable loss which typically penalises the discrepancy between the prediction by Model w and the ground truth contained in the data set S .

Given the training set S , let $F \subseteq S$ be the *forget set* which contains a subset of data points in S to be unlearned; and let $R := S \setminus F \in \mathcal{S}$ be the *retain set* which contains the remaining data points. This defines the *retrained* model

$$\mathcal{M}_{\text{retrain}} := M(R).$$

Let $U: \mathcal{W} \times \mathcal{S} \times \mathcal{S} \rightarrow \mathcal{W}$ be a (potentially random) *unlearning algorithm for M* which defines the *unlearned* model

$$\mathcal{M}_{\text{unlearn}} := U(\mathcal{M}_{\text{train}}, F, R).$$

Unlearning algorithms are normally expected to ensure that the unlearned model closely approximates the retrained model, i.e., $\mathcal{M}_{\text{unlearn}} \approx \mathcal{M}_{\text{retrain}}$, whilst the computational cost of unlearning – starting from $\mathcal{M}_{\text{train}}$ – should be less than retraining from scratch on R [19, 20]. While mimicking $\mathcal{M}_{\text{retrain}}$ aligns with Goal a., it does not enable controllable forgetting (Goal b.). Therefore, we will base our unlearning approach on the *teacher–student* framework (also known as *knowledge distillation*) from Chundawat et al. [15], which can achieve pre-specified degrees of forgetting by implementing different distillation strategies.

Informally, the teacher–student framework specifies the unlearning algorithm U as using stochastic gradient-descent – initialised from $\mathcal{M}_{\text{train}}$ – to minimize (or at least decrease)

$$L_{\mathcal{M}_F, F}(w) + L_{\mathcal{M}_R, R}(w), \quad (1)$$

where, for any dataset A , the objective $L_{\mathcal{M}, A}(w)$ penalises the difference between predictions made by the *student* model $w \in \mathcal{W}$ (which is unlearning) and some fixed *teacher* model \mathcal{M} on A and is typically specified as follows.

- Since $\mathcal{M}_{\text{unlearn}}$ should perform similarly to $\mathcal{M}_{\text{retrain}}$ on R which in turn should perform similar to $\mathcal{M}_{\text{train}}$ (on R),

it is common to take $\mathcal{M}_R := \mathcal{M}_{\text{train}}$ in (1). This can be interpreted as training w to obey the ‘competent’ teacher model $\mathcal{M}_{\text{train}}$ on R [21, 22, 15, 16].

- Since $\mathcal{M}_{\text{unlearn}}$ should achieve controllable forgetting, i.e., achieve a pre-specified performance δ that is worse than $\mathcal{M}_{\text{train}}$ on F , it is common to take

$$L_{\mathcal{M}_F, F} := -L_{\mathcal{M}_{\text{train}}, F},$$

in (1) which can be viewed as training w to disobey the ‘competent’ teacher $\mathcal{M}_{\text{train}}$ on F [16]; and then to stop the gradient-descent iterations when the accuracy on the forget set has dropped to the target level δ . Alternatively, if the goal is that the unlearned model should perform similarly to $\mathcal{M}_{\text{init}} := M(\emptyset)$ on F , one could simply take $\mathcal{M}_F := \mathcal{M}_{\text{init}}$ in (1), which can be viewed as training w to obey the ‘incompetent’ teacher model $\mathcal{M}_{\text{init}}$ on F [15]. Of course, $\mathcal{M}_{\text{init}}$ could be replaced by another model, e.g., by an adversarial model trained on with noisy data [21, 12, 22].

B. Unlearning in neural information retrieval

The goal of *information retrieval (IR)* is to identify and retrieve documents in response to a search query [23]. Let \mathcal{Q} be the universe of potential queries and let \mathcal{D} be the universe of potential documents. Queries are user inputs or requests for specific information, typically in the form of words, phrases, or questions; documents refer to units of content, such as web pages or articles.

Then a dataset for (neural) IR $S \in \mathcal{S}$ consists of tuples (x, y) , where

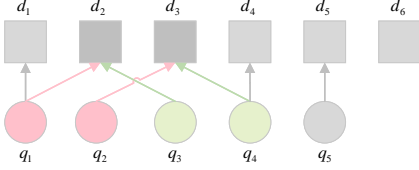
- $x = \langle q, d \rangle \in \mathcal{Q} \times \mathcal{D}$ is a query–document pair;
- $y \in \{+, -\}$ is the ground-truth relevance label of $\langle q, d \rangle$. Here, ‘+’ indicates that d is considered relevant to q ; ‘-’ indicates that d is irrelevant to q .

A *neural-ranking model* $w \in \mathcal{W}$ is then trained to predict a relevance score $f_w(x) \in \mathbb{R}$ of some query–document pair $x = \langle q, d \rangle$. Relevance scores output by neural-ranking models are used to rank documents. Each document associated with a query is sorted by its score in (descending) order, so that higher scores correspond to a higher rank and thus earlier positions in the search results. *Neural Machine UnRanking (NuMuR)* is then the task that this model unlearns either queries or documents (or both):

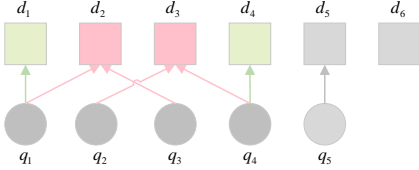
- *Query removal* refers to deleting a set of queries Q' (and associated relevance scores) from the dataset. In this case, $F := \{(\langle q, d \rangle, y) \in S \mid q \in Q'\}$.
- *Document removal* refers to deleting a set of documents D' (and associated relevance scores) from the dataset. In this case, $F := \{(\langle q, d \rangle, y) \in S \mid d \in D'\}$.

One of the difficulties encountered in NuMuR is that certain queries or documents may appear simultaneously in the retain set R and in the forget set F . For example, assume the query: “The best one-week itinerary for a trip to London” is associated with two recommended itineraries (i.e., documents). If one itinerary’s owner recalls their answer, we must unlearn one query–document pair whilst maintaining the other.

To formalise this issue, we split the retain set \mathbf{R} into an *entangled* set $\mathbf{E} := \{(\langle q, d \rangle, y) \in \mathbf{R} \mid \exists (\langle q', d' \rangle, y) \in \mathbf{F} : (q \in Q' \text{ or } d \in D')\}$ containing queries or documents that also appear in the forget set and a *disjoint set* $\mathbf{D} := \mathbf{R} \setminus \mathbf{E}$ containing all other queries and documents. Here, Q' and D' are again sets of queries and documents that should be unlearned. Figure 2 provides detailed illustrations of entangled sets and disjoint sets in both query removal and document removal.



(a) Query removal. Here, queries $Q' = \{q_1, q_2\}$ are to be unlearned. Thus, $\mathbf{F} = \{(\langle q_1, d_1 \rangle, y_{1,1}), (\langle q_1, d_2 \rangle, y_{1,2}), (\langle q_2, d_2 \rangle, y_{2,2})\}$, $\mathbf{E} = \{(\langle q_3, d_2 \rangle, y_{2,2}), (\langle q_4, d_3 \rangle, y_{4,3})\}$ and $\mathbf{D} = \{(\langle q_4, d_4 \rangle, y_{4,4}), (\langle q_5, d_5 \rangle, y_{5,5})\}$.



