Improving Neural Models for the Retrieval of Relevant Passages to Geographical Queries

João Coelho
INESC-ID
Instituto Superior Técnico
University of Lisbon
Lisbon, Portugal
joao.vares.coelho@tecnico.ulisboa.pt

João Magalhães
NOVA LINCS
Universidade NOVA de Lisboa
Lisbon, Portugal
jmag@fct.unl.pt

Bruno Martins
INESC-ID & LUMLIS (Lisbon ELLIS Unit)
Instituto Superior Técnico
University of Lisbon
Lisbon, Portugal
bruno.g.martins@tecnico.ulisboa.pt

ABSTRACT

People often ask questions about places, and this is reflected on the frequency of geo-spatial queries made to information retrieval and question answering systems. Recent developments associated to these two types of systems rely on deep neural networks, specifically on methods for passage retrieval based on Transformer models, trained on large datasets like MS-MARCO. Despite significant progress in approaches for retrieving (or re-ranking) passages from a document collection according to their relevance to an input query, few studies have specifically looked at geo-spatial queries (i.e., where-questions directly concerning locations, and also questions covering other informational needs relating to places, their types, and affordances). In this work, we explore neural retrieval models in the context of geo-spatial queries, using a subset of MS-MARCO with questions and passages containing place-names. After characterizing the subset of MS-MARCO, we analyzed a re-ranking strategy based on geographic distance, which we argue to be useful for selecting hard negative examples for model training. Then, we fine-tuned neural ranking models, following bi-encoder or cross-encoder strategies, using the MS-MARCO subset together with a geographically-aware negative sampling procedure. Experimental results show that the fine-tuned models can indeed achieve a superior performance. We also describe a simple knowledge distillation procedure to further improve the computationally more efficient bi-encoder models, using the results of the cross-encoder.

CCS CONCEPTS

• **Information systems → Information retrieval**; • **Computing methodologies → Machine learning**; **Natural language processing**.

KEYWORDS

Geographic Information Retrieval, Geographical Question Answering, Passage Retrieval, Neural Language Models, Transformers

ACM Reference Format:
João Coelho, João Magalhães, and Bruno Martins. 2021. Improving Neural Models for the Retrieval of Relevant Passages to Geographical Queries. In *29th International Conference on Advances in Geographic Information Systems (SIGSPATIAL '21), November 2–5, 2021, Beijing, China.* ACM, New York, NY, USA, 10 pages. https://doi.org/10.1145/3474717.3483960

1 INTRODUCTION

Question Answering (QA), i.e. the process of computing valid answers to questions formulated in natural language, is gaining increased attention both in industry and academia. Passage retrieval is a crucial component of QA systems, concerning the identification of top-ranked passages containing the answer for a given question, from a target document collection. Recent studies in the area have proposed a variety of passage retrieval methods based on neural language models [24], using large datasets such as MS-MARCO [3] for model training and evaluation.

Geographic questions (e.g., questions relating to where a particular event has taken place) are among the most frequent on QA systems, motivating the development of tailored approaches. Geographic questions are also featured prominently in the MS-MARCO dataset [12, 13], although current models for passage retrieval are not explicitly designed to explore geo-spatial properties when deciding on the matches between questions and passages.

This work explores the use of neural retrieval models in the context of geo-spatial queries. A subset of MS-MARCO, containing geo-spatial queries and passages, was first created and characterized. The construction of this subset leveraged an existing toponym resolution system to disambiguate place-names into geo-spatial coordinates. We then analyzed a re-ranking strategy based on the geographic distance between places mentioned in the queries, and places within the passages. This distance can also be employed within a geographically-aware negative sampling procedure, to be used during the fine-tuning of neural models for retrieval.

Starting from baseline passage re-ranking models pre-trained on the full MS-MARCO dataset, and considering either bi-encoder or cross-encoder architectures, we performed a fine-tuning of the models using the geographically-aware negative sampling procedure. We specifically select negative passages that score highly in terms of BM25 and poorly in terms of geographic proximity (i.e., the passages that are lexically similar to the query, and at the same time geographically distant, are perhaps more challenging for the passage re-ranking models).

Besides model training with a standard cross-entropy loss, we also explored a knowledge distillation procedure, based on a differentiable approximation to Spearman's rank correlation coefficient [2]. The idea was to try approximating the results of fine-tuned cross-encoders with computationally more efficient bi-encoders.

SIGSPATIAL '21, November 2–5, 2021, Beijing, China

ACM ISBN 978-1-4503-8664-7/21/11.
https://doi.org/10.1145/3474717.3483960

We evaluated the passage re-ranking models on the geo-spatial subset of the MS-MARCO development set. Results showed improvements over both types of baseline models (i.e., bi-encoders and cross-encoders), when considering fine-tuning with the proposed approaches. We also evaluated the fine-tuned models on the complete MS-MARCO development set, showing that the fine-tuned cross-encoder could achieve a slightly better performance, although bi-encoders can somewhat overfit the geo-spatial queries.

The rest of this document is organized as follows: Section 2 presents related work on passage retrieval and geographic question answering. Section 3 presents the geo-spatial subset of MS-MARCO. Section 4 introduces the retrieval models and the procedures involved in their fine-tuning. Section 5 presents the experimental evaluation. Finally, Section 6 provides concluding remarks, and discusses directions for future work.

2 RELATED WORK

This section describes previous work on neural passage retrieval in general, followed by an analysis on specific methods for geographic information retrieval and question answering.

2.1 Neural Passage Retrieval

Within the field of information retrieval, neural methods currently achieve state-of-the-art results, particularly on tasks that include ranking short text passages according to relevance towards user questions [23]. Most approaches are based on the usage of Transformer-based neural language models, either following a bi-encoder or a cross-encoder architecture.

Bi-encoders (Figure 2, left) encode queries and passages independently. This allows the offline indexing of individual passage representations through methods that support the fast execution of maximum inner product searches [18]. Conversely, cross-encoders (Figure 2, right) generate a representation for the concatenation of a query and a passage, directly modelling the interactions between these two components. These representations can be obtained through mean pooling of word token embeddings, or by considering only the [CLS] token from a model like BERT [8] or RoBERTa [25]. The representation can then be used to predict a relevance score for the passage to the query, for example by using a feed-forward layer with a sigmoid activation.

Usually, due to their superior computational performance, bi-encoders are used for full retrieval, i.e., to identify the top N relevant passages from a large background collection. Cross-encoders are mostly used for re-ranking a set of top N passages, retrieved initially through an efficient first-stage method, since they provide more accurate relevance estimates.

