Continual Learning of Long Topic Sequences in Neural Information Retrieval

Thomas Gerald, Laure Soulier

Continual Learning of Long Topic Sequences in Neural Information Retrieval

Thomas Gerald and Laure Soulier

CNRS-ISIR, Sorbonne University, Paris, France
{gerald, soulier}@isir.upmc.fr

Abstract. In information retrieval (*IR*) systems, trends and users' interests may change over time, altering either the distribution of requests or contents to be recommended. Since neural ranking approaches heavily depend on the training data, it is crucial to understand the transfer capacity of recent *IR* approaches to address new domains in the long term. In this paper, we first propose a dataset based upon the MSMarco corpus aiming at modeling a long stream of topics as well as *IR* property-driven controlled settings. We then in-depth analyze the ability of recent neural *IR* models while continually learning those streams. Our empirical study highlights in which particular cases catastrophic forgetting occurs (e.g., level of similarity between tasks, peculiarities on text length, and ways of learning models) to provide future directions in terms of model design.

Keywords: Continual learning, Information retrieval, Neural ranking models

1 Introduction

The information Retrieval (IR) field has seen a keen interest in neural approaches these last years [27, 14, 24, 12] thanks to recent advances in semantic and language understanding. However, these approaches are heavily data dependent, often leading to specialization for a certain type of corpus [28, 32]. If document retrieval remains a core task, many challenges revolve around, such as news detection [37], question answering [43] or conversational search [8]. In all these tasks, users' needs or document content might evolve through time; leading to evolving queries and/or documents and shifting the topic distribution at the inference step [3, 26, 37]. It is, therefore, crucial to understand whether *IR* models are able to change their ranking abilities to new topics/trends, but also to be still able to perform on previous topics/trends if these ones remain up to date. Accumulating and preserving knowledge is thus an important feature in *IR*, allowing to continuously adapt to new domains or corpora while still being effective on the old ones. This requirement refers to an emerging research field called *Continual learning* [17, 34, 40]. In practice, continual learning proposes to learn all tasks sequentially by guaranteeing that previous knowledge does not deteriorate through the learning process; this phenomenon is called *catastrophic forgetting*. To solve this issue, one might consider multi-task learning [30] in which models learn together all the sets of tasks. Another approach would consists in learning a model for each task, but, in this case, the knowledge is not transferred between

previous and current tasks. These two last settings are not always realistic in *IR*, since they consider that all tasks are available at the training step. In practice, content and users' needs may evolve throughout the time [26, 3].

To the best of our knowledge, only one previous work has addressed the continual learning setting in *IR* [22], highlighting the small weakness of the studied neural models to slightly forget knowledge over time. However this work has two limitations: 1) it only considers few tasks in the stream (2 or 3 successive datasets) and does not allow to exhibit neural model abilities in the more realistic scenario of long-term topic sequences (i.e., a larger number of users and topics implying evolving information needs/trends). 2) Although authors in [22] use datasets of different domains, there is no control of stream properties (e.g., language shift [1, 3], information update [26]) allowing to correlate the observed results with *IR* realistic settings, as done in [40] for classification tasks.

The objective of this paper is thus to provide a low-level analysis of the learning behavior of neural ranking models through a continual setting considering long sequences and *IR*-driven controlled topic sequences. In this aim, we propose to study different neural ranking models and to evaluate their abilities to preserve knowledge. To this end, we consider neural rankers successively fined tuned on each task of the sequence. More particularly, our contribution is threefold:

- We design a corpus derived from the MSMarco Passage ranking dataset [29] to address long sequences of topics for continual learning and *IR*-driven controlled topic sequences (Section 4).
- We compare the different neural ranking models in a long-term continual *IR* setting (Section 5.1) and the controlled settings (Section 5.3).
- We in-depth investigate the impact of task similarity level in the continual setting on the learning behavior of neural ranking models (Section 5.2).

2 Related Works

Neural Information Retrieval. Deep learning algorithms have been introduced in *IR* to learn representations of tokens/words/texts as vectors and compare query and document representations [10, 27, 42, 12, 13, 5]. With the advance of sequence-to-sequence models, semantic matching models have grown in popularity, particularly due to the design of new mechanisms such the well-known self-attention in transformer networks [39] or language models such as Bert [6].

Many *IR* approaches benefit from those advances as *CEDR* [24] that combines a Bert language model with relevance matching approaches including KNRM [42] and PACRR [12]. Moreover, recent works addressed ranking with sequence-to-sequence transformers based approach as the *Mono-T5* model [31] for re-ranking documents returned by a BM25 ranker. Using a weak initial ranker such as BM25 may be the bottleneck of reaching higher performances, some approaches are thus reconsidering dense retrieval [14, 15, 7, 44]. All these models are data-dependent, relying on word/topic/query distribution in the training dataset and their application to new domains is not always straightforward [28, 32]. While previous works addressed this issue by leveraging for instance fine-tuning techniques [23, 43], one can wonder whether these models are still effective

on the word/topic/query distribution of the training dataset. This condition is particularly crucial for open-domain *IR* systems (e.g., public search engines or future conversational search systems) since they should be able to face multiple users and solve both persistent information needs and event-related ones.

Continual Learning. Continual learning generally defines the setting in which a model is trained consecutively on a sequence of tasks and need to adapt itself to new encountered tasks. One main issue of continual learning is that models need to acquire knowledge throughout the sequence without forgetting the knowledge learnt on previous tasks (*catastrophic forgetting*). To solve the catastrophic forgetting issue, three main categories can be outlined [18]. First, regularisation approaches continually learn to address new tasks using soft or hard preservation of weights [17, 21, 41]. For instance, the *Elastic Weight Consolidation* model [17] softly updates weights for a new task according to their importance in the previous one. Second, replay approaches [34, 2, 25] (or *rehearsal approaches*), replay examples of previous tasks while training the model on a new one. Third, architecture-based approaches [4, 20, 40] rely on the decomposition of the inference function. For instance, new approaches leveraging techniques of neural architecture search [20, 40] have been proposed.