(b) Document removal. Here, documents $D' = \{d_2, d_3\}$ are to be unlearned. Thus, $\mathbf{F} = \{(\langle q_1, d_2 \rangle, y_{1,2}), (\langle q_2, d_3 \rangle, y_{2,3}), (\langle q_3, d_2 \rangle, y_{3,2}), (\langle q_4, d_3 \rangle, y_{4,3})\}$, $\mathbf{E} = \{(\langle q_1, d_1 \rangle, y_{1,1}), (\langle q_4, d_4 \rangle, y_{4,4})\}$ and $\mathbf{D} = \{(\langle q_5, d_5 \rangle, y_{5,5})\}$.

Fig. 2: Illustration of machine unranking for the dataset $\mathbf{S} = \{(\langle q_1, d_1 \rangle, y_{1,1}), (\langle q_1, d_2 \rangle, y_{1,2}), (\langle q_2, d_2 \rangle, y_{2,2}), (\langle q_3, d_2 \rangle, y_{2,2}), (\langle q_4, d_3 \rangle, y_{4,3}), (\langle q_4, d_4 \rangle, y_{4,4}), (\langle q_5, d_5 \rangle, y_{5,5})\}$.

C. Breakdown of existing knowledge-distillation based unlearning algorithms in neural information retrieval

The teacher–student approach from Chundawat et al. [15] (reviewed at the end of Section II-A) was primarily designed for classification models where the objectives $L_{\mathcal{M}, \mathcal{A}}(w)$ in (1) penalise the difference between class probabilities predicted by the student model w and by the reference (‘teacher’) model \mathcal{M} . The teacher–student approach thus exploits the fact that the outputs of neural classification models are class probabilities which are always normalized, so that forgetting (e.g., of a class) can always be ensured by simply lowering the associated class probabilities. Unfortunately, the relevance scores generated by neural ranking models cannot typically be normalized so that the ranking implications of modifying relevance-score distributions are unclear (see, e.g., Figure 8 in Appendix A); and this, as well as the fact that classification tasks do not involve entangled sets, causes the teacher–student approach to break down in neural IR. More specifically, we identify the following problems:

- 1) $\mathcal{M}_{\text{init}}$ cannot serve as ‘incompetent’ teacher. Due to the lack of normalization, the relevance scores on \mathbf{F} are not necessarily lower under the ‘incompetent’ teacher model (e.g., $\mathcal{M}_{\text{init}}$) than under the trained model. For example, a relevance score of 30, say, might imply a high rank under

the ‘competent’ teacher model, but a low rank under the incompetent teacher model (see Figure 8 in Appendix A). Therefore, using $\mathcal{M}_{\text{init}}$ as the teacher for the forget set while employing $\mathcal{M}_{\text{train}}$ as a ‘competent’ teacher for the retain set may be counterproductive.

- 2) *Controllable forgetting is challenging in neural ranking.* In a k -class classification model, forgetting a specific sample is achieved by adjusting the model output (i.e., class probabilities $\in \mathbb{R}^k$) so that the probability of the correct class falls below $1/k$. This type of forgetting can be quantitatively assessed using the Kullback–Leibler divergence from the outputs of the teacher model to the student model [21, 16, 22]. However, in neural ranking models, achieving forgetting by manipulating the model output (i.e., relevance score $\in \mathbb{R}$, an unnormalized scalar) is challenging, due to the absence of a clear threshold or benchmark for adjusting these scores.
- 3) *A naïve application of the teacher–student framework overlooks the entangled set.* A challenge in NuMuR is that some queries or documents can appear in both the retain and the forget set, as formalised by the entangled set. Conventional teacher–student frameworks implement distinct strategies for the forget and retain sets, as summarized in (1), without accounting for the entangled set. However, effectively decoupling the learned relevance estimation patterns between the forget set and the entangled set using such unlearning methods is problematic. This issue will be evidenced in Figure 6, where we illustrate that teacher–student approach that ignores the entangled set yields inferior performance on both the retain and test sets.

III. PROPOSED NEURAL MACHINE UNRANKING METHODOLOGY

In this section, we propose a new teacher–student framework for NuMuR, called *Contrastive and Consistent Loss (CoCoL)*. To address the three challenges discussed at the end of Section II, CoCoL introduces the following elements.

- 1) To overcome the problem of using an ‘incompetent’ teacher model (such as $\mathcal{M}_{\text{init}}$) in the presence of unnormalized relevance scores, we attempt to reduce the relevance scores on the forget set (relative to the trained model, $\mathcal{M}_{\text{train}}$) whilst seeking to maintain the relevance scores on the retain set.
- 2) Given that relevance scores are not normalized, to enable controllable forgetting, we stop the unlearning iterations when

$$\frac{1}{\#Q_{\mathbf{F}}} \sum_{q \in Q_{\mathbf{F}}} \frac{1}{\text{rank}_w(q)}, \quad (2)$$

is approximately equal to some pre-specified target $\delta > 0$ rather than basing the termination on the average relevance score reaching a predefined target level. Here, $Q_{\mathbf{F}} := \{q \in \mathcal{Q} \mid \exists (\langle q', d \rangle, y) \in \mathbf{F} : q' = q\}$ is the set of distinct queries in the forget set.

- In *query removal*, $\text{rank}_w(q)$ denotes the rank of the first relevant document for query q among all documents allocated to query q for ranking, evaluated by Model w .

Here, (2) simplifies to the classical mean reciprocal rank (MRR) as described by Liu et al. [24].

- In *document removal*, $\text{rank}_w(q)$ represents the rank of the first document marked for removal. This may differ from the rank of the first relevant document. For example, if Model w ranks the documents for Query q as $[d_1, d_3, d_4, d_2, \dots]$, where d_1 is the first relevant but d_2 is the first marked for removal, the reciprocal rank is recalculated as 0.25. While this differs from the classical MRR, we retain the ‘MRR’ notation for consistency in evaluation metrics.
- 3) To ensure that reducing the model accuracy on the forget set does not inadvertently damage the model performance on the entangled set, we pair a ‘forgetting sample’ with a random selection of a sample from the corresponding entangled set, as explained in the next section.

A. Objective

CoCoL uses gradient steps, started from the trained model $\mathcal{M}_{\text{train}}$, to decrease an objective of the form

$$L_{\mathcal{M}_{\text{FUEFUE}}}(w) + L_{\mathcal{M}_{\text{D,D}}}(w),$$

where $L_{\mathcal{M},\text{A}}(w)$ is again some objective which penalises the discrepancy between w and some reference model \mathcal{M} on some dataset A. Note that this objective differs from the standard teacher–student framework (1) in that the entangled set is moved into the first component. Note also that we say ‘decrease’ rather than ‘minimize’ because the unlearning is simply stopped when a pre-defined level of forgetting has been achieved (see below for details).

The components $L_{\mathcal{M}_{\text{FUEFUE}}}(w)$ and $L_{\mathcal{M}_{\text{D,D}}}(w)$ are implicitly defined through update rules which we now specify, where for some query–document pair $x = \langle q, d \rangle$:

$$\Delta_{w,\mathcal{M}}^{\alpha,\beta}(x) = -\frac{\alpha f_{\mathcal{M}}(x) - f_w(x) + \beta}{f_{\mathcal{M}}(x) + f_w(x)}, \quad (3)$$

measures the discrepancy between the relevance score $f_{\mathcal{M}}(x)$ returned by some fixed reference (‘teacher’) model \mathcal{M} and the relevance score $f_w(x)$ returned by the ‘student’ model w . Specifically, note that (3) decreases if the relevance score of the teacher model \mathcal{M} is much higher than that of the student model w . In (3), $\alpha > 0$ and $\beta \geq 0$ are tuning parameters whose choice will be discussed in Section III-B.

