frontiers in Big Data

ORIGINAL RESEARCH
published: 20 October 2021
doi: 10.3389/fdata.2021.762899

Towards Semantically-Rich Spatial Network Representation Learning *via* Automated Feature Topic Pairing

Dongjie Wang[1], *Kunpeng Liu*[1], *David Mohaisen*[1], *Pengyang Wang*[2], *Chang-Tien Lu*[3] *and Yanjie Fu*[1]*

[1]Computer Science Department, University of Central Florida, Orlando, FL, United States, [2]Computer Science Department, University of Macau, Macau, China, [3]Computer Science Department, Virginia Tech Falls Church, Falls Church, VA, United States

Automated characterization of spatial data is a kind of critical geographical intelligence. As an emerging technique for characterization, spatial Representation Learning (SRL) uses deep neural networks (DNNs) to learn non-linear embedded features of spatial data for characterization. However, SRL extracts features by internal layers of DNNs, and thus suffers from lacking semantic labels. Texts of spatial entities, on the other hand, provide semantic understanding of latent feature labels, but is insensible to deep SRL models. How can we teach a SRL model to discover appropriate topic labels in texts and pair learned features with the labels? This paper formulates a new problem: feature-topic pairing, and proposes a novel Particle Swarm Optimization (PSO) based deep learning framework. Specifically, we formulate the feature-topic pairing problem into an automated alignment task between 1) a latent embedding feature space and 2) a textual semantic topic space. We decompose the alignment of the two spaces into: 1) point-wise alignment, denoting the correlation between a topic distribution and an embedding vector; 2) pair-wise alignment, denoting the consistency between a feature-feature similarity matrix and a topic-topic similarity matrix. We design a PSO based solver to simultaneously select an optimal set of topics and learn corresponding features based on the selected topics. We develop a closed loop algorithm to iterate between 1) minimizing losses of representation reconstruction and feature-topic alignment and 2) searching the best topics. Finally, we present extensive experiments to demonstrate the enhanced performance of our method.

Keywords: feature-topic pairing, semantic space, spatial space, spatial representation learning, spatial graph

OPEN ACCESS

Edited by:
Xun Zhou,
The University of Iowa, United States

Reviewed by:
Yiqun Xie,
University of Minnesota Twin Cities,
United States
Amin Vahedian Khezerlou,
University of Wisconsin–Whitewater,
United States

****Correspondence:***
Yanjie Fu
yanjie.fu@ucf.edu

Specialty section:
This article was submitted to
Data Mining and Management,
a section of the journal
Frontiers in Big Data

Received: *23 August 2021*
Accepted: *20 September 2021*
Published: *20 October 2021*

Citation:
Wang D, Liu K, Mohaisen D, Wang P, Lu C-T and Fu Y (2021) Towards Semantically-Rich Spatial Network Representation Learning via *Automated Feature Topic Pairing. Front. Big Data 4:762899. doi: 10.3389/fdata.2021.762899*

1 INTRODUCTION

Critical infrastructures (e.g., transportation networks, power networks, social networks, water supply networks) often consist of spatially distributed entities that interact with each other, and have generated massive spatial-networked behavior data. Analyzing such data can identify trends, forecast future behavior, and detect anomalies. To enable effective analysis, it is critical to desire a new capability of automated characterization that effectively extract feature vectors from spatio-networked data.

As one of the emerging techniques, representation learning can be adapted to learn non-linear embedded features of spatial network data, which we call spatial representation learning (SRL). There has been a rich body in SRL, including node embedding, autoencoder, random walk, adversarial learning, generative learning based methods with spatial data (Wang and Li, 2017; Wang et al., 2018a;

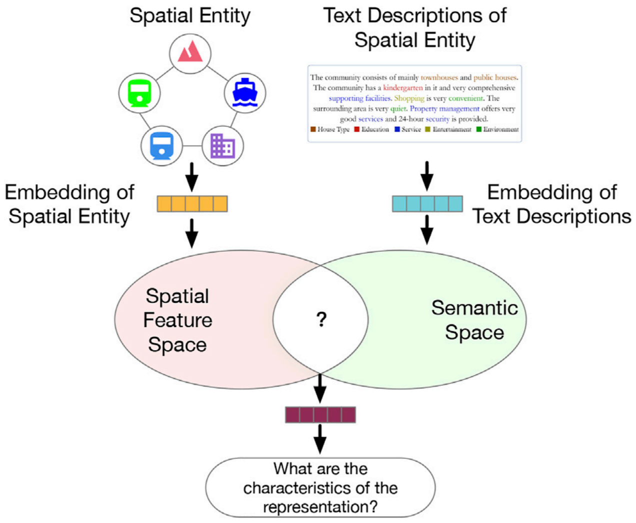

Spatial Entity
Text Descriptions of Spatial Entity
The community consists of mainly townhouses and public houses. The community has a kindergarten in it and very comprehensive supporting facilities. Shopping is very convenient. The surrounding area is very quiet. Property management offers very good services and 24-hour security is provided.
House Type
Education
Service
Entertainment
Environment
Embedding of Spatial Entity
Embedding of Text Descriptions
Spatial Feature Space
?
Semantic Space
What are the characteristics of the representation?

FIGURE 1 | The motivation of the feature-topic pairing problem: bridging the gap between feature embedding space and topic semantic space in representation learning.

Wang et al., 2018b; Chandra et al., 2019; Jean et al., 2019; Wang et al., 2019a; Wang et al., 2019b; Zhang Y. et al., 2019; Shan et al., 2020; Wang et al., 2020c; Wang et al., 2020d; Wang et al., 2021). Although these works achieved remarkable success, the model interpretability is still a big limitation that hinders these SRL methods from applying in more secure, fair, and rigorous scenarios.

Lacking model interpretability is possible to cause damaging or controversial consequences in incomplete scenarios that are not well-studied (Doshi-Velez and Kim, 2017). For instance, in the autonomous driving scenario, the end-to-end autopilot system brings high safety risks for drivers[1]. In 2015, Google's photo app classifies images of black people as gorillas, which exposes the limitation of algorithms[2]. More seriously, widely used crime prediction software prefers to provide higher risk scores of future crimes for black defendants[3]. Model interpretability is one of the most important approaches to overcome these limitations. Thus how to enhance the model interpretability attracts much attention of researchers (Elshawi et al., 2019; Hong et al., 2020; Stiglic et al., 2020; Poursabzi-Sangdeh et al., 2021). But, many existing works reflect that there is a trade-off between model performance and model interpretability (Mori and Uchihira, 2019; Saisubramanian et al., 2020). Can we not only improve the model interpretability but also keep the model performance becomes the research point of this paper.

To relieve the limitations of prior literature and expand the application scenarios of SRL approaches, a novel SRL model should understand not just which features are effective, but also what these effective features stand for. This issue relates to two tasks: 1) deep representation learning; 2) label generation and matching for latent embedded features. In response, we formulate the problem as a task of feature-topic pairing (**Figure 1**), which is to align a latent embedding feature space, consisting of multiple latent features, and a textual semantic topic space, consisting of multiple topic labels during SRL. The basic idea is to teach a machine to extract topic labels from texts, and then pair the labels with learned features. To that end, we propose to develop a novel deep learning framework to unify feature learning, topic selection, feature-topic matching.

There are three unique challenges (**Figure 2**) in addressing this problem. 1) Label Generation Challenge. The semantically-rich texts of spatial entities describe their types, functions, and attribute-related information. For instance, on a real estate website, the texts of a residential community describe crime rates and events, great school ratings, nearby transportation facilities, grocery stores, companies, and universities. These texts, if properly analyzed, will help to identify which underlying features truly attract residents to pay more to live. However, these spatial texts are all unstructured, how can we construct a textual semantic topic space for spatial entities to support feature-topic pairing? 2) Measurement Challenge. Be sure to note that we aim to teach a machine to automatically perform the automated pairing between embedded features and topic labels in a self-optimizing fashion. As a result, a measurement is needed to quantify the alignment or matching score between the topic label space and the embedding feature space, in order to guide the machine about how to search. However, there is no standard measurement for quantifying the topic-embedding space alignment. Thus, what does form of measurement should be adopted? And how can we integrate the suitable measurement into the whole self-optimizing framework? 3) Optimization Challenge. Since the model needs to decide an optimized topic label subset, the feature-topic pairing problem evolves multiple machine learning tasks, including feature learning, topic label selection, and feature-topic matching. If the three tasks are separately completed step by step, there is no guarantee that they are globally optimized. So, how can we develop a deep optimization framework to jointly and simultaneously unify the three tasks?

To solve the three challenges, we develop a new PSO-based framework (named AutoFTP) that enclose the optimizations of feature learning, topic selection, and feature-topic pairing in a loop. Specifically, our contributions are:

1) Formulating the feature-topic pairing problem. Motivated by lacking feature labels in SRL, we formulate and develop a new problem: feature topic pairing. In the proposed model, we propose a new strategy: we first let an optimizer to automatically select K topics; the optimizer then guides the representation learner to learn K latent features that optimally align with the K topics.
2) Generating candidate topic labels. We propose a three step mining method to generate candidate topic labels. Specifically, we first extract keywords from the texts of all spatial entities.

[1] https://www.brookings.edu/blog/techtank/2016/08/18/autopilot-fatality-reveals-risks-of-technology-testing/
[2] https://www.wsj.com/articles/BL-DGB-42522
[3] https://www.propublica.org/article/machine-bias-risk-assessments-in-criminal-sentencing

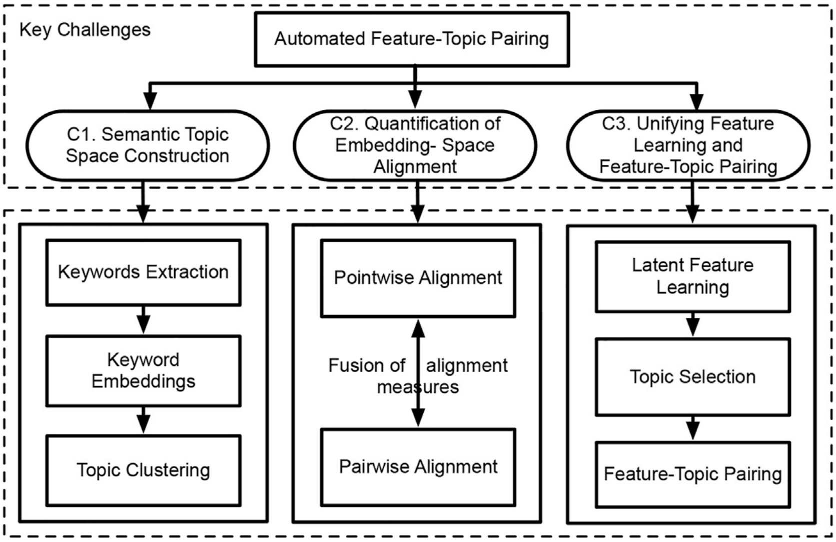

Key Challenges
Automated Feature-Topic Pairing
C1. Semantic Topic Space Construction
C2. Quantification of Embedding- Space Alignment
C3. Unifying Feature Learning and Feature-Topic Pairing
Keywords Extraction
Keyword Embeddings
Topic Clustering
Pointwise Alignment
Fusion of alignment measures
Pairwise Alignment
Latent Feature Learning
Topic Selection
Feature-Topic Pairing

FIGURE 2 | Key challenges and main tasks of the feature-topic pairing problem.

