SpaBERT: A Pretrained Language Model from Geographic Data for Geo-Entity Representation

Zekun Li[1], **Jina Kim**[1], **Yao-Yi Chiang**[1], **Muhao Chen**[2]
[1]Department of Computer Science and Engineering, University of Minnesota, Twin Cities
[2]Department of Computer Science, University of Southern California
{li002666,kim01479,yaoyi}@umn.edu, muhaoche@usc.edu

Abstract

Named geographic entities (geo-entities for short) are the building blocks of many geographic datasets. Characterizing geo-entities is integral to various application domains, such as geo-intelligence and map comprehension, while a key challenge is to capture the spatial-varying context of an entity. We hypothesize that we shall know the characteristics of a geo-entity by its surrounding entities, similar to knowing word meanings by their linguistic context. Accordingly, we propose a novel spatial language model, SpaBERT (), which provides a general-purpose geo-entity representation based on neighboring entities in geospatial data. SpaBERT extends BERT to capture linearized spatial context, while incorporating a spatial coordinate embedding mechanism to preserve spatial relations of entities in the 2-dimensional space. SpaBERT is pretrained with masked language modeling and masked entity prediction tasks to learn spatial dependencies. We apply SpaBERT to two downstream tasks: geo-entity typing and geo-entity linking. Compared with the existing language models that do not use spatial context, SpaBERT shows significant performance improvement on both tasks. We also analyze the entity representation from SpaBERT in various settings and the effect of spatial coordinate embedding.

1 Introduction

Interpreting human behaviors requires considering human activities and their surrounding environment. Looking at a stopping location, [Speedway,],[1] from a person's trajectory, we might assume that this person needs to use the location's amenities if Speedway implies a gas station, and is near a highway exit. We might predict a meetup at [Speedway,] if the trajectory travels through many other locations, , , ..., of the same name, Speedway, to arrive at [Speedway,] in the middle of farmlands. As humans, we are able to make such inferences using the name of a geographic entity (geo-entity) and other entities in a spatial neighborhood. Specifically, we contextualize a geo-entity by a reasonable surrounding neighborhood learned from experience and, from the neighborhood, relate other relevant geo-entities based on their name and spatial relations (e.g., distance) to the geo-entity. This way, even if two gas stations have the same name (e.g., Speedway) and entity type (e.g., 'gas station'), we can still reason about their spatially varying semantics and use the semantics for prediction.

Capturing this spatially varying location semantics can help recognizing and resolving geospatial concepts (e.g., toponym detection, typing and linking) and the grounding of geo-entities in documents, scanned historical maps, and a variety of knowledge bases, such as Wikidata, OpenStreetMap, and GeoNames. Also, the location semantics can support effective use of spatial textual information (geo-entities names) in many spatial computing task, including moving behavior detection from visiting locations of trajectories (Yue et al., 2021, 2019), point of interest recommendations (Yin et al., 2017; Zhao et al., 2022), air quality (Lin et al., 2017, 2018, 2020; Jiang et al., 2019) and traffic prediction (Yuan and Li, 2021; Gao et al., 2019) using location context.

Recently, the research community has seen a rapid advancement in pretrained language models (PLMs) (Devlin et al., 2019; Liu et al., 2019; Lewis et al., 2020; Sanh et al., 2019), which supports strong contextualized language representation abilities (Lan et al., 2020) and serves as the backbones of various NLP systems (Rothe et al., 2020; Yang et al., 2019). The extensions of these PLMs help NL tasks in different data domains (e.g., biomedicine (Lee et al., 2020; Phan et al., 2021), software engineering (Tabassum et al., 2020), fi-

[1]A geographic entity name Speedway and its location (e.g., latitude and longitude). Best viewed in color.

Findings of the Association for Computational Linguistics: EMNLP 2022, pages 2757 - 2769
December 7-11, 2022

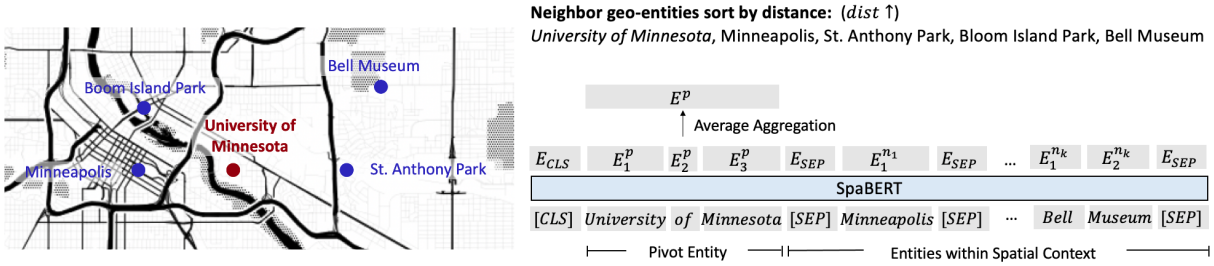

Figure 1: Overview for generating the pivot entity representation, E^p (●: pivot entity; ●: neighboring geo-entities). SpaBERT sorts and concatenate neighboring geo-entities by their distance to pivot in ascending order to form a pseudo-sentence. `[CLS]` is prepended at the beginning. `[SEP]` separates entities. SpaBERT generates token representations and aggregates representations of pivot tokens to produce E^p.

nance (Liu et al., 2021)) and modalities (e.g., tables (Herzig et al., 2020; Yin et al., 2020; Wang et al., 2022) and images (Li et al., 2020a; Su et al., 2019)). However, it is challenging to adopt existing PLMs or their extensions to capture geo-entities' spatially varying semantics. First, geo-entities exist in the physical world. Their spatial relations (i.e., distance and orientation) do not have a fixed structure (e.g., within a table or along a row of a table) that can help contextualization. Second, existing language models (LMs) pretrained on general domain corpora (Devlin et al., 2019; Liu et al., 2019) require fine-tuning for domain adaptation to handle names of geo-entities.

To tackle these challenges, we present SpaBERT (), a LM that captures the spatially varying semantics of geo-entity names using large geographic datasets for entity representation. Built upon BERT (Devlin et al., 2019), SpaBERT generates the contextualized representation for a geo-entity of interest (referred to as the *pivot*), based on its geographically nearby geo-entities. Specifically, SpaBERT linearizes the 2-dimensional spatial context by forming *pseudo sentences* that consist of names of the pivot and neighboring geo-entities, ordered by their spatial distance to the *pivot*. SpaBERT also encodes the spatial relations between the pivot and neighboring geo-entities with a continuous *spatial coordinate embedding*. The spatial coordinate embedding models horizontal and vertical distance relations separately and, in turn, can capture the orientation relations. These techniques make the inputs compatible with the BERT-family structures.