The update rules are then as follows.

- 1) *Contrastive loss: implicit definition of $L_{\mathcal{M}_{\text{FUEFUE}}}(w)$.* We employ a *contrastive* loss to modify the student model w such that it generates lower relevance scores on the forget set than the trained model $\mathcal{M}_{\text{train}}$ whilst ensuring that the relevance scores on the entangled set are maintained. Specifically, at each iteration, we randomly select a sample $(x, y) = (\langle q, d \rangle, y) \in \mathbf{F}$ from the forget set and a second sample $(x', y') = (\langle q', d' \rangle, y') \in \mathbf{E}_{\langle q, d \rangle}$, where $\mathbf{E}_{\langle q, d \rangle} := \{(\langle q'', d'' \rangle, y'') \in \mathbf{E} \mid q'' = q \text{ or } d'' = d\}$ contains the samples that are entangled with (x, y) , and then take a gradient step which reduces

$$\text{ReLU}(\Delta_{w,\mathcal{M}_{\text{train}}}^{\alpha,\beta}(x)) + |\Delta_{w,\mathcal{M}_{\text{train}}}^{1,0}(x')|. \quad (4)$$

Here, $\text{ReLU}(z) := \max(0, z)$. If $\mathbf{E}_{\langle q, d \rangle} = \emptyset$ then we take the second term in (4) to be zero.

- 2) *Consistent loss: implicit definition of $L_{\mathcal{M}_{\text{D,D}}}(w)$.* We employ a *consistent* loss to modify the student model w such that it generates relevance scores on the disjoint set that are similar to those from the trained model $\mathcal{M}_{\text{train}}$. Specifically, at each iteration, we randomly select a positive (i.e., relevant) sample $(x^+, y^+) = (\langle q^+, d^+ \rangle, y^+) \in \mathbf{D}$ and a negative (i.e., irrelevant) sample $(x^-, y^-) = (\langle q^-, d^- \rangle, y^-) \in \mathbf{D}$ from the disjoint set and then take a gradient step which reduces

$$|\Delta_{w,\mathcal{M}_{\text{train}}}^{1,0}(x^+)| + |\Delta_{w,\mathcal{M}_{\text{train}}}^{1,0}(x^-)|. \quad (5)$$

In summary, our CoCoL unranking approach is illustrated in Figure 3.

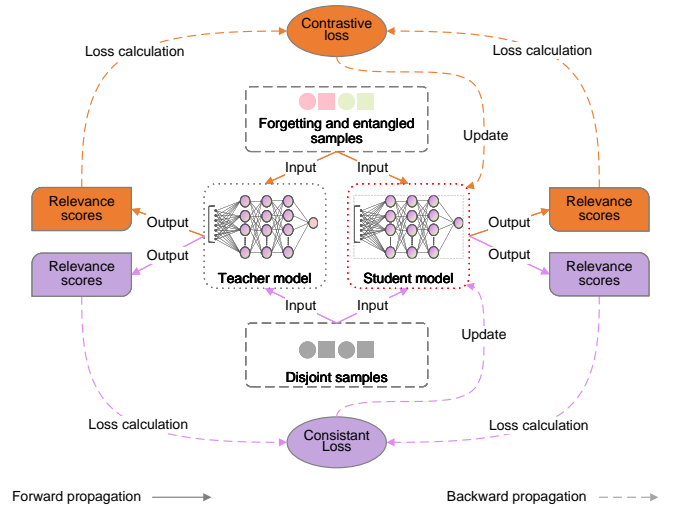


Fig. 3: Illustration of the proposed CoCoL method.

B. Choice of tuning parameters

The efficacy of CoCoL depends on the appropriate setting of parameters α and β in (4). From our empirical studies, we have found that $\alpha = 1$ and $\beta = 0$ works as a suitable default for most neural ranking models and datasets. However, adjusting α to a smaller value and β to a larger one can expedite forgetting, with minimal impact on the retain set. Specifically, for pretrained models, we recommend $\alpha \approx 1$ and β as a small integer, such as 5. For methods based on word embeddings, a significant reduction in α – for example, to 0.01 – can be beneficial. Detailed experimental results pertaining to various neural ranking models will be discussed in the subsequent sections.

C. Stopping criteria of unranking

With appropriate settings of α and β in (4), alternating between (4) and (5) ensures stable performance on both entangled and disjoint datasets, while performance on the forget set will progressively decline. Therefore, the optimal time to stop unlearning is when the performance on the forget set as measured by (2) reaches the pre-specified level $\delta > 0$.

IV. EXPERIMENTS

A. Datasets

Currently, there are no existing IR datasets specifically designed for machine unlearning research. To address this, we propose curating datasets derived from established benchmark IR datasets. In selecting the appropriate datasets for NuMuR, our selection criteria focused on datasets that feature extensive one-to-many relevant query–document and document–query pairings, essential for evaluating NuMuR methodologies. An in-depth review of resources listed on ir-datasets.com identified two sources that fulfil these requirements: MS MARCO [18] and TREC Complex Answer Retrieval (TREC CAR) [25]. These sources were selected due to their large sample sizes and the presence of overlapping queries and documents. The sample ratio of the forget set, entangled set, and disjoint set is approximately 1 : 1 : 2. Table I summarises the datasets.

TABLE I: Datasets created for this study.

Task	Item	MS MARCO	TREC CAR
Document Removal	Queries with multiple positive documents	2782	220271
	Positive documents per query	2.19	3.42
	To-be-ranked documents per query	2153	100
	Pairwise samples for training	5983761	1945509
	Pairwise samples for test	6668967	4710706
Query Removal	Positive passages with multiple queries	4035	19455
	Associated queries per positive documents	2.1	3.33
	To-be-ranked documents per query	1986	100
	Pairwise samples for training	8005618	752003
	Pairwise samples for test	6668967	4710706

B. Evaluation metrics

1) *Unlearning performance*: To evaluate ranking performance, we use the *MRR* metric as defined in (2) on the forget set. For the retain and test sets, *MRR* is similarly computed as the average of the reciprocals of the rank positions of the first retrieved relevant document for the queries in each set.

2) *Unlearning efficiency*: Unlearning efficiency is measured by the unlearn and relearn times.

a) *Unlearn time*: To ensure consistent measurement across different neural ranking models and unlearning methods we report the *normalized unlearn epoch duration*:

$$\begin{aligned} & \text{(normalized unlearn epoch duration)} \\ & := \frac{\text{(avg time per unlearning epoch of } \mathcal{M}_{\text{unlearn}})}{\text{(avg time per learning epoch of } \mathcal{M}_{\text{train}})}, \end{aligned}$$

as well as the *total unlearn time*:

$$\begin{aligned} \text{(total unlearn time)} & := \text{(normalized unlearn epoch duration)} \\ & \quad \times \text{(no of unlearn epochs)}. \end{aligned}$$

b) *Relearn time*: The relearn time measures the number of epochs required for the model to restore its pre-unlearning performance level. A longer relearn time is indicative of a more thorough unlearning process [13, 12, 22].

C. NIR models and unlearning baselines

We evaluate the proposed method alongside baseline approaches on multiple neural ranking models including two cutting-edge pretraining-based models, *ColBERT* [17] and *BERT with Dot Productions (BERTdot)* [26], along with two sophisticated word-embedding-based models, *Duet* [27], and *MatchPyramid* [28]. Table II lists the empirically chosen values of the tuning parameters α and β used for each model based on performance.