Then, we learn keyword embedding feature vectors with a pre-trained word model (He, 2014). Finally, we cluster all keyword embeddings by maximizing inter-topic distances and minimizing intra-topic distances to generate topics as candidate feature labels.

3) Quantifying feature-topic alignment measurement. We identify two types of feature-topic alignments: 1) point-wise alignment, and 2) pair-wise alignment. First, the point-wise alignment is to describe the correlation between an embedding feature vector and a categorical topic distribution. In particular, we maximize the correlation so that the distance between the distribution of the embedding vector space and the distribution of the topic semantic vector space can be minimized. The underlying motivation of point-wise alignment is originated from the fact that: if a topic density is high in describing a spatial entity, the topic's corresponding feature value is expected to be large to co-vary with the topic density. In this way, we align the distribution covariance of the two spaces. Second, the pair-wise alignment is to describe the consistency between a feature-feature similarity matrix and a topic-topic similarity matrix. In particular, we use the feature-feature similarity graph to describe the topology of the latent embedding feature space, and use the topic-topic similarity graph to describe the topology of the textual semantic topic space. If the two spaces are aligned, the two graphs (represented by matrices) are similar as well.
4) Optimization in the loop. We develop a Particle Swarm Optimization (PSO)-based algorithm. In this algorithm, we first simultaneously optimize the representation learning loss, point-wise alignment loss, pair-wise alignment loss, and downstream task loss as the feedback for PSO. Guided by the feedback, the PSO based algorithm selects a better K-sized topic subset for feature-topic pairing. In particular, based on the loss function value, PSO iteratively generates topic masks (i.e., 0–1 indicators to select or deselect) to search the optimal topics for space pairing until the learning objective converges. In this way, the PSO jointly achieves topic selection, feature-topic pairing, and latent feature learning.

Finally, we evaluate our method using Beijing's urban geography and mobility data. For comparison we implemented a broad range of other algorithms. Results showed that our method consistently outperformed the competing methods. We perform ablation study, interpretability, robustness check, stability, sensitivity to justify our technical insights.

2 PRELIMINARIES AND PROBLEM STATEMENT

In this section, we introduce key definitions of AutoFTP and the problem statement.

2.1 Particle Swarm Optimization

PSO is a heuristic optimization algorithm that finds an optimal solution in a dynamic environment, by imitating the social activity of a flock of birds. **Figure 3** shows the origin of PSO. A flock of eagles wants to capture a rabbit. To achieve the goal, all eagles exchange information related to the position of the rabbit. Each eagle updates its position based on its current status, velocity, the position where it knew is closest to the rabbit, and the position where the flock knew is closest to the rabbit, until the rabbit is captured.

Similarly, solving the feature-topic pairing problem can be analogized as a task of searching the optimal matching solutions in a dynamic environment. Specifically, we can view the eagles as a set of binary topic selector, which are to select the optimized subset of topics from a candidate topic set for feature-topic pairing. The choices of these binary topic selectors are iteratively updated in order to converge into the ultimate most matched topic-feature pairs.

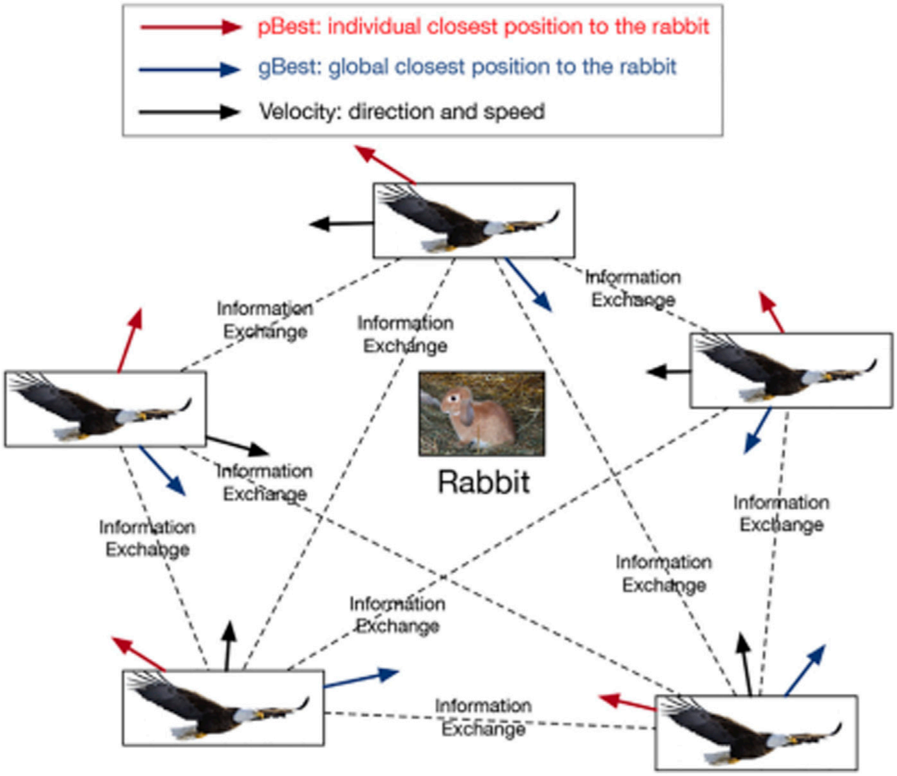


FIGURE 3 | The origin of PSO: a flock of eagles is preying on a rabbit. To capturing the rabbit quickly, each eagle records where is the closest position to the rabbit during its history exploration (pBest). Meanwhile, all eagles share the closest position to the rabbit where the flock knew (gBest). All eagles explore any position based on their velocity, pBest, and gBest until the rabbit is captured.

During the iterative process, all the binary topic selectors jointly share the changes of objective function losses (i.e., the losses of representation construction, feature-topic alignment, and downstream predictive task), so the topic selector knows how to update the topic selection next round.

2.2 Definitions

Definition 1: Spatial Entity. A spatial entity is a geographical concept that consists of a range (e.g. a circle area with a radius of 1 mile) and a location (i.e. the latitude and longitude of a center). The spatial entity also includes various Points-of-Interest (POIs) of different categories (e.g., buildings of education, shopping, medical, banking, etc.).

Definition 2: Point-wise Alignment. To tackle feature-topic pairing, we assume there are 1) an embedding vector that describes the features of a spatial entity and 2) a corresponding topic distribution associated to a spatial entity, which are extracted by optimization. To achieve feature-topic alignment, we propose a point-wise alignment to describe the correlation between features and topics. **Figure 4A** shows an example of point-wise alignment, we expect to maximize the correlation between the selected topic vector and the spatial

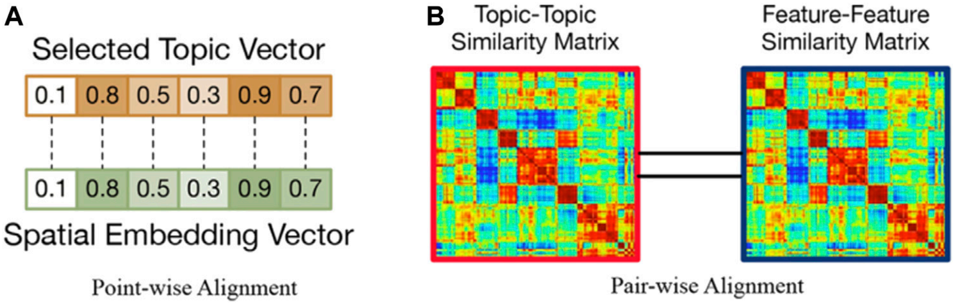


FIGURE 4 | Two kinds of space alignment.

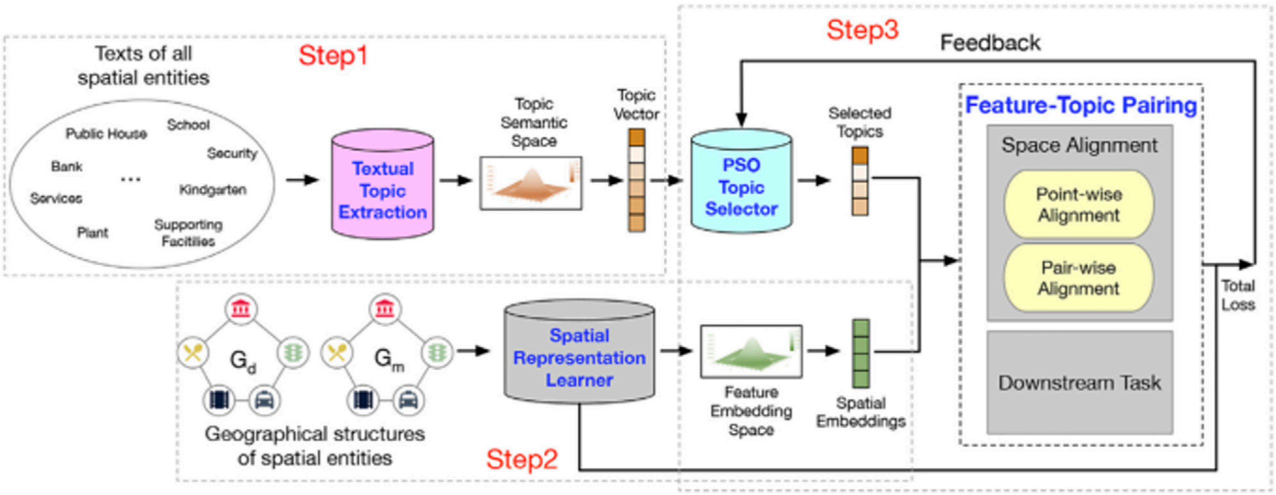


FIGURE 5 | An overview of AutoFTP. In the framework, we first construct a topic semantic space based on the texts of spatial entities. Then, we initialize a embedding feature space based on the geographical structures of spatial entities. Later, we employ a PSO-based framework to conduct feature-topic pairing through jointly optimizing representation learning, point-wise alignment, pair-wise alignment, and downstream task over learning iterations.

embedding vector. The larger the correlation between the two vectors is, the larger the similarity between the two vectors larger is.

Definition 3: Pair-wise Alignment. We propose another perspective (pair-wise) to model the feature-topic alignment. For each entity-entity pair, we compute their feature-feature similarity and topic-topic similarity, and obtain: 1) a topic-topic similarity matrix **S**; 2) a feature-feature similarity matrix $\mathbf{S}'$. We measure the consistency between the two matrices as the pairwise alignment.