In addition, the backbone LM, BERT, in SpaBERT is pretrained with general-purpose corpora and would not work well directly on geo-entity names because of the domain shift. We thus train SpaBERT using *pseudo sentences* generated from large geographic datasets derived from OpenStreetMap (OSM).[2] This pretraining process conducts Masked Language Modeling (MLM) and Masked Entity Prediction (MEP) that randomly masks subtokens and full geo-entity names in the pseudo sentences, respectively. MLM and MEP enable SpaBERT to learn from pseudo sentences for generating spatially varying contextualized representations for geo-entities.

SpaBERT provides a general-purpose representation for geo-entities based on their spatial context. Similar to linguistic context, spatial context refers to the surrounding environment of a geo-entity. For example, [Speedway,], [Speedway,], and [Speedway,] would have different representations since the surrounding environment of , , and could vary. We evaluate SpaBERT on two tasks: 1) geo-entity typing and 2) geo-entity linking to external knowledge bases. Our analysis includes the performance comparison of SpaBERT in various settings due to characteristics of geographic data sources (e.g., entity omission).

To summarize, this work has the following contributions. We propose an approach to linearize the 2-dimensional spatial context, encode the geo-entity spatial relations, and use a LM to produce spatial varying feature representations of geo-entities. We show that SpaBERT is a general-purpose encoder by supporting geo-entity typing and geo-entity linking, which are keys to effectively integrating large varieties of geographic data sources, the grounding of geo-entities as well as supports for a broad range of spatial computing applications. The experiments demonstrate that SpaBERT is effective for both tasks and outperforms the SOTA LMs.

[2]OpenStreetMap: https://www.openstreetmap.org/

2 SpaBERT

SpaBERT is a LM built upon a pretrained BERT and further trained to produce contextualized geo-entity representations given large geographic datasets. Fig. 1 shows the outline of SpaBERT, with the details below. This section first presents the preliminary (§2.1) and then describes the overall approach for learning geo-entities' contextualized representations (§2.2), the pretraining strategies (§2.3), and inference procedures (§2.4).

2.1 Preliminary

We assume that given a geographic dataset (e.g., OSM) with many geo-entities, $S = \{g_1, g_2, ..., g_l\}$, each geo-entity g_i (e.g., [Speedway, 📍]) has two attributes: name g^{name}(Speedway) and location g^{loc}(📍). The location attribute, g^{loc}, is a tuple $g^{loc} = (g^{locx}, g^{locy}, ...)$ that identifies the location in a coordinate system (e.g., x and y image pixel coordinates or latitude and longitude geo-coordinates with altitudes). WLOG, here we assume a 2-dimensional space. SpaBERT aims to generate a contextualized representation for each geo-entity g_i in S given its spatial context. We denote a geo-entity that we seek to contextualize as the *pivot entity*, g_p, p for short. The spatial context of p is $SC(p) = \{g_{n_1}, ..., g_{n_k}\}$ where $distance(p, g_{n_k}) < T$. T is a spatial distance parameter defining a local area. We call $g_{n_1}, ..., g_{n_k}$ as p's neighboring geo-entities and denote them as $n_1, ..., n_k$ when there is no confusion.

2.2 Contextualizing Geo-entities

Linearizing Neighboring Geo-entity Names For a pivot, p, SpaBERT first linearizes its neighboring geo-entitie names to form a BERT-compatible input sequence, called a *pseudo sentence*. The corresponding pseudo sentence for the example in Fig. 1 is constructed as:

```
[CLS] University of Minnesota [SEP]
Minneapolis [SEP] St.  Anthony Park [SEP]
Bloom Island Park [SEP] Bell Museum [SEP]
```

The pseudo sentence starts with the pivot name followed by the names of the pivot's neighboring geo-entities, ordered by their spatial distance to the pivot in ascending order. The idea is that nearby geo-entities are more related (for contextualization) than distant geo-entities. SpaBERT also tokenizes the pseudo sentences using the original BERT tokenizer with the special token [SEP] to separate entity names. The subtokens of a neighbor n_k is denoted as $T_j^{n_k}$ as in Fig. 2.

Encoding Spatial Relations SpaBERT adopts two types of position embeddings in addtion to the token embeddings in the pseudo sentences (Fig. 2). The sequence position embedding represents the token order, same as the original position embedding in BERT and other Transformer LMs. Also, SpaBERT incorporates a *spatial coordinate embedding* mechanism, which seeks to represent the spatial relations between the pivot and its neighboring geo-entities. SpaBERT's spatial coordinate embeddings encode each location dimension separately to capture the relative distance and orientation between geo-entities. Specifically, for the 2-dimensional space, SpaBERT generates normalized distance $dist_x^{n_k}$ and $dist_y^{n_k}$ for each neighboring geo-entity using the following equations:

$$dist_x^{n_k} = (g_{n_k}^{locx} - g_p^{locx})/Z$$
$$dist_y^{n_k} = (g_{n_k}^{locy} - g_p^{locy})/Z$$

where Z is a normalization factor, and (g_p^{locx}, g_p^{locy}), $(g_{n_k}^{locx}, g_{n_k}^{locy})$ are the locations of pivot p and neighboring entity n_k. Note that the tokens belong to the same geo-entity name have the same $dist_x^{n_k}$ and $dist_y^{n_k}$. Also, SpaBERT uses DSEP, a constant numerical value larger than $max(dist_x^{n_k}, dist_y^{n_k})$ for all neighboring entities to differentiate special tokens from entity name tokens.

SpaBERT encodes $dist_x^{n_k}$ and $dist_y^{n_k}$ using a continuous spatial coordinate embedding layer with real-valued distances as input to preserve the continuity of the output embeddings. Let $\mathcal{S}_{n_k}$ be the spatial coordinate embedding of the neighbor entity n_k, M be the embedding's dimension, and $\mathcal{S}_{n_k} \in \mathbf{R}^M$. We define $\mathcal{S}_{n_k}$ as:

$$\mathcal{S}_{n_k}^{(m)} = \begin{cases} sin(dist^{n_k}/10000^{2j/M}), m = 2j \\ cos(dist^{n_k}/10000^{2j/M}), m = 2j+1 \end{cases}$$

where $\mathcal{S}_{n_k}^{(m)}$ is the m-th component of $\mathcal{S}_{n_k}$. $dist^{n_k}$ represents $dist_x^{n_k}$ and $dist_y^{n_k}$ for the spatial coordinate embeddings along the horizontal and vertical directions, respectively.

The token embedding, sequence position embedding and spatial coordinate embedding are summed up then fed into the encoder. Similar to BERT, SpaBERT encoder calculates an output embedding for each token. Then SpaBERT averages the

Figure 2: Embedding modules. Inputs to the token-embedding are the tokenized geo-entity names in the pseudo-sentence. Inputs to the sequence position embedding are the sequence indices of the tokens. Inputs to the spatial coordinate embedding are the normalized distances from the pivot in each spatial dimension (details in §2.2).

pivot's token-level embeddings to produce a fixed-length embedding for the pivot's contextualized representation.