TABLE II: Values of (α, β) used in the experiments.

	MSMARCO query removal	MSMARCO document removal	TREC CAR query removal	TREC CAR document removal
COLBERT	(1, 5)	(1, 5)	(0.9, 0)	(0.9, 0)
BERTdot	(1, 5)	(1, 5)	(0.9, 0)	(0.9, 0)
DUET	(0.005, 0)	(0.01, 0)	(0.005, 0)	(0.01, 0)
MatchPyramid	(0.1, 0)	(0.1, 0)	(0.01, 0)	(0.01, 0)

Given the limited studies that exist in NuMuR, identifying comparable baselines is challenging. Therefore, the following task- and model-agnostic unlearning methods were selected as baselines:

- 1) *Amnesiac* [9, 10] continues training on $\mathcal{M}_{\text{train}}$ but with mislabeled samples in the forget set¹. To adapt this idea to NuMuR, we intentionally score several $\langle q, d \rangle$ pairs marked as ‘negative’ higher than those labelled as ‘positive’ in the forget set and then keep training $\mathcal{M}_{\text{train}}$ on the revised forget set and the original entangled set.
- 2) *NegGrad*, short for ‘negative gradient’, updates a learned model in the reverse direction of the original gradient on forget-set samples [11, 12].

We also report results for *retrain*, i.e., for retraining from scratch on only the retain set [19, 1, 20, 2] as this can be considered the ‘idealised’ approach (unless ‘controllable forgetting’ is sought). However, recall that as explained in Section II-A, obtaining $\mathcal{M}_{\text{retrain}}$ is typically prohibitively costly.

D. Experimental results

1) *Unlearning performance comparison*: The unlearning efficacy of the proposed CoCoL method compared to established baseline techniques is illustrated in Figure 4. Given this study’s focus on controllable forgetting, we conducted multiple experiments using CoCoL with various settings of (2).

First, aligning with Goal a., we set (2) to match the test performance of the $\mathcal{M}_{\text{retrain}}$. Additionally, aligning with Goal b., we set (2) to two other values:

- (1) the performance of $\mathcal{M}_{\text{retrain}}$ on the forget set (typically higher than Goal a.) to verify efficiency of CoCoL in achieving similar performance to the $\mathcal{M}_{\text{retrain}}$;
- (2) half of the performance of the $\mathcal{M}_{\text{retrain}}$ on the test set, to evaluate unlearning performance when requiring lower performance on the forget set.

¹Graves et al. [9] proposed two unlearning methods: the first method is as described, while the second requires gradient storage during the training of $\mathcal{M}_{\text{train}}$ and is challenging to apply in neural ranking tasks. Following Foster et al. [10], we use only the first method and refer to it as ‘Amnesiac’.

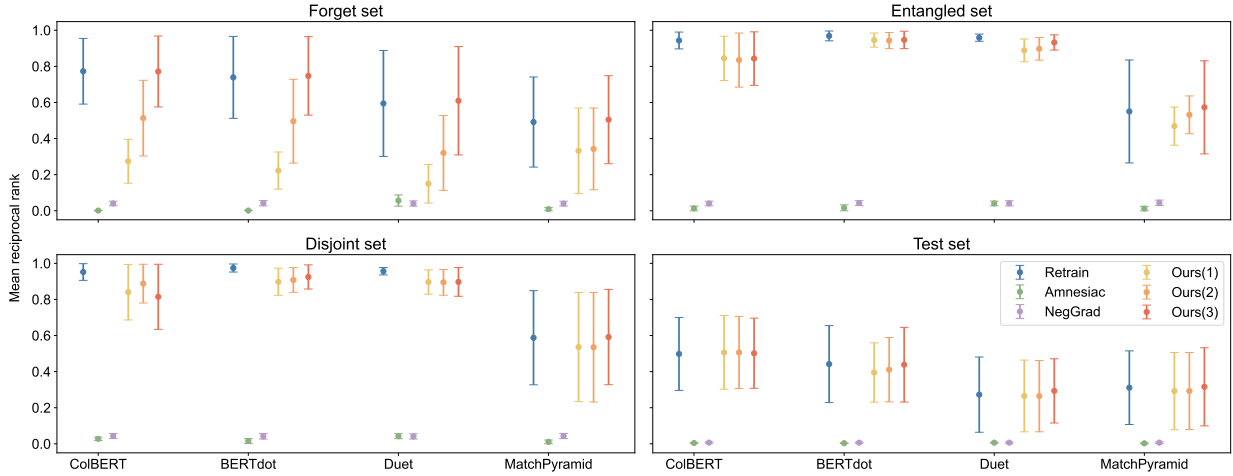


Fig. 4: Unlearning performance. The upper and lower bounds of each error bar denote the maximum and minimum MRR scores of the four tasks. See Figure 9 for detailed performance metrics for each task. Ours(1), Ours(2), and Ours(3) correspond to setting δ (i.e., the target for (2)) to half of the performance on the test set, the full performance on the test set, and the performance on the forget set, under $\mathcal{M}_{\text{retrain}}$, respectively. This figure illustrates that the proposed CoCoL method achieves controllable forgetting, maintaining varying degrees of performance on the forget set while maintaining performance close to the retrained model on the retain and test sets.

Notably, in the absence of the $\mathcal{M}_{\text{retrain}}$, we can use the $\mathcal{M}_{\text{train}}$ as a surrogate for evaluating performance on the retain and test sets.

For a detailed examination across four tasks and four neural ranking models, see Figure 9 in Appendix A. Pertinent to Goal b., additional details on the degrees of unlearning performance can be found in Figure 10 in Appendix A.

In terms of performance on the forget set, Amnesiac and NegGrad demonstrated the most substantial decreases. Aligning with Goal a., CoCoL consistently mirrored the performance metrics closest to the test set performance of $\mathcal{M}_{\text{retrain}}$.

Ideal unlearning performance on entangled and disjoint sets should mirror that of a retrained model. Therefore, we focus on Amnesiac, NegGrad, and CoCoL. As shown in Figure 4, NegGrad and Amnesiac consistently exhibited a decline in performance on both entangled and disjoint sets compared to the benchmark. In contrast, CoCoL, with different values of (2), maintained performance close to the benchmark across both sets, distinguishing it from NegGrad and Amnesiac, demonstrating controllable forgetting capacity.

For tet sets (unseen data), the performance of the retrained models serves as the benchmark, with higher performance indicating better inference ability. As shown in Figure 4, both Amnesiac and NegGrad exhibited significantly lower MRR scores compared to the benchmark, indicating their inability to maintain effective inference on unseen data. In contrast, CoCoL demonstrated not only controllable forgetting but also robust performance on unseen data.

2) *Unlearning efficiency*: The unlearning time for each experimental group is presented in Figure 5. When emulating the $\mathcal{M}_{\text{retrain}}$ (with (2) set to the performance of $\mathcal{M}_{\text{retrain}}$ on forget set), CoCoL consumed less time than Retrain in 13 out of 16 cases (four neural ranking models across four tasks).

Across the four neural ranking models evaluated, CoCoL

demonstrated a shorter unlearning time than Retrain for the pretrained models ColBERT and BERTdot in most cases. Conversely, CoCoL was less efficient with the Duet and MatchPyramid models, indicating potential areas for improving the efficiency of CoCoL in unlearning time for conventional models.

