

MGeo: Multi-Modal Geographic Language Model Pre-Training

Ruixue Ding*
Boli Chen*
ada.drx@alibaba-inc.com
boli.cbl@alibaba-inc.com
Damo Academy,
Alibaba Group,
Hangzhou, China

Pengjun Xie
Fei Huang
chengchen.xpj@alibaba-inc.com
f.huang@alibaba-inc.com
Damo Academy,
Alibaba Group,
Hangzhou, China

Xin Li
Qiang Zhang
Yao Xu
beilai.bl@alibaba-inc.com
muxi.zq@alibaba-inc.com
xuenuo.xy@alibaba-inc.com
Gaode Map,
Alibaba Group,
Beijing, China

ABSTRACT

Query and point of interest (POI) matching is a core task in location-based services (LBS), e.g., navigation maps. It connects users' intent with real-world geographic information. Lately, pre-trained language models (PLMs) have made notable advancements in many natural language processing (NLP) tasks. To overcome the limitation that generic PLMs lack geographic knowledge for query-POI matching, related literature attempts to employ continued pre-training based on domain-specific corpus. However, a query generally describes the *geographic context* (GC) about its destination and contains mentions of multiple geographic objects like nearby roads and regions of interest (ROIs). These diverse geographic objects and their correlations are pivotal to retrieving the most relevant POI. Text-based single-modal PLMs can barely make use of the important GC and are therefore limited. In this work, we propose a novel method for query-POI matching, namely Multi-modal Geographic language model (MGeo), which comprises a geographic encoder and a multi-modal interaction module. Representing GC as a new modality, MGeo is able to fully extract multi-modal correlations to perform accurate query-POI matching. Moreover, there exists no publicly available query-POI matching benchmark. Intending to facilitate further research, we build a new open-source large-scale benchmark for this topic, *i.e.*, Geographic Textual Similarity (GeoTES). The POIs come from an open-source geographic information system (GIS) and the queries are manually generated by annotators to prevent privacy issues. Compared with several strong baselines, the extensive experiment results and detailed ablation analyses demonstrate that our proposed multi-modal geographic pre-training method can significantly improve the query-POI matching capability of PLMs with or without users' locations. Our code and benchmark are publicly available at <https://github.com/PhantomGrapes/MGeo>.

*Both authors contributed equally to this research.

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. Copyrights for components of this work owned by others than the author(s) must be honored. Abstracting with credit is permitted. To copy otherwise, or republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee. Request permissions from permissions@acm.org.

SIGIR '23, July 23–27, 2023, Taipei, Taiwan

© 2023 Copyright held by the owner/author(s). Publication rights licensed to ACM.
ACM ISBN 978-1-4503-9408-6/23/07...\$15.00
<https://doi.org/10.1145/3539618.3591728>

CCS CONCEPTS

• **Information systems** → **Language models**; *Similarity measures*; *Business intelligence*.

KEYWORDS

query-POI matching, multi-modal, language model, geographic context, benchmark

ACM Reference Format:

Ruixue Ding, Boli Chen, Pengjun Xie, Fei Huang, Xin Li, Qiang Zhang, and Yao Xu. 2023. MGeo: Multi-Modal Geographic Language Model Pre-Training. In *Proceedings of the 46th International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR '23)*, July 23–27, 2023, Taipei, Taiwan. ACM, New York, NY, USA, 10 pages. <https://doi.org/10.1145/3539618.3591728>

1 INTRODUCTION

As an essential function of location-based services (LBS) like navigation maps (e.g., Google Maps), ride-hailing applications (e.g., Uber), and food delivery platforms (e.g., Uber Eats), query and point of interest (POI) matching aims to find a list of candidate POIs based on users' specific or implicit intent. The candidate results are crucial for providing users with real-world geographic information, which directly impacts the navigation, routing, and ordering process. Therefore, effective and accurate query-POI matching is indispensable for delivering a satisfactory user experience. A typical query-POI matching procedure is illustrated in Figure 1, which consists of a two-stage retrieve-then-rank pipeline [36, 38]. In specific, given a query, the lightweight retriever first produces an initial set of candidate POIs by searching a massive database, then the ranker sorts the most relevant candidate. This kind of architecture is widely adopted in information retrieval (IR) systems on account of the efficiency-effectiveness trade-off.

Recent literature on natural language processing (NLP) as well as IR shows a flourishing advancement of pre-trained language models (PLMs), notably in semantic textual similarity (STS) and open-domain question answering (QA) [3, 14, 17]. Continued self-supervised training on domain-specific corpus is shown to be effective for adapting generic PLMs to other domains [9]. To improve the capability of PLMs for tasks in LBS, various methods have lately been proposed to inject geographic knowledge based on textual data related to geography and user behavioral data [10, 11, 20, 29]. Although these methods are better at capturing semantic similarity than generic PLMs for query-POI matching, they can barely make

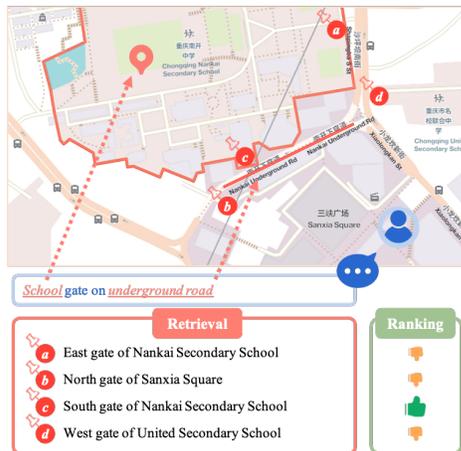


Figure 1: A typical query-POI matching procedure.

use of the more important circumstantial *geographic context* (GC), *i.e.*, the diverse geographic objects and their correlations from the geographic information system (GIS) (detailed in DEFINITION 2). Specifically, the geographic objects consist of roads represented as *lines* and regions of interest (ROIs) represented as *polygons*, the correlations include *near*, *covered*, and their *relative position*.

A query usually mentions multiple geographic objects in the background of the target POI. Fully capturing the information in the GC is necessary for accurate query-POI matching. For example, given the query "school gate on underground road", as shown in Figure 1, several relevant POIs are retrieved. The nearest "underground road" to the user is the "Nankai Underground Rd", and the "Nankai Secondary School" has a gate (c) on the "Underground Rd". Therefore, the most matched POI should be the gate (c). The problem is that the "Nankai Secondary School" is formally located on the "Shapingba S St" with its main gate (a). Its side gate (c) is not recorded in the GIS as located on the "Underground Rd". It should also be noticed that the user is currently in the "Sanxia Square", which has a gate (b) located on the "Underground Rd". The semantic textual similarity alone is not enough to distinguish these two hard negatives (a) and (b). Moreover, the gate (d) of the "United Secondary School" is the closest school gate to the user. Simply considering the relative position of the user and the POI will match the wrong gate (d). Only by taking the entire GC into consideration can we find the correct gate (c).

To this end, we propose a novel method that draws on GC for query-POI matching, namely Multi-modal Geographic language model (MGeo). MGeo bridges the modality gap between semantics and GC. MGeo consists of a geographic encoder and a multi-modal interaction module. The geographic encoder makes use of the GC by representing it as a new modality. The multi-modal interaction module then incorporates the geographic features with the semantics. MGeo makes use of the textual, geographic, and cross-modal interactions between queries and POIs. Since the interaction module is compatible with queries that have no GC, it is optional to provide the users' locations, as many applications may require. As a result, rich correlations among textual and geographic modalities can be fully extracted to ensure the quality of query-POI matching.

