Leveraging Language Foundation Models for Human Mobility Forecasting

Hao Xue*
University of New South Wales
Sydney, NSW, Australia
hao.xue1@unsw.edu.au

Bhanu Prakash Voutharoja*
University of Wollongong
University of New South Wales
NSW, Australia
bpv991@uowmail.edu.au

Flora D. Salim
University of New South Wales
Sydney, NSW, Australia

ABSTRACT

In this paper, we propose a novel pipeline that leverages language foundation models for temporal sequential pattern mining, such as for human mobility forecasting tasks. For example, in the task of predicting Place-of-Interest (POI) customer flows, typically the number of visits is extracted from historical logs, and only the numerical data are used to predict visitor flows. In this research, we perform the forecasting task directly on the natural language input that includes all kinds of information such as numerical values and contextual semantic information. Specific prompts are introduced to transform numerical temporal sequences into sentences so that existing language models can be directly applied. We design an AuxMobLCast pipeline for predicting the number of visitors in each POI, integrating an auxiliary POI category classification task with the encoder-decoder architecture. This research provides empirical evidence of the effectiveness of the proposed AuxMobLCast pipeline to discover sequential patterns in mobility forecasting tasks. The results, evaluated on three real-world datasets, demonstrate that pre-trained language foundation models also have good performance in forecasting temporal sequences. This study could provide visionary insights and lead to new research directions for predicting human mobility.

CCS CONCEPTS

• **Applied computing** → *Forecasting*; • **Computing methodologies** → *Natural language generation.*

KEYWORDS

human mobility, spatio-temporal prediction, language generation

ACM Reference Format:
Hao Xue, Bhanu Prakash Voutharoja, and Flora D. Salim. 2022. Leveraging Language Foundation Models for Human Mobility Forecasting. In *The 30th International Conference on Advances in Geographic Information Systems (SIGSPATIAL '22), November 1–4, 2022, Seattle, WA, USA.* ACM, New York, NY, USA, 9 pages. https://doi.org/10.1145/3557915.3561026

*Both authors contributed equally to this research.

SIGSPATIAL '22, November 1–4, 2022, Seattle, WA, USA

ACM ISBN 978-1-4503-9529-8/22/11…$15.00
https://doi.org/10.1145/3557915.3561026

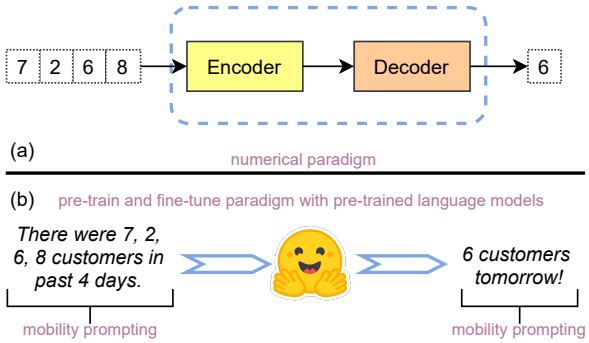

Figure 1: Conceptual comparison of: (a) the numerical paradigm for human mobility forecasting and (b) the proposed language foundation model based forecasting with mobility prompts.

1 INTRODUCTION

Nowadays, AI-powered digital assistants (*e.g.*, Alexa and Siri) have demonstrated advanced performance in answering general topics whereas they still could not well-responding human mobility forecasting problems. Mining spatio-temporal sequential patterns plays a critical component in many real-world applications including intelligent transportation and smart cities such as predicting the future customer flow of a shop to avoid the crowds during the pandemic. We observe that almost all of the existing deep learning based solutions for spatio-temporal and human mobility prediction tasks can be summarised as a *numerical paradigm* (Figure1 (a)) which takes a sequence of numerical values (history mobility observations) as input to generate a numerical value as future prediction (or a sequence of numerical values). Usually, only the numeric data can be well extracted and modelled within such a framework for future prediction. Furthermore, this *numerical paradigm* makes it difficult to seamlessly integrate the forecasting ability with the natural language processing models of the digital assistants.

More recently, the fast evolution of foundation models [4] such as BERT [6] and CLIP [13] has led to a paradigm shift in designing, training, and applying deep learning models. In the new *pre-train and fine-tune paradigm*, a foundation model is pre-trained with large-scale data and then adapted to solve various downstream tasks such as a well pre-trained BERT model for language translation, sentiment analysis, and question answering. However, in the literature, this shift only appears in Natural Language Processing (NLP) and Computer Vision (CV) fields. How to apply a foundation model for spatio-temporal forecasting and human mobility

prediction remains unexplored. In the time-series data forecasting domain especially with the human mobility data, due to the sequential numerical data format, there is no existing work on directly using pre-trained language foundation models for human mobility prediction. Although some network architecture designs such as Transformer [16] can be tweaked to the numerical paradigm for temporal sequence prediction (*e.g.*, [20]), it seems impossible to directly take advantage of existing pre-trained foundations models. The main reason is that the temporal sequence mining tasks require the prediction model to handle the numerical data format, whereas existing language foundation models are pre-trained with natural language sentences. This leads to a gap between the learned knowledge in pre-training and the downstream prediction.

In this paper, motivated by the above observations, we aim to investigate the following research question: *how can we directly leverage existing pre-trained language foundation models for sequential temporal data?* We hope that the exploration of this question in this vision paper could open new research directions and present novel insights in the spatio-temporal data forecasting field. The mining of temporal sequential patterns is exemplified by an aggregated human mobility forecasting task in this work. To address the aforementioned gap, we explore the idea of *mobility prompting*, which transfers the human mobility data into natural language sentences so that the pre-trained language foundation models can be directly applied in the fine-tuning phase for predicting human mobility. With mobility prompts that describe the historical mobility observations as input, we apply an encoder-decoder architecture to yield future mobility predictions (this forecasting paradigm is demonstrated in Figure1 (b)). Compared to the numerical paradigm, directly using natural language as input can seamlessly and naturally model the numerical sequences, the textual context, and the semantic information for generating future predictions. Furthermore, we propose a novel AuxMobLCast pipeline (**Aux**iliary **Mob**ility **L**anguage Fore**cast**ing) for forecasting human mobility, which introduces an auxiliary POI category classification task. The auxiliary task is specially designed to associate with the `[CLS]` token in the encoder part (*e.g.*, BERT's `[CLS]` token). This extra customised task does not modify the foundation model structure which makes it possible to directly utilise existing pre-trained models (*e.g.*, pre-trained weights provided by HuggingFace[1]). In particular, we conduct an empirical study to analyse the mobility forecasting performance of using different foundation models (such as BERT, RoBERTa, GPT-2, and XLNet) as the encoder/decoder of AuxMobLCast. The experimental results demonstrate that AuxMobLCast can further improve the prediction performance. In summary, our contributions are:

- To the best of our knowledge, this paper presents the first attempt to fine-tune existing pre-trained language foundation models for forecasting human mobility data. Mobility prompts are specifically used to address the gap between the human mobility data and the language formats.
- We empirically investigate and examine the performance of multiple pre-trained language foundation models for forecasting human mobility in the pre-train and fine-tune paradigm fashion. This work is the first demonstration of the power of pre-trained language models for forecasting tasks.
- We further introduce an auxiliary POI category classification task to improve the forecasting performance.

The rest of the paper is organised as follows. Section 2 introduces the related work. Section 3 presents a formal definition of the problem focused in this paper and describes the proposed AuxMobLCast. The details of datasets and our experimental settings are given in Section 4. It also analysis the performance of our AuxMobLCast in human mobility forecasting and conducts ablation studies. Section 5 concludes the paper and discusses the potential impact and future directions of this work.

2 RELATED WORK

2.1 Language Foundation Models

Recent rapid advances in self-supervised training techniques and Transformer [16] architectures facilitate the emerging foundation models and the pre-train and fine-tune paradigm in various NLP tasks. For example, BERT [6] is a Transformer-based model pre-trained in a self-supervised fashion. It utilises the masked language modelling task for pre-training on the unlabelled large-scale corpus of English text data, which makes the pre-trained model more scalable and can be adapted to different downstream tasks based on learned language understanding ability. In the masked language modelling pre-training process, the model randomly masks a certain percentage (*e.g.*, 15%) of the words in the input sentence. The model is then trained to predict the masked words. Through this technique, the model can be pre-trained on the raw texts only without any human labelling required. Since the emergence of BERT, almost all state-of-the-art models in NLP tasks are adapted and tweaked from one of a few foundation models such as BERT, GPT-2 [14] and RoBERTa. Beyond the NLP field, the foundation models, as well as the pre-train and fine-tune paradigm, also demonstrate superior performance in image classification [13, 23] and speech recognition [1]. This work aims to investigate how to apply foundation models for mining temporal sequential patterns, which further broadens the application of the pre-train and fine-tune paradigm. Although a similar numerical prediction task has been addressed with language models in [15] and [3], our work differs from this task in that we are interested in sequential numbers instead of separate numbers in the input text.

2.2 Time-series Forecasting

The human mobility forecasting task is much related to general time-series forecasting research. Deep learning based methods for time-series forecasting are mainly based on the Recurrent Neural Network (RNN) and its variants like Long Short Term Memory (LSTM) networks [8] and Gated Recurrent Units (GRU) [5]. Methods in this category often generate the future prediction in a sequence-to-sequence manner with the numerical history observations as the input sequence. Representative works under this numerical paradigm for sequential human behaviour prediction include ST-RNN [11] and DeepMove [7].

More recently, inspired by the success of Transformer [16] in language processing, various methods such as [10, 17, 19, 22] have been proposed for sequential time-series forecasting and human mobility prediction tasks. Although these methods are based on the

[1] https://huggingface.co/models

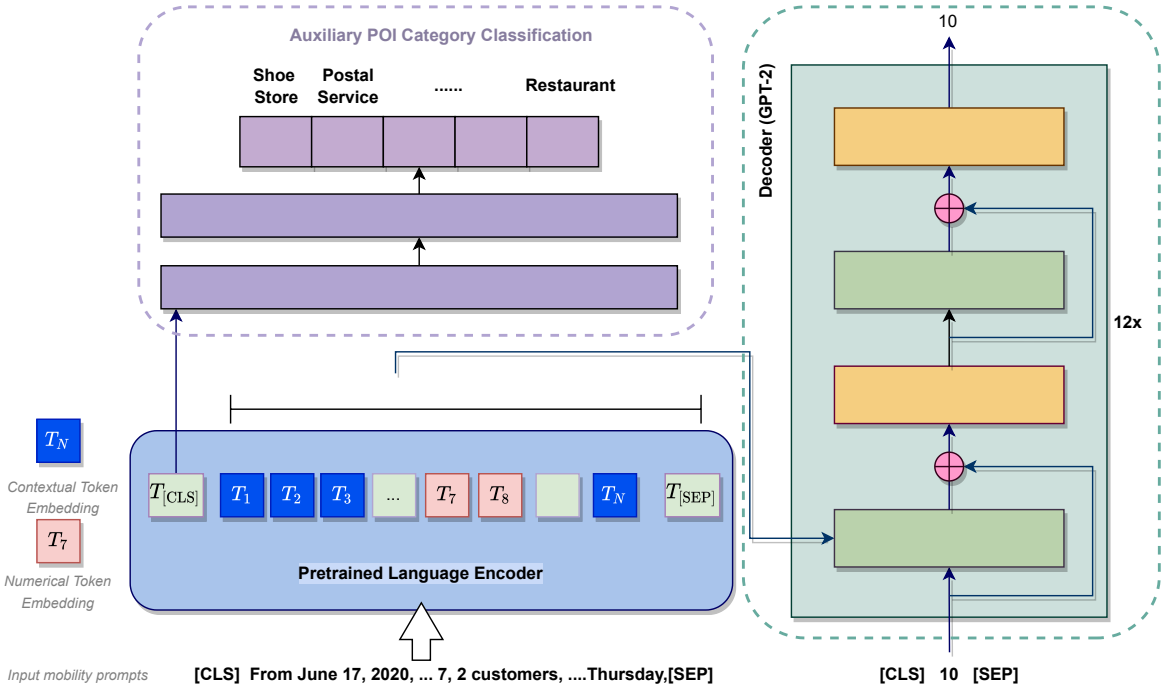

Figure 2: The illustration of our AuxMobLCast. Through the self-attention based language encoder, the interactions between the numerical tokens (red) and contextual tokens (blue, *e.g.*, the temporal information) are simultaneously extracted in prompt embeddings for the decoder. We also explicitly introduce an auxiliary POI category classification (the purple part).

effective Transformer, they cannot fully take advantage of the pre-train and fine-tune paradigm due to two main reasons. First, due to issues like privacy concerns in collecting human mobility data, there are no large scale time-series datasets that can be used for pre-training the model to learn general features of different time-series data types. Second, these methods introduce unique designs and tweaks at the network structure level for modelling the time-series characteristics, which results in that existing pre-trained language models cannot be directly applied.