Recently some works have addressed the continual learning setting for *NLP* tasks. LAMOL [38] for continual language modelling, [19] for conversational systems or [9] for translations tasks. While it exists *IR* approaches to perform on different domains such as using batch balanced topics [11], at the best of our knowledge, only one study addresses *IR* in the continual setting [22], comparing neural ranking models on three successive tasks (MSMarco, TREC-Microblog, and TREC CORD19). Our work follows this line by providing an analysis of the behavior of neural ranking models on longer sequences of topics. We also design *IR*-driven controlled sequences to highlight to what extent neural models face *IR*-specific divergences, such as language drift or documents collection update.

3 Research design for continual learning in *IR*

We address in this paper the following research questions aiming at analyzing the resilience of *IR* models to catastrophic forgetting:

- **RQ1:** How to design a sequence of tasks for continual learning in *IR*?
- **RQ2:** What are the performance of neural ranking models while learning long sequences of topics? Can we perceive signals of catastrophic forgetting?
- **RQ3:** Does the similarity level of tasks in the sequence impact the model effectiveness and their robustness to catastrophic forgetting?
- **RQ4:** How do neural ranking models adapt themselves to queries or documents distribution shifts?

3.1 Continual learning setting and metrics

We propose a continual learning setting based on long sequences. The latter consists in fine-tuning a model on different tasks successively. Following [22], we instantiate tasks by topics/domains, but we rather focus on long sequences of tasks with the perspective that such setting can be connected with long-term

trends/changes of user interests. In practice, we consider a sequence of n tasks $S = \{\mathcal{T}_1, \ldots, \mathcal{T}_i, \ldots, \mathcal{T}_n\}$, each task $\mathcal{T}_i$ corresponds to a set of queries and their associated relevant documents. We suppose that each task relies on different properties or distributions as in [33]. Neural ranking models are successively fine-tuned over the long sequence S of topics. The objective is to track each task and evaluate each of them at different timestep of the sequence (i.e., after the successive fine-tuning) to measure the model's abilities to adapt to new tasks and their resilience to catastrophic forgetting.

In practice, we propose to track in each sequence a subset of 5 randomly selected tasks (tracking whole tasks throughout the whole sequence is too computationally expensive). For each of these tasks, we will measure at each step of the topic sequence the MRR@K. To measure the catastrophic forgetting mf for a given task $\mathcal{T}_i$ at a training step θ_j (associated to task $\mathcal{T}_j$), we identify the maximum value obtained by the model along the sequence S and compare its performances at each training step θ_j with the maximum value:

$$mf(i, \theta_j) = \left(\max_{k \in 1,2,\ldots,|S|} score(i, \theta_k) \right) - score(i, \theta_j) \tag{1}$$

where $score(i, \theta_j)$ refers to a ranking metric for the task $\mathcal{T}_i$ using the model obtained training the j^{th} task $\mathcal{T}_j$ in the sequence. Looking to $mf(i, \theta_j)$ for all j in the sequence allows observing which tasks have a significant negative transfer impact on $\mathcal{T}_i$ (high value) and which have a low negative impact (low value).

3.2 Neural ranking models and learning

We evaluate two different state-of-the-art neural `IR` models:
• The vanilla Bert[6](noted **VBert**) estimating a ranking score based on a linear layer applied on the averaged output of the last layer of the Bert language model.
• The *Mono-T5-Ranker*[31] (noted **MonoT5**) based on a *T5-base* model fine-tuning and trained to generate a positive/negative token.

Implementation details: All models are trained with *Adam* optimizer [16], the optimizer state is not reinitialized for each task of the sequence. Indeed, re-initializing the optimizer will lead to observe a spike in the loss function whether addressing a same or a different task due to the state of *Adam* optimizer parameters. As previous work in *IR* [22, 31, 6], we perform sparse retrieval by re-ranking top-1000 most relevant documents retrieved by the BM25 model [36].

For MonoT5 we start with the *t5-base*[1] model with a learning rate of $10{-}3$ and batch size of 16. For the VBert model[2], the batch size is 16 with a learning rate of 2×10^{-5} for Bert parameters and 10^{-3} for scoring function parameters.

4 MSMarco Continual Learning corpus

Our continual learning framework is based on learning from a long sequence of tasks. One main difficulty is to create this sequence considering the availability

[1] https://huggingface.co/transformers/model_doc/t5.html
[2] using bert-base-uncased pretrain

of *IR* datasets. One method would be to build a sequence of datasets of different domains as in [22], but the number of datasets adapted to neural *IR* (with a sufficiently large number of queries and relevance judgments) is not sufficient for long sequences setting. We propose to model the task at a lower granularity level, namely topics, instead of the dataset granularity. In what follows, we present our methodology for creating long sequences of topics using the MSMarco dataset. Once this dataset is validated, it serves as a base for designing controlled settings related to particular *IR* scenarios (all settings and models are open-sourced [3]).

4.1 RQ1: Modeling the long topic sequence

To create the long sequence, we consider the MSMarco dataset [29]. Such dataset is based on real users' questions on Bing. Our intuition is that several queries might deal with the same user's interest (e.g., "what is the largest source of freshwater on earth?" or "what is water shortage mitigation"). These groups of queries denote what we call in the remaining paper *topics*. To extract topics, we propose a two-step method: extracting clusters from randomly sampled queries and populating those clusters with queries from the whole dataset. We use a similarity clustering[4] based on query representations obtained using the sentence-BERT model [35]. The clustering is based on a sample of 50,000 randomly picked queries and estimates the similarity cosine distance according to a threshold t to build clusters of a minimum size of s. We then populate clusters using other queries from the dataset according to threshold t. Finally, we produce the sequence of topics by randomly rearranging clusters to avoid bias of cluster size. Another sequencing method might be envisioned for future work, for instance considering a temporal feature by comparing topic trends in real search logs. In practice, the value of the threshold t differs in each step of clustering and populating, leading to the threshold t_1 and t_2 (with $t_2 < t_1$) to obtain clusters of reasonable size to be used for neural models. Depending on the value of those hyper-parameters (t_1, t_2, s), we obtain three datasets of topic sequences of different sizes (19, 27, and 74), resp. called *MS-TS*, *MS-TM* and *MS-TL* (for small, medium, large).