Figure 4B shows an example of pair-wise alignment, we aim to let the topic-topic similarity matrix S as close as the feature-feature similarity matrix S′ possible.

2.3 The Feature-topic Pairing Problem

The feature-topic pairing problem aims to pair the latent features extracted by representation learning, with the explicit topics of texts of a spatial entity. Formally, given a set of N spatial entities, the n-th entity is described by multiple graphs (e.g., a POI-POI distance graph G_n^d and a POI mobility connectivity G_n^m, defined in **Section 3.3**) and a topic distribution $\mathbf{t}_n$ extracted from textual descriptions $\mathcal{E}_n$. Let $\tilde{\mathbf{r}}_n$ be the embedding vector of the n-th entity. The objective is to optimize a function that measures representation loss and feature-topic alignment:

$$\tilde{\mathbf{R}} = \mathbf{argmax}_{\tilde{\mathbf{r}}_n} \sum_{n=1}^{N} f\left(\tilde{\mathbf{r}}_n | \mathbf{t}_n, G_n^d, G_n^m, K\right), \qquad (1)$$

where $\tilde{\mathbf{R}} = \{\tilde{\mathbf{r}}_n\}_{n=1}^{N} \in \mathbb{R}^{N\times K}$ are the embeddings of all spatial entities, K is the number features of an embedding vector.

3 THE PROPOSED METHOD-AUTOFTP

In this section, we first introduce an overview of our AutoFTP framework, then present its technical details.

3.1 Framework Overview

Figure 5 shows our proposed framework. First, we construct a semantic topic space by extracting topic distribution from the corresponding texts of spatial entities. Then, we initialize a feature embedding space based on the geographical structures of spatial entities. Next, we utilize a PSO-based topic selector to select the optimal K topics for pairing with the spatial embeddings coming from the feature embedding space. During the pairing process, the losses of spatial representation learner, point-wise alignment, pair-wise alignment, and downstream tasks are regarded as feedback to update the topic selector for the next pairing iteration. With the development of the learning iteration, the feature embedding space aligns to the topic semantic space gradually. Finally, the learned spatial embeddings of AutoFTP are effective and semantically rich. Here, to validate the effectiveness of AutoFTP, we apply the framework to predict the real estate price (downstream tasks) of the residential communities (spatial entities) based on spatial embeddings of the communities. The more accurate the prediction is, the more effective the learned embedding is. In addition, the AutoFTP can be generalized to other spatial representation learning problems with graphs and texts.

3.2 Textual Topic Extraction

To derive the textual semantic topic space, we extract the topic distributions of spatial entities from texts generated by location based social networks. Traditional topic models, such as LDA (Blei et al., 2003), PSLA (Hofmann, 2013), are implemented based on bag-of-words. These methods ignore word orders in sentences. To improve the performances of topic modeling, we employ a pre-trained deep word embedding model (He, 2014) to generate topics.

As illustrated in **Figure 6**, we first collect the text descriptions of all entities. Besides, we extract keywords from texts using the TextRank algorithm (Mihalcea and Tarau, 2004) and leverage a

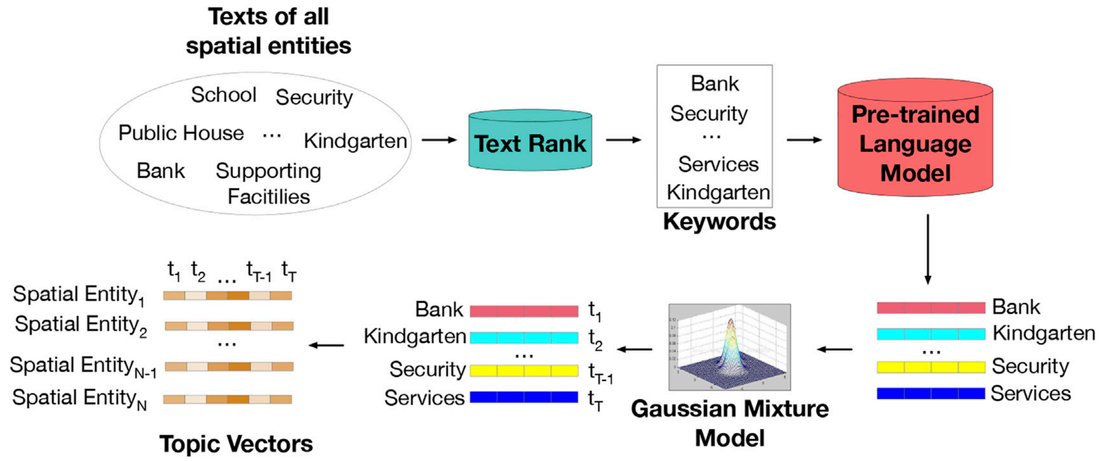


FIGURE 6 | Obtaining the topic vectors of spatial entities.

pre-trained language model (He, 2014) to learn the corresponding word embedding of each keyword. Moreover, we exploit a Gaussian Mixture Model (GMM) to cluster the keyword embeddings into T topics. The clustering model provides a topic label for each keyword. To explain the labeling process, we take the i-th keyword's embedding vector $\mathbf{x}_i$ as an example. First, we assume that the T topics obey a Gaussian Mixture Distribution (GMD). Then we randomly initialize the parameters of GMD. Next, we use the Expectation Maximization (EM) algorithm to find the optimal parameters of the GMD. Finally, we calculate the probability of $\mathbf{x}_i$ (a.k.a., membership), belonging to each topic based on the GMD, and select the topic with the highest probability as the label of $\mathbf{x}_i$. After that, we propose to construct the topic distribution vector of each spatial entity. In particular, for the n-th entity, the topic vector $\mathbf{t}_n$ is a T dimensional vector, where each dimension indicates a topic, and is filled by the number of associated keywords.

3.3 Graph Extraction of Spatial Entities

In order to learn the embedding feature vectors of spatial entities, we propose to construct the graph-structured topology of each spatial entity. This is because there is inherent spatial autocorrelation between each two spatial entities, according to the geographical first law. We describe a spatial entity in terms of its POIs, by building two graphs. 1) POI-POI distance graph: denoted by G^d, where POI categories are nodes and the average distances between POI categories are edge weights. 2) **POI**-POI mobility graph: denoted by G^m, where nodes are POI categories, and edge weights are human mobility connectivity. The number of POI categories in this paper is M, and the two graphs are extracted via the method in (Wang et al., 2018a). Specifically, we first use a parametric function to estimate POI visit probability based on a taxi GPS trace data: $P(\varsigma) = \frac{\beta_1}{\beta_2} \cdot \varsigma \cdot exp\left(1 - \frac{\varsigma}{\beta_2}\right)$, where ς denotes the distance between a POI and a drop-off position in a taxi trace, $\beta_1 = \max_\varsigma P(\varsigma)$, and $\beta_2 = \arg\max_\varsigma P(\varsigma)$. We calculate the visited probability of all POIs according to the formula. We sum up the probability of POIs belonging to the same POI category to calculate the visited probability of the POI category. Finally, we calculate the connectivity strength between POI categories as: $C_{\overrightarrow{ij}} = \begin{cases} P_i \cdot P_j, & if \;\; i \neq j \\ 0, & otherwise, \end{cases}$ where P_i and P_j represent the visited probability of POI category i and POI category j respectively; $C_{\overrightarrow{ij}}$ indicates the connectivity between POI category i and j.

3.4 Particle Swarm Optimization Based Feature-Topic Pairing

3.4.1 Spatial Representation Learner

To learn the representations of spatial entities, we utilize the Graph Auto Encoder (GAE) (Kipf and Welling, 2016) to construct latent embedding space. Specifically, to learn the embedding feature vector of the n-th entity, the encoder has two GCN layers. The encoding calculation process can be formulated as follows:

$$\begin{cases} \hat{\mathbf{A}}_n = \mathbf{A}_n + \mathbf{I}_n, \\ \tilde{\mathbf{A}}_n = \hat{\boldsymbol{D}}_n^{-\frac{1}{2}} \hat{\mathbf{A}}_n \hat{\boldsymbol{D}}_n^{-\frac{1}{2}}, \\ \mathbf{z}_n = \tilde{\mathbf{A}}_n \mathrm{Relu}\left(\tilde{\mathbf{A}}_n \mathbf{X}_n \mathbf{W}_n^{(0)}\right) \mathbf{W}_n^{(1)} \end{cases} \tag{2}$$

where $\mathbf{A}_n, \mathbf{I}_n, \tilde{\boldsymbol{A}}_n, \hat{\boldsymbol{D}}_n$ own the same shape $\mathbb{R}^{M \times M}$. Moreover, $\mathbf{A}_n$ is the adjacency matrix, $\mathbf{I}_n$ is the identity matrix, $\tilde{\boldsymbol{A}}_n$ is the symmetrically normalized adjacency matrix, $\hat{\boldsymbol{D}}_n$ is the degree matrix. In addition, $\mathbf{X}_n \in \mathbb{R}^{M \times U}$ is the feature matrix of the graph, in which U is the feature dimension; $\mathbf{W}_n^{(1)} \in \mathbb{R}^{U \times H}$ is the weight matrix of the first GCN layer, in which H is the output dimension of the layer; $\mathbf{W}_n^{(2)} \in \mathbb{R}^{H\times K}$ is the weight matrix of the second GCN layer; $\mathbf{z}_n \in \mathbb{R}^{M\times K}$ is the output embedding of the encoder. The decoder recovers the adjacency matrix according to $\mathbf{z}_n$:

$$\hat{\mathbf{A}}^* = \sigma\left(\mathbf{z}_n \mathbf{z}_n'\right). \tag{3}$$

The optimization objective is to minimize the reconstruction loss between the original graph, denoted by the adjacency matrix $\hat{\mathbf{A}}_n$, and the reconstructed graph, denoted by the adjacency matrix $\hat{\mathbf{A}}_n^*$:

$$\mathcal{L}_R = \sum_{n=1}^{N} \|\left(\hat{\mathbf{A}}_n - \hat{\mathbf{A}}_n^*\right)\|^2 \tag{4}$$

We apply the GAE to the POI-POI distance graph G_n^d and the POI-POI mobility graph G_n^m of the n-th spatial entity. After that, we obtain the node representations of G_n^d and G_n^m, denoted by $\mathbf{z}_{n^d} \in \mathbb{R}^{M\times K}$ and $\mathbf{z}_{n^m} \in \mathbb{R}^{M\times K}$. Then, we aggregate $\mathbf{z}_{n^d}$ and $\mathbf{z}_{n^m}$ by averaging all node embeddings together to attain the graph embedding of G_n^d and G_n^m respectively. Finally, we integrate the graph embeddings of G_n^d and G_n^m into the unified spatial embedding of the entity by averaging calculation, denoted by $\mathbf{r}_n \in \mathbb{R}^K$.