2.3 Pretraining

We train SpaBERT with two tasks to adapt the pre-trained BERT backbone to geo-entity pseudo sentences. One task is the masked language modeling (MLM) (Devlin et al., 2019), for which SpaBERT needs to learn how to complete the full names of geo-entities from pseudo sentences with randomly masked subtokens using the remaining subtokens and their spatial coordinates (i.e., partial names and spatial relations between subtokens). The block below shows an example of the masked input for MLM, where ### are the masked subtokens.

```
[CLS] ### of Minnesota [SEP] Minneapolis
[SEP] St.  ### Park [SEP] ### Island Park
### Bell Museum [SEP]
```

In addition, we hypothesize that given common spatial co-occurrence patterns in the real world, one can use neighboring geo-entities to predict the name of a geo-entity. Therefore, we propose and incorporate a masked entity prediction (MEP) task, which randomly masks all subtokens of an entity name in a pseudo sentence. For MEP, SpaBERT relies on the masked entity's spatial relation to neighboring entities to recover the masked name. The block below shows an example of the masked input for MEP.

```
[CLS] University of Minnesota [SEP]
Minneapolis [SEP] ### ### ### [SEP] Bloom
Island Park [SEP] Bell Museum [SEP]
```

Both MLM and MEP have a masking rate of 15% (without masking `[CLS]` and `[SEP]`). The sequence position and spatial coordinates are not masked.

Pretraining Data We construct a large training set from OSM covering the City of London and the State of California. Since OSM is crowd-sourced, we clean the raw data by removing non-alpha-numeric place names and geo-entities that do not have a place name or geocoordinates. We randomly select geo-entities as pivots and construct pseudo sentences as described in §2.2.

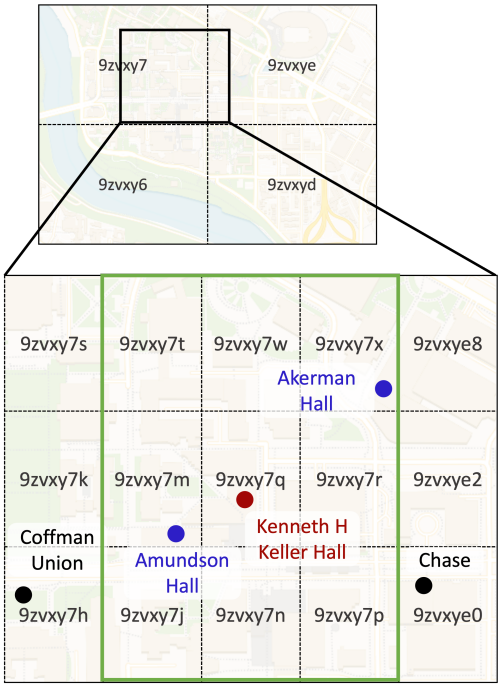

Figure 3: Example of the Geohash representations with a pivot entity named 'Kenneth H Keller Hall' and its neighboring entities from OpenStreetMap. (●: pivot entity; ●: neighboring entities; ●: entities not within the nearest nine grids). The upper hash grids have a string length of six, and the lower ones have a length of seven.

To efficiently find the neighboring entities from a pivot sorted by distances, we leverage the Geohash algorithm introduced by Gustavo Niemeyer in 2008. We first generate the Geohash string for geo-entities on OpenStreetMap, which encodes a location into a string of a maximum length of 20 with base 32. For example, a city '*Madelia*' with the geolocation (44.0508° N, 94.4183° W) has a Geohash string of '9zufe7nwjefpjksty0m8'. The length of the hash string is associated with the size of the geographic area that the string represents. The first character divides the entire Earth's surface into 32 large grids, and the second character further divides one large grid into 32 grids. Thus, by comparing the leading

characters of two hash strings, we can estimate the spatial proximity of two locations without calculating the exact distance between them. We use the hash strings to first filter out the geo-entities guaranteed to be outside the spatial context radius and keep the neighbors within the nearest nine hash grids for distance calculation. The use of Geohash reduces the computation time when constructing the pseudo sentences. Fig. 3 shows an example of Geohash representations. Given the pivot 'Kenneth H Keller Hall', we can calculate its hash string of different lengths to quickly retrieve the neighboring entities within a desired range (i.e., the green box).

This way, we generate 69,168 and 148,811 pseudo sentences in London and California, respectively, leading to a pretraining corpus of around 150M words. We use all of them for pretraining SpaBERT with MLM and MEP.

2.4 Inference

The pretrained SpaBERT can already generate spatially varying contextualized representations for geo-entities. The representations support various downstream applications, including contextualized geo-entity classification and similarity-based inference tasks, such as geo-entity typing and linking.

Contextualized Geo-entity Classification Classification of a geo-entity is crucial for recognizing and resolving geospatial concepts and the grounding of geo-entities in various data sources. Here, we use geo-entity typing as an example task to demonstrate the effectiveness of contextualized geo-entity representations. In this task, we aim to predict the geo-entity's semantic type (e.g., transportation and healthcare). We stack a softmax prediction head on top of the final-layer hidden states (i.e., geo-entity embeddings) to perform the classification.

Similarity-based Inference Integrating multi-source geographic datasets often involves finding the top K nearest geo-entities in some representation space. One example task, which we refer to as geo-entity linking, is to link geo-entities from a geographic information system (GIS) oriented dataset to graph-based knowledge bases (KBs). In such a task, we can directly use the contextualized geo-entity embeddings from SpaBERT and calculate the embedding similarity based on some metrics, such as the cosine distance, to match the corresponding geo-entities from separate data sources.

3 Experiments

We evaluate SpaBERT on supervised geo-entity typing and unsupervised geo-entity linking tasks (§3.1-§3.3). We also investigate how SpaBERT performs under various common characteristics of geographic data sources (e.g., entity omission from cartographic generalization) and how critical technical components and their parameters affect the performance (§3.4).

3.1 Experimental Setup

Datasets Open Street Map (OSM)[2] is a large-scale geographic database, and it is widely used in many popular map-related services. For *supervised geo-entity typing*, we randomly select 25,872 and 7,726 pivot geo-entities together with their neighboring entities from point features of nine semantic types on OSM in the State of California and the City of London, respectively (Tab. 1). Each geo-entity is associated with a name and its geocoordinates. We use 80% of the data in both regions for training and 20% for testing. The task is to predict the OSM semantic type for each geo-entity. (§3.2).

For *unsupervised geo-entity linking*, we randomly select 14 scanned historical maps in California from the United States Geological Survey (USGS) Historical Topographic Maps Collection. We manually transcribe text labels in these maps to generate 154 geo-entities, each containing a name (from the text label) and the image coordinates (text label's pixel location in the scanned map) The task is to find the corresponding Wikidata geo-entity[3] for each USGS geo-entity with only their spatial relations from pixel locations. Note that the geocoordinates of geo-entities in the scanned maps are unknown. We select 15K Wikidata geo-entities having an identical name to one of the USGS geo-entities[4] and then randomly add another 30K Wikidata geo-entities to construct the final Wikidata dataset of 45K geo-entities. For the ground truth, we manually identify the matched Wikidata geo-entity for each USGS geo-entity.