In addition, there is no public unencrypted benchmark for query-POI matching mostly due to privacy issues. Large publicly available corpus could lead to many breakthroughs in research, *e.g.*, MS MARCO [22]. Intending to facilitate further research on this topic, develop robust techniques, and track progress, we introduce Geographic TEXTual Similarity (GeoTES), which is an open-source large-scale benchmark for query-POI matching with GC (detailed in Section 4). The POIs come from the open-source GIS OpenStreetMap (OSM)¹. To prevent privacy issues, the queries are manually generated by annotators thus do not require encryption.

Our major contributions are highlighted as follows:

- We formalize the important concept GC for the query-POI matching problem and propose a novel method MGeo that uses geographic encoder to represent it as a new modality.
- A multi-modal interaction module is proposed to incorporate the correlations among textual and geographic modalities. It is compatible with queries that have no GC as well.
- A new open-source large-scale benchmark GeoTES is built to facilitate further research. The POIs come from an open-source GIS and the queries are manually generated by annotators to prevent privacy issues.
- Compared with strong baselines, the experiment results demonstrate that our proposed methods can significantly improve the query-POI matching capability of PLMs, even when no GC is provided for the queries.

2 RELATED WORK

2.1 Relevance Model

Traditional approaches for retrieving documents from large corpus generally use exact term-level matching, *e.g.*, Okapi Best Matching (BM25) [27]. Despite such heuristic retrievers having low latency via inverted list data structure, their measurement of similarity is only based on document statistics. Latterly, Deep neural network (DNN) models have been introduced to IR. For example, Deep Structured Semantic Model (DSSM) [12] measures the relevance of queries and documents in a semantic vector space by computing their cosine similarity. Along with the success of PLMs in NLP, studies on IR have also made remarkable progress [7, 14, 17]. On account of the efficiency-effectiveness trade-off, there are two major architectures, *i.e.*, bi-encoder and cross-encoder [31]. Bi-encoder allows efficient indexing [4, 26] and is usually used in the retrieval system. In contrast, cross-encoder concatenates the query and document to perform cross-interaction over all input terms. Although cross-encoder can provide a more accurate estimation of relevance, it needs more computing resources and is usually used only in the ranking system. MGeo can be applied on both bi-encoder and cross-encoder architecture.

2.2 Multi-Modal Representation Learning

Following the tremendous success of various pre-training techniques in NLP, a lot of Transformer-based models are proposed for other modalities, such as compute vision (CV) [1, 6, 16]. Except for single-modal, recent studies also show the derivative models have great potential in multi-modal representation learning [2, 16, 18, 28].

¹<https://www.openstreetmap.org>

For example, CLIP [25] converts classification to a retrieval task and enables zero-shot learning via large-scale multi-modal pre-training. In addition to image, layout of document and table can also be represented as different modalities [32, 35]. In this paper, our proposed MGeo use the geographic encoder to represents GC as a new modality for query-POI matching.

2.3 Query-POI Matching

Previous work on query-POI matching generally focuses on modeling the relative position between queries and POIs. Based on DSSM, PALM [38] obtains the positional relationship of queries and POIs from coordinate-based and kernel-based location embeddings, and incorporates the relationship with semantic similarity for POI retrieval. STDGAT [37] further takes multiple spatiotemporal factors into consideration via dual graph attention network when quantifying the query-POI relevance. On account of the ubiquity of PLMs in NLP, domain-adaptive pre-training methods have been proposed to inject extralinguistic knowledge into the generic PLMs [10, 20]. Typically, GeoL [11] makes use of the static geographic knowledge based on user behavior (search logs), e.g., geocoding [8]. Although the domain-adapted PLMs may be better at capturing the semantic similarity than generic PLMs for query-POI matching, they are still limited by ignoring GC in the background.

Moreover, to facilitate further research and promote the development of robust techniques, we also establish a reliable public large-scale query-POI matching benchmark named GeoTES.

3 PRELIMINARY

We first introduce the formal description of the query-POI matching problem, as well as some important definitions related to GC. Table 1 gives the frequently used notations.

Let P be the set of POIs p . P can either contain dozens of candidate POIs or a large number of POIs in the massive database. Each POI p consists of a textual description t^p and its geolocation l^p . The textual description of the POI t^p contains its formal address and name. Let q denote a query made by the user. The textual description of the query t^q belongs to three types, i.e., common *address* description, formal *street number* description, and casual *colloquial* description. The *street number* query contains standard numerical designation for a target POI, while the *address* query does not. The *colloquial* query uses spoken language and may contain colloquial noisy words. The query’s geolocation l^q can be the users’ geolocation. When the user searches for another area using the map, l^q is the center location displayed on screen. Furthermore, l^q may or may not be provided. We denote geolocation of a POI or query as l^{pq} .

PROBLEM 1. Query-POI matching problem. *Given the POI set P and a user’s query q in LBS, we aim to estimate the POI $p \in P$ that best matches the user’s intent.*

We define two tasks based on the size of P , i.e., *ranking* and *retrieval*. Specifically, for the ranking task, P is a list of candidate POIs with a limited number, where the best-matched one is included. As for the retrieval task, P is the massive database that contains all POIs, and the total number of POIs is large. Since cross-encoder is inefficient for large size of P , it only runs on the ranking task. Bi-encoder can run on both the ranking and retrieval tasks.

Table 1: Table of notations.

Notation	Description
P, p	The POI set and a POI.
q	A query given by the user.
o	A geographic object.
o^s	The shape of geographic object, $\in \{LINE, POLYGON\}$.
o^m	The position of o in the map.
r^t	The relation type $\in \{NEAR, COVERED\}$.
r^p	The relative position.
t	The textual description of POI or query.
$l = (lng, lat)$	The geolocation represented by longitude and latitude.
l^{pq}	The geolocation of a POI or query.
l^o	A vertex of o .
\tilde{o}	The rectangle that approximates the shape of o .

DEFINITION 1. Geographic object. *GIS is constructed on spatial data that defines the real-world geometric space. Let \mathcal{G} be the spatial database. Each geographic object $o \in \mathcal{G}$ with m vertices is described as a sequence of geolocation $\{l_1^o, l_2^o, \dots, l_m^o\}$. A geographic object is intrinsically characterized by its ID, absolute position in the map, and shape $o^s \in \{LINE, POLYGON\}$. Specifically, *LINE* represents the real-world road and *POLYGON* represents the ROI.*

Here we use m to denote the number of vertices in o . Note that given the geolocation of the POI or the query, we can form a list of nearby geographic objects $\{o_1, o_2, \dots, o_n\}$ sorted by distance, i.e., o_1 is the nearest geographic object to the POI or query. n is used to denote the number of geographic objects for a geolocation l^{pq} . We export OSM to PostGIS² and get the Geographic Context (GC) of a geolocation from it.