3.4.2 Measuring the Alignment of Embedding and Semantic Spaces

To pair features with topics, we conduct space alignment from the point-wise and pair-wise perspectives. Referring to definitions **Section 2.2** and **Section 2.3**, we aim to align the topic semantic space and feature embedding space from the coordinate system and information contents respectively. During the aligning process, we minimize the point-wise alignment loss $\mathcal{L}_P$ and pair-wise alignment loss $\mathcal{L}_C$. To be convenient, we take the n-th entity as an example to explain the calculation process.

1) Point-wise Alignment Loss: $\mathcal{L}_P$. We first select K values from the topic vector $\mathbf{t}_n$ as the vector $\check{\mathbf{t}}_n \in \mathbb{R}^K$, which contains the most representative semantics in the semantic space. Then, we maximize the correlation between $\check{\mathbf{t}}_n$ and the spatial embedding $\mathbf{r}_n$, which is equal to minimize the negative correlation between the two vectors. The formula of the minimizing process as follows:

$$\mathcal{L}_P = -\sum_{n=1}^{N} \frac{\operatorname{cov}\left(\check{\mathbf{t}}_n, \mathbf{r}_n\right)}{\delta\left(\check{\mathbf{t}}_n\right)\delta\left(\mathbf{r}_n\right)}, \tag{5}$$

where cov(.) denotes the covariance calculation; $\delta(.)$ denotes the standard deviation.

2) Pair-wise Alignment Loss: $\mathcal{L}_C$. We first construct the topic-topic similarity matrix $\mathbf{S}$ and the feature-feature similarity matrix $\mathbf{S}'$. Specifically, for $\mathbf{S} \in \mathbb{R}^{K\times K}$, we calculate the similarity between any two topics. For $\mathbf{S}' \in \mathbb{R}^{K\times K}$, we calculate the similarity between two features of spatial embeddings. We keep the pair-wise consistency between $\mathbf{S}$ and $\mathbf{S}'$ by minimizing the Frobenius norm, as follows:

$$\mathcal{L}_C = \|\mathbf{S} - \mathbf{S}'\|_F. \tag{6}$$

3.4.3 Supervised PSO For Automatic Topic Selection

As introduced above, we select K topics so the representation learner can learn a K-sized embedding vector in terms of K topics to achieve feature-topic alignment. However, how can the machine automatically identify the best K and select the most appropriate K topics?

A naive idea is that we can select K topics randomly at each iteration until we traverse all topic combinations and find the best topic subset based on the objective function. The searching process, however, is time-consuming and computationally expensive. Moreover, the topic selection problem belongs to the combinatorial optimization field, which is hard to solve by derivative-based optimization algorithms. Thus, a quickly and derivative-free optimization algorithm should be selected as our optimizer. Considering the high time complexity for traversing all possible subsets to find the optimal result, we propose to formulate the joint task of feature learning, topic selection, topic and feature matching into a PSO problem.

The PSO-based optimization framework is as illustrated in **Figure 7**. Specifically, we first randomly initialize a number of particles in PSO, where a particle is a binary topic mask (i.e., the mask value of 1 indicates “select” and the mask value of 0 indicates “deselect”). In other words, a set of particles select a subset of topics. A multi-objective deep learning model, whose objective function includes the losses of graph reconstruction, semantic alignment, and the regression estimator in the downstream task, is trained to learn spatial representations, using each selected topic subset. As an application, we use the embedding of spatial entities (residential communities) to predict their real estate prices, and the loss of the regression model $\mathcal{L}_{Reg}$ is:

$$\mathcal{L}_{Reg} = \frac{1}{N}\sum_{n=1}^{N} \left(c_n - c_n^*\right)^2, \tag{7}$$

where c_n is the golden standard real estate price and c_n^* is the predicted price. Next, we calculate the fitness of each particle according to the total loss of the deep model. The fitness can be calculated by:

$$Fitness = \mathcal{L}_C + \mathcal{L}_P + \mathcal{L}_R + \mathcal{L}_{Reg}. \tag{8}$$

Then, we utilize the fitness to inform all particles how far they are from the best solution. Next, each particle moves forward to the solution based on not only its current status but also all particles’ movement. After the fitness value of PSO converges, PSO identifies the best topic subset. Finally, the semantically-rich embeddings of spatial entities, given by: $\tilde{\mathbf{R}} = \{\tilde{\mathbf{r}}_n\}_{n=1}^N$.

4 EXPERIMENTAL RESULTS

In this section, we present extensive experiments with real world data to answer the following research questions: Q1. How

TABLE 1 | Statistics of the experimental data.

Data sources	Properties	Statistics
Taxi Traces	Number of taxis	13,597
	Time period	Apr.–Aug. 2012
Residential regions	Number of residential regions	2,990
	Time period of transactions	04/2011–09/2012
POIs	Number of POIs	328,668
	Number of POI categories	20
Texts	Number of textual descriptions	2,990
	Time Period	04/2011–09/2012
Real Estate Prices	Number of real estate prices	41,753
	Time Period	12/2011–06/2012

effective is our proposed AutoFTP in spatial representation learning? Q2. How effective is each loss function of AutoFTP? Are they all necessary for spatial representation learning? Q3. How well does AutoFTP work in space alignment? Do the learned spatial embeddings contain semantic meanings? Q4. How about the robustness of AutoFTP? Does it always outperform other baselines? Q5. How about the stability and sensitivity of AutoFTP?

4.1 Experimental Setup

4.1.1 Data Description

Table 1 shows the statistics of five data sources used in the experiments. Firstly, the taxi traces data describes the GPS trajectory of taxis in Beijing in 3 months. The format of each trace record is < trip id, distance, travel time, average speed, pick-up time, drop-off time, pick-up location, drop-off location >. Secondly, the residential regions, texts, and real estate price data sources are crawled from www.fang.com. In experiments, the residential regions are treated as spatial entities. The texts reflect the urban utilities and characteristics of spatial entities from multiple perspectives such as traffic condition, economic development, demographic situation, and etc. The real estate prices indicate the average value of the real estate of each spatial entity in 6 months. Thirdly, the POIs are extracted from www.dianping.com, which is a POI (small businesses such as restaurants, banks, gas stations, shopping markets) review website in China. Each POI is described in a format of < POI id, POI category, latitude, longitude >.

4.1.2 Application: Real Estate Price Prediction

Our proposed method (AutoFTP) can learn a list of vectorized representations for all spatial entities. Therefore, as a downstream application, we can apply these representations to train a regression model to predict the average real estate price of these spatial entities. Specifically, we first apply AutoFTP to learn a series of representations of spatial entities based on their geographical structural information and related text descriptions. Then, we build up a deep neural network (DNN) model for predicting average real estate price of each spatial entity according to its corresponding representation. To be convenient, we take the n-th spatial entity as an example to explain the regression model. The formulation of DNN is $f(\tilde{\mathbf{r}}_n, \mathbf{w}) = \mathbf{w} \cdot g(\tilde{\mathbf{r}}_n) + \mathbf{b}$, where $\tilde{\mathbf{r}}_n$ is the representation of the n-th spatial entity, $g(\tilde{\mathbf{r}}_n)$ is the nonlinear transformation of $\tilde{\mathbf{r}}_n$, $\mathbf{w}$ is the weight term, and $\mathbf{b}$ is the bias term. We want to minimize the difference between predicted price $f(\tilde{\mathbf{r}}_n, \mathbf{w})$ and real price y_n. Thus, the objective of the DNN is $\min \frac{1}{N}\sum_{n=1}^{N}(y_n - f(\tilde{\mathbf{r}}_n, \mathbf{w}))^2$, where N is the total number of spatial entities.

4.1.3 Evaluation Metrics

We evaluated our method using a real estate price prediction task (**Section 4.1.2**). We took the feature representation vectors of residential communities as inputs, and predicted their real estate prices. We compared the golden-standard prices y_n with the predicted prices $\hat{y}_n$ in terms of four metrics: 1) $\mathrm{RMSE} = \sqrt{\frac{1}{N}\sum_{n=1}^{N}(y_n - \hat{y}_n)^2}$; 2) $\mathrm{MAE} = \frac{1}{N}\sum_{n=1}^{N}|(y_n - \hat{y}_n)|$; 3) $\mathrm{MAPE} = \frac{100}{N}\sum_{n=1}^{N}|\frac{y_n - \widehat{y_n}}{y_n}|$; 4) $\mathrm{MSLE} = \frac{1}{N}\sum_{n=1}^{N}(log(1+y_n) - log(\hat{y}_n + 1))^2$. The regression loss and optimization algorithm are controlled to be the same. The lower the four metrics are, the more effective the spatial embedding features are.

4.1.4 Baseline Algorithms

We compared our proposed method with seven widely-used and robust representation learning (embedding) methods as follows: 1) AttentionWalk (Abu-El-Haija et al., 2018) utilizes a novel attention model to automatically learn the hyper-parameters of random-walk based network embedding methods, which improves the flexibility and performance of the model. We set the learning rate as 0.01, the regularization parameters as 0.5. 2) ProNE (Zhang J. et al., 2019) formulates the network embedding as sparse matrix factorization to improve the calculation speed, and conducts the propagation process in the spectrally modulated space to enhance the representation. We adopt the default parameter setting in (Zhang J. et al., 2019). 3) GatNE (Cen et al., 2019) is a random-walk based network embedding method, which considers the information of different attributes of nodes to enhance the graph representation. We set the number of walks as 20, walk length as 10, window size as 5, patience as 5. 4) GAE (Kipf and Welling, 2016) utilizes GCN to learn the node representations in the encode-decoder paradigm by minimizing the reconstruction loss. We set the number of GCN layers as 2 and the learning rate as 0.0001. 5) DeepWalk (Perozzi et al., 2014) is an extension of the word2vec model (Mikolov et al., 2013), which brings the idea of truncated random walks to a network embedding scenario. We set the number of walks as 80, walk length as 10, and window size as 5. 6) Node2Vec (Grover and Leskovec, 2016) is an enhanced version of DeepWalk, which considers the homogeneity and structural equivalence of networks during embedding process. We set the number of walks as 80, walk length as 10, window size as 5, return parameter p as 0.25 and in-out parameter q as 4. 7) Struc2Vec (Ribeiro et al., 2017) learns the node representation by considering the structural identity of nodes in the network. We set the number of walks as 80 and walk length as 10.

Besides, there are four losses in AutoFTP: reconstruction loss $\mathcal{L}_R$, point-wise alignment loss $\mathcal{L}_P$, pair-wise alignment loss $\mathcal{L}_C$, and regression loss $\mathcal{L}_{Reg}$. The four losses provide the optimization direction of AutoFTP. To study the benefits of each part, we

TABLE 2 | Overall Performance with respect to RMSE, MAE, MAPE and MSLE. (The smaller value is, the better performance is).

