Journal of Electronics and Information Science (2024)
Clausius Scientific Press, Canada

DOI: 10.23977/jeis.2024.090302
ISSN 2371-9524 Vol. 9 Num. 3

Trajectory Generation Method Based on Deep Learning

Jiaxiang Gao[a], Xiangjie He, Yiwei Liao

Institute of Computer Science and Information Engineering, Harbin, 150025, China
[a]gaojiaxiang7@163.com

Keywords: Variational Self-Encoder, Trajectory Data, Privacy Protection

Abstract: In recent years, trajectory data publishing has brought great convenience to our daily life, however, directly publishing real trajectory data can cause serious threats to users' privacy. In this paper, we focus on the trajectory generation problem, aiming at generating trajectory datasets similar to real trajectories to meet the demand for trajectory data for urban autopilot simulation and traffic analysis tasks, and at the same time, protect the privacy of users' trajectories. We propose a trajectory generation scheme incorporating a Variational Auto-Encoder (VAE), which is capable of generating trajectory data that is highly similar to the real trajectories, to replace the user's real sensitive trajectory data for the purpose of trajectory privacy protection. We test the proposed scheme in terms of trajectory similarity, and the results show that the proposed scheme can generate trajectory datasets more accurately and stably, and at the same time protect the user's trajectory privacy.

1. Introduction

With the continuous maturity of positioning technology and the popularization of Internet mobile devices, it has become easier to obtain the mobile trajectories of users in their daily activities, and more and more applications and devices provide services for users based on user location-based service (LBS) [1]. However, while providing services for users, LBS also generates a large amount of data with user privacy, which can cause the leakage of user privacy if the data is not handled properly. Therefore, how to improve the service quality of LBS mobile applications while protecting the privacy of users' trajectories has become an important issue for current researchers .

The trajectory generation problem is essentially a generative problem, and its main purpose is to model the features of the existing data-set, and then generate trajectory data similar to the real data, compared with the real trajectory data, the generated trajectory retains some of the characteristics of the trajectory at the same time can provide a certain degree of privacy protection. Up to now, researchers have studied a variety of generative models that can be used to solve the problem of trajectory privacy leakage. Auto-Encoder (AE) [2] is a traditional generative model that has been used to solve many generative problems. However, the Decoder in AE may still suffer from the disadvantage of not being able to generate high-quality trajectories, mainly due to the fact that the Decoder is not aware of the connection between the existing trajectories and the input noise. To solve this problem, in article [8], Kingma and Welling proposed an improved version of AE, called VAE, which can effectively improve the drawbacks of AE. VAE also consists of two parts, Encoder and Decoder. The Encoder part of VAE not only learns the features of the training trajectories, but also corrects the noise of the input of Decoder at the same time The Decoder. Unlike the Decoder in AE,

the input noise of VAE is sampled from a certain probability distribution of the existing trajectories, which improves the effectiveness of the Decoder. The contributions of this paper are summarized as follows: in order to solve the problems of privacy leakage of trajectory data and low quality of generated trajectory data, this paper proposes a trajectory generation model based on VAE, which is capable of extracting effective potential features from the trajectory data and obtaining the potential relationship between the user and the trajectory, and ultimately generates the quality-assured trajectory data with a certain degree of privacy protection.

Inspired by the above studies and considering the advantages of VAE in generating class problems, we propose a trajectory generation model that is stable and at the same time guarantees the availability of trajectories, which has certain advantages in generating similar trajectories and also provides a certain degree of privacy protection for sensitive trajectories. The model can effectively extract the features of existing trajectories and then generate trajectories that are similar to the existing trajectories, thus making it difficult to distinguish between the generated trajectories and the real trajectories.

2. Related work

As the application of deep learning in various aspects of the data generation field becomes more mature, some researchers have started to incorporate deep learning techniques to generate trajectory data. The generative model is mainly used to extract some important features in the existing trajectory data and then generate some trajectories similar to the original trajectory data. AutoEncoder (AE) [5] is a traditional generative model that contains an Encoder and a Decoder module. The Decoder is usually used as a generator to convert random noise into otherwise similar data. Since the AE does not learn the input noise of the decoder, the decoder may produce invalid results. Variational AutoEncoder can be seen as an improved version of AE. The input noise of VAE is sampled in the distribution of the existing data, which effectively solves this shortcoming of AE. Generative Adversarial Network (GAN) is another popular generative model, which is equivalent to a very small and very large two-player game. GAN is capable of generating high-quality data and continuously improving the realism of the generated results through adversarial learning during training. In contrast to GAN, VAE provide an explicit description of the latent space through a posteriori distributions, which helps to analyze the representational power of generative models. Thus, in our study, we adopt VAE as the basic generative model structure for trajectory data. In the literature, the authors proposed a spatio-temporal LSTM-based trajectory prediction model that embeds spatial interactions into the LSTM model. In literature [3], Zhang et al. proposed an algorithm to discover the most commonly used routes based on start-pair and end-pair through an ant colony optimization method. In literature [4] Jiang et al. introduced a novel algorithm that applies an absorbing Markov chain model to derive a transfer network to accurately find the most commonly used routes. However, a flexible model that can ensure that the generated trajectories are indistinguishable from real trajectories for privacy preservation purposes, while preserving the utility of the trajectories, is more needed in practical applications. In this paper, we design a VAE-based trajectory generation model that flexibly captures the features of existing trajectories without restricted inputs, generates synthetic trajectories that are indistinguishable from real trajectories, and ensures the utility of the trajectories while providing a certain degree of privacy protection for the users.