The relearn time for each model across different tasks is detailed in Figure III, considering only (2) set to meet Goal a.. The relearn process involves iterative training the unlearned model, $\mathcal{M}_{\text{unlearn}}$, with the forget set until the performance on this set matches the level of the originally trained model, $\mathcal{M}_{\text{train}}$. Consequently, the number of epochs in this iterative training is used to represent the relearn time.

Among the four neural ranking models examined, two pretraining-based models (ColBERT and BERTdot) and Duet could be relearned within two epochs. Specifically, for the TREC CAR query removal task, the relearn time of CoCoL was two epochs, compared to one epoch for the Retrain method. For the other three task groups, both CoCoL and Retrain demonstrated a relearn time of one epoch. In the case of MatchPyramid, CoCoL outperformed Retrain in two tasks of the MS MARCO dataset, whereas Retrain excelled in the two TREC CAR tasks.

E. Loss component effectiveness and parameter sensitivity analysis

This section describes the impact of individual loss components and parameter settings in CoCoL on the unlearning performance.

To assess the effectiveness of each component in contrastive and consistent loss, we conducted an experiment with both query removal and document removal tasks from the MS MARCO dataset using the state-of-the-art ColBERT neural-ranking model.

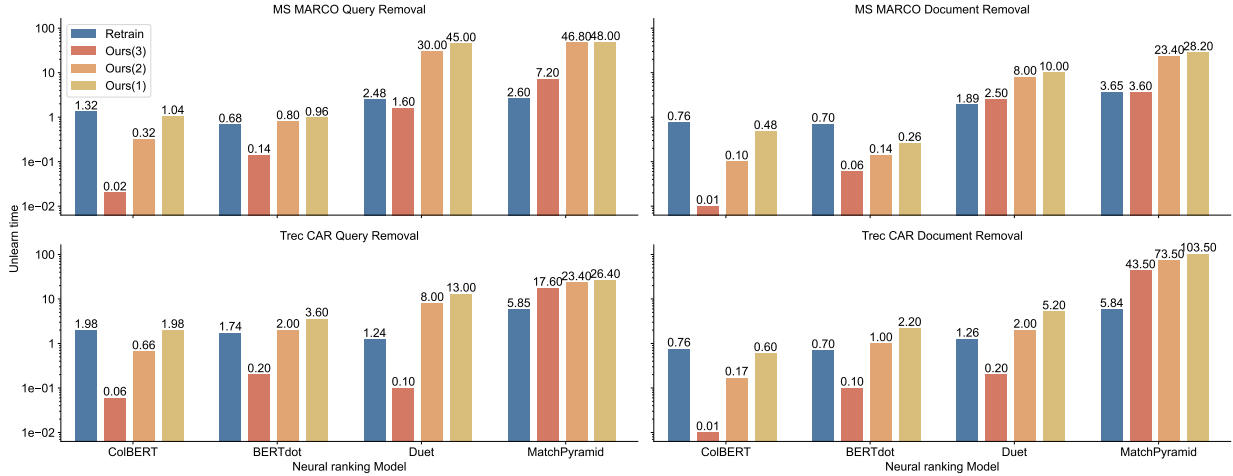


Fig. 5: Unlearn time (shorter is better). For advanced pretrained neural ranking models (ColBERT and BERTdot), CoCoL consumed significantly less time than Retrain when (2) was set to the performance on the forget set (Ours(3)), and took longer only in some groups for low forget set performance (Ours(1)). CoCoL was less efficient on pre-BERT models (MatchPyramid and Duet) compared to pretrained models. However, the former are less popular [29] and less effective (see Figure 9) than the latter.

TABLE III: Relearn time (higher is better).

Model	MS MARCO query removal		MS MARCO document removal		TREC CAR query removal		TREC CAR document removal	
	CoCoL	Retrain	CoCoL	Retrain	CoCoL	Retrain	CoCoL	Retrain
COLBERT	1	1	1	1	2	1	1	1
BERTdot	1	1	1	1	2	1	1	1
DUET	1	1	1	2	2	1	1	1
MatchPyramid	12	3	9	8	1	8	6	25

Figure 6 illustrates that omitting the consistent loss led to a more rapid decline in MRR scores on the forgetting set for both tasks, indicating that the consistent loss played a role in moderating the forgetting speed. The absence of consistent loss had a minimal impact on disjoint data: in query removal, the performance slightly underperformed the baseline model, whereas in document removal, it marginally surpassed the baseline.

The removal of the entangled component also resulted in an accelerated forgetting rate. However, this removal significantly diminished the performance of the unlearned model on all retained and unseen data, particularly in the document removal task, where there was a marked decline in model performance across all forget, retain, and test sets after just two epochs.

The second experiment examined the impact of parameters α and β in contrastive loss, which influenced the forgetting speed and the balance between forgetting and retaining performance. Using the ColBERT model tested on the MS MARCO dataset, the results, as depicted in Figure 7, provide intuitive observations. As α was progressively reduced from 1 to 0.1 in document removal or from 1 to 0.5 in query removal, the model exhibited a quicker forgetting speed without significantly damaging the performance on the retain and test sets. However, smaller values of α (e.g., 0.05, 0.01) resulted in a dramatic decline in MRR scores in the forget set and failed to maintain performance on the retain and unseen sets. The sensitivity analysis of β closely mirrored that of α . An optimal setting

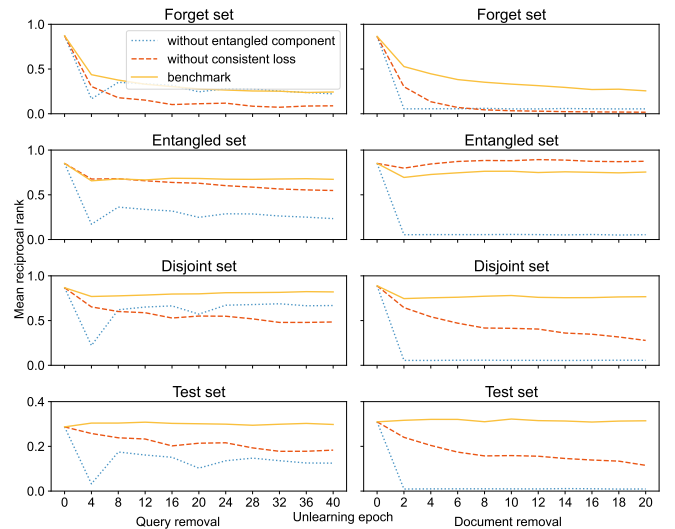


Fig. 6: The impact of omitting different loss components. The ‘benchmark’ indicates the benchmark performance using all loss components; ‘without entangled component’ and ‘without consistent loss’ denote the performance curves when excluding loss components associated with the entangled set in (4) and the disjoint set in (5), respectively. This figure demonstrates that both the entangled and consistent loss components are crucial for balancing forgetting performance with retention and inference capabilities.