DEFINITION 2. Geographic context (GC). *Given the geolocation l^{pq} of a POI or query, where l^{pq} is represented by a geographic coordinate (lng, lat) , the GC is characterized by the correlations between l^{pq} and its n geographic objects $\{o_1, o_2, \dots, o_n\}$. Formally, the relation type $r^t \in \{NEAR, COVERED\}$ indicates whether l^{pq} is inside o_i or at a distance. The relative position r^p depicts a more detailed positional relationship between l^{pq} and o_i .*

When searching for a target POI, a user usually explores the nearby circumstantial spatial data and mentions multiple related geographic objects in the query. The intrinsic characteristics of geographic objects are also important to extract GC information. Therefore, modeling the intrinsic characteristics of geographic objects is pivotal to capturing correlations in GC and ensuring the quality of query-POI matching. The methods used to encode GC are detailed in Section 5.1.1.

4 THE GEOTES BENCHMARK

In this section, we introduce our proposed large-scale benchmark GeoTES, which stands for Geographic TExtual Similarity. It is the first open-source benchmark for query-POI matching. The POIs are obtained from the open-source OSM and the queries are manually generated by annotators to prevent privacy issues.

In this version of GeoTES, all the POIs are located in Hangzhou and use Chinese text. Table 2 give examples from the benchmark. Since geographic objects in GIS and GC are language-agnostic, MGeo can be easily applied to multilingual situations. Section 5.1.1

²<https://postgis.net/>

Table 2: Examples from GeoTES. We only show the positive POI and one geographic object here for simplicity. Every query can have multiple negative POIs similar to the positive POI. Every query and candidate POI can have GC of multiple related geographic objects. Note that query GC is optional to simulate the absence of user location.

		Query	Positive POI
Text		滨海大厦对面江汉路	6号线江汉路(地铁站)
		Jiangnan Road opposite to Binhai Building	Line 6 Jiangnan Road (subway station)
Location (Longitude, Latitude)		(120.20435566081441, 30.210982121527547)	(120.20039749738959, 30.20525878443309)
Geographic Object	ID	41935	42599
	Shape	<i>POLYGON</i>	<i>LINE</i>
	Map Position	[1324, 1341, 1325, 1342]	[1322, 1335, 1323, 1336]
Geographic Context	Relation	<i>NEAR</i>	<i>COVERED</i>
	Relative Position	[31, 24, 22, 22]	[27, 27, 18, 19]

Table 3: Statistics of different query types.

Query Type	# Query
Address	81,286
Street No.	6,013
Colloquial	2,701
Total	90,000

Table 4: Statistics of geographic objects, which describe the average number of shapes with corresponding relation to queries and POIs.

	<i>LINE</i>		<i>POLYGON</i>	
	<i>NEAR</i>	<i>COVERED</i>	<i>NEAR</i>	<i>COVERED</i>
we Query	4.4	0.005	14.2	0.7
POI	3.7	0.003	10.4	0.6

details how to obtain map position and relative position. A query is equipped with a positive POI, and negative POIs with a limited number are provided for the ranking task.³

4.1 Annotation Process

We recruited 20 annotators and 4 experienced experts to annotate three types of queries based on POIs defined in Section 3. Table 3 gives the statistics of these query types, which follows the distribution of our online LBS. In OSM, each POI comes with a geographic location under the WGS84 coordinate system.⁴ Neighbouring POIs of the OSM POIs from several open-accessed map services are selected by the annotators to enrich the diversity of POI description and also serve as hard negatives. To simulate the queries' location in real scenes, the annotators are asked to randomly select a location within 1km of corresponding POI for 50% of the queries and randomly select a location in the city for the rest queries. All the annotators have adequate linguistic knowledge and educational/cultural background to produce appropriate queries. To eliminate biases during the annotation process, they are instructed with detailed

³Benchmark is available at <https://modelscope.cn/datasets/damo/GeoGLUE>.

⁴https://wiki.openstreetmap.org/wiki/Converting_to_WGS84

Table 5: Statistics of train/dev/test splits.

	# Query	# Candidate POI
Train	50,000	20
Dev	20,000	40
Test (Ranking)	20,000	40
Test (Retrieval)		2,849,754

annotation principles. A quality inspector verifies the annotations and confirms that each query has exactly one positive POI.

4.2 Benchmark Statistics

GeoTES has a total number of 90,000 queries with an average length of 17.2 and 2,849,754 POIs with an average length of 13.7. We extract the geographic surrounding objects for the queries and POIs from OSM. There are 21,950 lines and 65,722 polygons in our extracted geographic objects. Table 4 gives the averaged number of different shapes and relations, each query and POI has more *NEAR* relation and more relations to polygons (ROIs) than lines (roads). As shown in Table 5, the benchmark is randomly split in to train, development, and test sets. For the train, development, and ranking test sets, we provide a list of candidate POIs and ensure that one exact matched positive POI is contained. The retrieval test set use the same queries as the ranking test set while no candidate POI list should be provided. GeoTES is thus a reliable and challenging benchmark for evaluating both retrieval and ranking models.

5 METHODS

In this section, we present the detailed architecture and pre-training process of MGeo. Following state-of-the-art multi-modal methods [2, 16, 19], MGeo is composed of a geographic encoder and a multi-modal interaction module, as shown in Figure 2. The full training process of MGeo consists of three steps: (1) we first train geographic encoder alone to learn representations of GC, and the trained geographic encoder is fixed in the following stages; (2) text-GC pairs are then used to pre-train MGeo in a multi-modal way, by modeling geographic objects along with text and pre-training with massive text-GC pairs, and MGeo successfully aligns these

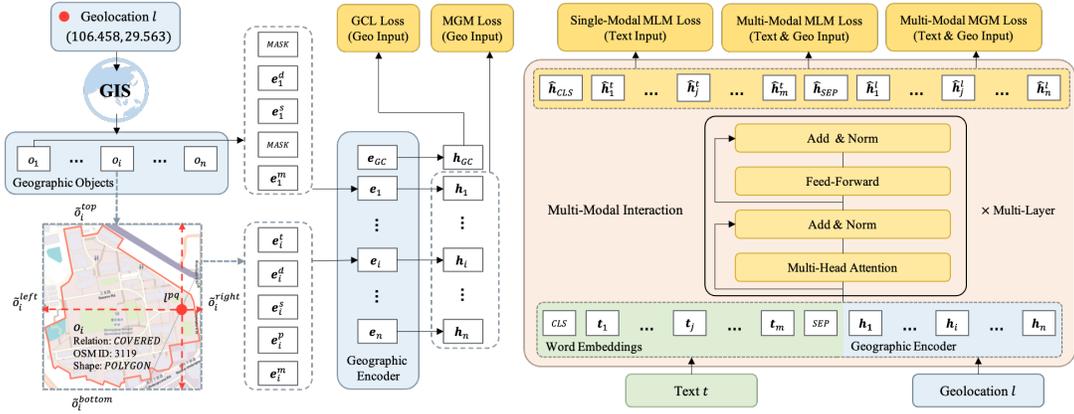


Figure 2: Architecture of MGeo. Left part shows encoding and pre-training process of geographic encoder and right part shows the multi-modal pre-training process of MGeo. Word embeddings of text t and GC representations of geographic encoder h are concatenated together and fed to multi-modal interaction module, which produces final representations h^t for each text token and h^l for each geographic object.

two modals into a same latent space; (3) MGeo is lastly fine-tuned on ranking and retrieval tasks and gains significant improvements.