3. Problems in the Economic Management of Modern Enterprises

This section focuses on the system architecture, variational self-encoder, differential privacy and trajectory dataset.

Definition 1. Spatio-temporal point: A spatio-temporal point is a GPS coordinate collected by

navigation applications, shared mobility platforms, and GPS-equipped vehicles. A spatio-temporal point is denoted as, and consists of a location at a time step value of.

Definition 2. A trajectory is a sequence of space-time points, denoted, where is a space-time point 2.

Definition 3. Trajectory dataset: a trajectory dataset is a collection of trajectories, denoted as, where is the number of trajectories.

Definition 4. Variable Auto-Encoder: Variable Auto-Encoder is an improvement of the traditional self-encoder model, which is one of the important generative models in the field of deep learning, and it mainly consists of two parts: encoder and decoder. Firstly, the encoder extracts the distributional features of the training data and maps them into the low-dimensional hidden variable space, then randomly samples the hidden variables in the variable space, then inputs the obtained hidden variables into the decoder, and finally the decoder re-maps them to the original data, and the final optimization objective function of the VAE is shown in Eq. (1).

$$\mathcal{L}(\phi,\theta;x) = \arg max\{E_{z\sim q(z|x;\phi)}[\log p(x\mid z;\theta)] - D_{kl}(p(z\mid x;\phi)\parallel p(x;\theta))\} \tag{1}$$

where ϕ and θ are the parameters of the encoder and decoder, respectively, $q(z\mid x;\phi)$ denotes the posterior probability distribution of z given the input data x, $p(x\mid x;\theta)$ is the conditional distribution, $p(z;\theta)$ is the prior distribution, $E_z \sim q(z\mid x;\phi)[\log p(x\mid z;\theta)]$ is the reconstruction error between the reconstructed data and the training data, and $D_{kl}(p(z\mid x;\phi)\parallel p(x;\theta))$ is the KL scatter between the posterior probability distribution $q(z\mid x;\phi)$ and the prior distribution $p(z;\theta)$.

3.1 Trajectory of VAE

VAE consists of two main modules: an encoder and a decoder. The encoder is mainly responsible for capturing the distribution of the training data and mapping it to lower dimensional potential vectors. The decoder then takes the potential vectors as input and converts them back to data. Finally, the trained decoder is used as a generator which decodes the noise sampled by a particular distribution into samples. In our scheme, we use a variational self-encoder to help users encode their input data into synthetic fakes. Since users usually do not have enough data to train the variational autocoder, we let the machine learning model service provider train the variational autocoder, i.e., the provider trains the machine learning model and the variational autocoder to provide privacy-preserving options. The variational autocoder can be fully published to and executed by the user. To reduce the workload of the user, in our scheme, the encoder of the variational self-encoder is released to the user and the decoder is operated by the provider. The user uses the encoder to encode its input data into an encoding vector. To prevent the encoded vector from recording the user's real input data or features, the user adds Gaussian noise to the encoded vector. The user then submits the noise-processed encoded vector to the machine learning model provider. The machine learning model provider uses the decoder of the variational self-encoder to reconstruct the noisy coded vector to generate fake data that is similar to the user's input data.

3.2 VAE Model Overview

In this paper, we design a VAE-based trajectory generation model, TrajVAE, which can efficiently extract the features of existing trajectories, and then add noise to the trajectories, which are finally converted into similar trajectories. TrajVAE consists of two main parts: Encoder and Decoder. To

extract the spatial information of the trajectories, we utilize a pre-trained embedding model to embed the relationship between neighboring intersections into a continuous latent space.

3.3 Embedded Modules

In order to extract spatial information in the road network, we first pre-train an embedding model via Deep Walk, which is able to encode the relationship between neighboring spatio-temporal points into a continuous vector space. Specifically, the distance between two spatio-temporal points can be modeled by the proximity of their embedding vectors in a low-dimensional space, as described in the literature, and we first iteratively perform a random traversal in a road network, the output of which is referred to as a "corpus". We then utilize word embedding methods to learn the latent representation of each spatio-temporal point in the road network. Finally, we obtain a continuous representation of the trajectories, and these embedding vectors are subsequently used in the encoder and decoder.