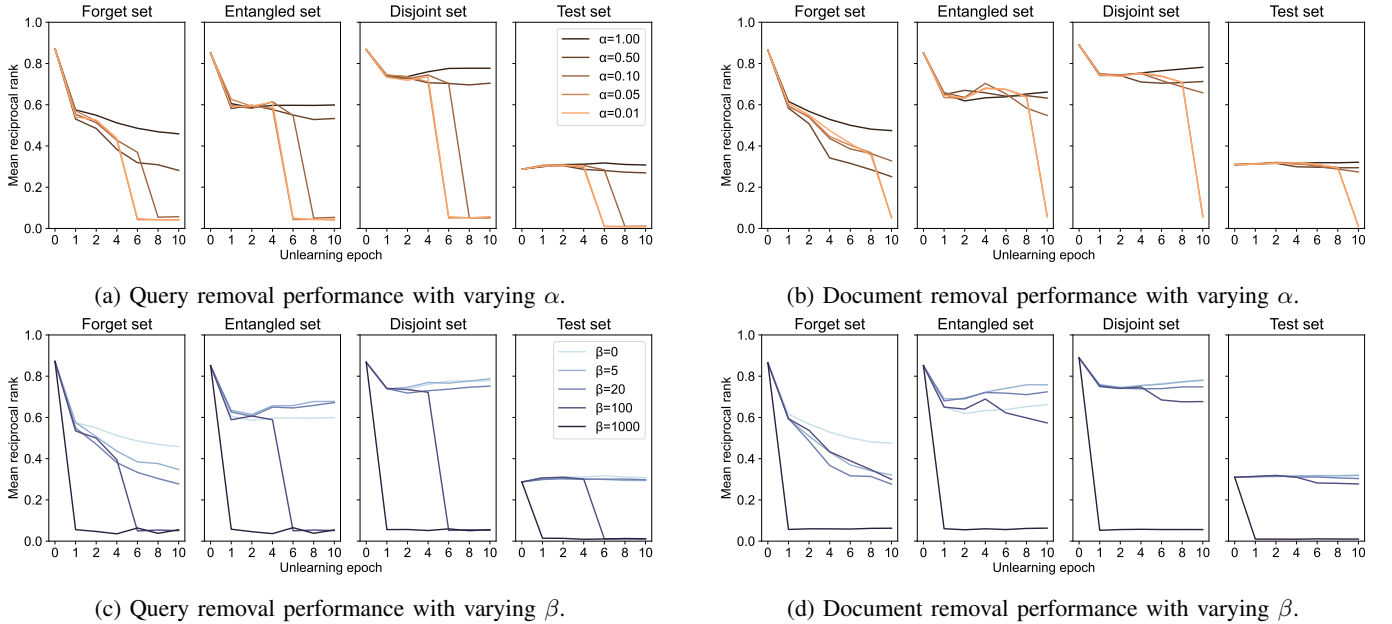


Fig. 7: The impact of parameters α and β in the contrastive loss. The figure shows that appropriate settings for α and β (specifically, $\alpha \geq 0.1$ or $\beta \leq 20$) enable effective forgetting while maintaining performance on the retain and test sets.

of β effectively accelerated forgetting.

F. Discussion

Our proposed method, CoCoL, was able to achieve controllable forgetting of targeted information while preserving the overall performance of neural ranking models on retained and unseen data. However, some limitations remain. Firstly, in the unlearning performance evaluation, CoCoL exhibited shortcomings in some TREC CAR groups, indicating an inconsistency in its effectiveness across different datasets and tasks. Secondly, the necessity of manual setting for parameters α and β poses a challenge. While these parameters influence the balance between forgetting and retaining performance, we provide experiential guidelines to streamline their tuning process. Finally, CoCoL does not differentiate between document removal and query removal, treating them as equivalent. A more tailored approach that recognizes and accommodates these differences could enhance the method’s precision and effectiveness. Addressing these limitations in future iterations of the method is crucial to improve its robustness and adaptability.

The exploration of CoCoL capabilities has opened avenues for future research in NuMuR. One critical area of focus should be on model-specific approaches, especially for models where CoCoL underperformed, such as MatchPyramid. CoCoL demonstrates proficiency with pretraining-based neural ranking models but the underperformance in other models suggests a need for strategies tailored to the unique characteristics of each model. Understanding and leveraging the specific features and mechanisms of different models can lead to more effective unranking approaches.

Additionally, future research should aim to distinguish between query removal and document removal more precisely. Recognizing and addressing the subtle differences between

these two sub-tasks could lead to the development of more nuanced and targeted unranking methods, enhancing the overall effectiveness and accuracy of the NuMuR task.

Another significant area for advancement is the automation of parameter settings, specifically for α and β . Automating these settings would not only streamline the unranking process but also potentially optimize the balance between forgetting and retaining performance, making the method more accessible and flexible.

Lastly, considering that neural ranking models typically comprise both embedding and ranking modules, it is imperative to investigate how unranking methods interact with these components differently. Future research should delve into the distinct impacts of unranking on embedding and ranking modules, and accordingly, develop improved unranking methods that treat these modules differently. Such an approach could lead to more effective unranking techniques, further advancing NuMuR.

V. CONCLUSION

In an era where data privacy and dynamic information landscapes are paramount, this study focuses on the field of machine unlearning, specifically within the context of neural ranking models for information retrieval (IR) systems. This research introduced the concept of Neural Machine UnRanking (NuMuR), presenting a novel method (Contrastive and Consistent Loss (CoCoL)) that effectively balances the delicate trade-off between controllable forgetting specific information and maintaining the overall performance of neural ranking models. CoCoL is particularly effective with pretraining-based neural ranking models, representing an advancement in addressing the unique challenges posed by machine unlearning in IR systems.

APPENDIX

RELEVANCE SCORE INTERVAL COMPARISON

This section provides examples showing that relevance score distributions vary across different neural ranking models. Figure 8 illustrates differences in the scale of relevance scores. Both BERTdot and ColBERT exhibit relatively lower relevance score ranges after training. Using $\mathcal{M}_{\text{init}}$ as the ‘incompetent’ teacher may result in higher relevance scores on forgetting samples. Additionally.

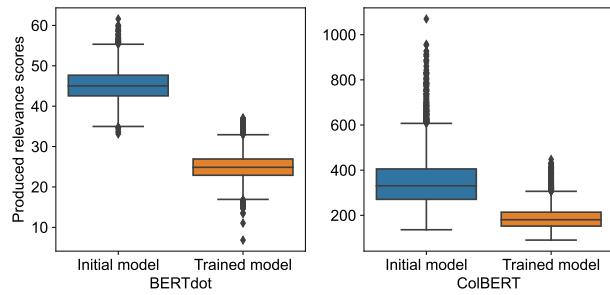


Fig. 8: Comparison between $\mathcal{M}_{\text{init}}$ and $\mathcal{M}_{\text{train}}$. The figure shows that neural ranking models trained on certain datasets may experience shifts in relevance score intervals.

DETAILED UNLEARNING PERFORMANCE

Figure 9 provides detailed unlearning performance (MRR scores of various neural ranking models across different tasks.

Figure 10 illustrates the unlearning performance of various neural ranking models across different tasks. The ‘Ideal epochs’ correspond to the stopping points that satisfy Goal a.. A clear trend is observed, where forgetting scores decline while performance on both entangled and disjoint sets remains stable (from Figure 10a to Figure 10l). Even when forgetting performance significantly diverges from test performance, CoCoL consistently maintains the stability of the retain set. This observation underscores the effectiveness of CoCoL in achieving Goal b..

In certain scenarios, CoCoL even enhances the convergence of retain sets throughout the unlearning process (as shown in Figure 10m and Figure 10n). Nevertheless, there are instances where it faces challenges in striking a precise balance between forgetting and retaining performances, evident in Figure 10o and Figure 10p.

DATA AVAILABILITY

To access the dataset and reproduce the experiments, please refer to the paper’s GitHub repository located at [github.com/\[whitespace\]](https://github.com/[whitespace]).