Cross-encoders, based on BERT [8] and directly producing the representations for the concatenation of queries and passages, have been extensively used in benchmarks such as MS-MARCO (i.e., perhaps the largest publicly available dataset for information retrieval experiments). As mentioned previously, a feed-forward layer, trained with the model, is commonly used to predict a relevance score, given a representation for the concatenation of the query and the passage [30]. This approach has also been extended, for example by ensembling scores of multiple language models (e.g., ELECTRA [6], RoBERTa [25] and BERT), considering an approximation to the reciprocal rank evaluation metric as a final score [14].

Recent developments in the area include systems like RocketQA [9], which consider approaches to optimize the training of bi-encoders, so as to try to achieve results closer to those of cross-encoders. First, when training on multiple GPUs, training examples between batches are shared, so as to increase the effective training data that is used to compute model updates. Then, as this method may increase the number of easy negatives, the authors considered a hard negative sampling procedure, where passages and queries are first scored with a pre-trained cross-encoder. The instances with high results below a threshold are considered as hard negatives. Data augmentation is also employed, by using the pre-trained cross-encoder to predict relevance estimates for unlabeled passages. By applying these techniques in the training of a bi-encoder, RocketQA achieved very strong results. When used together with an ensembling technique, this strategy achieved state-of-the-art results in the MS-MARCO passage re-ranking benchmark.

2.2 Geographic Information Retrieval

Some previous studies have also addressed the difficulties and inherent problems of Geographic Information Retrieval (GIR) [32] and Geographic Question Answering (GeoQA) [27]. The types of questions GeoQA systems aim to answer are very diverse, and different types of questions need data from different sources (e.g., instead of retrieving answers from document collections, several GeoQA studies have instead relied on structured knowledge bases describing geo-spatial information [10, 21, 31]). Also, the vagueness of some geographic concepts increases the difficulty of properly answering the questions formulated by users.

Some systems have combined textual representation models with structured geographic information, so as to try to overcome some of the issues posed by geo-spatial questions. This includes seminal GIR research based on heuristics [5, 28, 32], combining BM25 ranking together with geo-spatial criteria derived from gazetteers or general knowledge bases, and also more recent GIR/GeoQA methods based on machine learning and, more recently, neural networks.

For example, a spatial-reasoner has been proposed in previous research [7], which aims to answer questions where a geographical entity is the answer. This approach works by scoring a list of candidate entities against a query. To achieve that, it uses a distance aware query encoder, where contextual representations of query tokens are extended with information regarding a candidate entity. Each query token representation is appended with an one-hot encoding representing IOB tags for geo-spatial entities contained in the query (Inside, Outside and Beginning labels, identifying spatial tokens). Then, the spatial tokens (i.e., the ones labeled with B and I) are also concatenated with the Manhattan Distance from the candidate location to the location mention lm_k to which the tokens belong, while the remaining tokens are concatenated with 0. This whole concatenation is then fed to a bi-directional recurrent neural network to generate the encodings. Another element of the architecture is a distance-reasoning layer, that aims to learn a model that can infer both whether a location mentioned is needed to be considered for answering, and how it needs to be used for answering. A final network that focus only on textual properties is also used, through a bi-encoder that combines entity embeddings and question representations to generate a relevance score. Relevance scores and scores from the distance-reasoning layer are combined for producing a final score for the answer.

Other previous approaches have considered textual and geographic features directly for model training. One example is a learning-to-rank approach for geographical information retrieval over collections of news articles [29]. Given a query-document pair, textual features were extracted, including term frequencies, inverse document frequencies, document lengths, TF-IDF scores, and BM25 scores. These features were computed for the news headline only, and for the concatenation of headline and the document body. For the geographical features, a tool to annotate the name places and encompassing geographic scopes for the documents, with their coordinates and bounding boxes, was used. Multiple features were then computed, including the area of the geographic scope of the query and the document, the area and degree of overlap between the geographic scopes of the query and the document, and the normalized distance between the centroid point for the geographic scopes of the query and the document. Other combined features were also considered, balancing geographical and textual properties. The features were used to train a SVM^{map} model [38], i.e., a listwise learning-to-rank approach which aims to optimize an approximation to the Mean Average Precision (MAP) evaluation metric. As for results, the authors performed tests on data from a previous CLEF competition [28] and noted that using both textual and geographical features yielded the best results in terms of MAP. Using only the geographical features resulted in a poor performance, while using only the textual features was a competitive retrieval baseline.

On what regards the MS-MARCO dataset, some work has been done addressing its geographic contents. For instance, answers to where-questions in this dataset were investigated, e.g. by exemplifying templates that characterize and replicate their structures [12, 13]. When doing this, machine learning approaches for pattern learning were employed, introducing an encoding of questions and answers based on the type, scale, and prominence of the toponyms. Type was defined as a reference to a group of places with similar characteristics, scale as the hierarchical organization concerning size and relations between places, and prominence as a measure of how well-known a place is. These three items are used to characterize descriptions and capture relationships among places, i.e., relating toponyms in the questions to those in their answers. The encoding process starts by modeling the questions and respective answer as a sequence of toponyms. Then, the toponyms are encoded into type, scale, and prominence (TSP). A TSP encoding of a question and its answer can be seen as a generic form. As such, questions were compared to answers through type, scale and place distributions, deriving patterns through rule mining algorithms. The generic forms are used in prediction models which, given a question's generic form, try to predict the answer's generic form. The rule mining and predictive systems were evaluated on MS-MARCO [3]. Some remarks include that the scale in the answers is, in general, one level coarser than in the questions (i.e., questions often feature city-level toponyms, while answers contain country-level ones). The extracted rules were very representative, and the most frequent rule can be applied to 1277 question-answer pairs.

3 GEO-SPATIAL QUERIES AND PASSAGES

MS-MARCO [3] is a large benchmark dataset that can be used for multiple retrieval tasks. For passage retrieval, the available data comprises over one million queries from BING and eight million passages, with human annotations concerning relevance judgements (i.e., most queries are associated to one relevant passage).

Other authors [13] have studied the geographic contents of this dataset, identifying a total of 12548 geographic question-answer pairs (under a particular definition), and 22307 different place-names mentioned in these queries and passages.

Since one of our objectives is to study the impact of geographic distances in the re-ranking of passages to geo-spatial questions, we required the association of geo-spatial coordinates to place-names within MS-MARCO queries and passages. This section starts by describing a geoparsing procedure that led to the creation of a geographic subset of MS-MARCO, where place-names are explicitly mapped to coordinates. Then, we discuss a distance metric between queries and passages, that can be used for re-ranking.

3.1 Geoparsing Textual Data

In order to consider geographic distances between queries and passages, we needed a tool to recognize and assign coordinates to geographic entities (i.e., a geoparser). Recent work on this subject relies on the usage of deep neural networks for directly predicting geo-spatial coordinates from textual representations [4, 20], although for our specific purposes we required a very efficient method that could be used to process the entire MS-MARCO dataset (i.e., all the queries and the associated top-1000 passages).