3.4 Encoder

The trajectory dataset is learned and trained by the encoder so that the distribution of $\mathcal{T}$ is as close as possible to the normal distribution $N(\mu,\sigma^2)$, where μ and σ represent the mean vector and and standard deviation vectors of the training data, respectively. The Encoder consists of three modules that encode the discrete trajectories into a continuous space by pre-training an embedded model. The encoder workflow is roughly as follows, first inputting an ensemble of trajectory datasets and then passing them into the encoder to capture sequential features, while auxiliary features such as traffic patterns and associated attributes are merged into the model using a linear layer. Immediately after this the encoder generates a latent representation z, which is further transformed by a nonlinear function to produce a probability distribution on the latent space $p_\theta(z)$. Finally, the encoder samples a set of generated latent vectors from the distribution and passes them to the decoder module to synthesize the trajectory data.

3.5 Decoder

Through the process of encoding the training data, we obtain a certain data distribution $N(\mu,\sigma^2)$, and increasing the size of the value of $\mathcal{T}$ helps to improve the efficiency of the decoder. The feature extraction module in the decoder serves as the housekeeping part of the decoder, which mainly consists of two MLP that are responsible for arithmetic generation of the first hidden state c_0, h_0 and input x_0. The computational equations are shown below:

$$\begin{aligned} c_0 = h_0 &= \mathrm{Re}\,LU(z * w_z + b_z) \\ x_0 &= \mathrm{Re}\,LU(O * w_0 + b_0) \end{aligned} \tag{2}$$

where $\mathrm{Re}\,LU(*)$ is the activation function with $\mathrm{Re}\,LU$ nonlinearity w_z, b_z, w_o ,while b_o is the weights and deviations of the two MLPs mentioned above $z \sim N(\mu,\sigma^2)$.

The loss function of TrajVAE contains two main aspects, namely the reconstruction error $\mathcal{L}_r$ and the distribution error $\mathcal{L}_d$. The reconstruction error is responsible for calculating the Mean Square Error (MSE) between the training trajectory and the generated trajectory. The distribution error is measured and calculated by the KL dispersion between the spatial distribution of the latent variables

and the normal distribution[6].

3.6 Trajectory Generation

The following describes the process of generating trajectories through the TrajVAE model. First an empty list is initialized to store the generated trajectories. For each iteration, the noise needs to be sampled, and both it and the particular origin will be part of the input to the TrajVAE decoder[7]. The output of the Decoder is added to the result list T. This process will be repeated m times.

4. Conclusions

With the development of smart mobile devices, wireless communication and positioning technology, LBS provides users with services and also generates a large amount of data with user privacy, which can cause the leakage of user privacy if the data is not handled properly. Therefore, privacy protection methods based on deep learning have received more and more attention from researchers. Deep learning models can be used to generate trajectory data, using the generated trajectories to replace the user's real sensitive input data. In this paper, we focus on the trajectory generation problem, and we propose an effective trajectory generation scheme which utilizes the VAE deep learning model framework to generate trajectory data related to the user's original data. It is shown that the trajectory data generated by TrajVAE can provide a certain degree of protection for the user's sensitive trajectory data, and at the same time can retain the trajectory features in the original trajectory.

Acknowledgement

This present research work was supported by Harbin Normal University Higher Education Teaching Reform Research Project (No. XJGZ202409).

References

[1] Daraio E, Cagliero L, Chiusano S, et al. Comple- menting location-based social network data with mobility data: a pattern-based approach [J]. IEEE Transactions on Intelligent Transportation Systems, 2022, 23(11): 212-227.
[2] Srivastava R K, Koutn ́k J, et al. LSTM: A search space odyssey [J]. IEEE transactions on neural networks and learning systems, 2016, 28(10): 2222-2232.
[3] Zhang H, Huangfu W, Hu X, et al. Inferring the Most Popular Route Based on Ant Colony Optimization with Trajectory Data[C]//Wireless Sensor Networks: 11th China Wireless Sensor Network Conference, CWSN 2017,12-14.
[4] Jiang H, Li J, Zhao P, et al. Location privacy-preserving mechanisms in location-based services: A comprehensive survey[J]. ACM Computing Surveys (CSUR), 2021, 54(1): 1-36.
[5] S. Semeniuta, A. Severyn, E. Barth, et al. A hybrid convolutional variational autoencoder for text generation, EMNLP 2017 (2017) 627–637.
[6] Ivanovic B, Leung K, Schmerling E, et al. Multimodal deep generative models for trajectory prediction: A conditional variational autoencoder approach [J]. IEEE Robotics and Automation Letters, 2020, 6(2): 295-302.
[7] A. Gupta, J. Johnson, L. Fei-Fei, et al. Social GAN: socially acceptable trajectories with generative adversarial networks, CVPR (2018) 2255–2264.
[8] Li Y, Pan Q, Wang S, et al. Disentangled variational auto-encoder for semi-supervised learning[J]. Information Sciences, 2019, 482: 73-85.