5.1 Geographic Encoder

The geolocation alone is meaningless unless it has GC. Taking a geolocation l as input, geographic encoder maps the GC as a new modality to dense representations, which contains features of the surrounding geographic objects $\{o_1, o_2, \dots, o_n\}$.

5.1.1 Encoding. Geographic encoder can extract the correlations between query/POI geolocation (point) and their surrounding geographic objects (line or polygon). Geographic encoder respectively represents the intrinsic characteristics of geographic objects (*i.e.*, ID, shape, and map position), the relations (*i.e.*, *NEAR* or *COVERED*), and the relative position as embeddings.

ID. To extract the intrinsic features of geographic objects, the OSM IDs are mapped to embeddings in a similar way to word embeddings. The ID embeddings of o_i are denoted as e_i^d .

Shape. A one-hot function is used to encode the categorical shape type o_i^s as a numeric array and to obtain its corresponding embeddings e_i^s . The shape type embeddings are denoted as e_i^s .

Map position. The absolute position of o_i in the map e_i^m is pivotal to distinguishing itself from other geographic objects. The **entire** map area as a rectangle is split into a $N \times N$ grid to obtain its scale factors s_{lng} and s_{lat} for longitude and latitude respectively:

$$s_{lng} = \frac{lng^{m_{right}} - lng^{m_{left}}}{N}, s_{lat} = \frac{lat^{m_{top}} - lat^{m_{bottom}}}{N}, \quad (1)$$

where $lng^{m_{right}}$ denotes longitude of the map's right side and so on. The position of o_i in the map o_i^m can thus be calculate with the scale factors. For example, $o_i^{m_{left}}$ and $o_i^{m_{bottom}}$ are calculated as:

$$o_i^{m_{left}} = \lfloor \frac{\tilde{o}_i^{left} - lng^{m_{left}}}{s_{lng}} \rfloor \in \mathbb{N}, \quad (2)$$

$$o_i^{m_{bottom}} = \lfloor \frac{\tilde{o}_i^{bottom} - lat^{m_{bottom}}}{s_{lat}} \rfloor \in \mathbb{N}. \quad (3)$$

The discretized position feature of o_i in the map is then encoded as $e_i^m = \{e_i^{m_{left}}, e_i^{m_{bottom}}, e_i^{m_{right}}, e_i^{m_{top}}\}$.

Relation. Similar to the shape type, a one-hot function is used to encode the relation type r_i^t of o_i as a numeric array and to obtain its corresponding embeddings e_i^t .

Relative Position. To simplify the relative position r_i^p , we form a rectangle \tilde{o}_i of similar size to approximate the shape of o_i . Each side of the substituted rectangle (left, bottom, right, and top) is defined as:

$$\tilde{o}_i^{left} = \min(\{lng_j^{o_i}\}_{j \in \{1, \dots, m_i\}}), \quad (4)$$

$$\tilde{o}_i^{bottom} = \min(\{lat_j^{o_i}\}_{j \in \{1, \dots, m_i\}}), \quad (5)$$

$$\tilde{o}_i^{right} = \max(\{lng_j^{o_i}\}_{j \in \{1, \dots, m_i\}}), \quad (6)$$

$$\tilde{o}_i^{top} = \max(\{lat_j^{o_i}\}_{j \in \{1, \dots, m_i\}}), \quad (7)$$

where lng denotes longitude and lat denotes latitude of $l_j^{o_i}$ for simplicity. The relative position $r_i^p = \{r_i^{p_{left}}, r_i^{p_{bottom}}, r_i^{p_{right}}, r_i^{p_{top}}\}$ is then calculated by the normalized distances between l^{pq} and each side of the \tilde{o}_i . For example, $r_i^{p_{left}}$ is calculated as:

$$r_i^{p_{left}} = \text{sgn}(lng^{pq} - \tilde{o}_i^{left}) * \min(k, \lfloor k \frac{|lng^{pq} - \tilde{o}_i^{left}|}{\tilde{o}_i^{right} - \tilde{o}_i^{left}} \rfloor \rfloor) + k, \quad (8)$$

where $\text{sgn}(\cdot)$ is the sign function, and $\lfloor \cdot \rfloor$ is the floor function that outputs the greatest integer less than or equal to a number. $k \in \mathbb{N}$ is a discretization factor that maps the relative distance ratio to a discrete number. As a result, we have $r_i^{p_{left}} \in \{0, \dots, 2k\}$. The discretized relative position feature is then encoded as $e_i^p = \{e_i^{p_{left}}, e_i^{p_{bottom}}, e_i^{p_{right}}, e_i^{p_{top}}\}$.

Finally, the geographic encoder sums these features of o_i up as:

$$e_i = e_i^d + e_i^s + \sum e_i^m + e_i^t + \sum e_i^p \quad (9)$$

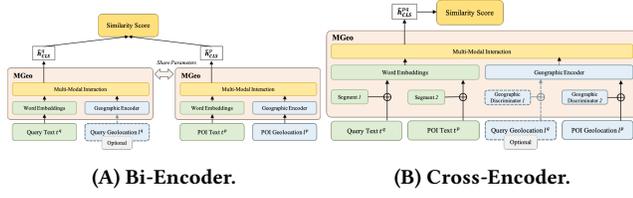


Figure 3: MGeo can use both (A) bi-encoder and (B) cross-encoder architectures to measure relevance between query and POI. Dashed line indicates that geolocation of query is optional. \oplus denotes element-wise addition.

The intrinsic characteristics of geographic objects are described by the three components (e^d , e^s , and e^m). e^d is the unique identifier of a geographic object, e^s distinguishes road from ROI, e^m depicts the positional relation among different geographic objects. The other two components (e^t and e^p) describe correlations between geolocation and geographic objects. After encoding surrounding geographic objects as a sequence $\{e_1, \dots, e_m\}$, geographic encoder employs multi-layer bidirectional transformers [33] to learn interactions among them. Following previous work [31], a *GC* token is prepended at the beginning like the *CLS* token. The outputs of geographic encoder are therefore denoted as $\{h_{GC}, h_1, \dots, h_m\}$.

5.1.2 Training. We design two tasks to train geographic encoder and it is fixed in later uses, *i.e.*, masked geographic modeling (MGM) and geographic contrastive learning (GCL).

MGM. Like the widely use masked language modeling (MLM) [5], MGM aims at predicting masked geographic features, *i.e.*, OSM IDs, geometric types, each side of the substituted rectangle, relation types, and relative positions. The MGM loss L_{MGM} is calculated by summing up the masked loss of all features.