In this paper, we hypothesise that language models pre-trained with very large scale text data could also be beneficial in learning sequential patterns. Instead of focusing on tweaking the model structure, we switch our focus to transforming the sequential data (*i.e.*, aggregated human mobility in this work) into language descriptions through mobility prompts. As a consequence, pre-trained language foundation models can be directly leveraged for forecasting human mobility.

3 METHODOLOGY

3.1 Problem Definition

In this paper, we focus on the aggregated human mobility forecasting task. Let $\{\text{POI}_1, \text{POI}_2, \cdots, \text{POI}_M\}$ denotes a set of M POIs in a certain area (*e.g.*, a city). For each POI_m, we can observe a history records of customer visits on n continuous days: $x^m_{t_1:t_n} = [x^m_{t_1}, x^m_{t_2}, \cdots, x^m_{t_n}]$ where x^m_t stands for the number of visits of POI m on day t. In addition, each POI corresponds to a semantic category class c^m such as a *Shoe shop* or a *Post office*. The focused human mobility forecasting problem can then be formulated as predicting the number of visits $x^m_{t_{n+1}}$ of the next day t_{n+1} given the history observation $x^m_{t_1:t_n}$.

In this paper, we are particularly interested in generating the future $x^m_{t_{n+1}}$ under the *pre-train and fine-tune paradigm* with language foundation models. For this purpose, we propose to develop mobility prompt X which translates the numerical history observations as natural language sentences instead of using the numerical sequence $x^m_{t_1:t_n}$ as the inputs of forecasting models in existing *numerical paradigm* based time-series forecasting methods. We note that the superscript m (POI index) is dropped for simplification from hereon.

3.2 Mobility Prompting

Inspired by other work that apply language models for non-NLP tasks through prompting such as CLIP [13] and CoOp [23], we introduce mobility prompting to convert the sequential observation $x_{t_1:t_n}$ into language description X to leverage pre-trained language models for forecasting human mobility. Such a transformation step is a key enabler for taking advantage of the *pre-train and fine-tune paradigm* in this work. The purpose of mobility prompting is to pre-process the numerical mobility data to generate meaningful sentences that can be fed to the pre-trained language foundation models. Specifically, we develop and investigate three different prompts as illustrated in Table1. In general, all three types of prompts contain key components related to mobility in the input text part, including the mobility data (*i.e.*, the number of visits on each day, the red

Table 1: Examples of three different mobility prompt types used in this study. Mobility data, temporal information, and the POI information are given in red, blue, and orange, respectively.

Prompt A	***Input***: "Place-of-Interest (POI) 385 is a Limited-Service Restaurant. From June 17, 2020, Wednesday to July 01, 2020, Wednesday, there were 11, 11, 10, 12, 9, 12, 6, 13, 10, 15, 16, 8, 8, 13, 19 people visiting POI 385 on each day. On July 02, 2020, Thursday," ***Target***: "there will be 11 people visiting POI 385."
Prompt B	***Input***: "From June 17, 2020, Wednesday to July 01, 2020, Wednesday, there were 11, 11, 10, 12, 9, 12, 6, 13, 10, 15, 16, 8, 8, 13, 19 people visiting POI Limited-Service Restaurant on each day. On July 02, 2020, Thursday," ***Target***: "there will be 11 people."
Prompt C	***Input***: "From June 17, 2020, Wednesday to July 01, 2020, Wednesday, there were 11, 11, 10, 12, 9, 12, 6, 13, 10, 15, 16, 8, 8, 13, 19 people visiting POI on each day. On July 02, 2020, Thursday," ***POI Category Target***: "Limited-Service Restaurant" ***Mobility Target***: "11"

parts), the temporal information (*i.e.*, the date, the blue parts), and information about the POI itself (the orange parts). The temporal information includes not only the history observations timestamps (*e.g.*, `From June 17, 2020, Wednesday to July 01, 2020, Wednesday`) but also the time of the prediction target (*e.g.*, `July 02, 2020, Thursday`) which provides an important cue for prediction. For the output target text (used as the ground truth for fine-tuning and evaluation), all prompts include the future prediction target $x_{t_{n+1}}$ (*e.g.*, 11 in the table).

Table1 lists an example input/target text for each prompt. Prompt A is the basic format. Compared to Prompt A, the POI index id is removed from the prompting sentences in Prompt B whereas the category information is kept. There are two rationales for this alteration: (1) As demonstrated in [19], POIs from different categories often have different visiting patterns. As a consequence, it is important to keep the category information. (2) Excluding a specific POI id could further alleviate privacy concerns in real-world applications. Based on Prompt B, we further design Prompt C for our AuxMobLCast pipeline. For this prompt type, the output target is explicitly divided into two parts: the POI category target for the auxiliary POI category classification in AuxMobLCast (details given in the next section) and the direct mobility prediction target.