REFERENCES

- [1] L. Bourtole, V. Chandrasekaran, C. A. Choquette-Choo, H. Jia, A. Travers, B. Zhang, D. Lie, and N. Papernot, “Machine unlearning,” in *2021 IEEE Symposium on Security and Privacy (SP)*, 2021, pp. 141–159.
- [2] H. Xu, T. Zhu, L. Zhang, W. Zhou, and P. S. Yu, “Machine unlearning: A survey,” *ACM Computing Surveys*, vol. 56, no. 1, pp. 1–36, 2023.
- [3] E. F. Villaronga, P. Kieseberg, and T. Li, “Humans forget, machines remember: Artificial intelligence and the right to be forgotten,” *Computer Law & Security Review*, vol. 34, no. 2, pp. 304–313, 2018.
- [4] H. Zhang, T. Nakamura, T. Isohara, and K. Sakurai, “A review on machine unlearning,” *SN Computer Science*, vol. 4, no. 4, p. 337, 2023.
- [5] M. Chen, Z. Zhang, T. Wang, M. Backes, M. Humbert, and Y. Zhang, “When machine unlearning jeopardizes privacy,” in *Proceedings of the 2021 ACM SIGSAC Conference on Computer and Communications Security*, ser. CCS ’21. New York, NY, USA: Association for Computing Machinery, 2021, p. 896–911.
- [6] J. Teevan, K. Collins-Thompson, R. W. White, S. T. Dumais, and Y. Kim, “Slow search: Information retrieval without time constraints,” in *Proceedings of the Symposium on Human-Computer Interaction and Information Retrieval*, ser. HCIR ’13. New York, NY, USA: Association for Computing Machinery, 2013.
- [7] R. Campos, G. Dias, A. M. Jorge, and A. Jatowt, “Survey of temporal information retrieval and related applications,” *ACM Computing Surveys*, vol. 47, no. 2, aug 2014.
- [8] European Commission, “Relations with the United Kingdom,” https://commission.europa.eu/strategy-and-policy/relations-non-eu-countries/relations-united-kingdom_en, 2024, accessed: 2024-06-29.
- [9] L. Graves, V. Nagisetty, and V. Ganesh, “Amnesiac machine learning,” *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 35, no. 13, pp. 11 516–11 524, May 2021.
- [10] J. Foster, S. Schoepf, and A. Brintrup, “Fast machine unlearning without retraining through selective synaptic dampening,” *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 38, no. 11, pp. 12 043–12 051, Mar. 2024.
- [11] P.-F. Zhang, G. Bai, Z. Huang, and X.-S. Xu, “Machine unlearning for image retrieval: A generative scrubbing approach,” in *Proceedings of the 30th ACM International Conference on Multimedia*, ser. MM ’22. New York, NY, USA: Association for Computing Machinery, 2022, p. 237–245.
- [12] A. K. Tarun, V. S. Chundawat, M. Mandal, and M. Kankanhalli, “Fast yet effective machine unlearning,” *IEEE Transactions on Neural Networks and Learning Systems*, 2023.
- [13] Y. Cao and J. Yang, “Towards making systems forget with machine unlearning,” in *2015 IEEE Symposium on Security and Privacy*. IEEE, 2015, pp. 463–480.
- [14] J. Xu, Z. Wu, C. Wang, and X. Jia, “Machine unlearning: Solutions and challenges,” *IEEE Transactions on Emerging Topics in Computational Intelligence*, vol. 8, no. 3, pp. 2150–2168, 2024.
- [15] V. S. Chundawat, A. K. Tarun, M. Mandal, and M. Kankanhalli, “Can bad teaching induce forgetting? unlearning in deep networks using an incompetent teacher,”

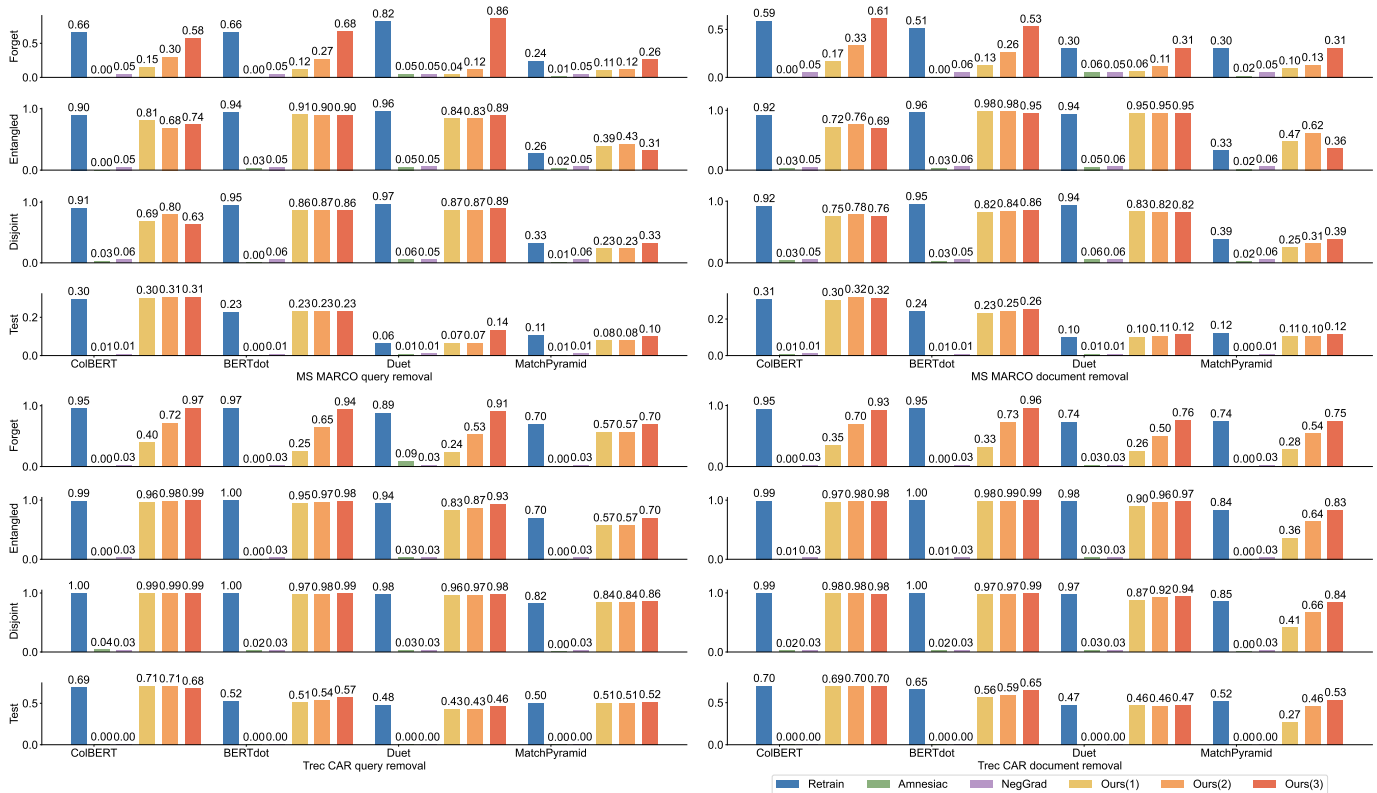


Fig. 9: Detailed unlearning performance (MRR scores). Ours(1), (2), and (3) in legends correspond to setting (2) to half of the performance on the test set, the full performance on the test set, and the performance on the forget set, generated by CoCoL, respectively.

Proceedings of the AAAI Conference on Artificial Intelligence, vol. 37, no. 6, pp. 7210–7217, Jun. 2023.