	RMSE	Outperform	MAE	Outperform	MAPE	Outperform	MSLE	Outperform
AutoFTP	18.646	—	16.192	—	58.851	—	0.2267	—
AttentionWalk	21.418	+14.9%	19.712	+21.7%	68.590	+16.6%	0.2907	+28.2%
ProNE	21.830	+17.1%	19.929	+23.1%	69.188	+17.6%	0.2949	+30.1%
GatNE	21.229	+13.9%	19.288	+19.1%	67.043	+13.9%	0.2854	+25.9%
GAE	21.338	+14.4%	19.676	+21.5%	68.579	+16.5%	0.2894	+27.6%
DeepWalk	23.561	+26.4%	21.987	+35.8%	76.038	+29.2%	0.3321	+46.5%
Node2Vec	22.688	+21.7%	21.084	+30.2%	73.135	+24.3%	0.3152	+39.0%
Struc2Vec	21.589	+15.8%	19.937	+23.1%	69.423	+17.9%	0.2942	+29.7%
AutoFTP^{R}	21.965	+17.8%	20.283	+25.3%	70.991	+20.6%	0.2928	+29.1%
AutoFTP$^{(R+P)}$	20.509	+9.99%	18.921	+16.8%	66.477	+12.9%	0.2681	+18.3%
AutoFTP$^{(R+C)}$	21.014	+12.7%	19.413	+19.8%	67.920	+15.4%	0.2773	+22.3%
AutoFTP$^{(R+P+C)}$	20.211	+8.39%	18.676	+15.3%	65.685	+11.6%	0.2636	+16.3%

develop four internal variants of AutoFTP: 1) **AutoFTP**^{R}, which only keeps $\mathcal{L}_R$ of AutoFTP; 2) **AutoFTP**$^{(R+P)}$, which keeps $\mathcal{L}_R$ and $\mathcal{L}_P$ of AutoFTP; 3) **AutoFTP**$^{(R+C)}$, which keeps $\mathcal{L}_R$ and $\mathcal{L}_C$ of AutoFTP; 4) **AutoFTP**$^{(R+P+C)}$, which keeps $\mathcal{L}_R$, $\mathcal{L}_P$, and $\mathcal{L}_C$ of AutoFTP. The dimension of embeddings in all models is 20.

4.1.5 Hyperparameters, Source Code, and Reproducibility

We detailed the hyperarameters and the steps of our algorithm in the **Appendix**. We released our code[4] to help to reproduce experimental results.

4.1.6 Environmental Settings

The experimental studies were conducted in the Ubuntu 18.04.3 LTS operating system, plus Intel(R) Core(TM) i9-9920X CPU@ 3.50GHz, 1 way SLI Titan RTX and 128GB of RAM, with the framework of Python 3.7.4, Tensorflow 2.0.0, and Pyswarm 1.3.0.

4.2 Overall Performance (Q1)

Table 2 shows the comparison of all the 11 models. As can be seen, AutoFTP, in overall, outperforms the baseline algorithms in terms of RMSE, MAE, MAPE and MSLE. A possible reason for this observation is that compared with other baseline algorithms, AutoFTP not just captures geographical structural information but also preserves rich semantics of spatial entity. Besides, the regression estimator (the downstream task) of AutoFTP provides a clear learning direction (accuracy) for spatial representation learning. Thus, in the downstream predictive task, the spatial embedding features learned by AutoFTP beats all baselines.

In addition, another interesting observation is that among all baseline models, GatNE outperforms others in terms of all evaluation metrics. Such observation shows that GatNE considers different attributed information of nodes in spatial graphs of spatial entities. Thus, the spatial embedding features learned by GatNE are more effective compared with other baseline models. Moreover, after further observing **Table 2**, we can find that the predictive performances of GAE are better than most random-walk based approaches, except GatNE. Such observation indicates that the graph convolution-based methods (GAE, AutoFTP) are more suitable than the random-walk based approaches (other baselines) in modeling geographical structure information. In summary, the overall performance experiment shows the superiority and effectiveness of AutoFTP compared with other baseline models.

4.3 Study of AutoFTP Variants (Q2)

To validate the necessity of each loss of AutoFTP, we internally compared the performances of AutoFTP with the performances of the variants of AutoFTP. **Table 2** shows the ranking orders of the predictive accuracies of the compared methods are: AutoFTP > AutoFTP$^{(R+P+C)}$ > AutoFTP$^{(R+P)}$ > AutoFTP$^{(R+C)}$ > AutoFTP^{R}. A potential interpretation for the observation is that with the increase of optimization objective (loss), AutoFTP captures more characteristics of spatial entities from representation learning, point-wise alignment, pair-wise alignment, and regression task. In addition, compared with AutoFTP$^{(R+P)}$ and AutoFTP$^{(R+C)}$, we find that the predictive performance of AutoFTP$^{(R+P)}$ is better than AutoFTP$^{(R+C)}$. A plausible reason for the observation is that the features of spatial entities captured by point-wise alignment are more indicative for spatial entities compared with them learned by pair-wise alignment. Moreover, another interesting observation is that AutoFTP outperforms other variants by a large margin. Such observation indicates that the regression loss $\mathcal{L}_{Reg}$ provides a clear optimization direction for AutoFTP, which preserves the features related to the downstream task into spatial embeddings. To sum up, the ablation study experiment demonstrates the four loss functions of AutoFTP are necessary for capturing the representative features in spatial entities during spatial representation learning process.

4.4 Study of the Interpretability of Spatial Embeddings (Q3)

The space alignment in AutoFTP is implemented from two perspectives: point-wise alignment and pair-wise alignment. The two kinds of alignment make the learned spatial embeddings contain more semantic meaning and interpretability.

[4] https://www.dropbox.com/sh/cst7yegcp1yvwax/AAB_3xBtgGDod10ULgncI-4La?dl=0

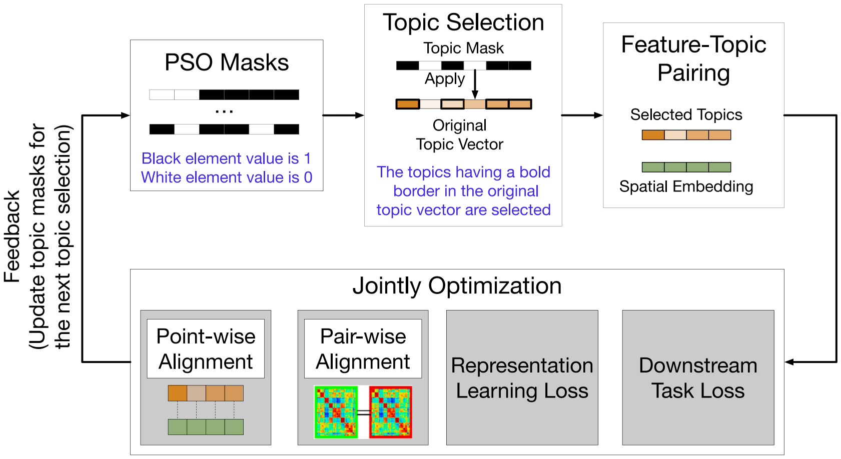


FIGURE 7 | The optimization process of the PSO topic selector.

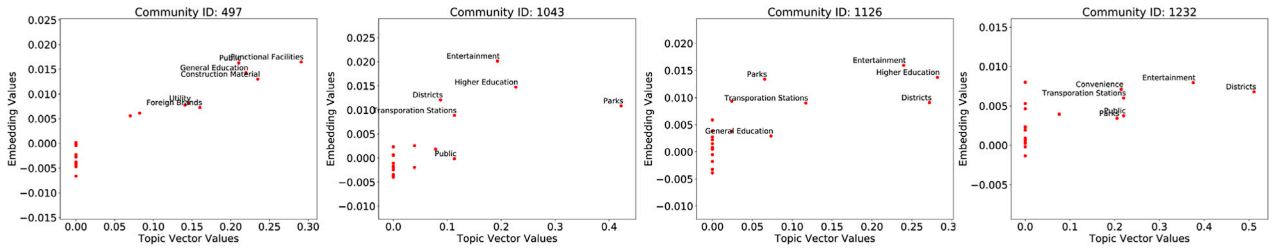


FIGURE 8 | Illustration of Point-wise Alignment with sample communities.

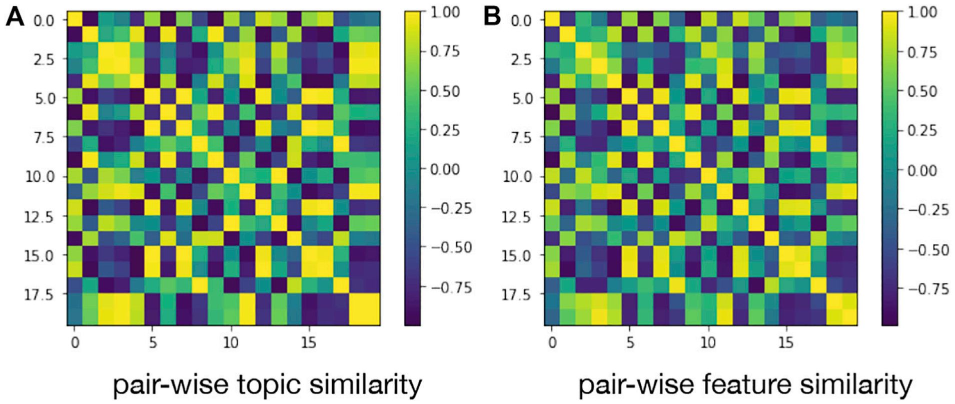


FIGURE 9 | Illustration of pair-wise alignment.

4.4.1 Study of the Point-wise Alignment

To analyze the point-wise alignment, we picked communities (spatial entities) 497, 1,043, 1,126, and 1,232 as examples to plot their corresponding embedding vectors against their corresponding topic vectors. Meanwhile, we extracted the topic names of the most significant 6 topics. **Figure 8** shows AutoFTP keeps the point-wise consistency between the semantic feature space and the embedding space. Moreover, the learned spatial embeddings contain abundant semantic meanings. We can infer the urban functions for each community based on

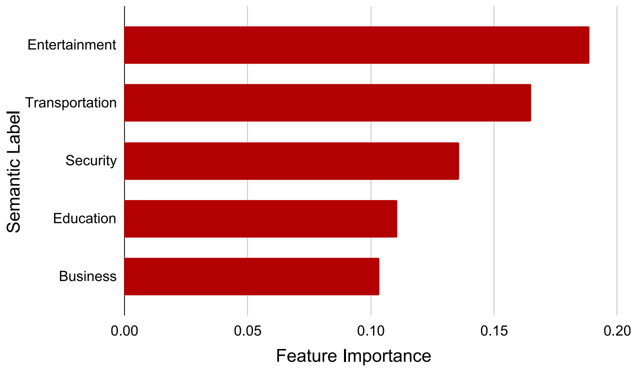


FIGURE 10 | Feature importance of spatial embeddings.

TABLE 3 | Semantic labels and top 3 keywords.