GCL. This task is related to multiple geolocations $\{l_1^{pq}, \dots, l_{bs}^{pq}\}$ in a batch of size bs . We begin with the definition of the real-world geographic distance matrix $\mathbf{H} \in \mathbb{R}^{bs \times bs}$ defined as:

$$\mathbf{H}_{i,j} = \sigma(-\|haversine(l_i^{pq}, l_j^{pq})\|_{\mathcal{N}}), i, j \in \{1, \dots, bs\}, i \neq j, \quad (10)$$

where *haversine* is the haversine function [23] that calculates spherical distance between geolocations, $\|\cdot\|_{\mathcal{N}}$ is gaussian normalization function, and σ is sigmoid function that maps distance to range (0, 1). As the latent distance between embeddings in the output space should correspond to their real-world geographic distance, we use h_{GC} as the representation of geolocation l^{pq} with GC and calculate the latent distance matrix $\tilde{\mathbf{H}} \in \mathbb{R}^{bs \times bs}$ as:

$$\tilde{\mathbf{H}}_{i,j} = \langle \|h_{GC}^i\|_{L^2}, \|h_{GC}^j\|_{L^2} \rangle \quad (11)$$

where $\langle \cdot \rangle$ denotes the dot-product function and $\|\cdot\|_{L^2}$ is L^2 normalization function. We use KL-divergence to measure the similarity between \mathbf{H} and $\tilde{\mathbf{H}}$. GCL loss L_{GCL} is then calculated by:

$$L_{GCL} = \sum_{i=1}^{bs} D_{KL}(\text{softmax}(\mathbf{H}_i) \parallel \text{softmax}(\tilde{\mathbf{H}}_i)) \quad (12)$$

where $D_{KL}(\cdot \parallel \cdot)$ denotes the KL-divergence, and the *softmax* function is applied to transform \mathbf{H}_i and $\tilde{\mathbf{H}}_i$ to a distribution.

The training loss L_g of geographic encoder is thus calculated by:

$$L_g = L_{MGM} + L_{GCL} \quad (13)$$

Using such an training process, geographic encoder is capable of modeling GC in a given GIS.

5.2 Multi-Modal Pre-Training

The input of MGeo pre-training is a pair of text and geolocation (t, l). The pre-training data can come from diverse sources, *e.g.*, click of users or position of delivery clerks. The multi-modal training aims at aligning these two modals into one latent space. Word embeddings are used to map text into a sequence of vectors. The geographic encoder provides the GC embeddings given l . The two embeddings are then concatenated together and fed into multi-layer bidirectional Transformers.

We use three tasks to learn interaction between GC and text, *i.e.*, single-modal MLM, multi-modal MLM, and multi-modal MGM. These tasks are trained in turns. Single-modal MLM is the original MLM task used in BERT, which randomly masks and replaces the input text with *MASK* token. The outputs of geographic encoder are removed for single-modal MLM. While multi-modal MLM predicts the masked token relying on the entire GC and part of textual information. Multi-modal MGM randomly masks and replaces the input geographic features with *MASK* and predicts them relying on entire textual information and part of GC.

5.3 Relevance Measurement

MGeo can use both bi-encoder and cross-encoder architectures, as shown in Figure 3. Bi-encoder encodes query and POI separately for efficiency issues. It can be used in both retrieval and ranking phases. In practice, the GC of a POI or query is encoded by geographic encoder. Since user location is not always available due to privacy issues or limited hardware, the GC of query can be absent. The outputs are then concatenated with word embeddings. Transformer-based multi-modal interaction module then produces hidden states as final representations. We compute the similarity score of a query and POI pair by the cosine similarity between their *CLS* representations, *i.e.*, $\hat{\mathbf{h}}^Q$ and $\hat{\mathbf{h}}^P$. Bi-encoder calculates similarity scores between a query and all the POIs for retrieval task.

Different from bi-encoder, cross-encoder concatenates every query-POI pair together before being fed to multi-modal interaction module. Cross-encoder allows fine-grained token-level interaction between query and POI, it usually provides a more accurate estimation of relevance but is less efficient. Therefore, cross-encoder is only used in ranking phase as usual. The GC of query or POI is encoded separately by geographic encoder. The GC of query is also optional. We concatenate query textual embeddings, POI textual embeddings, query GC embeddings (optional), and POI GC embeddings together, which are then fed to multi-modal interaction module. Particularly, we use geographic discriminator to facilitate geographic comparison between GC of query and POI. Geographic discriminator adds embeddings to outputs of geographic encoder to distinguish query GC from POI GC. Like the segment embeddings in BERT, embeddings of geographic discriminator are randomly initialized and trainable. We fed the hidden states of *CLS* $\hat{\mathbf{h}}_{CLS}^{pq}$ to a multi-layer perceptron (MLP) to produce similarity scores.

Table 6: Ranking results of bi-encoder and cross-encoder PLMs. Bold indicates the best of each column.

PLM	Bi-Encoder				Cross-Encoder			
	Recall@1	Recall@3	Recall@5	MRR@5	Recall@1	Recall@3	Recall@5	MRR@5
BERT	58.83	79.40	86.24	69.60	81.52	91.11	94.10	86.53
RoBERTa	68.52	85.41	90.25	76.15	83.20	93.09	95.77	88.30
ERNIE	58.18	81.86	88.96	70.43	81.82	91.79	94.73	87.01
StructBERT	69.09	86.29	91.09	77.96	83.51	93.21	95.67	88.53
BERT	72.49	89.18	93.48	81.03	83.24	92.92	95.63	88.25
StructBERT	74.30	89.78	94.06	82.28	83.65	93.33	95.92	88.61
BERT	74.86	90.61	94.53	82.93	85.11	94.42	96.75	89.86
StructBERT	75.37	89.99	93.96	82.89	84.72	93.85	96.16	89.40
BERT	76.04	91.24	95.18	83.85	85.89	95.48	97.48	90.74
StructBERT	76.07	90.68	94.50	83.57	86.49	95.55	97.62	91.10

Table 7: Model sizes of pre-trained and fine-tuned models.

	Pre-Train	Fine-Tune
BERT-DA	118M	102M
BERT-MGeo	213M	129M

6 EXPERIMENTS

In this section, we compare the proposed MGeo with several strong baselines on GeoTES.

6.1 Setup

6.1.1 Tasks. The experiments are conducted on two tasks, *i.e.*, *ranking* and *retrieval*. The two tasks use the same train, development, and test sets as shown in Table 5. A list of candidate POIs that contains the positive POI is provided for the ranking test set. Both bi-encoder and cross-encoder are evaluated on ranking task. Since retrieval task requires searching the full POI corpus, and cross-encoder needs too much computing resources to complete retrieval task, only bi-encoder is evaluated on retrieval task.

6.1.2 Evaluation metrics. Following previous IR work [24], we use Recall and Mean Reciprocal Rank (MRR) at top k ranks to evaluate the performance on both tasks. Recall@ k calculates the proportion of queries that have the positive POI contained in the top- k candidates, and MRR@ k calculates the averaged reciprocal of the rank at which the positive POI is placed. We report the evaluation scores on the test set of models that perform best on the development set during training.

6.1.3 PLM Baselines. We first evaluate the performance of four widely used PLMs with the base model size on GeoTES, including BERT [5], RoBERTa⁵ [21], ERNIE 3.0 [30], and StructBERT [34]. We further apply domain-adaptive pre-training techniques (DA) on BERT and another top-performing model. DA is a widely used single-modal pre-training baseline [9]. For a fair comparison, domain corpus used in DA is the same as that used in our proposed multi-modal geographic pre-training (MGeo), except that MGeo has additional GC along with query and POI.

⁵https://huggingface.co/clue/roberta_chinese_base

6.1.4 Hyperparameters. The architecture of the multi-modal interaction module is multi-layer transformers. The model sizes are listed in Table 7.

Geographic Encoder. All geographic feature embeddings are set to 256. The discretization factor k is 10 and the grid number N is 2000. Geographic encoder has 4 layers of transformer with 256 hidden sizes. The mask probability is 0.15. The training batch size is 512. We use AdamW as optimizer with learning rate being 1e-4, weight decay being 0.02. We train geographic encoder for 30 epochs and take the last epoch checkpoint.