For geo-entity linking task, the annotation details are described below. We hire three undergraduate students (not the co-authors) to transcribe text labels on the USGS maps. They draw bounding boxes/polygons around text labels and transcribe

[3]Entities with the coordinate location attribute (https://www.wikidata.org/wiki/Property:P625)

[4]Two geo-entities can have the same name (e.g., Los Angeles, CA vs. Los Angeles, TX)

Classes	California	London
Education	6,222	618
Entertainment_Arts_Culture	1,380	601
Facilities	574	179
Financial	2,590	769
Healthcare	3,779	1,779
Public_Service	2,658	393
Sustenance	4,276	1,693
Transportation	4,226	1,618
Waste_Management	167	76
Total	25,872	7,726

Table 1: OSM Geo-entity typing dataset statistics. The first column shows the nine OSM semantic types, following by the numbers of samples for each class in California and London.

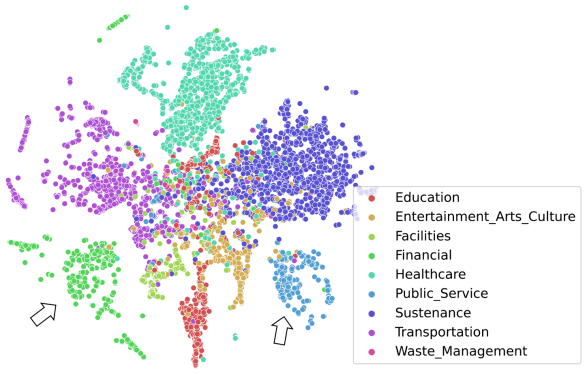

Figure 4: TSNE visualization for entity features produced by SpaBERT$_{Base}$. The color indicates the ground-truth labels. *Financial* and *Public_Service* (pointed by the arrows) are well separated from other classes which conforms to the high F1 scores in Tab. 2 (best viewed in color).

the text string to indicate geo-entity names and their locations. The results are verified and corrected by another undergraduate student and one Ph.D. student. Then, we run an automatic script to find the potential corresponding geo-entity in Wikidata by matching their names. The collected Wikidata URIs are noisy, and we have one Ph.D. student verify the URIs by marking them as True or False linking. The verification results are randomly further reviewed by a senior digital humanities scholar specialized in the representation and reception of historical places.

Model Configuration We create two variants of SpaBERT, namely SpaBERT$_{base}$ and SpaBERT$_{large}$ with weights and tokenizers initialized from the uncased versions of the BERT$_{base}$ and BERT$_{large}$, respectively. During pretraining, we mix the masked instances for MLM and MEP tasks in the same ratio. We primarily use the same optimization parameters as in (Devlin et al., 2019) and train the model with the AdamW optimizer. We save the model checkpoints every 2K iterations. The learning rate is 5×10^{-5} for SpaBERT$_{Base}$ and 1×10^{-6} for SpaBERT$_{Large}$ for stability. Distance normalization factor Z is 0.0001, and distance separator DSEP is 20. The batch size is 12 to fit in one NVIDIA RTX A5000 GPU with 24GB memory.

3.2 Supervised Geo-Entity Typing

Task Description We tackle geo-entity typing as a supervised classification task using the modified SpaBERT architecture in §2.4 and finetune the model end-to-end by optimizing the cross-entropy loss. Specifically, the training set of this task contains $\{(g_i, SC(g_i), y_i)\}_{i=1}^{N}$, where g_i is the pivot entity; y_i is its OSM semantic type label; $SC(g_i)$ contains its neighboring entities in the nearest nine hash grids. Note that during training, the model does not use the neighboring geo-entities' OSM semantic types. We report the F1 score for each class (i.e., an OSM semantic type) and the micro-F1 for all samples in the test set.

Model Comparison We compare SpaBERT with several strong PLMs, including BERT (Devlin et al., 2019), RoBERTa (Liu et al., 2019), SpanBERT (Joshi et al., 2020), LUKE (Yamada et al., 2020), and SimCSE (Gao et al., 2021). For all these models, we use the pretrained weights with the uncased version. For SimCSE, we use both the BERT and RoBERTa versions with the weights obtained from unsupervised training. We further finetune these baseline LMs end-to-end on the same pseudo-sentence dataset as SpaBERT (Tab. 1) with a softmax prediction head appended after the baseline model. Note that these baseline models only use the token and position embeddings but not the spatial coordinate embeddings.

Result Analysis Tab. 2 summarizes the geo-entity typing results on the OSM test set with finetuned baselines and SpaBERT. The last column shows the micro-F1 scores (per-instance average). The remaining columns are the F1 scores for individual classes. For all baseline models, BERT$_{Base}$ and SimCSE$_{BERT\text{-}Large}$ (BERT$_{Large}$ as a close second) have the highest F1 in the base and large groups, respectively. In addition, SpaBERT shows the best performance over the existing context-aware LMs for both base and large groups. We hypothesize that SpaBERT's advantage comes from 1) the pretraining tasks, especially MEP, and 2) the spatial

Classes →	Edu.	Ent.	Fac.	Fin.	Hea.	Pub.	Sus.	Tra.	Was.	Micro Avg
BERT$_{Base}$	.674	.634	**.763**	.929	.856	.872	.856	.862	.678	.835
RoBERTa$_{Base}$	.626	.627	.605	.951	**.869**	.818	.838	.850	.475	.820
SpanBERT$_{Base}$	.633	.589	.608	.916	.859	.882	.824	.867	**.735**	.819
LUKE$_{Base}$	.648	.608	.598	.945	.857	.867	.854	.851	.517	.825
SimCSE$_{BERT\text{-}Base}$	.623	.590	.504	.925	.867	.852	.857	.810	.470	.810
SimCSE$_{RoBERTa\text{-}Base}$	.621	.629	.499	.951	.841	.853	.828	.856	.500	.814
SpaBERT$_{Base}$	**.674**	**.653**	.680	**.959**	.865	**.900**	**.883**	**.888**	.703	**.852**
BERT$_{Large}$	.707	.661	.647	.937	.874	.850	.873	.864	.526	.841
RoBERTa$_{Large}$	.657	.626	.682	.907	.855	.805	.831	.859	.587	.817
SpanBERT$_{Large}$	.683	.652	.661	.931	.868	.853	.851	.848	.624	.829
LUKE$_{Large}$	.665	.607	.660	.899	.855	.809	.813	.844	.587	.808
SimCSE$_{BERT\text{-}Large}$	.693	.661	**.713**	.940	.880	.871	.864	.867	.564	.844
SimCSE$_{RoBERTa\text{-}Large}$	.683	.630	.648	.916	.865	.802	.807	.848	.587	.811
SpaBERT$_{Large}$	**.731**	**.690**	.710	**.956**	**.901**	**.892**	**.893**	**.903**	**.677**	**.871**

Table 2: Geo-entity typing with the state-of-the-art LMs and SpaBERT. Column names are the OSM classes. Bold numbers are the highest scores in each column. Underlined numbers are the highest scores among baselines.

coordinate embeddings. Additionally, compared to its backbone, BERT, SpaBERT shows performance improvement in almost all classes for both base and large models. The results demonstrate that SpaBERT can effectively contextualize geo-entities with their spatial context.