- [16] M. Kurmanji, P. Triantafillou, J. Hayes, and E. Triantafillou, “Towards unbounded machine unlearning,” in *Advances in Neural Information Processing Systems*, A. Oh, T. Neumann, A. Globerson, K. Saenko, M. Hardt, and S. Levine, Eds., vol. 36. Curran Associates, Inc., 2023, pp. 1957–1987.
- [17] O. Khattab and M. Zaharia, “ColBERT: Efficient and effective passage search via contextualized late interaction over bert,” in *Proceedings of the 43rd International ACM SIGIR Conference on Research and Development in Information Retrieval*, ser. SIGIR ’20. New York, NY, USA: Association for Computing Machinery, 2020, p. 39–48.
- [18] N. Craswell, B. Mitra, E. Yilmaz, D. Campos, and J. Lin, “MS MARCO: Benchmarking ranking models in the large-data regime,” in *Proceedings of the 44th International ACM SIGIR Conference on Research and Development in Information Retrieval*, 2021, pp. 1566–1576.
- [19] A. Sekhari, J. Acharya, G. Kamath, and A. T. Suresh, “Remember what you want to forget: Algorithms for machine unlearning,” *Advances in Neural Information Processing Systems*, vol. 34, pp. 18 075–18 086, 2021.
- [20] E. Chien, C. Pan, and O. Milenkovic, “Efficient model updates for approximate unlearning of graph-structured data,” in *The Eleventh International Conference on Learning Representations*, 2022.
- [21] J. Kim and S. S. Woo, “Efficient two-stage model retraining for machine unlearning,” in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2022, pp. 4361–4369.
- [22] V. S. Chundawat, A. K. Tarun, M. Mandal, and M. Kankanhalli, “Zero-shot machine unlearning,” *IEEE Transactions on Information Forensics and Security*, vol. 18, pp. 2345–2354, 2023.
- [23] S. Ceri, A. Bozzon, M. Brambilla, E. Della Valle, P. Fraternali, and S. Quarteroni, *An Introduction to Information Retrieval*. Berlin, Heidelberg: Springer, 2013, pp. 3–11.
- [24] T.-Y. Liu *et al.*, “Learning to rank for information retrieval,” *Foundations and Trends® in Information Retrieval*, vol. 3, no. 3, pp. 225–331, 2009.
- [25] L. Dietz and J. Foley, “TREC CAR Y3: Complex answer retrieval overview,” in *Proceedings of Text REtrieval Conference (TREC)*, 2019.
- [26] S. Hofstätter, S.-C. Lin, J.-H. Yang, J. Lin, and A. Hanbury, “Efficiently teaching an effective dense retriever with balanced topic aware sampling,” in *Proceedings of the 44th International ACM SIGIR Conference on Research and Development in Information Retrieval*, ser. SIGIR ’21. New York, NY, USA: Association for Computing Machinery, 2021, p. 113–122.
- [27] B. Mitra, F. Diaz, and N. Craswell, “Learning to match using local and distributed representations of text for web

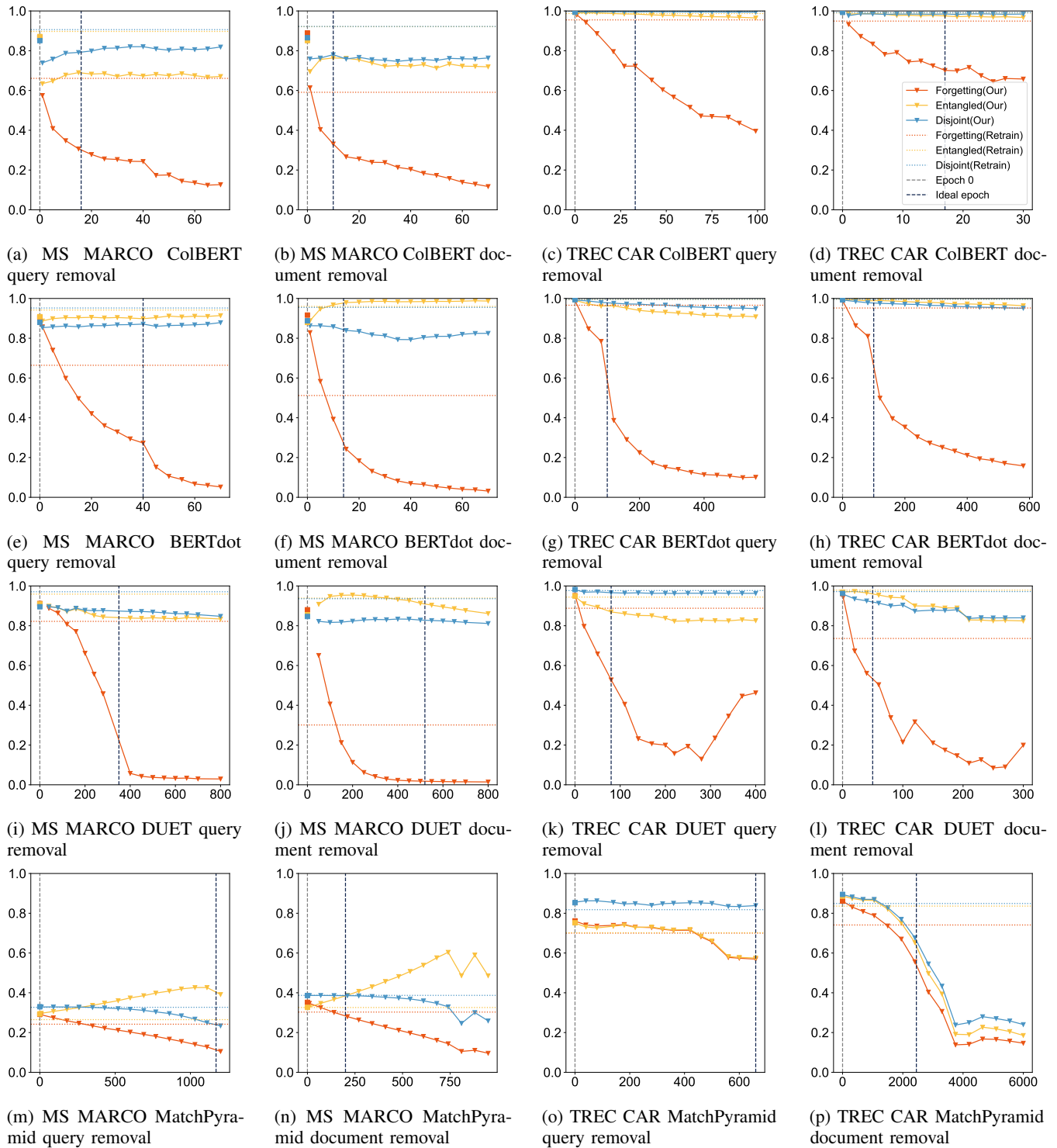


Fig. 10: Unlearning performance of CoCoL across various neural ranking models on different tasks.

- search,” in *Proceedings of the 26th International Conference on World Wide Web*, ser. WWW '17. Republic and Canton of Geneva, CHE: International World Wide Web Conferences Steering Committee, 2017, p. 1291–1299.
- [28] L. Pang, Y. Lan, J. Guo, J. Xu, S. Wan, and X. Cheng, “Text matching as image recognition,” *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 30, no. 1, Mar. 2016.
- [29] W. X. Zhao, J. Liu, R. Ren, and J.-R. Wen, “Dense text retrieval based on pretrained language models: A survey,” *ACM Transactions on Information Systems*, vol. 42, no. 4, feb 2024.