Statistics of these three topic sequences are described in Table 1. To build the train/validation/test sets, we constraint the validation and the test set to be composed of approximately 40 queries by topic. Notice that we do not use the original split as it remains difficult to consider enough testing examples falling into the created topics.

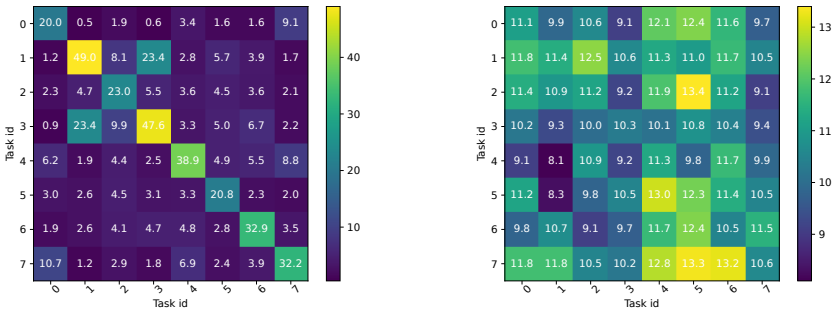
4.2 Evaluating the long topic sequence

To verify the relevance of the clusters, we aim at measuring retrieval evidence within and between clusters (i.e., queries within clusters might have similar retrieval evidence and queries between clusters might have different ones). As retrieval evidence, we use the retrieved documents for each query using the *BM25* model with default parameters [5]. Our intuition is that similar queries

[3] https://github.com/tgeral68/continual_learning_of_long_topic
[4] https://www.sbert.net/examples/applications/clustering (fast clustering)
[5] Implemented in pyserini: https://github.com/castorini/pyserini

Table 1: Parameters and statistics of the generated dataset and their inter/intra task similarity metric ($c-score$). The intra-score is the mean $c-score$ when comparing a task with itself, and the inter score when comparing different tasks.

Name	t_1	s	t_2	$\|\mathcal{T}\|$	#queries by topics	inter	intra
MS-TS	0.7	40	0.5	19	$3,650 \pm 1,812$	3.8%	31.4%
MS-TM	0.75	20	0.5	27	$3,030 \pm 1,723$	4.1%	32.1%
MS-TL	0.75	10	0.55	74	$1,260 \pm 633$	3.3%	34.6%
MS-RS	-	-	-	19	$3,650 \pm 1,812$	10.3%	10.2%
MS-RM	-	-	-	27	$3,030 \pm 1,723$	9.9%	9.8%
MS-RL	-	-	-	74	$1,260 \pm 633$	8.7%	8.8%



(a) Topics (b) Random

Fig. 1: Matrix of similarities between topics for 8 tasks of MS-S (1a) and MS-RL (1b) datasets. The c-score ($\times 100$) is processed on all topic pairs, a high value (yellow) denotes the level of retrieved document overlap between queries of topics.

should share retrieved documents (and vice versa). To compare queries within and between clusters, we randomly select two pools (noted A_i and B_i) of 250 queries within each cluster associated to task $\mathcal{T}_i$. Let $D^{A_i} = \{D_q | q \in A_i\}$ (resp. $D^{B_i} = \{D_q | q \in B_i\}$) the documents returned by the ranker for the queries in A_i (resp. B_i).

We thus compute the $c-score$ which measures the ratio of common documents between two tasks $\mathcal{T}_i$, $\mathcal{T}_j$ (or same task if $i = j$) as follows:

$$c-score(\mathcal{T}_i, \mathcal{T}_j) = \frac{|D^{A_i} \cap D^{B_j}|}{|D^{A_i}|} \tag{2}$$

This score is then averaged over pairs of topics within the sequence (intra when comparing topics with their-selves and inter when comparing different topics).

To evaluate our topic sequence methodology, for each of the three datasets we create a long topic sequence baseline in which clusters are extracted randomly from the queries of topics based corpora. We obtain three randomized datasets denoted *MS-RS*, *MS-RM* and *MS-RL*.

Table 1 reports for each of the generated datasets the intra and inter c-scores. By comparing the inter metric between both corpus settings (around 3/4% for the clustering-based ones and around 9/10% for randomized ones), one can conclude that our long topic sequence includes clusters that are more different than

the ones created in the randomized corpus. The trend is opposite when looking at the intra, meaning that our sequence relies on clusters gathering similar queries but dissimilar from each other. This statement is reinforced in Figure 1 which depicts the $c - score$ matrix for all couples $(i, j) \in \{1, 2, \ldots, |S|\}^2$ for a subset of 8 tasks (for clarity) of the $MS - S$ and $MS - RS$ corpora. We observe that for the randomized matrix (Figure 1(b)), the metric value is relatively uniform. In contrast, in the matrix obtained from our long topic sequence based on clustering (Figure 1(a)), the c-score is very small when computed for different topic clusters (low inter similarity) and higher in the diagonal line (high intra similarity).

4.3 *IR*-driven controlled stream-based scenario

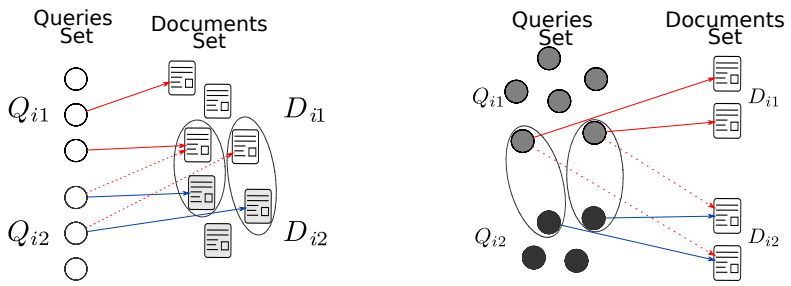
In this section, we focus on local peculiarities of the long topic sequence by analyzing *IR*-driven use cases, such as documents or queries distribution shifts. Typically, the available documents may change over time, or even some can be outdated (for instance documents relevant at a certain point in time). Also, it happens that the queries evolve, either by new trends, the emergence of new domains, or shifts in language formulation. To model those scenarios, we propose three different short topic streams to fit the local focus. Topics are based on our long topic sequence $S = \{\mathcal{T}_1, \ldots, \mathcal{T}_i, \ldots, \mathcal{T}_n\}$ built on MSMarco (Section 4.1). For each scenario, we consider an initial setting $\mathcal{T}_{init}$ modeling the general knowledge before analyzing particular settings. In other words, it constitutes the data used for the pre-training of neural ranking models before fine-tuning on a specific sequence. The proposed controlled settings are:

- **Direct Transfer [40]:** The task sequence is $(\mathcal{T}_{init}, \mathcal{T}_i^+, \mathcal{T}_j, \mathcal{T}_i^-)$ where tasks $\mathcal{T}_i^+$ and $\mathcal{T}_i^-$ belong to the topic task $\mathcal{T}_i$ and have different sizes ($|\mathcal{T}_i^-| \ll |\mathcal{T}_i^+|$). This setting refers to the case when the same topic comes back in the stream with new available data (new queries and new relevant documents).
- **Information Update:** The task sequence is $(\mathcal{T}_{init}, \mathcal{T}_i', \mathcal{T}_i'')$ where $\mathcal{T}_i'$ and $\mathcal{T}_i''$ have dissimilar document distributions and a similar query distribution. Intuitively, it can be interpreted as a shift in the required documents, such as new trends concerning a topic or an update of the document collection.
- **Language Drift:** The task sequence is $(\mathcal{T}_{init}, \mathcal{T}_i^*, \mathcal{T}_i^{**})$ where $\mathcal{T}_i^*$ and $\mathcal{T}_i^{**}$ have similar document distributions and a dissimilar query distribution. This can correspond to a change of query formulation or focus in a same topic.

To build those sequences, the initial task $\mathcal{T}_{init}$ aggregates k different tasks available in the original sequence topics S. We set $k = 5$ which is a good balance between considering enough tasks for the pre-training and considering not too many tasks to allow an impact of model fine-tuning on our controlled settings.

For the **Direct Transfer**, we randomly select a set of three topics (metrics are then averaged), 75% of the queries are used for $\mathcal{T}_i^+$ and 25% for $\mathcal{T}_i^-$. $\mathcal{T}_j$ is a topic selected randomly.

For **Information Update**, we consider that, for persistent queries, relevant documents might evolve. To do so, we randomly select three topics $\mathcal{T}_i$. For each topic $\mathcal{T}_i$, we cluster the associated relevant documents using a constrained



(a) `IU` scenario (b) `LD` scenario

Fig. 2: Both Information Update (IU) and Language Drift (LD) scenarios. The circle of documents or queries represent the pair of documents or queries of different clusters, mapped using the closest neighborhood algorithm. This mapping is used to infer query-relevant documents of different clusters (dotted lines). Solid lines correspond to original query-relevant documents pairs. The red arrows build the training sets of tasks $\mathcal{T}'$ and $\mathcal{T}^*$ while blue arrows compose the one of tasks $\mathcal{T}''$ and $\mathcal{T}^{**}$.

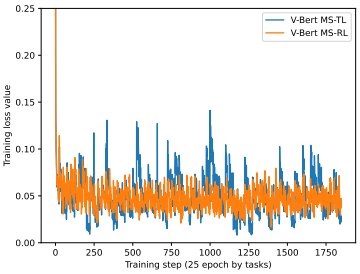
2-means algorithm[6] based on the cosine similarity metric of Sentence Bert embeddings (used in the section 4.1). We obtain two document sets D_{i1} and D_{i2}: the initial and final information distribution. Since queries in MSMarco passages have in a vast majority one relevant document[7], we can easily obtain the set of queries Q_{i1} and Q_{i2} associated to document sets D_{i1} and D_{i2} (see Figure 2(a) - solid lines being the query-document relevance pairs). To model the information update, we map documents D_{i2} relevant for queries in Q_{i2} (final distribution) to most similar documents in D_{i1} (initial distribution) in the embedding space (circles in Figure 2(a)). The task $\mathcal{T}'_i$ considers the whole set of queries Q_{i1} and Q_{i2} but only the document set D_{i1} as initial information (red arrows in Figure 2(a)). The task $\mathcal{T}''_i$ corresponds to the update of the information (namely, documents). We thus only consider the query set Q_{i2} for persistent queries with the document set D_{i2} as information update (blue arrows in Figure 2(a)). We expect that Q_{i1} performs similarly after information update if models do not suffer from catastrophic forgetting and that Q_{i2} improves its performance with the information update. We also consider the reversed setting in which we first consider D_{i2} as the initial information and then update the information with D_{i1}, Q_{i1} (persistent queries).

For the **Language Drift** scenario, we use a similar protocol by clustering queries instead of documents to obtain the sets of queries Q_{i1} and Q_{i2}, and then the associated relevant document sets D_{i1} and D_{i2}. To model the language drift in queries, we consider that one query set will change its query formulation. To do so, let consider that sets Q_{i1} and Q_{i2} reflect resp. the initial and final language distribution of same information needs, and thus, requiring same/similar relevant documents. To observe the language drift, we map pairs of queries $(q_{i1}, q_{i2}) \in Q_{i1} \times Q_{i2}$ according to their similarity in the embedding

[6] https://pypi.org/project/k-means-constrained/
[7] if not the case, we sample one document to build the query-relevant document pairs

Model	Dataset	Learning protocol		
		Random	clustering	Multi-task
VBert	SMALL	18.4/19.6	16.3/17.5	**18.5**/**19.7**
	MEDIUM	**17.9**/**19.0**	17.8/18.9	17.5/18.7
	LARGE	**18.8**/**19.9**	17.3/18.5	18.5/19.7
MonoT5	SMALL	**16.1**/**17.3**	13.1/14.4	15.5/16.8
	MEDIUM	15.4/16.7	13.4/14.7	**15.7**/**17.1**
	LARGE	13.9/15.1	13.8/15.1	**15.7**/**17.0**
BM25	SMALL		10.8/11.7	
	MEDIUM		10.5/11.4	
	LARGE		11.7/12.7	

(a) Mean performances on all the tasks reporting $mrr@10/mrr@100$ for the different models.