We can also observe that the *Financial* class has high F1 scores for all models. The reason is that when entity names are strongly associated with the semantic type (e.g., U.S. Bank), pretrained LMs could produce meaningful entity representation even without the spatial context. However, the typing performance can still be further improved after encoding the spatial context with SpaBERT. Fig. 4 visualizes the geo-entity features from SpaBERT$_{Base}$. It shows that *Financial* and *Public_Service* have the most separable features from other classes. The non-separable region in the middle is mainly due to the similar spatial context and semantics of geo-entities (e.g. "nursing home" can be close to both *Facilities* and *Healthcare*).

3.3 Unsupervised Geo-Entity Linking

Task Description We define the problem as follows. For a query set $Q = \{(q_i, SC(q_i))\}_{i=1}^{|Q|}$, the goal is to find the corresponding geo-entity for each q_i in the candidate set $C = \{(c_i, SC(c_i))\}_{i=1}^{|C|}$ (typically $|C| \gg |Q|$). Here, Q and C are the USGS and Wikidata geo-entities (§3.1). $SC(q_i)$ contains q_i's nearest 20 geo-entities, and $SC(c_i)$ includes c_i's neighboring geo-entities within a 5km radius.[5] For this task, we use the evaluation metrics of Recall@K and Mean-Reciprocal-Rank (MRR).

Model Comparison We use the same baseline models as in the typing task without finetuning.

[5]Because Q is not georeferenced, we use the top K nearest neighbors instead of pixel distances.

Model	MRR	R@1	R@5	R@10
BERT$_{Base}$	.400	.289	.559	.635
RoBERTa$_{Base}$	.326	.232	.446	.540
SpanBERT$_{Base}$	.164	.138	.201	.213
LUKE$_{Base}$	.306	.188	.440	.547
SimCSE$_{BERT\text{-}Base}$	.453	**.371**	.547	.628
SimCSE$_{RoBERTa\text{-}Base}$	.227	.188	.264	.301
SpaBERT$_{Base}$	**.515**	.338	**.744**	**.850**
BERT$_{Large}$	.337	.245	.459	.509
RoBERTa$_{Large}$	.379	.220	.603	.704
SpanBERT$_{Large}$	.229	.176	.308	.339
LUKE$_{Large}$	.402	.232	.635	.767
SimCSE$_{BERT\text{-}Large}$	.475	**.402**	.559	.616
SimCSE$_{RoBERTa\text{-}Large}$	.214	.176	.239	.283
SpaBERT$_{Large}$	**.537**	.383	**.744**	**.864**

Table 3: Geo-entity linking result. Bold and underlined numbers are the highest scores in each column and the highest scores among the baselines, respectively.

Result Analysis Tab. 3 shows the geo-entity linking results. Large models perform better than their base counterparts for both baseline models and SpaBERT. In particular, among the baseline models, SimCSE$_{BERT}$ has the highest score for MRR and R@1. SpaBERT has significantly higher R@5 and R@10 than SimCSE. Also, SpaBERT outperforms all baselines on MRR, R@5, and R@10. For both base and large versions, SpaBERT outperforms its backbone model BERT. The difference between BERT and SpaBERT shows the effectiveness of the spatial coordinate embeddings and domain adaptation using MLM and the proposed MEP. Also, SpanBERT shows the worst performance among all models. This could be that SpanBERT utilizes masked random spans in pretraining, which does not work well for pseudo sentences of geo-entity names. In contrast, SpaBERT's MEP specifically masks tokens from the same

Map Set	Scale ⇓	MRR	R@1	R@5	R@10
15-CA	1:62500	0.503	0.422	0.680	0.914
30-CA	1:125000	0.639	0.599	0.862	0.932
60-CA	1:250000	0.404	0.133	0.678	0.800

Table 4: Impact of entity omission for linking (USGS and Wikidata). Results from SPABERT$_{Large}$. 1:625000 means 1 centimeter on a map represents 625 meters in the physical world.

geo-entity and could effectively contextualize geo-entity names distributed in the 2D space with the spatial coordinate embeddings.

3.4 Spatial Context and Ablation Study

Impact of Entity Omission for Linking We analyze how SPABERT handles different USGS map scales for geo-entity linking with entity omission from cartographic generalization (e.g., some entities only exist in certain scales). Here SPABERT performs the best for the 1:125K maps (30-CA) (Tab. 4). Our hypothesis is that the 1:125K maps contain denser geo-entities than the test maps at other scales. When selecting the top K nearest neighbors for a pivot in other map scales, some neighbors could be far away and would not provide relevant context.

Since the entity omission criteria of the USGS maps are unknown and Wikidata mostly contain important landmark geo-entities, we also simulate omission using the OSM dataset by gradually removing random neighbors of a pivot entity within a fixed neighborhood. We test this scenario for geo-entity linking to see if the same pivot entity would have similar representations from decreasing neighbor densities (i.e., the original set of neighbors vs. after random omission at varying rates). We use the same evaluation metric as in §3.3, as they reflect the similarity of the learned geo-entity representation (Fig. 5). The linking scores decrease with additional entities removed, indicating the similarity between the same pivot entity's representations learned from the original data and the omitted data declines when omission percentage becomes larger. The elbow point is at about 30%, showing that SPABERT is reasonably robust if the percentage of the removed neighbors is within 30%.

Impact of Pseudo Sentence Length We analyze the impact of pseudo sentence length on geo-entity typing by varying the number of neighboring entities. Tab. 5 shows the results of SPABERT$_{Base}$ and SPABERT$_{Large}$ with an increasing number of neighbors. We observe that as #Neighbor increases,

#Neighbor	1	2	3	4	5	6	7	8	9
Base	.773	.808	.814	.827	.835	.831	.835	.836	.840
Large	.795	.822	.834	.838	.843	.847	.852	.848	.854
#Neighbor	**10**	**20**	**30**	**40**	**50**	**60**	**70**	**80**	**90**
Base	.843	.844	.846	.851	.852	.850	.849	.851	.852
Large	.850	.857	.862	.858	.858	.868	.863	.867	.869

Table 5: Impact of the pseudo sentence length. Numbers in the table are micro-F1 scores on the OSM typing dataset. *Base* and *Large* are short for SPABERT$_{Base}$ and SPABERT$_{Large}$.