We resorted to Mordecai [11], an open source tool which is able to resolve entities to geographic coordinates. This system starts by using a pre-trained named entity recognition model to identify place-names in the input texts. Then, a large coverage gazetteer (i.e., a toponym index) is used to find the potential coordinates of the recognized place-names, by matching the place-name strings against candidate gazetteer entries. Neural networks, trained on annotated English data, are used to infer the correct country and gazetteer entry for each place name, combining heuristics based on prominence (i.e., prefer capital cities and important places) and contextual similarity.

Mordecai was used to identify place-names, and for mapping place-names to coordinates, in both queries and passages. To parse queries in the training data, and since queries are usually small, their concatenation with the relevant passage within MS-MARCO was used as input, so as to better contextualize the query and minimize errors. However, only the entities contained within the length of the query were kept. We considered a query to be geographic if at least one entity is mapped to coordinates. Passages were parsed with no extra pre-processing.

The MS-MARCO benchmark provides separate training and development sets, both with relevance judgements. The training set was parsed first with Mordecai, identifying 27104 geo-spatial queries. From those, only the 16833 queries with at least one relevant passage were considered. From the development set, 292 queries were identified as geo-spatial. A total of 292 random queries were also sampled from the training set so as to compose a separate development set (i.e., given that only the development set of

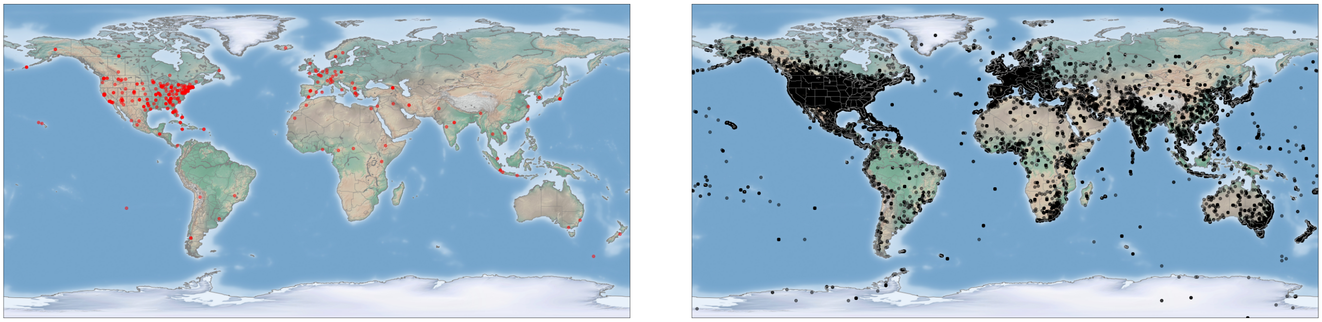

Figure 1: Geo-spatial distribution for places within queries (left) and passages (right) of the MS-MARCO development set.

MS-MARCO is available publicly, we used data from this development set as the test set to evaluate our models, and constructed a separate development set for tuning hyper-parameters). Hence, our geo-spatial subset of MS-MARCO is as follows:

- Training set: 16541 queries, together with the top-1000 passages as retrieved by BM25 for each query;
- Validation set: 292 queries, together with the top-1000 passages as retrieved by BM25 for each query;
- Test set: 292 queries, together with the top-1000 passages as retrieved by BM25 for each query;

Every query in each set has at least one relevant passage. Every passage associated with a query was also processed by Mordecai. Ultimately, we built a dataset for model training with approximately 1.3 million geographic query-passage pairs.

3.2 Distances Between Queries and Passages

Figure 1 shows the geo-spatial distribution of place-names in queries and passages within our test set, represented by their coordinates (as assigned by Mordecai) in a point map. Most of the entities are concentrated in North America and Europe, although the data has a global geo-spatial distribution.

Following the intuition that passages mentioning places that are geographically close to the places mentioned within a query should, in principle, be more relevant, we can consider a re-ranking method which leverages the distance between the set of toponyms in a query, T_q, and the set of toponyms in a passage, T_p:

$$\text{distance}(T_q, T_p) = \min_{t_q \in T_q, t_p \in T_p} \text{h}(t_q, t_p) \ . \qquad (1)$$

In the previous equation, t_q, t_p are the toponyms in the sets T_q and T_p, and $\text{h}(\cdot)$ is the haversine distance, given by:

$$\text{h}(x, y) = 2r \arcsin \sqrt{\sin^2\left(\frac{\Delta\phi}{2}\right) + \cos\phi_x \cos\phi_y \sin^2\left(\frac{\Delta\lambda}{2}\right)}, \qquad (2)$$

where r is Earth's radius, λ_x, ϕ_x and λ_y, ϕ_y are the geographical latitude and longitude of points x and y, and $\Delta\lambda$ and $\Delta\phi$ are their absolute differences.

Considering the top-1000 BM25 passages for each query, we can re-rank the passages by distance (and if distance is tied, the BM25 score can be considered). Results for this re-ranking are depicted in Table 1, and discussed latter in Section 5. The fact that this re-ranking slightly surpasses pure BM25 retrieval, in the geo-spatial subset, led us to hypothesise that geographical distances can indeed play an important role in the training of passage ranking models.

4 GEO-SPATIAL PASSAGE RE-RANKING

Our work focused on fine-tuning bi-encoders and cross-encoders, whose general architectures were described in Section 2.1, for the specific task of geo-spatial passage re-ranking. In this section, we introduce a hard negative sampling procedure, which takes the distance between geographic entities in queries and passages into consideration, also focusing on how the training batches can be built. Then, we cover the fine-tuning setup for the cross-encoders and bi-encoders. Finally, a knowledge distillation process from cross-encoders to bi-encoders is also described.

4.1 Hard Negative Sampling and Data Batches

Previous studies have reported on improved results by neural retrieval models trained with hard negative sampling [9, 36]. As such, we consider the use of a geographically-aware negative sampling procedure. For this, the minimum haversine distance between the geo-spatial entities in a query and a passage is used (Equation 1).

Given the top-25 BM25 passages for each query in the training set, the distance between the query and passages is computed. Then, to sample N hard negatives for a query, the N passages in the top-25 list provided by BM25, with highest distance, are chosen. This way, passages that are lexically similar to the query, yet geographically distant, are considered as hard negatives. An initial set of tests was used to fine-tune the value of 25, and it should be noted that, for each query, we end up using only a smaller set of sampled passages (e.g., a total of 10 negative passages) in association to each query.

When building batches, triples are first considered by associating a query with its positive passage and a negative passage, sampled as described above. Then, to maximize the effective training data, batch-wise negative pairing is applied, by using the positive and negative passages of a given query as negatives for the others. This way, for a batch of N triples, $N \times N \times 2$ pairs can be extracted.