Pre-Training. The training batch size is 512. We use AdamW as optimizer with learning rate being 5e-5, weight decay being 0.02. We train for 10 epochs and take the last epoch checkpoint.

Fine-Tuning. For bi-encoder models, every training step has 56 queries, each has 20 candidates. We use AdamW as optimizer with learning rate being 5e-5, weight decay being 0.02. Specifically, ERNIE and StructBERT don't converge in this learning rate, we change it to 5e-6. We train geographic encoder for 10 epochs.

For cross-encoder models, every training step has 24 queries and the learning rate for RoBERTa is 5e-6. Other settings are the same as bi-encoder.

6.2 Ranking Results

Table 6 gives the ranking results of both bi-encoder and cross-encoder PLMs. As the original StructBERT outperforms the other generic PLMs, it is used for further DA. The generic PLMs directly fine-tuned on the downstream tasks show a low performance, which indicates that these two tasks are challenging. Since cross-encoder can make fine-grained interactions among input features, while bi-encoder only interacts with the *CLS* representations for the sake of efficiency, cross-encoder generally outperforms bi-encoder by a large margin.

By applying DA on bi-encoder, PLMs could gain an advantage over the generic ones. However, DA models consider only the textual modality and neglect the geographic modality. Through multi-modal pre-training, MGeo without query GC raises 2.37% (*resp.*, 1.07%) point of Recall@1 on BERT-DA (*resp.*, StructBERT-DA) by

Table 8: Ranking results of more baselines.

		Recall@1
Bi-Encoder	DSSM [38]	34.59
	DPAM [38]	44.15
	PALM [38]	45.51
	BERT	58.83
	ColBERT [15]	62.36
	Poly-Encoder [13]	49.87
	BERT-MGeo	76.04
Cross-Encoder	BERT	81.52
	ERNIE-GeoL [11]	82.94
	BERT-MGeo	85.89

Table 9: Retrieval results of bi-encoder.

	BERT	BERT-DA	BERT-MGeo
Recall@1	21.70	51.76	52.70
Recall@5	29.32	60.82	63.39
Recall@20	35.70	67.08	70.49
Recall@50	40.30	71.61	75.00
MRR@5	24.58	55.29	56.79

bridging the gap between query text and POI GC. After being accompanied by query GC, MGeo further shows a 3.55% (*resp.*, 1.77%) improvement in Recall@1 over DA models with the help of incorporating correlations between query GC and POI text, as well as between query GC and POI GC. It is worth noting that GC of half the training and test queries use random locations to simulate the arbitrary geolocation of users, as described in Section 4. The results show MGeo is robustness and it may gain more improvements if the queries have more precise geolocations.

In cross-encoder, MGeo also shows superiority over baselines. DA brings fewer benefits on PLMs than it does in bi-encoder, *i.e.*, 1.72% on BERT and 0.14% on StructBERT. However, improvements brought by incorporating the new geographic modal are consistent. MGeo without query GC gains 1.87% (*resp.*, 1.07%) Recall@1 on BERT-DA (*resp.*, StructBERT-DA). Together with query GC, MGeo boost DA models by 2.65% (*resp.*, 2.84%) in Recall@1, showing effectiveness of multi-modal interaction.

6.2.1 More Baseline Comparisons. Besides the PLM baselines, we also compare with more query-POI matching baselines, including two SOTA text-matching models, *i.e.*, ColBERT [15] and Poly-Encoder [13]. ColBERT uses a late interaction architecture to enhance bi-encoder model. Similarly, Poly-Encoder uses attention mechanism to capture richer interactions between query and POI. Detailed introductions of DSSM, DPAM, and PALM can be found in [38]. ERNIE-GeoL is a strong PLM cross-encoder baseline introduced in [11]. Since the data and code of ERNIE-GeoL are not released, we only adopt the pre-training objectives. The results on the ranking task are shown in Table 8. For bi-encoder, BERT-MGeo still outperforms ColBERT and Poly-Encoder, which capture more fine-grained interactions between query and POI. For cross-encoder, ERNIE-GeoL uses specific pre-training objectives to capture static

Table 10: Inference time in seconds.

	Bi-Encoder	Cross-Encoder
BERT-DA	0.0219	0.0396
BERT-MGeo w/o query GC	0.0205	0.0414
BERT-MGeo	0.0269	0.0466

Table 11: Influence of different geographic object types.

	Bi-Encoder		Cross-Encoder	
	Recall@1	MRR@5	Recall@1	MRR@5
Line	74.56	82.57	83.71	88.88
Polygon	74.26	82.51	84.84	89.85
Both	76.04	83.85	85.89	90.74

geographic knowledge and outperforms BERT. While BERT-MGeo capture dynamic GC and outperforms ERNIE-GeoL.

6.3 Retrieval Results

Bi-encoder is also evaluated on the retrieval task, which focuses on finding the relevant POIs rather than ranking the correct POI at the top. Table 9 reports Recall and MRR metrics. Compared to BERT-DA, MGeo improves 3.41% Recall@20. The results demonstrate that the effectiveness of MGeo in bi-encoder architecture stays consistent when the size of candidates becomes 100,000 times larger.

6.4 Inference Time

The inference time in seconds on 1 NVIDIA V100 GPU of bi-encoder and cross-encoder models is listed in Table 10. For bi-encoder, we only count the time of query encoding, since the document can be encoded in advance in many industrial scenarios. We use 26 queries and 1040 documents for inference.

6.5 Ablation Study

Since we use the same bi-encoder models for both retrieval and ranking tasks, the ablation study is mainly conducted on ranking task of BERT-based models.

6.5.1 Geographic Object. We first study the influence of training queries with GC. We randomly remove GC of the same proportion from the training, development, and test queries. As shown in Figure 4, the performance is impaired when a small proportion of queries contain GC. This decrease comes from a larger proportion of noise. Taking 30% of queries having GC as example, there are already 15% of GC inputs are random (half GC are randomly selected). Since it is difficult to distinguish query without GC from query without geographic object (but with geolocation), the rest queries without GC can be considered as noise too. Thus we have in total 75% queries with noisy GC, which damages model performance. When noises proportion becomes smaller than 65% (70% query with GC), the performance is better than training without query GC.

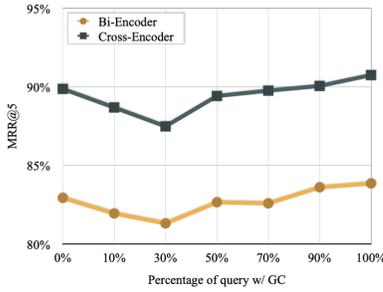


Figure 4: Ranking MRR@5 for different percentage of queries that have GC.