Semantic label	Keywords
Entertainment	Work Out, Tennis Court, Golf Court
Transportation Facilities	Highways, High Speed Rail, Bus Stations
Security	Surveillance, Firefighting, Emergency
Education	Primary School, High School, University
Business	Commercial Street, Canal, Satellite City

Figure 8. For instance, the community #497 exhibits high weights on some specific topics, such as, functional facilities, general education, and construction materials. Such observation indicates that this community is probably a large residential area with well-decorated apartments and general education institutions. The community #1043 and #1126 all have high weights in entertainment, higher education, parks, etc. We can speculate that they are both residential regions nearby universities. This is because the facilities belonging to these topics indicates the two communities are very likely to be in a college town. For the community #1232, it exhibits high weights in district, entertainment and convenience related categories. We can infer that the community is a commercial district with many transportation facilities.

4.4.2 Study of the Pair-wise Alignment

To observe the pair-wise alignment, we visualized the pair-wise topic similarity matrix and pair-wise feature matrix by heat map respectively. As illustrated in **Figure 9**, we can find that the two matrices are similar with only minor differences. The observation indicates that the embedding feature space is well-matched with the semantic feature space.

4.4.3 Study of the Interpretability

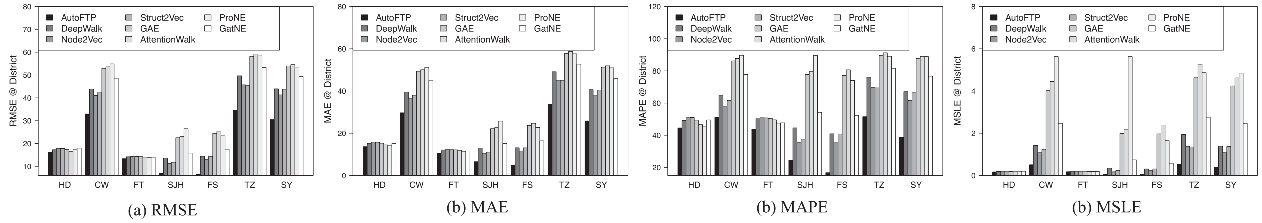
The results of **section 4.4.1** and **section 4.4.2** shows that the feature embedding space and the topic semantic embedding space are aligned well. To study the interpretability of spatial embeddings further, we built up a tree model for real estate price prediction and then analyze the feature importance based on the semantic labels of the spatial embeddings. Specially, we exploited a random forest model to predict the real estate price of spatial entities based on the corresponding embeddings. Then, we collected the feature importance of the model as illustrated in **Figure 10**. We can find that the semantic labels of top 5 dimensions in the embeddings that affects the real estate price prediction are "Entertainment", "Transportation", "Security", "Education", and "Business". The three most representative keywords in each semantic label, as shown in **Table 3**. In common sense, the 5 semantic labels are the most important factors that people consider for buying an estate (Boiko et al., 2020). In other words, they affect the real estate price heavily. Thus, the feature importance analysis experimental results are reasonable. In summary, this experiment validates that AutoFTP can select the most significant topic semantics for feature-topic automatically. In addition, the semantic labels of the spatial embeddings can be regarded as an auxiliary information to improve the interpretability of the embeddings.

4.5 Robustness Check (Q4)

To evaluate the robustness of AutoFTP, we divided the embeddings into 5 groups (HaiDian, ChongWen, FengTai, ShiJingShan, FangShan) according to the geographical district of spatial entities. **Figure 11** shows that AutoFTP consistently outperforms the baselines, and performs more stably than the baselines across the five districts. Such observation indicates that AutoFTP captures the unique local features of different spatial groups. There are two possible reasons for the observation: 1) the semantic alignment of AutoFTP injects the distinct semantic characteristics of spatial entities into the learned embeddings; and 2) the customized regression estimator provides a clear optimization objective for AutoFTP. Overall, the robustness check experiment demonstrates that AutoFTP outperforms other baseline models in not only the global zone but also each local spatial sub-areas.

4.6 Study of the Stability and Sensitivity (Q5)

In this section, we fully evaluated the stability and parameter sensitivity of AutoFTP. We first examined the stability of AutoFTP by analyzing the training losses of AutoFTP and convergence of PSO optimization part. To observe the changing trend of each loss objectively, we scaled the value of losses into $[0 \sim 1]$ and visualized them in **Figure 12A**. We can find that all losses (reconstruction loss $\mathcal{L}_R$, regression loss $\mathcal{L}_{Reg}$, point-wise loss $\mathcal{L}_P$, pair-wise loss $\mathcal{L}_C$) reach convergence over training iterations. Especially, $\mathcal{L}_R$ and $\mathcal{L}_{Reg}$ reach equilibrium quickly only after 10 epochs. This observation validates the training stability of AutoFTP. We also analyzed the convergence of PSO. As shown in **Figure 12B**, the PSO optimization part reaches convergence after 65 epochs, which further indicates the stable performance of AutoFTP. For the parameter sensitivity evaluation, we investigated the influence of the parameter K (the dimension of final embeddings and the number of significant topics) for the model performance and the training time. The same to **Figure 12A**, we scaled the value of all metrics into $[0 \sim 1]$ and visualized them in **Figure 12C**. We can find that the value of K affects the model performance heavily. The observation is reasonable because K determines the information content of the final learned embeddings. The plots in **Figure 12D** show that the larger K is, the longer the



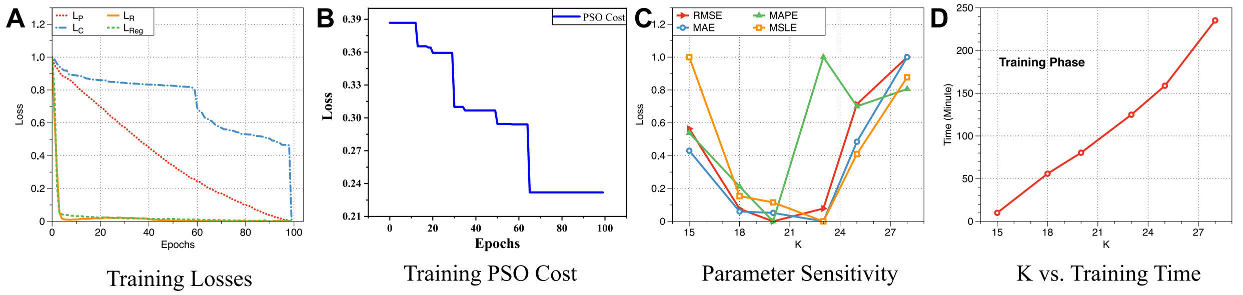
(a) RMSE

(b) MAE

(b) MAPE

(b) MSLE

FIGURE 11 | Robustness check according to geographical district.



Training Losses

Training PSO Cost

Parameter Sensitivity

K vs. Training Time

FIGURE 12 | Study the stability and sensitivity of AutoFTP. **(A)** Training Losses. **(B)** Training PSO Cost. **(C)** Parameter Sensitivity. **(D)** K vs. Training Time.

training time is. A potential reason for the observation is that the larger K means that we need to try more topic subsets for feature-topic pairing.

5 RELATED WORK

Graph Representation Learning with Latent Semantics. Graph representation learning refers to techniques that preserve the structural information of a graph into a low-dimensional vector (Wang et al., 2016; Abu-El-Haija et al., 2018; Zhang J. et al., 2019; Cen et al., 2019; Wang et al., 2020b). However, owing to traditional graph representation learning models are implemented by deep neural networks, the learned embeddings lack interpretability. Recently, to overcome this limitation, researchers leveraged the texts related to graphs to learn semantically rich representations. For instance, Mai et al. implemented an entity retrieval academic search engines that incorporate the text embedding and knowledge graph embedding for accelerating retrieving speed (Mai et al., 2018). Xiao et al. improved the semantic meaning of knowledge graph's embedding by integrating both graph triplets and textual descriptions of spatial entities (Xiao et al., 2017). Different from these studies, in this paper, based on spatial entities data composing by spatial graphs and related texts, we propose a new representation learning framework that unifies feature embedding learning and feature-topic pairing together in a closed-loop manner by a PSO based optimization method.

Topic Models in Spatio-temporal Domain. Topic models aim to automatically cluster words and expressions patterns for characterizing documents (Xun et al., 2017; Lee and Kang, 2018; Hu et al., 2019). Recently, to understand the hidden semantics of spatial entities, many researchers applied topic models in the spatio-temporal data mining domain (Zheng et al., 2017; Huang et al., 2019; Huang et al., 2020). For instance, Zhao et al. discovered representative and interpretable human activity patterns from transit data automatically by a spatio-temporal topic model (Zhao et al., 2020). Yao et al. tracked spatio-temporal and semantic dynamics of urban geo-topics based on an improved dynamic topic model that embeds spatial factors of pairwise distances between tweets (Yao and Wang, 2020). These successful applications validate the effectiveness of topic models for extracting semantics in spatio-temporal domains. However, traditional topic models only focus on word frequency in texts but neglect the semantics of words. Recently, the success of many pre-trained language models (Vaswani et al., 2017; Kenton and Toutanova, 2019; Yang et al., 2019) brings hope for producing more reasonable topic distribution. Thus, in this paper, we employ a pre-trained language model to get the embeddings of keywords and utilize Gaussian Mixture Model to extract topic distribution based on the embeddings.

Explainable Artificial Intelligence (XAI) With artificial intelligence methods are applied in multiple scenarios successfully, how to improve the model explainability becomes a big challenge. In the traditional machine learning domain, researchers employ some simple models that own the explainability naturally such as linear models, decision trees, rule-based models, and etc to explain the modeling process (Burkart and Huber, 2021; Lakkaraju et al., 2016; Lakkaraju et al., 2017). For instance (Lundberg et al., 2020), improved the global interpretability of tree models by combining many local feature explanations of each prediction and obtained good performance on three medical machine learning problems by applying these models (Wang and Rudin, 2015). provided a Bayesian framework for learning falling rule lists that do not rely on traditional greedy decision tree learning approaches to improve the explainability of classification models. Although these approaches can improve the model interpretability, the model performance often is sacrificed. Recently, the excellent predictive performance of deep learning models leads the techniques have been applied in many scenarios such as fraud detection, credit evaluation, healthcare, etc. But explainability is the key limitation of the deep learning models. To improve the model explainability, XAI on deep learning attracts much attention from researchers (Gunning, 2017; Selvaraju et al., 2017; Samek and Müller, 2019; Agarwal et al., 2020). For instance (Selvaraju et al., 2017), proposed a gradient-weighted class activation mapping method to highlight the import regions in the image for predicting the concept. (Agarwal et al., 2020). proposed neural additive models that learns a linear combination of neural networks for depicting the complex relationships between input features and the output. However, these models focus on studying the relationship between the embeddings and outputs, but cannot provide explicit semantic meanings. Different from these studies, we try to give explicit semantic labels for the learned embeddings through the alignment between the feature embedding space and topic semantic space.