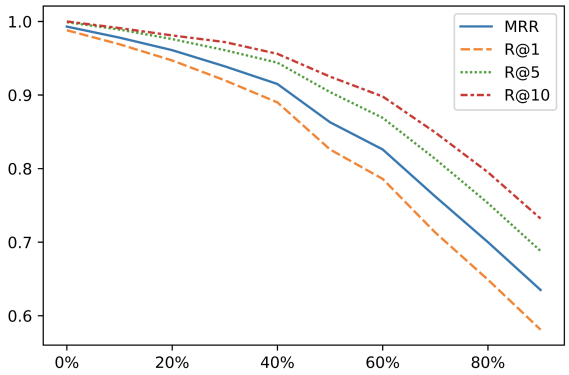

Figure 5: Geo-entity feature similarity with a decreasing neighbor density. X-axis indicates the percentage of the neighbors removed for the pivot entities.

the micro-F1 score increases accordingly, as expected. The increment is more evident at the beginning, illustrating that the spatial context helps contextualize the pivot entity, especially in a local neighborhood. The benefit of adding additional neighboring entities decreases as distance increases beyond a certain point. This conforms to the well-studied spatial autocorrelation.

Effect of Spatial Coordinate Embedding We train two additional SPABERT variants that do not include the spatial coordinate embedding during MLM and MEP pretraining and use them for geo-entity linking. For MRR, SPABERT$_{Base}$ and SPABERT$_{Large}$ drop from 0.515 to 0.458 and from 0.537 to 0.478 compared to their original SPABERT version. Also, the variants have lower recall scores compared to their original SPABERT version. The most significant drop in the recall is on R@1 from 0.383 to 0.283 for SPABERT$_{Large}$. The results demonstrate the effectiveness of the spatial coordinate embedding.

4 Related Work

Pretrained Language Models PLMs have been the dominant paradigm for language representation. Following the success of MLMs (Devlin et al., 2019; Liu et al., 2019) and autoregressive PLMs

(Peters et al., 2018; Radford et al., 2019), more recent work has extended the pretraining process with more tasks or learning objectives for span prediction (Joshi et al., 2020), cross-encoders (Reimers and Gurevych, 2019), contrastive learning (Gao et al., 2021), and massively multi-task learning (Raffel et al., 2020). To support entity representation, several approaches propose to perform mention detection (Yamada et al., 2020), incorporate mention memory cells (de Jong et al., 2021), or injecting structural knowledge representations (Wang et al., 2021; Zhang et al., 2019; Peters et al., 2019; Zhou et al., 2022). Due to the large body of work, we refer readers to recent surveys summarizing this line of work (Qiu et al., 2020; Wei et al., 2021).

To extend the use of PLMs beyond language, much exploration has also been conducted to represent other modalities. For example, a significant amount of vision-language models (Kim et al., 2021; Zhai et al., 2022; Li et al., 2020a; Lu et al., 2019) have been developed by jointly pretraining on co-occurring vision and language corpora, and are in the support of grounding (Zhang et al., 2020; Chen et al., 2020), generation (Cho et al., 2021; Yu et al., 2022) and retrieval tasks (Zhang et al., 2021; Li et al., 2020b) on the vision modality. Other PLMs capture semi-structure tabular data by linearizing table cells (Herzig et al., 2020; Yin et al., 2020; Iida et al., 2021) or incorporating structural prior in the attention mechanism (Wang et al., 2022; Trabelsi et al., 2022; Eisenschlos et al., 2021). To the best of our knowledge, none of the prior studies have extended pretrained LMs to geo-entity representation, nor do they support a suitable mechanism to capture the geo-entities' spatial relations, which do not have structural prior. This is exactly the focus of our work.

Domain-specific Language Modeling Another line of studies has been conducted to adapt PLMs to specific domains typically by pretraining on domain-specific corpora. For example, in the biomedicine domain, a series of models (Lee et al., 2020; Peng et al., 2019; Alsentzer et al., 2019; Phan et al., 2021) have been developed by training PLMs on corpora derived from PubMed. Similarly, PLMs have been trained on corpora specific to software engineering (Tabassum et al., 2020), finance (Liu et al., 2021), and proteomics (Zhou et al., 2020) domains. In this context, SpaBERT represents a pilot study in the geographical domain by allowing the PLM to learn from spatially distributed text.

5 Conclusion

This paper presented SpaBERT (☕), a language model trained on geographic datasets for contextualizing geo-entities. SpaBERT utilizes a novel spatial coordinate embedding mechanism to capture spatial relations between 2D geo-entities and linearizes the geo-entities into 1D sequences, compatible with the BERT-family structures. The experiments show that the general-purpose representations learned from SpaBERT achieve better or competitive results on the geo-entity typing and geo-entity linking tasks compared to SOTA pretrained LMs. We plan to evaluate SpaBERT on additional related tasks, such as geo-entity to natural language grounding. Also, we plan to extend SpaBERT to support other geo-entity geometry types, including lines and polygons.

Limitations

The current model design only considers points but not polygon and line geometries, which could also help provide meaningful spatial relations for contextualizing a geo-entity. Training of SpaBERT also requires considerable GPU resources which might produce environmental impacts.

Ethical Consideration

SpaBERT was evaluated on the English-spoken regions, so the model and results could have a bias towards these regions and their commonly used languages. Replacing the backbone of SpaBERT with a multi-lingual model and training SpaBERT with diverse regions could mitigate the bias.

Acknowledgement

We thank the reviewers for their insightful comments and suggestions. We thank Dr. Valeria Vitale for her valuable input. This material is based upon work supported in part by NVIDIA Corporation, the National Endowment for the Humanities under Award No. HC-278125-21 and Council Reference AH/V009400/1, the National Science Foundation of United States Grant IIS 2105329, a Cisco Faculty Research Award (72953213), and the University of Minnesota, Computer Science & Engineering Faculty startup funds.

References

Emily Alsentzer, John Murphy, William Boag, Wei-Hung Weng, Di Jindi, Tristan Naumann, and Matthew McDermott. 2019. Publicly available clinical BERT embeddings. In *Proceedings of the 2nd Clinical Natural Language Processing Workshop*, pages 72–78, Minneapolis, Minnesota, USA. Association for Computational Linguistics.

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 *European Conference on Computer Vision*, pages 104–120. Springer.

Jaemin Cho, Jie Lei, Hao Tan, and Mohit Bansal. 2021. Unifying vision-and-language tasks via text generation. In *Proceedings of the 38th International Conference on Machine Learning*, volume 139 of *Proceedings of Machine Learning Research*, pages 1931–1942. PMLR.

Michiel de Jong, Yury Zemlyanskiy, Nicholas FitzGerald, Fei Sha, and William W Cohen. 2021. Mention Memory: incorporating textual knowledge into Transformers through entity mention attention. In *International Conference on Learning Representations*.