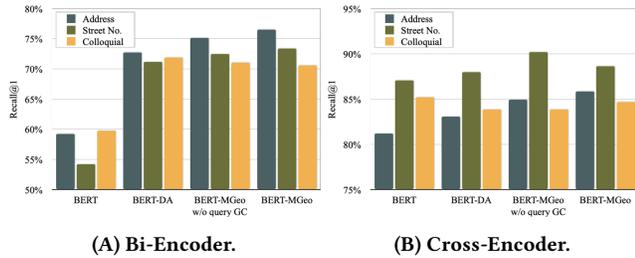


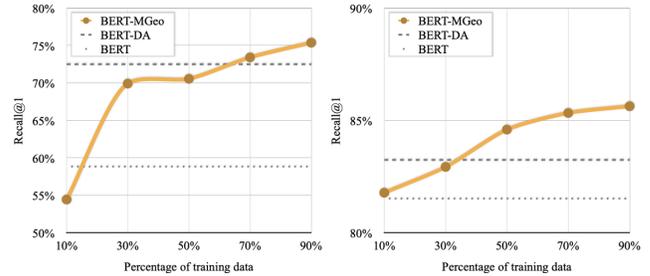
Figure 5: Ranking Recall@1 for (A) bi-encoder and (B) cross-encoder with different query types.

The influence of different geographic object types is reported in Table 11. There is not a huge gap between line and polygon for bi-encoder, while cross-encoder can perform better with only polygon than only line, as there are more polygons than lines in the GIS. This also suggests that cross-encoder is better at capturing the fine-grained correlations than bi-encoder. Nevertheless, using either line or polygon is better than the single-modal baselines. Besides, bi-encoder and cross-encoder can have a better performance when the two types of geographic objects both present.

6.5.2 Query Type. Figure 5 shows the performance on three query types, *i.e.*, address, street number, and colloquial. Bi-encoder models perform best on address description, while cross-encoder models perform best on street number description. This suggests that cross-encoder is better at capturing fine-grained correlations. Colloquial query contains many daily expressions, which rarely appear in domain corpus. Thus BERT-DA is even worse than BERT on it. However, the use of GC help reduce this shortcoming of DA.

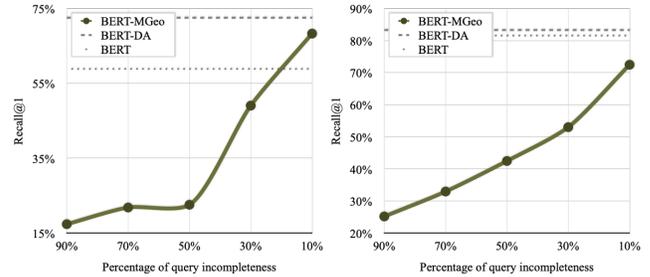
6.5.3 Amount of Training Data. We study the performance of MGeo with different amounts of training data. As shown in Figure 6, the dashed line is used for representing BERT-DA and the dotted line for the original BERT. With only 30% of training data, the bi-encoder and cross-encoder using MGeo can outperform the BERT baseline by a large margin.

6.5.4 Query Incompleteness. POI suggestion also plays an important role in LBS, where the name of POIs are listed when the input is unfinished. To simulate such scenario, we also evaluate MGeo



(A) Bi-Encoder. (B) Cross-Encoder.

Figure 6: Ranking Recall@1 for (A) bi-encoder and (B) cross-encoder with different amounts of training data.



(A) Bi-Encoder. (B) Cross-Encoder.

Figure 7: Ranking Recall@1 for (A) bi-encoder and (B) cross-encoder with different percentage of query incompleteness.

on incomplete queries by truncating the trailing characters. Figure 7 shows the performance with different truncation ratio of the test queries. The results demonstrate that bi-encoder using MGeo could outperform the BERT baseline with full queries with a small truncation ratio. Whereas the cross-encoder could not, since the semantic similarity is more important for cross-encoder.

7 CONCLUSION

In this paper, we formalize the important concept of *Geographic Context* (GC), which is indispensable for real-world human POI exploration process. We propose a multi-modal geographic language model MGeo, which considers GC as a new modality. Therefore, GC can be represented together with text. In addition, we build a new open-source large-scale benchmark GeoTES to facilitate further research on the query-POI matching topic. Extensive experiments are conducted to evaluate our proposed method on the state-of-the-art PLMs, and the detailed analyses demonstrate that MGeo can significantly outperform other baselines. Even though geolocation of user may be absent and query has no GC, MGeo can still obtain improvements over the baselines, showing its capability of modeling text-to-text, GC-to-GC and text-to-GC correlations. For future work, other modalities like POI image can be further explored, as well as more inventive geographic encoder. Besides, our proposed GC modeling has the potential to boost all geography-related tasks.