3.3 Proposed AuxMobLCast

In this study, in addition to investigating existing pre-trained language models' ability on human mobility forecasting, we also propose a novel pipeline AuxMobLCast based on the general encoder-decoder framework. As demonstrated in Figure2, in AuxMobLCast, we explicitly introduce an auxiliary classification task (the purple part) to classify the POI category. This design is motivated by the observation that the POI category is correlated to its visiting pattern. For example, Figure3 shows the averaged weekly visiting pattern of

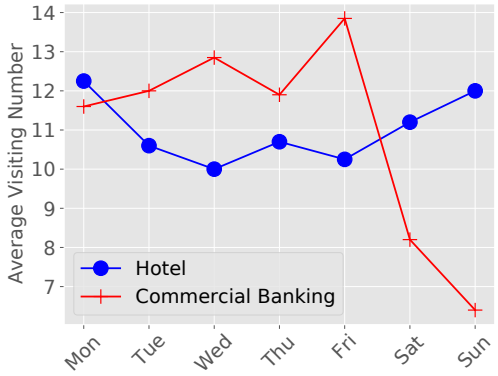

Figure 3: The average weekly visiting patterns of two different categories POIs.

two POIs located in Miami during the entire data collection period (details given in Section 4.1). One POI belongs to the *Hotel* category and the other POI is the *Commercial Banking* category. From this plot, we can clearly see that these two POIs have different visiting patterns. The Hotel POI would expect higher customer volumes during weekends than weekdays, whereas the visiting count of the Commercial Banking POI has a significant drop during weekends. Learning how to distinguish different categories from input prompts could be beneficial in forecasting future visiting volumes. In addition, another important characteristic of this introduced auxiliary operation is that it does not affect the encoder structure. Thus, available pre-trained weights of encoders/decoders can be directly used as initialisation, which is a key enabler for forecasting under the *pre-train and fine-tune paradigm*.

The overall pipeline works in a similar fashion as the typical sequence-to-sequence task. The encoder encodes the input mobility prompt sentences which describe the history mobility observation. The decoder then takes the encoded features as input to generate the predicted sentence tokens. This process can be described as:

$$Token_{\text{pred}} = \text{DECODER}(\text{ENCODER}(X)), \tag{1}$$

where ENCODER$(\cdot)$ and DECODER$(\cdot)$ represent the encoder and the decoder, respectively. In the proposed AuxMobLCast, we vary and investigate multiple Transformer based encoder structures (details given in Section 4.5). With the help of the self-attention mechanism inside the Transformer, not only the intra-relation of numerical tokens at different time steps (*i.e.*, the mobility data $x^m_{t_1:t_n}$) in the input prompts but also the inter-relation between numerical and contextual tokens (*e.g.*, tokens for the date information) could be learned simultaneously to generate future predictions. GPT-2 [14] is also a Transformers-based model architecture designed to yield the next word in sentences pre-trained with very large corpus of English data (general English text data without any specific mobility data). It stacks multiple decoder layers (normally 12 - 48 layers), the GPT-2 with 12 decoder layers is selected as the decoder in our AuxMobLCast. The auto-regressive nature of the GPT-2 decoder makes it a preferred choice for text generation tasks from a given input prompt.. The numerical prediction $\hat{x}_{t_{n+1}}$ can then be detokenized from the generated token $Token_{\text{pred}}$.

Table 2: Pre-trained language models explored in the proposed AuxMobLCast.

	Model	HuggingFace Configuration
Encoder	BERT	https://huggingface.co/bert-base-uncased
	RoBERTa	https://huggingface.co/roberta-base
	XLNet	https://huggingface.co/xlnet-base-cased
Decoder	GPT-2	https://huggingface.co/gpt2

Specifically, for the auxiliary classification, after encoding through an encoder (*e.g.*, BERT), the feature embedding of the `[CLS]` token is passed through a fully connected layer followed by a softmax layer for the POI category classification. This token is selected for classification as it is a global token and attends to all the other input tokens that cover the mobility information.

3.4 Training Objective

Due to the introduced auxiliary POI category classification, there are two types of losses in AuxMobLCast. The first loss is a cross-entropy loss denoted by $\mathcal{L}_{\text{CE}}$ for guiding the sequence generation for our main purpose and the second loss $\mathcal{L}_{\text{POI}}$ is the auxiliary category classification loss. During the training, the proposed AuxMobLCast is fine-tuned end-to-end with the loss $\mathcal{L}$ given by:

$$\mathcal{L} = \lambda_{\text{CE}}\mathcal{L}_{\text{CE}} + \lambda_{\text{POI}}\mathcal{L}_{\text{POI}}, \tag{2}$$

where λ_{CE}, λ_{POI} are two factors used to combine two losses and the sum of these factors equals 1.

4 EXPERIMENTS

In our experiments, we aim to investigate and answer the following research questions:

- **RQ1**: Compared to the conventional *numerical paradigm* forecasting methods, what is the performance of language based encoder-decoder methods with mobility prompts as input?
- **RQ2**: Under the AuxMobLCast pipeline, can we apply different pre-trained language foundation models as encoders, and what is the forecasting performance for each of them?
- **RQ3**: Could we achieve a further mobility forecasting performance gain through the introduced auxiliary POI category classification?

4.1 Datasets

In this study, the datasets used for evaluation are from real-world human mobility data provided by SafeGraph.[2] It contains daily visit counting records and the category information of each POI. To enhance privacy, the provided data is aggregated and anonymised. We access the raw SafeGraph Weekly Patterns data through SafeGraph Data for Academics[3]. The data from three representative cities (New York City (NYC), Dallas, and Miami) was collected from 2020-06-15 to 2020-11-08 to form three datasets. These datasets include 479 POIs from 39 categories, 1374 POIs from 65 categories, and 1007 POIs from 51 categories, respectively. Based on these statistics, we can see that Dallas has the most POIs and categories, while NYC has the least number of POIs/categories. We split each dataset into training set (70%), validation set (10%), and testing set (20%). For the input data, the history observation length is defined as 15 days (*i.e.*, $n = 15$). Mobility prompts given in the sentence format are generated based on the templates illustrated in Table1. For the comparison purpose, the train/val/test sets in the numerical format are also maintained for each city to evaluate methods in the *numerical paradigm*.

[2]https://docs.safegraph.com/docs/weekly-patterns
[3]https://www.safegraph.com/academics

4.2 Implementations

The proposed AuxMobLCast follows the *pre-train and fine-tune paradigm* and we leverage the existing pre-trained language models. The encoders and decoders are configured and initialised with pre-trained weights provided by HuggingFace Models. We then carry out fine-tuning using the generated mobility. The pre-trained language models are fine-tuned using mobility prompts with the following implementation settings. The total epoch number is selected as 50 with early-stopping. For factors in loss function (Equation (2)), $\lambda_{\text{CE}} = 0.8, \lambda_{\text{POI}} = 0.2$ is also applied on NYC and Miami datasets, whereas the setting of $\lambda_{\text{CE}} = 0.9, \lambda_{\text{POI}} = 0.1$ is used for Dallas dataset. The models are optimised with Adam optimiser with a 5×10^{-5} initial learning rate (weight decay is set to 5×10^{-4}). These parameters are selected based on the performance on the validation set. The *ReduceLROnPlateau* decay is adopted with patience 6 and cooldown 2 during fine-tuning. The experiments are performed with PyTorch on a Linux server equipped with Nvidia V100 GPUs. For the configurations and pre-trained weights of the pre-trained language models used in our experiments, Table 2 lists the corresponding HuggingFace links. In the experiments, we used the standard pre-trained weights of these decoder/encoders which can be accessed through the listed links. Our codes are available at https://github.com/cruiseresearchgroup/AuxMobLCast.