To avoid repeated pairs, there are no duplicate queries within a batch. However, following ideas related to those addressed in other recent work [16], similar queries are grouped together in the same batch, since hard negative passages sampled for a given query are probably also challenging for similar queries. To divide the queries in groups of N, a corpus with all queries is first considered. Then, one query is sampled randomly. BM25 is used to compare all

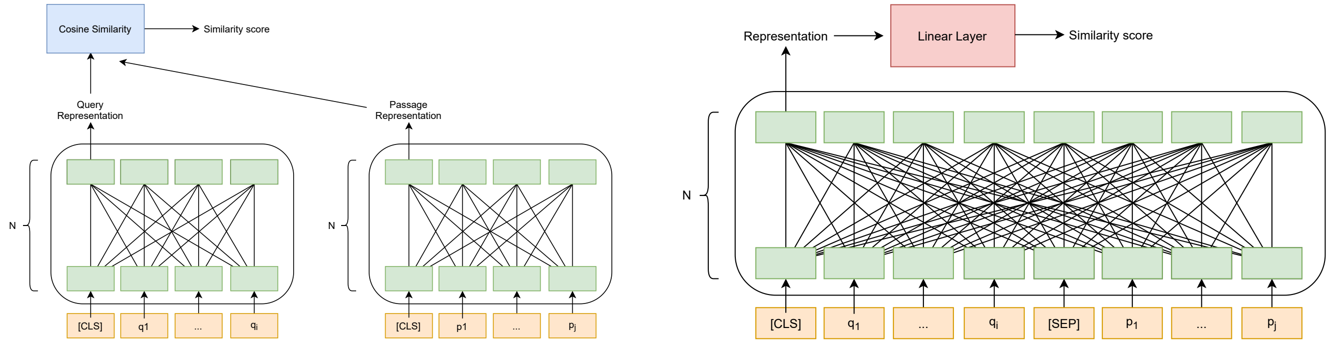

Figure 2: Architecture for bi-encoder (left) and cross-encoder (right) retrieval models.

queries in the corpus to the sampled one. The top $N - 1$ queries are extracted, building the first group. The N queries from the first group are removed from the corpus, and the process is repeated until all queries are grouped. The aforementioned procedure aims to balance easy-negatives and hard-negatives for the batch-wise paring, since the similarity of the queries in the first groups will, in principle, be higher than the similarity in the final groups.

4.2 Fine-Tuning Transformer Models for Geo-spatial Passage Re-ranking

We now describe the fine-tuning of cross-encoder and bi-encoder models based on pre-existing Transformers.

4.2.1 Cross-Encoders. The cross-encoder that we considered for fine-tuning is built on top of ELECTRA [6], and it was pre-trained on the full MS-MARCO dataset. The model is provided with the SentenceTransformers library [33]. We specifically used the model named `ms-marco-electra-base`, which was the best MS-MARCO model from this library at the time of preparing the manuscript. Batches were built as described in the previous section, considering 4 different queries per batch. This yields a total of 32 pairs being compared per batch (i.e., each of the 4 queries is matched to its positive passage, to its hard negative passage, and to the passages from the other 3 queries). Gradients were accumulated for 10 steps, and the standard binary cross-entropy was used as the loss function:

$$L_{BCE} = -\sum_{p \in P_q^+} \log(\text{score}(q,p)) - \sum_{p \in P_{q}{}^-} \log(1 - \text{score}(q,p)) \ . \quad (3)$$

In the previous equation, P_q^+, P_q^- are the sets of positive and negative passages for query q, respectively, provided within the same training batch. To obtain the $\text{score}(q,p)$, i.e, an estimate of the relevancy of passage p to query q, the representation of the [CLS] token is fed to a linear layer with a sigmoid activation.

4.2.2 Bi-Encoders. The model to be fine-tuned as a bi-encoder for passage re-ranking is based on a distilled version of RoBERTa [25]. It is also provided through the SentenceTransformers library, and pre-trained on the whole MS-MARCO dataset [33]. We specifically used the model named `msmarco-distilroberta-base-v2`, which was the best MS-MARCO bi-encoder within the SentenceTransformers library, at the time of preparing the manuscript.

The general setup in the previous section was kept the same when training bi-encoders. The only difference is on computing the relevance of passage p for query q, $\text{score}(q,p)$. Contrary to cross-encoders, the transformer model in bi-encoders receives a single sentence as input. As such, representations are generated for queries and passages independently, through mean pooling of the token embeddings. Within a batch, there are 4 queries and 8 passages (i.e., one positive and one hard negative passage selected per query). Hence, a similarity matrix $M_{4,8}$ can be built, where the value for $M_{i,j}$ is given by the cosine similarity between the representations of query i and passage j.

When inspecting the cosine similarity values, we noticed that their ranges and magnitudes were such that they did not provide a good separation between positive and negative passages. Thus, the values are normalized by first applying the softmax operation over the rows of the similarity matrix independently. Then, the columns of the matrix are also processed through a similar approach. Both values are summed, and we then divided the result by two, so that the final scores range from 0 to 1. The scores are provided to a standard binary cross-entropy loss, during model training.

4.3 Knowledge Distillation

As previously stated, the properties of bi-encoders make them computationally more efficient. However, cross-encoders are better at capturing query-passage interactions, usually performing better than bi-encoders, in terms of result quality.

Previous studies have addressed distillation processes that go from larger models to smaller versions, for example by using the outputs of the large model as targets [17, 35]. However, when distilling from a cross-encoder to a bi-encoder, trying to fit the cross-encoder's outputs directly is not optimal, since the range and magnitude of the cross-encoder scores (sigmoid of logits) differ from those of the bi-encoder (cosine similarity of vector embeddings). Approaches for cross-architecture distillation have been attempted, for example by optimizing the margin between scores [15].

In order to try to approximate the results achieved by bi-encoders to those of cross-encoders, a distillation process based on the Spearman rank correlation coefficient was used in this work, which considers the ranked lists of scores instead of the scores themselves.

Rank-based metrics are not differentiable, which means that it is not possible to use gradient-based optimizers for model training

Table 1: MRR@10 and R@{1,5,10,100, 500, 1000} for the geo-spatial subset of MS-MARCO, using the different models.