REFERENCES

- [1] Nicolas Carion, Francisco Massa, Gabriel Synnaeve, Nicolas Usunier, Alexander Kirillov, and Sergey Zagoruyko. 2020. End-to-End Object Detection with Transformers. In *Computer Vision – ECCV 2020*.
- [2] Yen-Chun Chen, Linjie Li, Licheng Yu, Ahmed El Kholy, Faisal Ahmed, Zhe Gan, Yu Cheng, and Jingjing Liu. 2020. UNITER: UNiversal Image-Text Representation Learning. In *Computer Vision – ECCV 2020*.
- [3] Nick Craswell, Bhaskar Mitra, Emine Yilmaz, Daniel Campos, and Jimmy Lin. 2021. MS MARCO: Benchmarking Ranking Models in the Large-Data Regime. In *Proceedings of the 44th International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR '21)*.
- [4] Zhuyun Dai and Jamie Callan. 2019. Deeper Text Understanding for IR with Contextual Neural Language Modeling. In *Proceedings of the 42nd International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR '19)*.
- [5] 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 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*.
- [6] Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, Jakob Uszkoreit, and Neil Houlsby. 2021. An Image is Worth 16x16 Words: Transformers for Image Recognition at Scale. In *International Conference on Learning Representations*.
- [7] Luyu Gao, Zhuyun Dai, and Jamie Callan. 2021. Rethink Training of BERT Rerankers in Multi-Stage Retrieval Pipeline. In *Advances in Information Retrieval: 43rd European Conference on IR Research, ECIR 2021*.
- [8] Daniel W. Goldberg, John P. Wilson, and Craig A. Knoblock. 2007. From Text to Geographic Coordinates: The Current State of Geocoding. *Urisa Journal* (2007).
- [9] Suchin Gururangan, Ana Marasović, Swabha Swayamdipta, Kyle Lo, Iz Beltagy, Doug Downey, and Noah A. Smith. 2020. Don't Stop Pretraining: Adapt Language Models to Domains and Tasks. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*.
- [10] Valentin Hofmann, Goran Glavas, Nikola Ljubesic, Janet B. Pierrehumbert, and Hinrich Schütze. 2022. Geographic Adaptation of Pretrained Language Models. *CoRR* (2022). arXiv:2203.08565
- [11] Jizhou Huang, Haifeng Wang, Yibo Sun, Yunsheng Shi, Zhengjie Huang, An Zhuo, and Shikun Feng. 2022. ERNIE-GeoL: A Geography-and-Language Pre-Trained Model and Its Applications in Baidu Maps. In *Proceedings of the 28th ACM SIGKDD Conference on Knowledge Discovery and Data Mining (KDD '22)*.
- [12] Po-Sen Huang, Xiaodong He, Jianfeng Gao, Li Deng, Alex Acero, and Larry Heck. 2013. Learning Deep Structured Semantic Models for Web Search Using Clickthrough Data. In *Proceedings of the 22nd ACM International Conference on Information & Knowledge Management (CIKM '13)*.
- [13] Samuel Humeau, Kurt Shuster, Marie-Anne Lachaux, and Jason Weston. 2020. Poly-encoders: Architectures and Pre-training Strategies for Fast and Accurate Multi-sentence Scoring. In *International Conference on Learning Representations*.
- [14] Vladimir Karpukhin, Barlas Oguz, Sewon Min, Patrick Lewis, Ledell Wu, Sergey Edunov, Danqi Chen, and Wen-tau Yih. 2020. Dense Passage Retrieval for Open-Domain Question Answering. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*.
- [15] Omar Khattab and Matei Zaharia. 2020. ColBERT: Efficient and Effective Passage Search via Contextualized Late Interaction over BERT. In *Proceedings of the 43rd International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR '20)*.
- [16] Wonjae Kim, Bokyung Son, and Ildoo Kim. 2021. ViLT: Vision-and-Language Transformer Without Convolution or Region Supervision. In *Proceedings of the 38th International Conference on Machine Learning (Proceedings of Machine Learning Research)*.
- [17] Jimmy Lin, Daniel Campos, Nick Craswell, Bhaskar Mitra, and Emine Yilmaz. 2021. Significant Improvements over the State of the Art? A Case Study of the MS MARCO Document Ranking Leaderboard. In *Proceedings of the 44th International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR '21)*.
- [18] Junyang Lin, Rui Men, An Yang, Chang Zhou, Yichang Zhang, Peng Wang, Jingren Zhou, Jie Tang, and Hongxia Yang. 2021. M6: Multi-Modality-to-Multi-Modality Multitask Mega-Transformer for Unified Pretraining. In *Proceedings of the 27th ACM SIGKDD Conference on Knowledge Discovery & Data Mining (KDD '21)*.
- [19] Junyang Lin, Rui Men, An Yang, Chang Zhou, Yichang Zhang, Peng Wang, Jingren Zhou, Jie Tang, and Hongxia Yang. 2021. M6: Multi-Modality-to-Multi-Modality Multitask Mega-Transformer for Unified Pretraining. In *Proceedings of the 27th ACM SIGKDD Conference on Knowledge Discovery & Data Mining (KDD '21)*.
- [20] Xiao Liu, Juan Hu, Qi Shen, and Huan Chen. 2021. Geo-BERT Pre-training Model for Query Rewriting in POI Search. In *Findings of the Association for Computational Linguistics: EMNLP 2021*.
- [21] Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. 2019. RoBERTa: A Robustly Optimized BERT Pretraining Approach. *CoRR* (2019). arXiv:1907.11692
- [22] Tri Nguyen, Mir Rosenberg, Xia Song, Jianfeng Gao, Saurabh Tiwary, Rangan Majumder, and Li Deng. 2016. MS MARCO: A Human Generated MACHine Reading Comprehension Dataset. In *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) (CEUR Workshop Proceedings)*.
- [23] Dwi Arman Prasetya, Phong Thanh Nguyen, Rinat Faizullin, Iswanto Iswanto, and Edmond Febrinicko Armay. 2020. Resolving the shortest path problem using the haversine algorithm. *Journal of Critical Reviews* (2020).
- [24] Yingqi Qu, Yuchen Ding, Jing Liu, Kai Liu, Ruiyang Ren, Wayne Xin Zhao, Daxiang Dong, Hua Wu, and Haifeng Wang. 2021. RocketQA: An Optimized Training Approach to Dense Passage Retrieval for Open-Domain Question Answering. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*.
- [25] Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, Gretchen Krueger, and Ilya Sutskever. 2021. Learning Transferable Visual Models From Natural Language Supervision. In *Proceedings of the 38th International Conference on Machine Learning*.
- [26] Nils Reimers and Iryna Gurevych. 2019. Sentence-BERT: Sentence Embeddings using Siamese BERT-Networks. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*.
- [27] Stephen E. Robertson, Steve Walker, Susan Jones, Micheline Hancock-Beaulieu, and Mike Gatford. 1994. Okapi at TREC-3. In *Proceedings of The Third Text Retrieval Conference*.
- [28] Chen Sun, Austin Myers, Carl Vondrick, Kevin Murphy, and Cordelia Schmid. 2019. VideoBERT: A Joint Model for Video and Language Representation Learning. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*.
- [29] Yibo Sun, Jizhou Huang, Chunyuan Yuan, Miao Fan, Haifeng Wang, Ming Liu, and Bing Qin. 2021. GEDIT: Geographic-Enhanced and Dependency-Guided Tagging for Joint POI and Accessibility Extraction at Baidu Maps. In *Proceedings of the 30th ACM International Conference on Information & Knowledge Management (CIKM '21)*.
- [30] Yu Sun, Shuohuan Wang, Shikun Feng, Siyu Ding, Chao Pang, Junyuan Shang, Jiaxiang Liu, Xuyi Chen, Yanbin Zhao, Yuxiang Lu, Weixin Liu, Zhihua Wu, Weibao Gong, Jianzhong Liang, Zhizhou Shang, Peng Sun, Wei Liu, Xuan Ouyang, Dianhai Yu, Hao Tian, Hua Wu, and Haifeng Wang. 2021. ERNIE 3.0: Large-scale Knowledge Enhanced Pre-training for Language Understanding and Generation. *CoRR* (2021). arXiv:2107.02137
- [31] Nandan Thakur, Nils Reimers, Johannes Daxenberger, and Iryna Gurevych. 2021. Augmented SBERT: Data Augmentation Method for Improving Bi-Encoders for Pairwise Sentence Scoring Tasks. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*.
- [32] Mohamed Trabelsi, Zhiyu Chen, Shuo Zhang, Brian D. Davison, and Jeff Heflin. 2022. StruBERT: Structure-Aware BERT for Table Search and Matching. In *Proceedings of the ACM Web Conference 2022 (WWW '22)*.
- [33] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. 2017. Attention is All you Need. In *Advances in Neural Information Processing Systems*.
- [34] Wei Wang, Bin Bi, Ming Yan, Chen Wu, Jiangnan Xia, Zuyi Bao, Liwei Peng, and Luo Si. 2020. StructBERT: Incorporating Language Structures into Pre-training for Deep Language Understanding. In *International Conference on Learning Representations*.
- [35] Yang Xu, Yiheng Xu, Tengchao Lv, Lei Cui, Furu Wei, Guoxin Wang, Yijuan Lu, Dinei Florencio, Cha Zhang, Wanxiang Che, Min Zhang, and Lidong Zhou. 2021. LayoutLMv2: Multi-modal Pre-training for Visually-rich Document Understanding. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*.
- [36] Andrew Yates, Rodrigo Nogueira, and Jimmy Lin. 2021. Pretrained Transformers for Text Ranking: BERT and Beyond. In *Proceedings of the 44th International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR '21)*.
- [37] Zixuan Yuan, Hao Liu, Yanchi Liu, Denghui Zhang, Fei Yi, Nengjun Zhu, and Hui Xiong. 2020. Spatio-Temporal Dual Graph Attention Network for Query-POI Matching. In *Proceedings of the 43rd International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR '20)*.
- [38] Ji Zhao, Dan Peng, Chuhan Wu, Huan Chen, Meiyu Yu, Wanji Zheng, Li Ma, Hua Chai, Jieping Ye, and Xiaohu Qie. 2019. Incorporating Semantic Similarity with Geographic Correlation for Query-POI Relevance Learning. In *Proceedings of the AAAI Conference on Artificial Intelligence*.