(b) VBert loss values for both random and clustering-based large corpus.

Fig. 3: General performance of neural ranking models on the long topic sequence.

space (circles in Figure 2(b)). Thus, we can associate documents of D_{i2} (document relevant for queries of Q_{i2}) to the query set Q_{i1}: q_{i1} has two relevant documents (d_{i1} and d_{i2}) (red arrows in Figure 2(b)). The $\mathcal{T}_i^*$ is composed of the query set Q_{i1} and the associated relevant documents belong to both D_{i1} and D_{i2} (red arrows). The $\mathcal{T}_i^{**}$ is based on the query set Q_{i2} (new language for similar information needs) associated to the relevant documents D_{i2} (blue arrows). We also consider the reversed setting in which query sets Q_{i2} and Q_{i1} are resp. used for the initial and final language.
For those two last scenarios (information update and language drift), metrics are respectively averaged over initial and reversed settings.

5 Model performance and learning behavior on long topic sequences

In this section, we report the experiments on the continual settings proposed in Section 4. We first analyze the overall retrieval performance of the different models applied on long topic sequences. We then present a fine-grained analysis of the different models with a particular focus on catastrophic forgetting regarding the similarity of topics in the sequence. Finally, we analyze specific *IR* use cases through our controlled settings.

5.1 RQ2: Performances on the MSMarco long topic sequence

We focus here on the global performance of neural ranking models after having successively been fine-tuned on topics in our MSMarco-based long sequence setting (Table 3a). For comparison, we use different sequence settings (i.e., the randomized and the topic clustering ones) of different sizes (i.e., small, medium, and large). We also run the multi-task baseline in which models are trained on all the tasks of the sequence jointly (without sequence consideration). At a first glance, we can remark that, in a large majority, neural models after fine-tuning on random sequences or multi-task learning obtain better results than after the fine-tuning on our long topic sequences. This can be explained by the fact that,

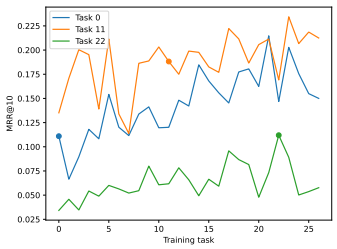
within our setting, the topic-driven sequence impacts the learning performance: a supplementary effort is needed by the model to adapt to new domains, which is not the case in the random setting. In this latter, the diversity is at the instance level. This trend is depicted in Figure 3b, highlighting peaks in the clustering-based setting (blue line) referring to topic/cluster changes. This result confirms that catastrophic forgetting might occur with neural ranking models.

5.2 Fine-grained analysis

To get a deeper understanding of model behavior, we aim here to analyze the model performance throughout the learning of the sequence. We are particularly interested in explaining the possible behavior of catastrophic forgetting according to the similarity level between tasks in the sequence. For computational reasons, we were not able to track all tasks throughout the whole sequence, we thus considered 5 randomly selected tasks (as described in Section 3.1). For each of these 5 tasks $\mathcal{T}_i$, we estimate the catastrophic forgetting using the mf score (Equation 1) regarding each task $\mathcal{T}_j$ of the sequence (with $i \neq j$). For the similarity metric, we use the $c - score$ (Equation 2) computed between both tasks $\mathcal{T}_i$ and $\mathcal{T}_j$. In Table 4a, we group together similarity by quartiles and estimate the average of the mf score for tracked tasks in each similarity quartile. We first remark that the mean similarity values of quartiles are relatively small (except the 4^{th} quartile), reinforcing the validation of our dataset building methodology. Also, we observe the following general trends. First, neural ranking models suffer from catastrophic forgetting (positive mf score), particularly the MonoT5 model. The difference in terms of model on both the global effectiveness (Figure 3(a)) and the similarity analysis suggests that MonoT5 is more sensible to new domains than the VBert model. This can also explain by the difference in the way of updating weights (suggested in the original papers [6, 31]). In VBert, two learning rates are used: a small one for the Bert model and a larger for the scorer layer; implying that the gradient descent mainly impacts the scorer. In contrast, the MonoT5 is learnt using a single learning rate leading to modify the whole model. Second, more tasks are similar (high $c-score$), less neural ranking models forget (low mf). In contrast to continual learning in other application domains [17, 34] in which fine-tuning models on other tasks always deteriorates task performance, our analysis suggests that tasks might help each other (particularly when they are relatively similar), at least in lowering the catastrophic forgetting. Moreover, as discussed in [10], relevance matching signals play an important role in the model performance, often more than semantic signals. The task sequence may lead to a synergic effect to perceive these relevance signals. Figure 4b shows the VBert performance for three tasks located at different places in the sequence (circle point). To perceive catastrophic forgetting, we look at one part of the curve after the point. One can see that task performances increase after their fine-tuning (higher increase when the task is at the beginning of the sequence), highlighting this synergic effect. In brief, continual learning in *IR* differs from usual classification/generation lifelong learning setting. It is more likely to have different tasks allowing to "help" each other, either by having closely related topics or by learning a similar structure in the query-document matching.

Model	Dataset	1^{st}	2^{nd}	3^{rd}	4^{th}
Mean Similarity by quartile	MS-S	1.4	2.6	3.9	13.8
	MS-M	1.5	2.8	4.7	15.3
VBert	MS-S	6.3	6.4	5.4	**4.6**
	MS-M	4.2	4.4	5.1	**3.8**
MonoT5	MS-S	9.2	7.0	6.5	**6.3**
	MS-M	6.5	5.3	6.0	**4.5**

(a) Mean mf score grouped by similarities between tasks (mean of 5 selected topics). The results are averaged according to quartile based on the task similarity metric. The mean value of grouped similarity are reported in the head of the table.



(b) MRR@10 results for three tasks tracked along the training sequence

Fig. 4: Fine-grained analysis of neural ranking model in the long topic sequence.