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)*, pages 4171–4186, Minneapolis, Minnesota. Association for Computational Linguistics.

Julian Eisenschlos, Maharshi Gor, Thomas Müller, and William Cohen. 2021. MATE: Multi-view attention for table transformer efficiency. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 7606–7619, Online and Punta Cana, Dominican Republic. Association for Computational Linguistics.

Tianyu Gao, Xingcheng Yao, and Danqi Chen. 2021. SimCSE: Simple contrastive learning of sentence embeddings. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 6894–6910, Online and Punta Cana, Dominican Republic. Association for Computational Linguistics.

Y Gao, Z Duan, W Shi, J Feng, and Y-Y Chiang. 2019. Personalized Recommendation Method of POI Based on Deep Neural Network. In *Proceedings of the 2019 6th International Conference on Behavioral, Economic and Socio-Cultural Computing (BESC)*, pages 1–6.

Jonathan Herzig, Pawel Krzysztof Nowak, Thomas Müller, Francesco Piccinno, and Julian Eisenschlos. 2020. TaPas: Weakly supervised table parsing via pre-training. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 4320–4333, Online. Association for Computational Linguistics.

Hiroshi Iida, Dung Thai, Varun Manjunatha, and Mohit Iyyer. 2021. TABBIE: Pretrained representations of tabular data. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 3446–3456, Online. Association for Computational Linguistics.

Jyun-Yu Jiang, Xue Sun, Wei Wang, and Sean Young. 2019. Enhancing air quality prediction with social media and natural language processing. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 2627–2632, Florence, Italy. Association for Computational Linguistics.

Mandar Joshi, Danqi Chen, Yinhan Liu, Daniel S. Weld, Luke Zettlemoyer, and Omer Levy. 2020. SpanBERT: Improving pre-training by representing and predicting spans. *Transactions of the Association for Computational Linguistics*, 8:64–77.

Wonjae Kim, Bokyung Son, and Ildoo Kim. 2021. Vilt: Vision-and-language transformer without convolution or region supervision. In *International Conference on Machine Learning*, pages 5583–5594. PMLR.

Zhenzhong Lan, Mingda Chen, Sebastian Goodman, Kevin Gimpel, Piyush Sharma, and Radu Soricut. 2020. ALBERT: A lite BERT for self-supervised learning of language representations. In *8th International Conference on Learning Representations, ICLR 2020, Addis Ababa, Ethiopia, April 26-30, 2020*. OpenReview.net.

Jinhyuk Lee, Wonjin Yoon, Sungdong Kim, Donghyeon Kim, Sunkyu Kim, Chan Ho So, and Jaewoo Kang. 2020. Biobert: a pre-trained biomedical language representation model for biomedical text mining. *Bioinformatics*, 36(4):1234–1240.

Mike Lewis, Yinhan Liu, Naman Goyal, Marjan Ghazvininejad, Abdelrahman Mohamed, Omer Levy, Veselin Stoyanov, and Luke Zettlemoyer. 2020. BART: Denoising sequence-to-sequence pre-training for natural language generation, translation, and comprehension. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 7871–7880, Online. Association for Computational Linguistics.

Liunian Harold Li, Mark Yatskar, Da Yin, Cho-Jui Hsieh, and Kai-Wei Chang. 2020a. What does BERT with vision look at? In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 5265–5275, Online. Association for Computational Linguistics.

Xiujun Li, Xi Yin, Chunyuan Li, Pengchuan Zhang, Xiaowei Hu, Lei Zhang, Lijuan Wang, Houdong Hu, Li Dong, Furu Wei, et al. 2020b. OSCAR: Object-Semantics Aligned Pre-training for Vision-Language Tasks. In *European Conference on Computer Vision*, pages 121–137. Springer.

Yijun Lin, Yao-Yi Chiang, Meredith Franklin, Sandrah P. Eckel, and José Luis Ambite. 2020. Building Autocorrelation-Aware Representations for Fine-Scale Spatiotemporal Prediction. In *2020 IEEE International Conference on Data Mining (ICDM)*, pages 352–361.

Yijun Lin, Yao-Yi Chiang, Fan Pan, Dimitrios Stripelis, Jose Luis Ambite, Sandrah P Eckel, and Rima Habre. 2017. Mining Public Datasets for Modeling Intra-City PM2.5 Concentrations at a Fine Spatial Resolution. In *Proceedings of the 25th ACM SIGSPATIAL International Conference on Advances in Geographic Information Systems*, volume 8, pages 1–10, New York, NY, USA. ACM.

Yijun Lin, Nikhit Mago, Yu Gao, Yaguang Li, Yao-Yi Chiang, Cyrus Shahabi, and José Luis Ambite. 2018. Exploiting spatiotemporal patterns for accurate air quality forecasting using deep learning. In *Proceedings of the 26th ACM SIGSPATIAL International Conference on Advances in Geographic Information Systems*, SIGSPATIAL '18, pages 359–368, New York, NY, USA. Association for Computing Machinery.

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. *arXiv preprint arXiv:1907.11692*.

Zhuang Liu, Degen Huang, Kaiyu Huang, Zhuang Li, and Jun Zhao. 2021. FinBERT: A Pre-trained Financial Language Representation Model for Financial Text Mining. In *Proceedings of the Twenty-Ninth International Conference on International Joint Conferences on Artificial Intelligence*, pages 4513–4519.

Jiasen Lu, Dhruv Batra, Devi Parikh, and Stefan Lee. 2019. Vilbert: Pretraining task-agnostic visiolinguistic representations for vision-and-language tasks. In *Advances in Neural Information Processing Systems*, volume 32. Curran Associates, Inc.

Yifan Peng, Shankai Yan, and Zhiyong Lu. 2019. Transfer learning in biomedical natural language processing: An evaluation of BERT and ELMo on ten benchmarking datasets. In *Proceedings of the 18th BioNLP Workshop and Shared Task*, pages 58–65, Florence, Italy. Association for Computational Linguistics.

Matthew E. Peters, Mark Neumann, Mohit Iyyer, Matt Gardner, Christopher Clark, Kenton Lee, and Luke Zettlemoyer. 2018. Deep contextualized word representations. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*, pages 2227–2237, New Orleans, Louisiana. Association for Computational Linguistics.

Matthew E. Peters, Mark Neumann, Robert Logan, Roy Schwartz, Vidur Joshi, Sameer Singh, and Noah A. Smith. 2019. Knowledge enhanced contextual word representations. 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)*, pages 43–54, Hong Kong, China. Association for Computational Linguistics.

Long N Phan, James T Anibal, Hieu Tran, Shaurya Chanana, Erol Bahadroglu, Alec Peltekian, and Grégoire Altan-Bonnet. 2021. Scifive: a text-to-text transformer model for biomedical literature. *arXiv preprint arXiv:2106.03598*.