Comparison with Prior Literature As an emerging feature extraction technique, deep SRL has demonstrated the power in automated geographic and spatial feature extraction. However, SRL inherits drawbacks of traditional DNNs, such as: the embedding feature space lacks semantic interpretation. Texts can provide more interpretation, but spatial text mining has developed separately. Now, there is cross and increasing interests in both fields to benefit from the advances of the other. Our study targets at an unexplored area at the intersection between representation learning in geospatial data and topic label mining in texts. We develop and formulate a new problem: feature-topic pairing, to address the alignment challenges of the feature embedding space and the semantic topic space. The self-optimizing solution unifies representation learning, topic label selection, feature-topic matching in a PSO framework. This framework can be generalized to other integrated tasks, such as, representation learning integrated with not just topic based selection, but also causal selection, or other constrained selection over features, in various application senarios. This is how this study differentiates from and advances prior literature.

6 CONCLUSION

We presented a novel spatial representation learning (SRL) framework, namely AutoFTP. The spatial embeddings produced by traditional SRL models lack semantic meaning. To overcome this limitation, we formulated the feature-topic paring problem. We proposed a novel deep learning framework to unify representation learning, topic label selection, and feature-topic pairing. Specifically, we designed a segmentation-embedding-clustering method to generate candidate feature topic labels from texts. We developed an integrated measurement to measure the pointwise and pairwise alignment between topic label and embedding feature space. We devised a PSO based optimization algorithm to effectively solve the joint task of feature learning and feature-topic pairing. Our method integrated spatial graphs and associated texts to learn effective embedding features with visible labels. Extensive experiments demonstrated the effectiveness of AutoFTP by comparing it with other baseline models. The topic labels of the learned features were shown by many case studies and the feature importance analysis of a downstream task. For future work, we plan to extend our approach from geospatial networks to other applications that consist of graphs and texts, such as social media and software code safety.

DATA AVAILABILITY STATEMENT

Publicly available datasets were analyzed in this study. This data can be found here: https://www.dropbox.com/sh/woqh4qvuzq1788r/AAB5Vz1DSeJiLKxq-POHLMAVa?dl=0.

AUTHOR CONTRIBUTIONS

DW proposes the main idea, finishes major experiments, writes the paper. KL helps accomplish partial experiments and writes some paragraphs in the paper. DM helps improve the presentation of the paper PW helps modify some typos and errors in the paper C-TL improves the presentation and language of the paper YF improves the presentation of the paper and provides the experimental data and devices.

FUNDING

This research was partially supported by the National Science Foundation (NSF) via the grant numbers: 1755946, 2040950, 2006889, 2045567, 2141095.

REFERENCES

Abu-El-Haija, S., Perozzi, B., Al-Rfou, R., and Alemi, A. A. (2018). Watch Your Step: Learning Node Embeddings via Graph Attention. *Adv. Neural Inf. Process. Syst.* 31, 9180–9190.

Agarwal, R., Frosst, N., Zhang, X., Caruana, R., and Hinton, G. E. (2020). *Neural Additive Models: Interpretable Machine Learning with Neural Nets*.

Blei, D., Ng, A., and Jordan, M. (2003). *Latent Dirichlet Allocation Journal of Machine Learning Research*.3

Boiko, D., Parygin, D., Savina, O., Golubev, A., Zelenskiy, I., and Mityagin, S. (2020). “Approaches to Analysis of Factors Affecting the Residential Real Estate Bid Prices in Case of Open Data Use,” in *Electronic Governance and Open Society: Challenges in Eurasia*. Editors A. Chugunov, I. Khodachek, Y. Misnikov, and D. Trutnev (Cham: Springer International Publishing), 360–375. doi:10.1007/978-3-030-39296-3_27

Burkart, N., and Huber, M. F. (2021). A Survey on the Explainability of Supervised Machine Learning. *jair* 70, 245–317. doi:10.1613/jair.1.12228

Cen, Y., Zou, X., Zhang, J., Yang, H., Zhou, J., and Tang, J. (2019). “Representation Learning for Attributed Multiplex Heterogeneous Network,” in Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining, 1358–1368. doi:10.1145/3292500.3330964

Chandra, D. K., Wang, P., Leopold, J., and Fu, Y. (2019). “Collective Representation Learning on Spatiotemporal Heterogeneous Information Networks,” in Proceedings of the 27th ACM SIGSPATIAL International Conference on Advances in Geographic Information Systems, 319–328. doi:10.1145/3347146.3359104

Doshi-Velez, F., and Kim, B. (2017). *Towards a Rigorous Science of Interpretable Machine Learning*. arXiv preprint arXiv:1702.08608.

Elshawi, R., Al-Mallah, M. H., and Sakr, S. (2019). On the Interpretability of Machine Learning-Based Model for Predicting Hypertension. *BMC Med. Inform. Decis. Mak.* 19, 146–232. doi:10.1186/s12911-019-0874-0

Grover, A., and Leskovec, J. (2016). “node2vec: Scalable Feature Learning for Networks,” in Proceedings of the 22nd ACM SIGKDD international conference on Knowledge discovery and data mining, New York, NY, August 2016, 855–864. doi:10.1145/2939672.2939754*KDD*2016

Gunning, D. (2017). “Explainable Artificial Intelligence (Xai),” in Defense Advanced Research Projects Agency (DARPA), Nd Web *2*.

He, H. (2014). *HanLP: Han Language Processing*. [Dataset].

Hofmann, T. (2013). *Probabilistic Latent Semantic Analysis*. arXiv preprint arXiv: 1301.6705.

Hong, S. R., Hullman, J., and Bertini, E. (2020). Human Factors in Model Interpretability: Industry Practices, Challenges, and Needs. *Proc. ACM Hum.-Comput. Interact.* 4, 1–26. doi:10.1145/3392878

Hu, N., Zhang, T., Gao, B., and Bose, I. (2019). What Do Hotel Customers Complain about? Text Analysis Using Structural Topic Model. *Tourism Manage.* 72, 417–426. doi:10.1016/j.tourman.2019.01.002

Huang, L., Wen, Y., Guo, W., Zhu, X., Zhou, C., Zhang, F., et al. (2020). Mobility Pattern Analysis of Ship Trajectories Based on Semantic Transformation and Topic Model. *Ocean Eng.* 201, 107092. doi:10.1016/j.oceaneng.2020.107092

Huang, Q., Huang, C., Huang, J., and Fujita, H. (2019). Adaptive Resource Prefetching with Spatial-Temporal and Topic Information for Educational Cloud Storage Systems. *Knowledge-Based Syst.* 181, 104791. doi:10.1016/j.knosys.2019.05.034

Jean, N., Wang, S., Samar, A., Azzari, G., Lobell, D., and Ermon, S. (2019). “Tile2vec: Unsupervised Representation Learning for Spatially Distributed Data,” in Proceedings of the AAAI Conference on Artificial Intelligence, 33. 3967–3974. doi:10.1609/aaai.v33i01.33013967

Kenton, J. D. M.-W. C., and Toutanova, L. K. (2019). “Bert: Pre-training of Deep Bidirectional Transformers for Language Understanding,” in Proceedings of NAACL-HLT, 4171–4186.

Kipf, T. N., and Welling, M. (2016). *Variational Graph Auto-Encoders*. arXiv preprint arXiv:1611.07308.

Lakkaraju, H., Bach, S. H., and Jure, L. (2016). “Interpretable Decision Sets: A Joint Framework for Description and Prediction,” in Proceedings of the 22nd ACM SIGKDD international conference on knowledge discovery and data mining, 2016. 1675–1684. doi:10.1145/2939672.2939874

Lakkaraju, H., Kamar, E., Caruana, R., and Leskovec, J. (2017). *Interpretable & Explorable Approximations of Black Box Models*. arXiv preprint arXiv: 1707.01154.

Lee, H., and Kang, P. (2018). Identifying Core Topics in Technology and Innovation Management Studies: A Topic Model Approach. *J. Technol. Transf.* 43, 1291–1317. doi:10.1007/s10961-017-9561-4

Lundberg, S. M., Erion, G., Chen, H., DeGrave, A., Prutkin, J. M., Nair, B., et al. (2020). From Local Explanations to Global Understanding with Explainable Ai for Trees. *Nat. Mach. Intell.* 2, 56–67. doi:10.1038/s42256-019-0138-9

Mai, G., Janowicz, K., and Yan, B. (2018). “Combining Text Embedding and Knowledge Graph Embedding Techniques for Academic Search Engines,” in *Semdeep/NLIWoD@ ISWC*, 77–88.

Mihalcea, R., and Tarau, P. (2004). “Textrank: Bringing Order into Text,” in Proceedings of the 2004 conference on empirical methods in natural language processing.

Mikolov, T., Sutskever, I., Chen, K., Corrado, G. S., and Dean, J. (2013). “Distributed Representations of Words and Phrases and Their Compositionality,” in *Advances in Neural Information Processing Systems*, 3111–3119.

Mori, T., and Uchihira, N. (2019). Balancing the Trade-Off between Accuracy and Interpretability in Software Defect Prediction. *Empir. Softw. Eng.* 24, 779–825. doi:10.1007/s10664-018-9638-1

Perozzi, B., Al-Rfou, R., and Skiena, S. (2014). “Deepwalk: Online Learning of Social Representations,” in Proceedings of the 20th ACM SIGKDD international conference on Knowledge discovery and data mining (New York: ACM), 701–710.

Poursabzi-Sangdeh, F., Goldstein, D. G., Hofman, J. M., Wortman Vaughan, J. W., and Wallach, H. (2021). “Manipulating and Measuring Model Interpretability,” in Proceedings of the 2021 CHI Conference on Human Factors in Computing Systems, 1–52. doi:10.1145/3411764.3445315

Ribeiro, L. F., Saverese, P. H., and Figueiredo, D. R. (2017). “struc2vec: Learning Node Representations from Structural Identity,” in Proceedings of the 23rd ACM SIGKDD international conference on knowledge discovery and data mining, 385–394.

Saisubramanian, S., Galhotra, S., and Zilberstein, S. (2020). “Balancing the Tradeoff between Clustering Value and Interpretability,” in Proceedings of the AAAI/ACM Conference on AI, Ethics and Society (New York: ACM), 351–357. doi:10.1145/3375627.3375843

Samek, W., and Müller, K.-R. (2019). “Towards Explainable Artificial Intelligence,” in Explainable AI: interpreting, explaining and visualizing deep learning (Springer), 5–22. doi:10.1007/978-3-030-28954-6_1

Selvaraju, R. R., Cogswell, M., Das, A., Vedantam, R., Parikh, D., and Batra, D. (2017). “Grad-cam: Visual Explanations from Deep Networks via Gradient-Based Localization,” in Proceedings of the IEEE international conference on computer vision, 618–626. doi:10.1109/iccv.2017.74

Shan, S., Li, Z., Yang, Q., Liu, A., Zhao, L., Liu, G., et al. (2020). Geographical Address Representation Learning for Address Matching. *World Wide Web* 23, 2005–2022. doi:10.1007/s11280-020-00782-2

Stiglic, G., Kocbek, P., Fijacko, N., Zitnik, M., Verbert, K., and Cilar, L. (2020). Interpretability of Machine Learning-Based Prediction Models in Healthcare. *Wiley Interdiscip. Rev. Data Mining Knowledge Discov.* 10, e1379. doi:10.1002/widm.1379

Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., et al. (2017). “Attention Is All You Need,” in Proceedings of the 31st International Conference on Neural Information Processing Systems, 6000–6010.