4.3 Evaluation

For evaluating the performance of each method, considering we focus on the numerical prediction task, the Root Mean Square Error (RMSE) and the Mean Absolute Error (MAE) are selected as evaluation metrics. These errors are calculated based on the predicted $\hat{x}_{t_{n+1}}$ and the ground truth $x_{t_{n+1}}$ to measure the closeness of the predicted values. In our experiments, the average performance and the standard deviation of multiple runs (5 runnings with different random seeds) of each method are reported excluding the basic linear regression (LR) method.

4.4 Results and Analysis

In this part of experiments, we compare the performance of using mobility prompts against the following *numerical paradigm* based time-series forecasting methods:

- Basic Linear Regression (LR);
- GRU [5]: one of the most popular variant of Recurrent Neural Networks;
- GRUAtt [2]: introducing the attention mechanism into the GRU architecture;

Table 3: Prediction results of the numerical paradigm based methods and methods using mobility prompts. Both Prompt A and Prompt B are compared.

Prompt	Model	NYC		Dallas		Miami		Average	
		RMSE	MAE	RMSE	MAE	RMSE	MAE	RMSE	MAE
N/A Numerical based	LR	9.131	5.639	24.544	6.601	13.081	6.082	15.585	6.107
	GRU	7.547±0.098	4.550±0.038	23.987±0.262	5.400±0.016	12.125±0.160	5.413±0.026	14.553	5.121
	GRUAtt	7.704±0.107	4.464±0.037	22.562±0.433	5.276±0.048	11.465±0.417	5.045±0.107	13.910	4.928
	Transformer	6.714±0.072	4.279±0.058	18.820±0.278	5.166±0.125	10.995±0.181	5.130±0.117	12.176	4.858
N/A Numerical based With Temporal Embedding	Transformer	6.452±0.055	4.250±0.057	18.796±0.338	5.337±0.183	10.004±0.022	5.053±0.066	11.751	4.880
	Reformer	6.645±0.040	4.377±0.018	17.423±0.200	5.518±0.066	10.411±0.151	5.116±0.046	11.493	5.004
	Informer	6.279±0.140	4.134±0.074	18.061±0.205	5.441±0.052	9.526±0.098	4.823±0.043	11.289	4.799
	Autoformer	6.433±0.103	4.323±0.108	18.033±0.896	7.021± 0.977	9.852± 0.731	6.321±0.701	11.439	5.888
A	GRUAtt-A	6.901±0.212	4.290±0.042	19.914±1.259	5.165±0.067	9.964±0.632	5.009±0.055	12.260	4.821
	Transformer-A	6.657±0.070	4.286±0.075	18.212±1.422	5.036±0.096	9.672±0.605	5.034±0.105	11.514	4.785
B	GRUAtt-B	6.887±0.105	4.355±0.059	19.743±0.884	5.212± 0.227	10.066±0.520	5.124±0.036	12.232	4.897
	Transformer-B	6.648±0.190	4.273±0.054	18.189±1.382	5.087±0.023	9.563±0.406	4.991±0.164	11.467	4.784

Table 4: Prediction results of different configurations of our AuxMobLCast on three datasets. Prompt C is applied for this part of experiments.

Encoder	Aux	NYC		Dallas		Miami		Average	
		RMSE	MAE	RMSE	MAE	RMSE	MAE	RMSE	MAE
BERT	✓	6.312±0.253	4.114±0.038	15.304±0.835	5.168±0.210	10.307±1.698	4.804±0.084	10.641	4.695
	×	6.291±0.010	4.144±0.024	18.125±1.509	5.111±0.096	12.197±1.057	4.871±0.060	12.204	4.708
RoBERTa	✓	6.277±0.218	4.106±0.048	16.902±1.621	4.964±0.062	10.744±0.793	4.926±0.127	11.307	4.665
	×	6.336±0.259	4.117±0.049	15.821±1.114	5.294±0.193	11.804±0.652	5.228±0.172	11.320	4.879
XLNet	✓	6.586±0.177	4.289±0.085	16.566±0.998	5.305±0.094	12.683±1.127	5.075±0.161	11.945	4.889
	×	6.605±0.253	4.223±0.033	15.602±0.285	5.202±0.123	13.071±2.561	5.254±0.059	11.759	4.893

- Transformer [16]: the vanilla Transformer network with the multi-head self-attention mechanism;
- Reformer [9]: a variant of Transformer focused on the efficiency. It has been used as a baseline for forecasting in [22] and [18].
- Informer [22]: a specific variant of Transformer that is proposed for long sequence forecasting task.
- Autoformer [18]: a recent variant of Transformer for predicting temporal sequences.

For Transformer based methods (including Transformer, Reformer, Informer, and Autoformer), we consider incorporating temporal information. Specifically, the temporal embeddings (e.g., time-of-day, day-of-week, month-of-year) are injected as the Transformer position embeddings so that the date information is used as input features for prediction. All the above methods take the numerical sequences as input and generate a number as the prediction. For methods using natural language based mobility prompts as input/output, we start with two widely used architectures for language sequence-to-sequence tasks, namely, GRUAtt and Transformer. Both methods are in the encoder-decoder structure. Prompt A and Prompt B defined in Table1 are adopted respectively. To make it clear, we use "-A" and "-B" to differentiate methods trained with two prompts (*e.g.*, GRUAtt-A, Transformer-B).