	MRR@10	R@1	R@5	R@10	R@100	R@500	R@1000
BM25	0.2560	0.1433	0.3893	0.5040	0.7797	0.8990	0.9366
BM25 + Geo. Distance Re-Rankings	0.2633	0.1518	0.4115	0.5143	0.7380	0.8887	0.9366
Base Bi-Encoder	0.4019	0.2631	0.5776	0.6792	0.8921	0.9366	0.9366
Base Bi-Encoder + Geo. Distance Re-Rankings	0.3841	0.2546	0.5451	0.6313	0.7962	0.8990	0.9366
Fine-tuned Bi-Encoder	0.4208	0.2878	0.5982	0.6832	0.8973	0.9366	0.9366
Base Cross-Encoder	0.4959	0.3333	0.6964	0.8002	0.9281	0.9366	0.9366
Base Cross-Encoder + Geo. Distance Re-Rankings	0.4652	0.3316	0.6142	0.7123	0.8116	0.8990	0.9366
Fine-tuned Cross-Encoder	0.5103	0.3607	0.6861	0.7968	0.9247	0.9366	0.9366
Distilled Bi-Encoder	0.4291	0.2997	0.5776	0.6809	0.9075	0.9366	0.9366

directly in this scenario. Nonetheless, methods for fully differentiable soft sorting and ranking have already been explored in the literature, which allows for a differentiable implementation of the Spearman rank correlation coefficient [2].

In this case, the cross-encoder scores for each pair that is used during training are pre-computed. Considering the same setup of the previous subsections, there are 4 lists of scores per batch (i.e., 4 different queries), containing the scores for each of the 8 passages. The Spearman rank correlation coefficient (r) is computed between the ranked lists of cross-encoder scores and the scores of the model under training. The average of the 4 lists is then considered and, to use this value as a loss, the following transformation is applied:

$$L_{SRC} = \frac{-r+1}{2} \ . \tag{4}$$

This loss is combined with the binary cross-entropy:

$$L = L_{SRC} + L_{BCE} \ . \tag{5}$$

5 EXPERIMENTAL RESULTS

In this section, we start by describing our experimental setup. Then, we discuss the obtained results, by evaluating the models on both the geo-spatial test set and on the full MS-MARCO development set. We also use SHAP [26] as a model explainer, to interpret and compare the fine-tuned and the base models. All the code supporting our experiments is publicly available in a GitHub repository[1].

5.1 Experimental Setup and Evaluation Metrics

For our evaluation, we consider both the full MS-MARCO development set, and the geo-spatial subset (i.e., our geo-spatial test set, described in Section 3.1). Besides evaluating geographic queries, we can access the impact of fine-tuning on geographic data over other types of queries. The top-1000 passages to be re-ranked were obtained by Pyserini [22] (i.e., a Python interface for Anserini), using a BM25 [34] approach that is tuned for MS-MARCO:

$$\text{score}_{\text{BM25}}(q,d) = \sum_{i \in q} \text{idf}(i) \times \frac{\text{tf}(i,d) \times (k_1+1)}{\text{tf}(i,d) + k_1 \times \left(1 - b + b \times \frac{|d|}{\text{avgdl}}\right)} \ . \tag{6}$$

In the previous equation, b and k_1 are hyperparameters (in this case, tuned to the values of 0.68 and 0.82, respectively), avgld is the average length of the passages in the collection, $\text{tf}(i,d)$ is the frequency of term i within passage d, and $\text{idf}(i)$ is the inverse document (i.e., passage) frequency for term i.

To re-rank passages with bi-encoders, representations are first obtained for all queries and passages independently. Then, the cosine similarities between a query and its 1000 passages are computed. For cross-encoders, each pair (query, passage) is used as input for the model, directly yielding a similarity value. In both cases, the 1000 passages are then re-ranked by similarity.

The official evaluation measure for MS-MARCO is the Mean Reciprocal Rank at the 10th passage (MRR@10), given by:

$$\text{MRR@10} = \frac{1}{|Q|} \sum_{i=1}^{|Q|} \frac{1}{\text{rank}(i,10)} \ . \tag{7}$$

In the previous equation, Q is the set of queries, and $\text{rank}(i,n)$ is a function that, given the top-n results for the i^{th} query, returns the position (rank) of the first relevant passage.

The Recall at k^{th} position (R@k) was also considered:

$$\text{R@k} = \frac{1}{|Q|} \sum_{i=1}^{|Q|} \frac{|\text{Rel}_\text{i} \cap \text{Top}_{\text{k,i}}|}{|\text{Rel}_i|} \ , \tag{8}$$

where Rel_i is the set of relevant passages for the i^{th} query, and $\text{Top}_{k,i}$ are the top k retrieved passages for the i^{th} query.

5.2 Results

5.2.1 Geo-spatial Queries. Table 1 shows the results achieved by the fine-tuned models, along those obtained from the base models, for the queries in the geo-spatial subset of MS-MARCO. The table also shows the result of a lexical BM25 baseline, together with a re-ranking based on geographic distance.

While re-ranking based on geographic distance slightly improves the BM25 results, it worsens them when re-ranking the top-1000 passages sorted by the base neural models (i.e., re-ranking passages according to distance, and use the scores from the neural models in case of ties). However, by applying this distance to re-rank passages when sampling hard negatives for fine-tuning the models, the results improved overall. This suggests that while a naive

[1] https://github.com/JMVCoelho/geo-passage-retrieval

Table 2: MRR@10 and R@{1,5,10,100, 500, 1000} for the full MS-MARCO development set, using the different models.

	MRR@10	R@1	R@5	R@10	R@100	R@500	R@1000
BM25	0.1874	0.1008	0.2944	0.3916	0.6701	0.8116	0.8573
Base Bi-Encoder	0.2839	0.1667	0.4304	0.5416	0.7957	0.8535	0.8573
Fine-tuned Bi-Encoder	0.2679	0.1519	0.4124	0.5243	0.7840	0.8526	0.8573
Base Cross-Encoder	0.3714	0.2384	0.5385	0.6431	0.8289	0.8565	0.8573
Fine-tuned Cross-Encoder	0.3776	0.2504	0.5330	0.6375	0.8270	0.8564	0.8573
Hybrid with Cross-Encoder	0.3719	0.2395	0.5381	0.6429	0.8288	0.8565	0.8573
Distilled Bi-Encoder	0.2746	0.1692	0.3985	0.5097	0.7862	0.8526	0.8573
Hybrid with Distilled Bi-Encoder	0.2857	0.1696	0.4304	0.5416	0.7963	0.8535	0.8573

Table 3: Results in MRR@10 for the full MS-MARCO development set, grouping queries by their answer type.

		Cross-Encoders		Bi-Encoders	
Query Type	Queries	Base	Fine-Tuned	Base	Fine-tuned
LOCATION	498	0.4550	0.4738	0.3538	0.3693
NUMERIC	1665	0.3690	0.3823	0.3022	0.2915
PERSON	461	0.4370	0.4365	0.2988	0.2932
DESCRIPTION	3725	0.3599	0.3607	0.2720	0.2568
ENTITY	631	0.3318	0.3461	0.2395	0.2471

use of geo-spatial distance may fail to improve over strong neural baselines, the information may be used to improve model training.