	DT scenario			B
	$\mathcal{T}_i^+$	$\mathcal{T}_j$	$\mathcal{T}_i^-$	
MonoT5	26.6	24.9	26.6	27.2
VBert	28.5	26.7	27.3	28.9

(a) MRR@10 for task $\mathcal{T}_i$ in the Direct Transfer (DT) scenario. See Section 4.3 for building $\mathcal{T}_i$ and $\mathcal{T}_j$.

		IU			LD		
		$\mathcal{T}_i'$	$\mathcal{T}_i''$	B	$\mathcal{T}_i^*$	$\mathcal{T}_i^{**}$	B
MonoT5	$Q_{i1}D_{i1}$	28.15	29.6	-	15.6	23.0	-
	$Q_{i2}D_{i2}$	7.75	26.0	-	16.8	26.5	-
	$Q_{i1}D_{i1} \cup Q_{i2}D_{i2}$	18.2	27.8	27.2	15.6	23.8	27.2
VBert	$Q_{i1}D_{i1}$	23.7	30.2	-	28.2	30.1	-
	$Q_{i2}D_{i2}$	14.5	31.4	-	25.5	25.5	-
	$Q_{i1}D_{i1} \cup Q_{i2}D_{i2}$	19.1	30.9	28.9	26.6	27.0	28.9

(b) MRR@10 for the Information Update (IU) and Language Drift (LD) scenarios. See Section 4.3 for the explanation of sets.

Fig. 5: Model performances on *IR*-driven controlled settings. B stands for the baseline.

5.3 RQ3: Behavior on *IR*-driven controlled settings

In this section we review the different scenario described in the section 4.3: **Direct Transfer** (`DT`), **Information Update** (`IU`) and **Language Drift** (`LD`). For all the different settings, we estimate the average metric of the different tracked tasks after each sequence step.

Table 5a reports the effectiveness of neural models on task $\mathcal{T}_i$ ($\mathcal{T}_i^+$ and $\mathcal{T}_i^-$ being subsets of $\mathcal{T}_i$) after each fine-tuning step in the **Direct Transfer scenario**. One can see that fine-tuning on a foreign domain ($\mathcal{T}_2$), the performance of both models on task $\mathcal{T}_i$ drop, highlighting a behavior towards catastrophic forgetting. However, both models are able to slightly adapt their retrieval performance after the fine-tuning of task $\mathcal{T}_i^-$. This final performance is however lower than the baseline model (training on both $\mathcal{T}_{init}$ and $\mathcal{T}_i$) and for the VBert model lower than its initial performance in the beginning of the learning sequence. These two last statements suggest the ability of neural models to quickly reinject a part of the retrained knowledge learnt in the early sequence to adapt to new query/document distributions in the same topic.

Table 5b reports the average effectiveness metrics for both **Information Update** (`IU`) and **Language Drift** (`LD`) scenarios on different sets, $Q_{ik}D_i$ (k=1,2) denoting the sets used to build relevant pairs of query-document (see Section 4.3). In `IU` scenario, relevant documents of certain queries (Q_{i2}) evolve over time ($D_{i1} \rightarrow D_{i2}$). For both $Q_{i1}D_{i1}$ and particularly $Q_{i2}D_{i2}$ whose queries have encountered the information update, evaluation performances increase throughout the fine-tuning process over the sequence. This denotes the ability of models to adapt to new document distributions (i.e., new information in documents). The adaptation is more important for the MonoT5 model (7.75 vs. 26.0 for the $Q_{i2}D_{i2}$ set), probably explained by its better adaptability to new tasks (as discussed in section 5.2). Interestingly, the performance at the end of the learning sequence overpasses the result of the baseline (fine-tuning on $\mathcal{T}_i$): contrary to the direct transfer scenario, this setting has introduced pseudo-relevant documents in task $\mathcal{T}'_i$ which might help in perceiving relevance signals.

For the **Language Drift** `LD` scenario, the behavior is relatively similar in terms of adaptation: performances increase throughout the sequence and MonoT5 seems more flexible in terms of adaptation. However, it seems more difficult to sufficiently acquire knowledge to reach the baseline performance (although pseudo-relevant documents have also been introduced). This might be due to the length of queries, concerned by the distribution drift: when the vocabulary changes in a short text (i.e., queries), it is more difficult to capture the semantics for the model and to adapt itself in terms of knowledge retention than when the change is carried out on long texts (i.e., documents as in the information update).

6 Conclusion and future work

In this paper, we proposed a framework for continual learning based on long topic sequences and carried out a fined-grained evaluation, observing a catastrophic forgetting metric in regards to topic similarity. We also provided specific stream of tasks, each of them addressing a likely scenario in case of *IR* continual learning. Our analysis suggests different design implications for future work: 1) catastrophic forgetting in *IR* exists but is low compared to other domains [17, 40], 2) when designing lifelong learning strategy, it is important to care of task similarity, the place of the task in the learning process and of the type of the distribution that needs to be transfered (short vs. long texts). We are aware that results are limited to the experimented models and settings and that much remains to be accomplish for more generalizable results. But, we believe that our in-depth analysis of topic similarity and the controlled settings is a step forward into the understanding of continual *IR* model learning.

Acknowledgements. We thank the ANR JCJC SESAMS project (ANR-18-CE23-0001) for supporting this work. This work was performed using HPC resources from GENCI-IDRIS (Grant 2021-101681).