The results of the above methods on the testing set are reported in Table3. The last two columns list the average RMSE and MAE across three cities for each method. In general, for each method, it can be noticed that NYC has the lowest prediction error whereas Dallas has the worst performance. This observation can be explained by the fact that Dallas is relatively harder to predict due to the largest number of POIs and categories in all three datasets. For methods under the numerical paradigm, Transformer outperforms other baselines while the linear regression has the worst performance, which is as expected. When the temporal information is considered, Transformer with temporal embedding yields a better RMSE performance compared to Transformer without using temporal embedding. Autoformer and Reformer also demonstrate good prediction performance. Among these numerical methods with temporal embedding, Informer achieves the best performance on both RMSE and MAE against all the other numerical based methods. This shows that the temporal information can be used as a strong cue for forecasting and it should be included the mobility prompts.

Comparing the methods with the same backbone model (GRUAtt vs. GRUAtt-A/B and Transformer vs. Transformer-A/B), we observe that using language based mobility prompts could lead to better prediction performance. Furthermore, with the help of mobility prompts, the methods even using the vanilla Transformer as

Table 5: Prediction results of different configurations (with or without using temporal date information in the input prompts).

Encoder	Date Info	NYC		Dallas		Miami		Average	
		RMSE	MAE	RMSE	MAE	RMSE	MAE	RMSE	MAE
BERT	✓	6.312±0.253	4.114±0.038	15.304±0.835	5.168±0.210	10.307±1.698	4.804±0.084	10.641	4.695
	×	6.764±0.092	4.461±0.040	16.633±0.552	5.550±0.230	11.017±1.348	5.331±0.159	11.471	5.114
RoBERTa	✓	6.277±0.218	4.106±0.048	16.902±1.621	4.964±0.062	10.744±0.793	4.926±0.127	11.307	4.665
	×	6.498±0.089	4.345±0.036	17.091±0.798	5.289±0.047	12.030±0.355	5.353±0.117	11.873	4.996
XLNet	✓	6.586±0.177	4.289±0.085	16.566±0.998	5.305±0.094	12.683±1.127	5.075±0.161	11.945	4.889
	×	7.434±0.371	4.944±0.226	17.647±1.702	6.834±1.540	16.605±3.717	7.578±1.494	13.895	6.452

backbone (Transformer-A and Transformer-B) could have similar prediction performance (worse performance on average RMSE but better on MAE) as the state-of-the-art numerical based methods (Informer and Autoformer with temporal embedding). In addition, comparing results using Prompt A and results using Prompt B (the last four rows), it can be seen that the two prompt types have very close performance. Considering that Prompt B further removes the sensitive POI id information from Prompt A, Prompt B is a preferable prompt type. The above results and analysis could answer the RQ1 and indicate that using mobility prompts has better prediction performance compared to the basic numerical based forecasting methods and is able to achieve comparable performance as the state-of-the-art numerical based methods with temporal embedding.

4.5 Performance of AuxMobLCast and Ablation Study

In this section, we focus on answering RQ2 and RQ3. In the proposed AuxMobLCast, autoregressive language model GPT-2 is selected as the decoder and we explore the performance of using the following language foundation models as the encoder:

- BERT [6]: bert-base-uncased, 12-layer, 768-hidden, 12-heads, 110M parameters.
- RoBERTa [12]: roberta-base, 12-layer, 768-hidden, 12-heads, 125M parameters.
- XLNet [21]: xlnet-base-cased, 12-layer, 768-hidden, 12-heads, 110M parameters.

The details of these models and corresponding pre-trained weights are available through the links given in Table 2.

The prediction results of using different encoders on three datasets are listed in Table 4. In this table, a ✓ under the Aux column indicates that the auxiliary POI category classification is enabled, while a × means that the auxiliary classification is removed and AuxMobLCast is the basic encoder-decoder structure. In this part of the experiments, Prompt C given in Table1 is applied as inputs and output targets. When the auxiliary classification is disabled (rows marked with ×), the POI category target is also dropped from the Prompt C and only the mobility target part is kept during the fine-tuning process.

4.5.1 Different Encoders. If we jointly compare the results (rows with ✓) reported in Table4 with the top performers in Table3, we observe that using BERT as the encoder in our AuxMobLCast outperforms the performance of Informer and Transformer-B on average. Although the results of BERT and RoBERTa on Miami are not the best, they achieve significant performance improvements on Dallas, compared to all the baselines in Table3. More specifically, BERT is almost on par with RoBERTa on NYC dataset, while using BERT further reduces the RMSE by 9.5% on Dallas. Compared to BERT and RoBERTa, using XLNet seems not optimal under our AuxMobLCast pipeline. Its average forecasting performance is slightly worse than Transformer-B. However, using XLNet as the encoder still has a good forecasting performance on the most challenging Dallas dataset. These results suggest that our AuxMobLCast pipeline can be adapted to multiple language foundation models and BERT is the most suitable encoder structure. It also justifies our hypothesis that applying pre-trained language foundation models with mobility prompts could be a new direction for addressing the human mobility forecasting task.

4.5.2 With or Without Auxiliary Classification. Table4 also reports the ablation study results of comparing the row with a ✓ against the row with a × under each encoder setting. On average, introducing the auxiliary POI category classification task brings performance gain for the frameworks using BERT or RoBERTa as the encoder. For the configuration using XLNet as the encoder, enabling the auxiliary task results in a better forecasting performance on the Miami dataset but has a worse performance (both RMSE and MAE) on the Dallas dataset. Therefore, XLNet is not the optimal encoder under the proposed AuxMobLCast pipeline. For the RoBERTa encoder setting, we witness a larger RMSE but a smaller MAE performance on using the auxiliary task on the Dallas dataset. From the table, we can also notice that when BERT is chosen as the encoder, using the auxiliary task outperforms the setting without the auxiliary part by a relatively large margin on the RMSE of Dallas and Miami datasets. Based on the above analysis, although the auxiliary POI category classification does not reduce the forecasting errors for every encoder structure, it still yields substantial improvements for the BERT encoder setting. This further confirms that BERT is a favourable encoder selection for AuxMobLCast.

4.5.3 With or Without Date Information in Prompts. Based on the previous experiments (results given in Table 3), we have noticed that the temporal embedding could contribute to prediction performance gain for the numerical based forecasting methods. In this part of the experiment, we would also like to investigate the importance of temporal information (*i.e.*, the date information of the mobility data) under the proposed AuxMobLCast framework. To this end, we further evaluate the prompts without including

Table 6: The comparison of numerical forecasting methods and our AuxMobLCast under the zero-shot setting.