For cross-encoders, the fine-tuned version achieves a better MRR@10 when compared to the base model. Looking at the multiple recall cuts, the R@1 value also improves, while the R@{5, 10, 100} values slightly decrease. As for bi-encoders, the results on this subset also improved when compared to the base model, but this time all the recall cuts were superior. The distilled bi-encoder achieved the best bi-encoder MRR@10 result, although the performance is still far from that of the cross-encoder model. The cross-encoder behavior of improving R@1 and decreasing other cuts is also transferred in this setting to in the distilled bi-encoder.

5.2.2 General Queries. Table 2 shows the results achieved by the fine-tuned models, along those for the base models, and for a lexical BM25 baseline, for all queries in the full MS-MARCO development set. The fine-tuned cross-encoder managed to achieve a superior MRR@10 and R@1, again slightly decreasing for other cuts. However, the bi-encoders seem to overfit to the specific geographic context, as the performance of the fine-tuned models decreases in this set when compared to the base models.

We also provide the results for hybrid approaches, considering the base cross/bi-encoder when ranking non-geographic queries, and the best cross/bi-encoder otherwise. With this, we are able to achieve results that slightly surpass the base models, since the fine-tuned models are better at ranking passages for geographic queries. It is worth mentioning that the hybrid cross-encoder performed worse on the full development set when compared to the fine-tuned cross-encoder, which means that the latter is also better at some non-geographical queries than the original model. This may be due to the fact that geographical queries do not necessarily mention place-names explicitly, despite implicitly including geo-spatial criteria. The model that was fine-tuned with our procedure may be performing better on these cases.

Finally, Table 3 shows the results achieved by the base and fine-tuned models for all queries in the full MS-MARCO development set, where the queries are grouped by their original types within the dataset (e.g., a PERSON query is a query for which the answer is a person). LOCATION queries are the ones with larger variation between the base and fine-tuned models. For cross-encoders, the scores improve for all query types, except for a slight decrease in PERSON queries. For bi-encoders, the overfitting to the geo-spatial context is again noticeable, since the fine-tuned model only improves the scores for LOCATION and ENTITY queries.

5.2.3 Qualitative Analysis. To further understand the results, Table 4 presents 3 example queries and their relevant passages. For both the base and the fine-tuned cross-encoders, SHAP was used to assign shapely values to tokens, so as to depict which ones are contributing more for the ranking results.

In the examples, both models are taking the geographic entities into consideration, which is probably why the base models already achieve strong results in the geographic subset. However, the fine-tuned model concentrates the attention in those entities, while the base model distributes attention to other tokens.

Similar behavior is present in the bi-encoder models. Table 5 shows one of the examples used for the cross-encoder, where the fine-tuned bi-encoder also gives more attention to the relevant toponym when compared to the base model.

Table 6 presents the individual reciprocal rank scores for 20 example queries from the MS-MARCO development set, showing the results obtained with (a) the BM25 baseline, (b) BM25 complemented with the geo-spatial re-ranking heuristic, (c) cross-encoder models, and (d) bi-encoder models. The queries correspond to those with the highest difference in the reciprocal rank between the base models (i.e., average between the base cross-encoder and dual encoder) and the fine-tuned models. In all the example queries, the fine-tuned models performed equally or better than the base models, despite the fact that some of the queries express a relation between two locations (e.g., our models performed better on the query *how far is it from chantilly va to baltimore*, even though the proposed approach does not model spatial relations), and despite some errors in the text geo-parsing step (e.g., in the case of the query *how*

Table 4: Tokens that contribute to classification for the cross-encoder. The higher the shade of red, the higher the contribution.

	Query	Relevant Passage
Base Cross-Encoder	average winter temperature in kent co. delaware	Kent County has a moderate but distinct four-season climate. Average annual temperature: 55° Fahrenheit. January low average temperature: 26.1° Fahrenheit. July high average temperature: 87.2° Fahrenheit. The average annual rainfall is: 44.6 inches. The average annual snowfall is: 14.9 inches.
Fine-Tuned Cross-Encoder	average winter temperature in kent co. delaware	Kent County has a moderate but distinct four-season climate. Average annual temperature: 55° Fahrenheit. January low average temperature: 26.1° Fahrenheit. July high average temperature: 87.2° Fahrenheit. The average annual rainfall is: 44.6 inches. The average annual snowfall is: 14.9 inches.
Base Cross-Encoder	what county is lumberton, nc	Lumberton is a city in Robeson County, North Carolina, United States. The population has grown to 21,542 in the 2010 census from 20,795 in the 2000 census. It is the county seat of Robeson County, the largest county in the state. Lumberton, located in southern North Carolina's Inner Banks region, is located on the Lumber River.
Fine-Tuned Cross-Encoder	what county is lumberton, nc	Lumberton is a city in Robeson County, North Carolina, United States. The population has grown to 21,542 in the 2010 census from 20,795 in the 2000 census. It is the county seat of Robeson County, the largest county in the state. Lumberton, located in southern North Carolina's Inner Banks region, is located on the Lumber River
Base Cross-Encoder	what is prime rate in canada	What is the Prime Rate? In Canada, the prime rate is a guideline interest rate used by banks on loans for their most creditworthy, best, or prime clients. The prime rate rises and falls with the ebb and flow of the Canadian economy, influenced significantly by the overnight rate, which is set by the Bank of Canada.
Fine-Tuned Cross-Encoder	what is prime rate in canada	What is the Prime Rate? In canada, the prime rate is a guideline interest rate used by banks on loans for their most creditworthy, best, or prime clients. The prime rate rises and falls with the ebb and flow of the Canadian economy, influenced significantly by the overnight rate, which is set by the Bank of Canada.

much money will americans spend for easter, the word *easter* was incorrectly recognized as a reference to Easter Island on both the query and on the relevant passage, and only the fine-tuned models could place the correct passage on the top result for this query).

6 CONCLUSIONS AND FUTURE WORK

In this work, we analysed a geographic subset of the MS-MARCO passage retrieval dataset, and proposed a fine-tuning setup for geospatial passage re-ranking (i.e., for retrieving relevant passages to questions involving place names). Our model fine-tuning setup focuses on batch construction through a geographically-aware hard negative sampling procedure. We fine-tuned both bi-encoders and cross-encoders, and the results show that both types of neural ranking architectures benefit from the fine-tuning for the geographic context. Also, we explored a cross-architecture knowledge distillation method based on the Spearman rank correlation coefficient, which further improved the bi-encoder's performance.

Table 5: Tokens that contribute to classification for the bi-encoders. The higher the shade of red, the higher the contribution.