References

1. Albakour, M.D., Macdonald, C., Ounis, I.: On sparsity and drift for effective real-time filtering in microblogs. In: Proceedings of the 22nd ACM International Conference on Information and Knowledge Management, CIKM '13, p. 419–428. Association for Computing Machinery, New York, NY, USA (2013). DOI 10.1145/2505515.2505709. URL https://doi.org/10.1145/2505515.2505709
2. Asghar, N., Mou, L., Selby, K.A., Pantasdo, K.D., Poupart, P., Jiang, X.: Progressive memory banks for incremental domain adaptation. In: ICLR, vol. abs/1811.00239 (2020)
3. Cai, F., Liang, S., de Rijke, M.: Time-sensitive personalized query auto-completion. In: Proceedings of the 23rd ACM International Conference on Conference on Information and Knowledge Management, CIKM '14, p. 1599–1608. Association for Computing Machinery, New York, NY, USA (2014). DOI 10.1145/2661829.2661921. URL https://doi.org/10.1145/2661829.2661921
4. Cai, H., Chen, H., Zhang, C., Song, Y., Zhao, X., Yin, D.: Adaptive parameterization for neural dialogue generation. In: EMNLP-IJCNLP, pp. 1793–1802 (2019)
5. Dai, Z., Xiong, C., Callan, J., Liu, Z.: Convolutional neural networks for soft-matching n-grams in ad-hoc search. In: WSDM, pp. 126–134 (2018)
6. Devlin, J., Chang, M., Lee, K., Toutanova, K.: BERT: pre-training of deep bidirectional transformers for language understanding. In: NAACL-HLT, pp. 4171–4186 (2019)
7. Formal, T., Piwowarski, B., Clinchant, S.: SPLADE: sparse lexical and expansion model for first stage ranking. In: F. Diaz, C. Shah, T. Suel, P. Castells, R. Jones, T. Sakai (eds.) SIGIR '21: The 44th International ACM SIGIR Conference on Research and Development in Information Retrieval, Virtual Event, Canada, July 11-15, 2021, pp. 2288–2292. ACM (2021). DOI 10.1145/3404835.3463098. URL https://doi.org/10.1145/3404835.3463098
8. Gao, J., Xiong, C., Bennett, P.: Recent advances in conversational information retrieval. In: Proceedings of the 43rd International ACM SIGIR Conference on Research and Development in Information Retrieval, SIGIR '20, p. 2421–2424. Association for Computing Machinery, New York, NY, USA (2020). DOI 10.1145/3397271.3401418. URL https://doi.org/10.1145/3397271.3401418
9. Garcia, X., Constant, N., Parikh, A.P., Firat, O.: Towards continual learning for multilingual machine translation via vocabulary substitution. In: NAACL-HLT, pp. 1184–1192 (2021)
10. Guo, J., Fan, Y., Ai, Q., Croft, W.B.: A deep relevance matching model for ad-hoc retrieval. In: CIKM, pp. 55–64 (2016)
11. Hofstätter, S., Lin, S., Yang, J., Lin, J., Hanbury, A.: Efficiently teaching an effective dense retriever with balanced topic aware sampling. In: F. Diaz, C. Shah, T. Suel, P. Castells, R. Jones, T. Sakai (eds.) SIGIR '21: The 44th International ACM SIGIR Conference on Research and Development in Information Retrieval, Virtual Event, Canada, pp. 113–122. ACM (2021)
12. Hui, K., Yates, A., Berberich, K., de Melo, G.: PACRR: A position-aware neural IR model for relevance matching. In: EMNLP, pp. 1049–1058 (2017)
13. Hui, K., Yates, A., Berberich, K., de Melo, G.: Co-pacrr: A context-aware neural IR model for ad-hoc retrieval. In: WSDM, pp. 279–287 (2018)
14. Karpukhin, V., Oguz, B., Min, S., Lewis, P.S.H., Wu, L., Edunov, S., Chen, D., Yih, W.: Dense passage retrieval for open-domain question answering. In: B. Webber, T. Cohn, Y. He, Y. Liu (eds.) Proceedings of the 2020 Conference on Em-

pirical Methods in Natural Language Processing, EMNLP 2020, Online, November 16-20, 2020, pp. 6769–6781. Association for Computational Linguistics (2020). DOI 10.18653/v1/2020.emnlp-main.550. URL https://doi.org/10.18653/v1/2020.emnlp-main.550

15. Khattab, O., Zaharia, M.: Colbert: Efficient and effective passage search via contextualized late interaction over BERT. In: SIGIR, pp. 39–48. ACM (2020)
16. Kingma, D.P., Ba, J.: Adam: A method for stochastic optimization. In: Y. Bengio, Y. LeCun (eds.) ICLR 2015 (2015)
17. Kirkpatrick, J., Pascanu, R., Rabinowitz, N.C., Veness, J., Desjardins, G., Rusu, A.A., Milan, K., Quan, J., Ramalho, T., Grabska-Barwinska, A., Hassabis, D., Clopath, C., Kumaran, D., Hadsell, R.: Overcoming catastrophic forgetting in neural networks. CoRR **abs/1612.00796** (2016). URL http://arxiv.org/abs/1612.00796
18. Lange, M.D., Aljundi, R., Masana, M., Parisot, S., Jia, X., Leonardis, A., Slabaugh, G.G., Tuytelaars, T.: Continual learning: A comparative study on how to defy forgetting in classification tasks. CoRR **abs/1909.08383** (2019). URL http://arxiv.org/abs/1909.08383
19. Lee, S.: Toward continual learning for conversational agents. CoRR **abs/1712.09943** (2017). URL http://arxiv.org/abs/1712.09943
20. Li, X., Zhou, Y., Wu, T., Socher, R., Xiong, C.: Learn to grow: A continual structure learning framework for overcoming catastrophic forgetting. In: ICML, vol. 97, pp. 3925–3934 (2019)
21. Li, Z., Hoiem, D.: Learning without forgetting. IEEE Transactions on Pattern Analysis and Machine Intelligence (12), 2935–2947 (2018)
22. Lovón-Melgarejo, J., Soulier, L., Pinel-Sauvagnat, K., Tamine, L.: Studying catastrophic forgetting in neural ranking models. In: ECIR, pp. 375–390 (2021)
23. Ma, X., dos Santos, C.N., Arnold, A.O.: Contrastive fine-tuning improves robustness for neural rankers. In: C. Zong, F. Xia, W. Li, R. Navigli (eds.) Findings of the Association for Computational Linguistics: ACL/IJCNLP 2021, Online Event, August 1-6, 2021, *Findings of ACL*, vol. ACL/IJCNLP 2021, pp. 570–582. Association for Computational Linguistics (2021). DOI 10.18653/v1/2021.findings-acl.51. URL https://doi.org/10.18653/v1/2021.findings-acl.51
24. MacAvaney, S., Yates, A., Cohan, A., Goharian, N.: CEDR: contextualized embeddings for document ranking. In: SIGIR, pp. 1101–1104 (2019)
25. de Masson d'Autume, C., Ruder, S., Kong, L., Yogatama, D.: Episodic memory in lifelong language learning. CoRR **abs/1906.01076** (2019)
26. McCreadie, R., Deveaud, R., Albakour, M., Mackie, S., Limsopatham, N., Macdonald, C., Ounis, I., Thonet, T., Dinçer, B.T.: University of glasgow at TREC 2014: Experiments with terrier in contextual suggestion, temporal summarisation and web tracks. In: E.M. Voorhees, A. Ellis (eds.) Proceedings of The Twenty-Third Text REtrieval Conference, TREC 2014, Gaithersburg, Maryland, USA, November 19-21, 2014, *NIST Special Publication*, vol. 500-308. National Institute of Standards and Technology (NIST) (2014). URL http://trec.nist.gov/pubs/trec23/papers/pro-uogTr_cs-ts-web.pdf
27. McDonald, R.T., Brokos, G., Androutsopoulos, I.: Deep relevance ranking using enhanced document-query interactions. In: EMNLP, pp. 1849–1860 (2018)
28. Mitra, B., Craswell, N.: An introduction to neural information retrieval. Found. Trends Inf. Retr. **13**(1), 1–126 (2018). DOI 10.1561/1500000061. URL https://doi.org/10.1561/1500000061