Training	Test	Method	RMSE	MAE
NYC	Miami	Transformer	15.867±0.202	5.220±0.084
		Reformer	15.488±0.169	5.401±0.016
		Informer	16.333±0.297	5.181±0.067
		Autoformer	9.445±0.095	5.020±0.049
		XLNet	18.801±2.840	7.228±3.960
		BERT	20.272±1.432	5.949±0.223
		RoBERTa	17.834±0.284	5.598±0.030
	Dallas	Transformer	31.207±0.304	5.721±0.098
		Reformer	30.502±0.313	5.897±0.022
		Informer	31.314±0.827	5.615±0.077
		Autoformer	19.239±0.564	5.327±0.065
		XLNet	21.341±1.733	8.291±1.008
		BERT	17.396±0.995	5.472±0.027
		RoBERTa	17.415±0.224	5.309±0.021
Miami	NYC	Transformer	6.656±0.044	4.341±0.023
		Reformer	7.514±0.056	4.770±0.035
		Informer	6.429±0.074	4.236±0.036
		Autoformer	6.525±0.065	4.432±0.048
		XLNet	7.158±0.178	4.304±0.015
		BERT	6.295±0.066	4.204±0.019
		RoBERTa	6.289±0.061	4.209±0.032
	Dallas	Transformer	21.405±0.373	5.316±0.033
		Reformer	25.205±0.832	5.723±0.056
		Informer	21.688±0.510	5.198±0.045
		Autoformer	21.267±0.990	5.350±0.037
		XLNet	16.747±0.150	5.149±0.019
		BERT	15.546±0.241	5.723±0.224
		RoBERTa	20.920±1.245	5.202±0.048
Dallas	NYC	Transformer	6.733±0.753	4.447±0.066
		Reformer	7.556±0.036	4.823±0.023
		Informer	6.766±0.078	4.497±0.075
		Autoformer	6.939±0.204	4.855±0.167
		XLNet	7.202±0.371	4.702±0.225
		BERT	6.231±0.066	4.162±0.017
		RoBERTa	6.291±0.144	4.249±0.090
	Miami	Transformer	10.904±0.129	5.995±0.037
		Reformer	11.259±0.715	5.287±0.059
		Informer	9.657±0.422	5.076±0.043
		Autoformer	10.321±0.665	5.457±0.128
		XLNet	15.801±2.490	5.771±0.291
		BERT	14.014±0.741	5.342±0.055
		RoBERTa	16.031±0.626	5.330±0.123

the temporal date information. By removing the temporal information (the blue parts in Table 1), the input prompt is then simplified as: `there were 11, 11, 10, 12, 9, 12, 6, 13, 10, 15, 16, 8, 8, 13, 19 people visiting POI on each day.`[4]

[4]Using the same example as shown in Table 1.

Based on this simplified prompt, Table 5 lists and compares the performance of our AuxMobLCast under the settings of with or without temporal date information. Similar to previous experiments, we also evaluate AuxMobLCast with different pre-trained language models (*i.e.*, BERT, RoBERTa, and XLNet). From the table, it can be observed that the performance of all three models have decreased after removing the date information. More specifically, the performance reduction of using RoBERTa is smaller than the configurations of using BERT and XLNet. Without the date information in the input prompts, it is difficult for language models to capture the relationships (*e.g.*, weekly patterns) between numerical tokens and temporal contextual tokens. The lacking of modelling such relationships could explain the performance reduction.

4.6 Zero-shot Performance

To further investigate the performance of AuxMobLCast, we conduct an experiment under the zero-shot setting. Specifically, we train each method on one dataset and test the trained model on the test set of the rest two datasets (*e.g.*, train on NYC, test on Miami and Dallas). The comparison results of the state-of-the-art numerical based forecasting methods (including Transformer, Reformer, Informer, and Autoformer) and our AuxMobLCast (using XLNet, BERT, RoBERTa as encoders) are reported in Table 6. For numerical based methods, the temporal embeddings are enabled to consider the temporal information. The best performer under each training/test configuration has been shown in red in the table. From this table, it can be observed that AuxMobLCast achieves 4 top performers in a total of 6 configurations. The numerical based methods have better performance when the test set is Miami (for both training with NYC and training with Dallas situations). Although using the language foundation models in our AuxMobLCast does not demonstrate superior performance for all settings, it still shows that AuxMobLCast has a relatively good and promising generalisation ability. With further research, we believe that using pre-trained language models would achieve better and more robust performance in the task of human mobility forecasting.

5 DISCUSSION

In this work, we studied the application of language models especially pre-trained language foundation models for mining temporal sequential patterns to predict human mobility. To this end, mobility prompts are applied to transform the numerical mobility data into sentences as inputs and output targets. To utilise pre-trained language models, we also propose the AuxMobLCast based on the encoder-decoder architecture, in which an auxiliary task is explicitly introduced for classifying POI categories. The results show that addressing sequential numerical data prediction through mobility prompts and applying pre-trained language foundation models (*e.g.*, BERT) under the proposed AuxMobLCast pipeline could improve the forecasting performance.

Why It Works.

In recent years, Transformer and its variants have emerged as powerful models in addressing NLP tasks. Through the inherent attention mechanisms, Transformers can well discovering the latent relationship of the input sequence tokens. More recently, we have also witnessed the booming of Vision Transformers for processing

images or videos. Through the patching transformation, images are divided into patch sequences to fit the Transformer structure. Thus, for the time-series sequences (*e.g.*, human mobility data), we believe pre-trained language Transformers could also suitable in handling time-series data with proper transformation from the raw numerical data to language sequences (*i.e.*, the mobility prompts). Through the mobility prompts, the intra-relation of numerical values at different time steps as well as the inter-relation between numerical values and contextual information (*e.g.*, temporal date information) would be better modelled simultaneously by Transformers, which leads to good forecasting performance.

Broader Impact.

The findings of this research provide visionary ideas and novel insights for human mobility forecasting tasks. The human mobility forecasting task analysed in this paper shows a glimpse of the potential applications in the direction of applying pre-trained language models with prompts. How to leverage pre-trained language models for various numerical forecasting applications (*e.g.*, weather forecasting, demand forecasting) could lead to new research directions. Although the improvement gain demonstrated in this paper is not very large, we believe that with further and deeper research, using pre-trained language models could be a new trend for the spatio-temporal and human mobility forecasting areas. Further, we hope this work would also facilitate the related research of multimedia assistant systems (*e.g.*, integrating spatio-temporal forecasting ability with Siri or Cortana).