	Query	Relevant Passage
Base Bi-Encoder	what county is lumberton, nc	Lumberton is a city in Robeson County, North Carolina, United States. The population has grown to 21,542 in the 2010 census from 20,795 in the 2000 census. It is the county seat of Robeson County, the largest county in the state. Lumberton, located in southern North Carolina's Inner Banks region, is located on the Lumber River.
Distilled Bi-Encoder	what county is lumberton, nc	Lumberton is a city in Robeson County, North Carolina, United States. The population has grown to 21,542 in the 2010 census from 20,795 in the 2000 census. It is the county seat of Robeson County, the largest county in the state. Lumberton, located in southern North Carolina's Inner Banks region, is located on the Lumber River.

As for future work, the same negative sampling procedure can also be adapted for different domains. For example, temporal questions (e.g., queries focusing on when a given event has taken place) also involve an inherent distance between temporal entities [37].

When building the batches, instead of using BM25 scores when choosing the negatives, a cross-encoder can be used, so that semantic similarity is taken into account, rather than lexical similarity. Also, the geographic distance can be considered when clustering similar queries together, in an attempt to improve the batching of the data and the sharing of negatives across queries.

The knowledge distillation strategy can also be further tested, since it is domain independent. Our preliminary results were somewhat encouraging, motivating additional research.

Finally, other efficient and well-performing retrieval strategies, different from the standard bi-encoders, can also be considered. This includes approaches such as SparTerm [1] or ColBERT [19].

ACKNOWLEDGMENTS

This research was partially supported through the Fundação para a Ciência e Tecnologia (FCT), namely through the project grant with reference PTDC/CCI-CIF/32607/2017 (MIMU), as well as through the INESC-ID and NOVA LINCS multi-annual funding from the PIDDAC programme, respectively with references UIDB/50021/2020 and UIDP/04516/2020. We also gratefully acknowledge the support of NVIDIA Corporation, with the donation of the two Titan Xp GPUs used in our experiments.

REFERENCES

[1] Yang Bai, Xiaoguang Li, Gang Wang, Chaoliang Zhang, Lifeng Shang, Jun Xu, Zhaowei Wang, Fangshan Wang, and Qun Liu. 2020. SparTerm: Learning Term-based Sparse Representation for Fast Text Retrieval. *ArXiv* abs/2010.00768 (2020).

[2] Mathieu Blondel, Olivier Teboul, Quentin Berthet, and Josip Djolonga. 2020. Fast differentiable sorting and ranking. In *Proceedings of the International Conference on Machine Learning*.

[3] Daniel Fernando Campos, T. Nguyen, M. Rosenberg, Xia Song, Jianfeng Gao, Saurabh Tiwary, Rangan Majumder, L. Deng, and Bhaskar Mitra. 2016. MS MARCO: A Human Generated MAchine Reading COmprehension Dataset. In *Proceedings of the Workshop on Cognitive Computation, co-located with the Annual Conference on Neural Information Processing Systems*.

[4] Ana Cardoso, Bruno Martins, and Jacinto Estima. 2019. Using Recurrent Neural Networks for Toponym Resolution in Text. In *Proceedings of the EPIA Conference on Artificial Intelligence*.

[5] Nuno Cardoso, Bruno Martins, Marcirio Chaves, Leonardo Andrade, and Mário J Silva. 2005. The XLDB group at GeoCLEF 2005. In *Proceedings of the International Conference on Cross-Language Evalution Forum*.

[6] Kevin Clark, Minh-Thang Luong, Quoc V. Le, and Christopher D. Manning. 2020. ELECTRA: Pre-training Text Encoders as Discriminators Rather Than Generators. In *Proceedings of the International Conference on Learning Representations*.

[7] Danish Contractor, Shashank Goel, Mausam, and Parag Singla. 2020. Joint Spatio-Textual Reasoning for Answering Tourism Questions. *ArXiv* abs/2009.13613 (2020).

[8] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. In *Proceedings of the Conference of the North American Chapter of the Association for Computational Linguistics*.

[9] Yingqi Qu Yuchen Ding, Jing Liu, Kai Liu, Ruiyang Ren, Xin Zhao, Daxiang Dong, Hua Wu, and Haifeng Wang. 2020. RocketQA: An Optimized Training Approach to Dense Passage Retrieval for Open-Domain Question Answering. *ArXiv* abs/2010.08191 (2020).

[10] Carolin Haas and Stefan Riezler. 2016. A Corpus and Semantic Parser for Multilingual Natural Language Querying of OpenStreetMap. *Proceedings of the Annual Conference of the North American Chapter of the Association for Computational Linguistics*.

[11] Andrew Halterman. 2017. Mordecai: Full Text Geoparsing and Event Geocoding. *The Journal of Open Source Software* 2, 9 (2017).

[12] Ehsan Hamzei, Stephan Winter, and Martin Tomko. 2019. Initial Analysis of Simple Where-Questions and Human-Generated Answers. In *Proceedings of the International Conference on Spatial Information Theory*.

[13] Ehsan Hamzei, Stephan Winter, and Martin Tomko. 2021. Templates of generic geographic information for answering where-questions. *International Journal of Geographical Information Science* (2021).

[14] Shuguang Han, Xuanhui Wang, Mike Bendersky, and Marc Najork. 2020. Learning-to-Rank with BERT in TF-Ranking. *ArXiv* abs/2004.08476 (2020).

[15] Sebastian Hofstätter, Sophia Althammer, Michael Schröder, Mete Sertkan, and Allan Hanbury. 2020. Improving Efficient Neural Ranking Models with Cross-Architecture Knowledge Distillation. *ArXiv* abs/2010.02666 (2020).

[16] Sebastian Hofstätter, Sheng-Chieh Lin, Jheng-Hong Yang, Jimmy Lin, and Allan Hanbury. 2021. Efficiently Teaching an Effective Dense Retriever with Balanced Topic Aware Sampling. *ArXiv* abs/2104.06967 (2021).

[17] Xiaoqi Jiao, Yichun Yin, Lifeng Shang, Xin Jiang, Xiao Chen, Linlin Li, Fang Wang, and Qun Liu. 2020. TinyBERT: Distilling BERT for Natural Language Understanding. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing*.

[18] Jeff Johnson, Matthijs Douze, and Hervé Jégou. 2017. Billion-scale similarity search with GPUs. *ArXiv* abs/1702.08734 (2017).

[19] Omar Khattab and Matei Zaharia. 2020. ColBERT: Efficient and Effective Passage Search via Contextualized Late Interaction over BERT. In *Proceedings of the International ACM SIGIR Conference on Research and Development in Information Retrieval*.

Table 6: Individual Reciprocal Rank (RR) scores for the positive passages associated to the 20 test queries with the highest difference in the scores obtained by the base and the fine-tuned models (averaged scores for cross-encoders and bi-encoders).