Xipeng Qiu, Tianxiang Sun, Yige Xu, Yunfan Shao, Ning Dai, and Xuanjing Huang. 2020. Pre-trained models for natural language processing: A survey. *arXiv preprint arXiv:2003.08271*.

Alec Radford, Jeffrey Wu, Rewon Child, David Luan, Dario Amodei, Ilya Sutskever, et al. 2019. Language models are unsupervised multitask learners. *OpenAI blog*, 1(8):9.

Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, Peter J Liu, et al. 2020. Exploring the limits of transfer learning with a unified text-to-text transformer. *J. Mach. Learn. Res.*, 21(140):1–67.

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)*, pages 3982–3992, Hong Kong, China. Association for Computational Linguistics.

Sascha Rothe, Shashi Narayan, and Aliaksei Severyn. 2020. Leveraging pre-trained checkpoints for sequence generation tasks. *Transactions of the Association for Computational Linguistics*, 8:264–280.

Victor Sanh, Lysandre Debut, Julien Chaumond, and Thomas Wolf. 2019. Distilbert, a distilled version of bert: smaller, faster, cheaper and lighter. *arXiv preprint arXiv:1910.01108*.

Weijie Su, Xizhou Zhu, Yue Cao, Bin Li, Lewei Lu, Furu Wei, and Jifeng Dai. 2019. VL-BERT: Pre-training of Generic Visual-Linguistic Representations. In *International Conference on Learning Representations*.

Jeniya Tabassum, Mounica Maddela, Wei Xu, and Alan Ritter. 2020. Code and named entity recognition in StackOverflow. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 4913–4926, Online. Association for Computational Linguistics.

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*, pages 442–451.

Fei Wang, Zhewei Xu, Pedro Szekely, and Muhao Chen. 2022. Robust (controlled) table-to-text generation with structure-aware equivariance learning. In *Proceedings of the 2022 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 5037–5048, Seattle, United States. Association for Computational Linguistics.

Ruize Wang, Duyu Tang, Nan Duan, Zhongyu Wei, Xuanjing Huang, Jianshu Ji, Guihong Cao, Daxin Jiang, and Ming Zhou. 2021. K-Adapter: Infusing Knowledge into Pre-Trained Models with Adapters. In *Findings of the Association for Computational Linguistics: ACL-IJCNLP 2021*, pages 1405–1418, Online. Association for Computational Linguistics.

Xiaokai Wei, Shen Wang, Dejiao Zhang, Parminder Bhatia, and Andrew Arnold. 2021. Knowledge enhanced pretrained language models: A compreshensive survey. *arXiv preprint arXiv:2110.08455*.

Ikuya Yamada, Akari Asai, Hiroyuki Shindo, Hideaki Takeda, and Yuji Matsumoto. 2020. LUKE: Deep contextualized entity representations with entity-aware self-attention. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 6442–6454, Online. Association for Computational Linguistics.

Sen Yang, Dawei Feng, Linbo Qiao, Zhigang Kan, and Dongsheng Li. 2019. Exploring pre-trained language models for event extraction and generation. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 5284–5294, Florence, Italy. Association for Computational Linguistics.

Hongzhi Yin, Weiqing Wang, Hao Wang, Ling Chen, and Xiaofang Zhou. 2017. Spatial-aware hierarchical collaborative deep learning for poi recommendation. *IEEE Transactions on Knowledge and Data Engineering*, 29(11):2537–2551.

Pengcheng Yin, Graham Neubig, Wen-tau Yih, and Sebastian Riedel. 2020. TaBERT: Pretraining for joint understanding of textual and tabular data. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 8413–8426, Online. Association for Computational Linguistics.

Jiahui Yu, Yuanzhong Xu, Jing Yu Koh, Thang Luong, Gunjan Baid, Zirui Wang, Vijay Vasudevan, Alexander Ku, Yinfei Yang, Burcu Karagol Ayan, Ben Hutchinson, Wei Han, Zarana Parekh, Xin Li, Han Zhang, Jason Baldridge, and Yonghui Wu. 2022. Scaling Autoregressive Models for Content-Rich Text-to-Image Generation.

Haitao Yuan and Guoliang Li. 2021. A survey of traffic prediction: from spatio-temporal data to intelligent transportation. *Data Science and Engineering*, 6(1):63–85.

M Yue, Y Li, H Yang, R Ahuja, Y Chiang, and C Shahabi. 2019. DETECT: Deep Trajectory Clustering for Mobility-Behavior Analysis. In *Proceedings of the 2019 IEEE International Conference on Big Data (Big Data)*, pages 988–997.

Mingxuan Yue, Yao-Yi Chiang, and Cyrus Shahabi. 2021. VAMBC: A Variational Approach for Mobility Behavior Clustering. In *Machine Learning and Knowledge Discovery in Databases. Applied Data Science Track*, pages 453–469. Springer International Publishing.

Xiaohua Zhai, Xiao Wang, Basil Mustafa, Andreas Steiner, Daniel Keysers, Alexander Kolesnikov, and Lucas Beyer. 2022. LiT: Zero-Shot Transfer With Locked-Image Text Tuning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 18123–18133.

Bowen Zhang, Hexiang Hu, Vihan Jain, Eugene Ie, and Fei Sha. 2020. Learning to represent image and text with denotation graph. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 823–839, Online. Association for Computational Linguistics.

Bowen Zhang, Hexiang Hu, Linlu Qiu, Peter Shaw, and Fei Sha. 2021. Visually grounded concept composition. In *Findings of the Association for Computational Linguistics: EMNLP 2021*, pages 201–215, Punta Cana, Dominican Republic. Association for Computational Linguistics.

Zhengyan Zhang, Xu Han, Zhiyuan Liu, Xin Jiang, Maosong Sun, and Qun Liu. 2019. ERNIE: Enhanced language representation with informative entities. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 1441–1451, Florence, Italy. Association for Computational Linguistics.

Pengpeng Zhao, Anjing Luo, Yanchi Liu, Jiajie Xu, Zhixu Li, Fuzhen Zhuang, Victor S. Sheng, and Xiaofang Zhou. 2022. Where to Go Next: A Spatio-Temporal Gated Network for Next POI Recommendation. *IEEE Transactions on Knowledge and Data Engineering*, 34(5):2512–2524.

Guangyu Zhou, Muhao Chen, Chelsea JT Ju, Zheng Wang, Jyun-Yu Jiang, and Wei Wang. 2020. Mutation effect estimation on protein–protein interactions using deep contextualized representation learning. *NAR genomics and bioinformatics*, 2(2):lqaa015.

Wenxuan Zhou, Fangyu Liu, Ivan Vulić, Nigel Collier, and Muhao Chen. 2022. Prix-LM: Pretraining for multilingual knowledge base construction. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long*

Papers), pages 5412–5424, Dublin, Ireland. Association for Computational Linguistics.