Future Work.

As this is the first research that leverages pre-trained language models for discovering temporal sequential patterns in mobility forecasting tasks, there are still several limitations. One limitation of this study is about the mobility prompt generation. In the future, we plan to fully investigate mobility prompts based on the recent prompt learning techniques. An automatic approach for transforming diverse sequential numerical behaviour data, as well as various types of time-series data, will be beneficial in exploring the forecasting ability of pre-trained language models. In addition, how to explore pre-trained language models for multi-variate time-series data forecasting could be another interesting future direction.

ACKNOWLEDGMENTS

This work was supported by Australian Research Council (ARC) Discovery Project *DP190101485*. This paper is also a contribution to the IEA EBC Annex 79.

REFERENCES

[1] Alexei Baevski and Abdelrahman Mohamed. 2020. Effectiveness of self-supervised pre-training for asr. In *ICASSP 2020-2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 7694–7698.

[2] Dzmitry Bahdanau, Kyunghyun Cho, and Yoshua Bengio. 2015. Neural Machine Translation by Jointly Learning to Align and Translate. In *3rd International Conference on Learning Representations, ICLR 2015, San Diego, CA, USA, May 7-9, 2015, Conference Track Proceedings*, Yoshua Bengio and Yann LeCun (Eds.).

[3] Taylor Berg-Kirkpatrick and Daniel Spokoyny. 2020. An empirical investigation of contextualized number prediction. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*. 4754–4764.

[4] Rishi Bommasani, Drew A Hudson, Ehsan Adeli, Russ Altman, Simran Arora, Sydney von Arx, Michael S Bernstein, Jeannette Bohg, Antoine Bosselut, Emma Brunskill, et al. 2021. On the Opportunities and Risks of Foundation Models. *arXiv preprint arXiv:2108.07258* (2021).

[5] Junyoung Chung, Caglar Gulcehre, KyungHyun Cho, and Yoshua Bengio. 2014. Empirical Evaluation of Gated Recurrent Neural Networks on Sequence Modeling. *arXiv preprint arXiv:1412.3555* (2014).

[6] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, NAACL-HLT*, Jill Burstein, Christy Doran, and Thamar Solorio (Eds.). Association for Computational Linguistics, 4171–4186.

[7] Jie Feng, Yong Li, Chao Zhang, Funing Sun, Fanchao Meng, Ang Guo, and Depeng Jin. 2018. Deepmove: Predicting human mobility with attentional recurrent networks. In *Proceedings of the 2018 world wide web conference*. 1459–1468.

[8] Sepp Hochreiter and Jürgen Schmidhuber. 1997. Long Short-term Memory. *Neural computation* 9, 8 (1997), 1735–1780.

[9] Nikita Kitaev, Lukasz Kaiser, and Anselm Levskaya. 2020. Reformer: The Efficient Transformer. In *8th International Conference on Learning Representations, ICLR 2020, Addis Ababa, Ethiopia, April 26-30, 2020*. OpenReview.net.

[10] Shiyang Li, Xiaoyong Jin, Yao Xuan, Xiyou Zhou, Wenhu Chen, Yu-Xiang Wang, and Xifeng Yan. 2019. Enhancing the locality and breaking the memory bottleneck of transformer on time series forecasting. *Advances in Neural Information Processing Systems* 32 (2019), 5243–5253.

[11] Qiang Liu, Shu Wu, Liang Wang, and Tieniu Tan. 2016. Predicting the next location: A recurrent model with spatial and temporal contexts. In *Thirtieth AAAI conference on artificial intelligence*.

[12] Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. 2019. Roberta: A robustly optimized bert pretraining approach. *arXiv preprint arXiv:1907.11692* (2019).

[13] Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, Gretchen Krueger, and Ilya Sutskever. 2021. Learning Transferable Visual Models From Natural Language Supervision. In *ICML (Proceedings of Machine Learning Research, Vol. 139)*, Marina Meila and Tong Zhang (Eds.). PMLR, 8748–8763.

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

[15] Georgios Spithourakis and Sebastian Riedel. 2018. Numeracy for Language Models: Evaluating and Improving their Ability to Predict Numbers. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*. 2104–2115.

[16] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. In *Advances in neural information processing systems*. 5998–6008.

[17] Xian Wu, Chao Huang, Chuxu Zhang, and Nitesh V Chawla. 2020. Hierarchically structured transformer networks for fine-grained spatial event forecasting. In *Proceedings of The Web Conference 2020*. 2320–2330.

[18] Jiehui Xu, Jianmin Wang, Mingsheng Long, et al. 2021. Autoformer: Decomposition transformers with auto-correlation for long-term series forecasting. *Advances in Neural Information Processing Systems* 34 (2021).

[19] Hao Xue, Flora Salim, Yongli Ren, and Nuria Oliver. 2021. MobTCast: Leveraging Auxiliary Trajectory Forecasting for Human Mobility Prediction. *Advances in Neural Information Processing Systems* 34 (2021).

[20] Hao Xue and Flora D. Salim. 2021. TERMCast: Temporal Relation Modeling for Effective Urban Flow Forecasting. In *Advances in Knowledge Discovery and Data Mining - 25th Pacific-Asia Conference, PAKDD 2021, Virtual Event, May 11-14, 2021,*, Vol. 12712. Springer, 741–753.

[21] Zhilin Yang, Zihang Dai, Yiming Yang, Jaime Carbonell, Russ R Salakhutdinov, and Quoc V Le. 2019. Xlnet: Generalized autoregressive pretraining for language understanding. *Advances in neural information processing systems* 32 (2019).

[22] Haoyi Zhou, Shanghang Zhang, Jieqi Peng, Shuai Zhang, Jianxin Li, Hui Xiong, and Wancai Zhang. 2021. Informer: Beyond efficient transformer for long sequence time-series forecasting. In *Proceedings of AAAI*.

[23] Kaiyang Zhou, Jingkang Yang, Chen Change Loy, and Ziwei Liu. 2021. Learning to prompt for vision-language models. *arXiv preprint arXiv:2109.01134* (2021).