Wang, D., Cui, P., and Zhu, W. (2016). “Structural Deep Network Embedding,” in Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (New York: ACM), 1225–1234. doi:10.1145/2939672.2939753

Wang, D., Wang, P., Liu, K., Zhou, Y., Hughes, C. E., and Fu, Y. (2021). “Reinforced Imitative Graph Representation Learning for mobile User Profiling: An Adversarial Training Perspective,” in Proceedings of the AAAI Conference on Artificial Intelligence, 35. 4410–4417.

Wang, D., Wang, P., Zhou, J., Sun, L., Du, B., and Fu, Y. (2020a). “Defending Water Treatment Networks: Exploiting Spatio-Temporal Effects for Cyber Attack Detection,” in 2020 IEEE International Conference on Data Mining (ICDM) (IEEE), 32–41. doi:10.1109/icdm50108.2020.00012

Wang, F., and Rudin, C. (2015). “Falling Rule Lists,” in Proceedings of the Eighteenth International Conference on Artificial Intelligence and Statistics of Proceedings of Machine Learning Research, San Diego, California, USA.

Editors G. Lebanon and S. V. N. Vishwanathan (San Diego, CA: PMLR), 38. 1013–1022.

Wang, H., and Li, Z. (2017). "Region Representation Learning via Mobility Flow," in Proceedings of the 2017 ACM on Conference on Information and Knowledge Management, 237–246. doi:10.1145/3132847.3133006

Wang, P., Fu, Y., Xiong, H., and Li, X. (2019a). "Adversarial Substructured Representation Learning for mobile User Profiling," in Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining, 130–138. doi:10.1145/3292500.3330869

Wang, P., Fu, Y., Zhang, J., Li, X., and Lin, D. (2018a). Learning Urban Community Structures. *ACM Trans. Intell. Syst. Technol.* 9, 1–28. doi:10.1145/3209686

Wang, P., Fu, Y., Zhang, J., Wang, P., Zheng, Y., and Aggarwal, C. (2018b). "You Are How You Drive: Peer and Temporal-Aware Representation Learning for Driving Behavior Analysis," in Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining, 2457–2466.

Wang, P., Fu, Y., Zhou, Y., Liu, K., Li, X., and Hua, K. A. (2020b). "Exploiting Mutual Information for Substructure-Aware Graph Representation Learning," in *IJCAI*, 3415–3421. doi:10.24963/ijcai.2020/472

Wang, P., Li, X., Zheng, Y., Aggarwal, C., and Fu, Y. (2019b). "Spatiotemporal Representation Learning for Driving Behavior Analysis: A Joint Perspective of Peer and Temporal Dependencies," in *IEEE Transactions on Knowledge and Data Engineering*. doi:10.1109/tkde.2019.2935203

Wang, P., Liu, K., Jiang, L., Li, X., and Fu, Y. (2020c). "Incremental mobile User Profiling: Reinforcement Learning with Spatial Knowledge Graph for Modeling Event Streams," in Proceedings of the 26th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining, 853–861. doi:10.1145/3394486.3403128

Wang, S., Cao, J., Chen, H., Peng, H., and Huang, Z. (2020d). SeqST-GAN. *ACM Trans. Spat. Algorithms Syst.* 6, 1–24. doi:10.1145/3378889

Xiao, H., Huang, M., Meng, L., and Zhu, X. (2017). "Ssp: Semantic Space Projection for Knowledge Graph Embedding with Text Descriptions," in Thirty-First AAAI Conference on Artificial Intelligence.

Xun, G., Li, Y., Zhao, W. X., Gao, J., and Zhang, A. (2017). "A Correlated Topic Model Using Word Embeddings," in Proceedings of the Twenty-Sixth International Joint Conference on Artificial Intelligence (Melbourne: IJCAI), 4207–4213. doi:10.24963/ijcai.2017/588

Yang, Z., Dai, Z., Yang, Y., Carbonell, J., Salakhutdinov, R. R., and Le, Q. V. (2019). Xlnet: Generalized Autoregressive Pretraining for Language Understanding. *Adv. Neural Inf. Process. Syst.* 32, 5753–5763.

Yao, F., and Wang, Y. (2020). Tracking Urban Geo-Topics Based on Dynamic Topic Model. *Comput. Environ. Urban Syst.* 79, 101419. doi:10.1016/j.compenvurbsys.2019.101419

Zhang, J., Dong, Y., Wang, Y., Tang, J., and Ding, M. (2019a). Prone: Fast and Scalable Network Representation Learning. *IJCAI* 19, 4278–4284. doi:10.24963/ijcai.2019/594

Zhang, Y., Fu, Y., Wang, P., Li, X., and Zheng, Y. (2019b). "Unifying Inter-region Autocorrelation and Intra-region Structures for Spatial Embedding via Collective Adversarial Learning," in Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining, 1700–1708. doi:10.1145/3292500.3330972

Zhao, Z., Koutsopoulos, H. N., and Zhao, J. (2020). Discovering Latent Activity Patterns from Transit Smart Card Data: A Spatiotemporal Topic Model. *Transportation Res. C: Emerging Tech.* 116, 102627. doi:10.1016/j.trc.2020.102627

Zheng, R., Liu, Q., Rao, W., Yuan, M., Zeng, J., and Jin, Z. (2017). "Topic Model-Based Road Network Inference from Massive Trajectories," in 2017 18th IEEE International Conference on Mobile Data Management (MDM) (IEEE), 246–255. doi:10.1109/mdm.2017.41

Conflict of Interest: The authors declare that the research was conducted in the absence of any commercial or financial relationships that could be construed as a potential conflict of interest.

Publisher's Note: All claims expressed in this article are solely those of the authors and do not necessarily represent those of their affiliated organizations, or those of the publisher, the editors and the reviewers. Any product that may be evaluated in this article, or claim that may be made by its manufacturer, is not guaranteed or endorsed by the publisher.

APPENDIX

Reproducing the Algorithm

To claim AutoFTP clearly, we provide the pseudo-code of the learning process of AutoFTP. As illustrated in **Algorithm 1**, the framework includes three steps: (i) initializing the parameters of PSO, (ii) optimizing multiple objectives of AutoFTP, (iii) and outputting the final spatial embeddings. The framework takes topic vectors, POI-POI distance graphs, and POI-POI mobility graphs of spatial entities as input, and final semantically-rich embeddings as output.

For initializing the parameters of PSO (Line 1–5 in **Algorithm 1**), we first generate *M* particles as a particle swarm. Then, we initialize the position (topic mask) and velocity of each particle. Specifically, we sample *K* values from the uniform distribution *U*(0, 1) as the position vector, and sample *K* values from the uniform distribution *U*(−1, 1) as the velocity vector. Next, we update each particle's best known position (pBest) and the swarm's best known position (gBest) based on each particle's position.

For optimizing multiple objectives of AutoFTP (Line 6–20 in **Algorithm 1**), we first check that if the optimization process achieves the termination conditions. If best topic mask is not found or the training iteration does not surpass the max iteration limitation, we optimize the objectives continually. Otherwise, we output the final spatial representations. During the optimization process, for one iteration, we utilize one particle to do feature-topic pairing. Specifically, we first update the velocity of the particle based on the old velocity, the gap between the current position and pBest, and the gap between the current position and gBest. Then, we generate a new position vector (topic mask) based on the velocity vector (Line 10–11 in **Algorithm 1**). In the two lines, ω, ϕ_p, ϕ_g are weights, and γ is the learning rate of the corresponding items. Then, we filter *K* topics by the topic mask and generate the basic embedding of a spatial entity. In addition, we align the semantics of the *K* topics and the features of the basic embedding, and accomplish a downstream task simultaneously (Line 12–15 in **Algorithm 1**). Moreover, we evaluate the performance of the particle, and update the value of pBest and gBest for next optimization iteration (Line 16–20, in **Algorithm 1**).

For outputting final spatial embeddings (Line 21–23 in **Algorithm 1**), we copy the learned spatial representation as the final semantically-rich representations of spatial entities.

Algorithm 1 | Automatic Feature-Topic Pairing (AutoFTP).

```
   Input  : Topic vectors $[\mathbf{t}_1, \mathbf{t}_2, \cdots, \mathbf{t}_N]$;
            POI-POI distance graphs $[G_1^d, G_2^d, \cdots, G_N^d]$;
            POI-POI mobility graphs $[G_1^m, G_2^m, \cdots, G_N^m]$;
   Output : Semantically-rich embeddings $[\tilde{\mathbf{r}}_1, \tilde{\mathbf{r}}_2, \cdots, \tilde{\mathbf{r}}_n]$.
 1 // Initializing the parameters of PSO
 2 for each particle $i = 1, \cdots, M$ do
 3     $x_i \sim U(0,1)$; $v_i \sim U(-1,1)$; $pBest_i \leftarrow x_i$;
 4     if Fitness($gBest$) > Fitness($pBest_i$) then
 5         $gBest \leftarrow pBest_i$;
 6 // Optimizing multiple objectives of AutoFTP
 7 while (Fitness($gBest$) > $\epsilon$) and (Iteration < maxIter) do
 8     for each entity $n = 1, \cdots, N$ do
 9         for each particle $i = 1, \cdots, M$ do
10             for each dimension $k = 1, \cdots, K$ do
11                 $v_{i,d} \leftarrow \omega \cdot v_{i,d} + \phi_p \cdot (pBest_{i,d} - x_{i,d}) + \phi_g \cdot (gBest_d - x_{i,d})$;
12             $x_i \leftarrow x_i + \gamma v_i$;
13             $\check{\mathbf{t}}_n \leftarrow \mathbf{t}_n \oplus x_i$;
14             $\mathbf{r}^n = GCN(G_n^d, G_n^m)$;
15             Calculate the value of $\mathcal{L}_R, \mathcal{L}_P, \mathcal{L}_C, \mathcal{L}_{Reg}$.
16             if Fitness($pBest_i$) > Fitness($x_i$) then
17                 $pBest_i \leftarrow x_i$;
18                 if Fitness($gBest$) > Fitness($pBest_i$) then
19                     $gBest \leftarrow pBest_i$;
20     Iteration++;
21 // Outputting final spatial embeddings
22 for each entity $n = 1, \cdots, N$ do
23     $\tilde{\mathbf{r}}_n \leftarrow \mathbf{r}_n$
```