	BM25 Baselines		Cross-Encoders		Bi-Encoders	
Query	Text Alone	Geo. Re-Ranking	Base	Fine-Tuned	Base	Fine-Tuned
hot air balloon festival in maryland	1.0000	0.0000	0.3333	1.0000	0.0000	1.0000
benefits management fairport, ny	0.1429	0.1667	0.5000	1.0000	0.2500	1.0000
how much money will americans spend for easter	0.0000	0.1429	0.2000	1.0000	0.2000	0.5000
where is way st. binghamton	0.0000	0.0000	0.5000	1.0000	0.5000	1.0000
what happened in europe as a result of the cooling in climate that occurred in the early fourteenth century	0.5000	0.0000	0.5000	1.0000	0.5000	1.0000
what is the zip code for helena mt	0.1250	0.1250	0.2500	1.0000	0.3333	0.5000
wenatchee washington population	0.1111	0.1111	0.1667	1.0000	1.0000	1.0000
what do partnerships file tax in michigan	1.0000	1.0000	1.0000	1.0000	0.1667	1.0000
which county is greenwood indiana	0.5000	0.5000	0.1600	1.0000	0.5000	0.5000
honolulu chinese new year celebration	1.0000	1.0000	1.0000	1.0000	0.2000	1.0000
population of waukesha wisconsin	0.0000	0.0000	1.0000	1.0000	0.2000	1.0000
what town in kansas is home to boot hill	0.5000	0.5000	0.2500	0.3333	0.3333	1.0000
captain of israel's host	0.0000	0.0000	0.2500	0.5000	0.5000	1.0000
what year was the masstricht treaty	0.1667	0.2500	0.5000	0.5000	0.2500	1.0000
what is the population of perryville missouri	0.0000	0.0000	0.1429	0.3333	0.5000	1.0000
how far is it from chantilly va to baltimore	1.0000	1.0000	0.3333	1.0000	1.0000	1.0000
what is the current time in lagos nigeria	0.0000	0.0000	1.0000	1.0000	0.3333	1.0000
average gas costs in kentucky	0.0000	0.0000	0.5000	0.5000	0.3333	1.0000
driving distance littleton co to ft. collins co	0.0000	0.0000	0.5000	0.5000	0.3333	1.0000
what are the best plants for connecticut gardens	0.0000	0.1667	1.0000	1.0000	0.3333	1.0000

[20] Sayali Kulkarni, Shailee Jain, Mohammad Javad Hosseini, Jason Baldridge, Eugene Ie, and Li Zhang. 2020. Spatial Language Representation with Multi-Level Geocoding. *ArXiv* arXiv:2008.09236 (2020).

[21] Carolin Lawrence and Stefan Riezler. 2016. NLMaps: A Natural Language Interface to Query OpenStreetMap. *Proceedings of the International Conference on Computational Linguistics.*

[22] Jimmy Lin, Xueguang Ma, Sheng-Chieh Lin, Jheng-Hong Yang, Ronak Pradeep, and Rodrigo Nogueira. 2021. Pyserini: An Easy-to-Use Python Toolkit to Support Replicable IR Research with Sparse and Dense Representations. *ArXiv* abs/2102.10073 (2021).

[23] Jimmy Lin, Rodrigo Nogueira, and A. Yates. 2020. Pretrained Transformers for Text Ranking: BERT and Beyond. *ArXiv* abs/2010.06467 (2020).

[24] Qi Liu, Matt J. Kusner, and P. Blunsom. 2020. A Survey on Contextual Embeddings. *ArXiv* abs/2003.07278 (2020).

[25] Y. Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, M. Lewis, Luke Zettlemoyer, and Veselin Stoyanov. 2019. RoBERTa: A Robustly Optimized BERT Pretraining Approach. *ArXiv* abs/1907.11692 (2019).

[26] Scott M Lundberg and Su-In Lee. 2017. A Unified Approach to Interpreting Model Predictions. In *Proceedings of the Annual Meeting on Neural Information Processing Systems 30.*

[27] Gengchen Mai, Krzysztof Janowicz, Rui Zhu, Ling Cai, and Ni Lao. 2021. Geographic Question Answering: Challenges, Uniqueness, Classification, and Future Directions. *ArXiv* abs/2105.09392 (2021).

[28] Thomas Mandl, Paula Carvalho, Giorgio Maria Di Nunzio, Fredric C. Gey, Ray R. Larson, Diana Santos, and Christa Womser-Hacker. 2008. GeoCLEF 2008: The CLEF 2008 Cross-Language Geographic Information Retrieval Track Overview. In *Proceedings of the Workshop of the Cross-Language Evaluation Forum.*

[29] Bruno Martins and Pável Calado. 2010. Learning to Rank for Geographic Information Retrieval. In *Proceedings of the ACM Workshop on Geographic Information Retrieval.*

[30] Rodrigo Nogueira and Kyunghyun Cho. 2019. Passage Re-ranking with BERT. *ArXiv* abs/1901.04085 (2019).

[31] Dharmen Punjani, Kuldeep Singh, Andreas Both, Manolis Koubarakis, Iosif Angelidis, Konstantina Bereta, Themis Beris, Dimitris Bilidas, Theofilos Ioannidis, Nikolaos Karalis, et al. 2018. Template-based question answering over linked geospatial data. In *Proceedings of the ACM Workshop on Geographic Information Retrieval.*

[32] Ross S. Purves, Paul D. Clough, Christopher B. Jones, Mark M. Hall, and Vanessa Murdock. 2018. Geographic Information Retrieval: Progress and Challenges in Spatial Search of Text. *Foundations and Trends in Information Retrieval* 12, 2-3 (2018).

[33] Nils Reimers and Iryna Gurevych. 2020. The Curse of Dense Low-Dimensional Information Retrieval for Large Index Sizes. *ArXiv* abs/2012.14210 (2020).

[34] Stephen Robertson and Hugo Zaragoza. 2009. The Probabilistic Relevance Framework: BM25 and Beyond. *Foundations and Trends in Information Retrieval* 3, 4 (2009).

[35] Victor Sanh, Lysandre Debut, Julien Chaumond, and Thomas Wolf. 2019. DistilBERT, a distilled version of BERT: smaller, faster, cheaper and lighter. *ArXiv* abs/1910.01108 (2019).

[36] Nandan Thakur, Nils Reimers, Johannes Daxenberger, and Iryna Gurevych. 2020. Augmented SBERT: Data Augmentation Method for Improving Bi-Encoders for Pairwise Sentence Scoring Tasks. *ArXiv* abs/2010.08240 (2020).

[37] Jiexin Wang, Adam Jatowt, and Masatoshi Yoshikawa. 2021. ArchivalQA: A Large-scale Benchmark Dataset for Open Domain Question Answering over Archival News Collections. *ArXiv* abs/2109.03438 (2021).

[38] Yisong Yue, Thomas Finley, Filip Radlinski, and Thorsten Joachims. 2007. A support vector method for optimizing average precision. In *Proceedings of the International ACM SIGIR Conference on Research and Development in Information Retrieval.*