29. Nguyen, T., Rosenberg, M., Song, X., Gao, J., Tiwary, S., Majumder, R., Deng, L.: MS MARCO: A human generated machine reading comprehension dataset. In: T.R. Besold, A. Bordes, A.S. d'Avila Garcez, G. Wayne (eds.) Proceedings of the Workshop on Cognitive Computation: Integrating neural and symbolic approaches 2016 co-located with the 30th Annual Conference on Neural Information Processing Systems (NIPS 2016), Barcelona, Spain, December 9, 2016, *CEUR Workshop Proceedings*, vol. 1773. CEUR-WS.org (2016). URL http://ceur-ws.org/Vol-1773/CoCoNIPS_2016_paper9.pdf
30. Nishida, K., Saito, I., Otsuka, A., Asano, H., Tomita, J.: Retrieve-and-read: Multi-task learning of information retrieval and reading comprehension. In: Proceedings of the 27th ACM International Conference on Information and Knowledge Management, CIKM '18, p. 647–656. Association for Computing Machinery, New York, NY, USA (2018). DOI 10.1145/3269206.3271702. URL https://doi.org/10.1145/3269206.3271702
31. Nogueira, R., Jiang, Z., Pradeep, R., Lin, J.: Document ranking with a pretrained sequence-to-sequence model. In: EMNLP, pp. 708–718 (2020)
32. Onal, K.D., Zhang, Y., Altingovde, I.S., Rahman, M.M., Karagoz, P., Braylan, A., Dang, B., Chang, H., Kim, H., McNamara, Q., Angert, A., Banner, E., Khetan, V., McDonnell, T., Nguyen, A.T., Xu, D., Wallace, B.C., de Rijke, M., Lease, M.: Neural information retrieval: at the end of the early years. Inf. Retr. J. **21**(2-3), 111–182 (2018). DOI 10.1007/s10791-017-9321-y. URL https://doi.org/10.1007/s10791-017-9321-y
33. Pan, S.J., Yang, Q.: A survey on transfer learning. IEEE Transactions on Knowledge and Data Engineering **22**(10), 1345–1359 (2010)
34. Rebuffi, S., Kolesnikov, A., Sperl, G., Lampert, C.H.: icarl: Incremental classifier and representation learning. In: CVPR, pp. 5533–5542 (2017)
35. Reimers, N., Gurevych, I.: Sentence-bert: Sentence embeddings using siamese bert-networks. In: EMNLP (2019). URL http://arxiv.org/abs/1908.10084
36. Robertson, S.E., Walker, S., Hancock-Beaulieu, M., Gull, A., Lau, M.: Okapi at TREC. In: TREC, vol. 500-207, pp. 21–30 (1992)
37. Sankepally, R.: Event information retrieval from text. In: Proceedings of the 42nd International ACM SIGIR Conference on Research and Development in Information Retrieval, SIGIR'19, p. 1447. Association for Computing Machinery, New York, NY, USA (2019). DOI 10.1145/3331184.3331415. URL https://doi.org/10.1145/3331184.3331415
38. Sun, F., Ho, C., Lee, H.: LAMOL: language modeling for lifelong language learning. In: ICLR (2020)
39. Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A.N., Kaiser, L., Polosukhin, I.: Attention is all you need. In: NeurIPS, pp. 5998–6008 (2017)
40. Veniat, T., Denoyer, L., Ranzato, M.: Efficient continual learning with modular networks and task-driven priors. CoRR **abs/2012.12631** (2020). URL https://arxiv.org/abs/2012.12631
41. Wiese, G., Weissenborn, D., Neves, M.: Neural domain adaptation for biomedical question answering. In: CoNLL 2017, pp. "281–289" (2017)
42. Xiong, C., Dai, Z., Callan, J., Liu, Z., Power, R.: End-to-end neural ad-hoc ranking with kernel pooling. In: SIGIR, pp. 55–64 (2017)
43. Yang, W., Xie, Y., Tan, L., Xiong, K., Li, M., Lin, J.: Data augmentation for BERT fine-tuning in open-domain question answering. CoRR **abs/1904.06652** (2019). URL http://arxiv.org/abs/1904.06652

44. Zhao, T., Lu, X., Lee, K.: SPARTA: efficient open-domain question answering via sparse transformer matching retrieval. In: K. Toutanova, A. Rumshisky, L. Zettlemoyer, D. Hakkani-Tür, I. Beltagy, S. Bethard, R. Cotterell, T. Chakraborty, Y. Zhou (eds.) Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, NAACL-HLT 2021, Online, June 6-11, 2021, pp. 565–575. Association for Computational Linguistics (2021). DOI 10.18653/v1/2021.naacl-main.47. URL https://doi.org/10.18653/v1/2021.naacl-main.47