Joint Gated Co-Attention Based Multi-Modal Networks for Subregion House Price Prediction

Pengkun Wang, Chuancai Ge, Zhengyang Zhou, *Student Member, IEEE*, Xu Wang, Yuantao Li, and Yang Wang, *Senior Member, IEEE*

Abstract—Urban housing price is widely accepted as an economic indicator which is of both business and research interest in urban computing. However, due to the complex nature of influencing factors and the sparse property of transaction records, to implement such a model is still challenging. To address these challenges, in this work, we study an effective and fine-grained model for urban subregion housing price predictions. Compared to existing works, our proposal improves the forecasting granularity from city-level to mile-level, with only publicly released transaction data. We employ a feature selection mechanism to select more relevant features. Then, we propose an integrated model, JGC_MMN (Joint Gated Co-attention Based Multi-modal Network), to learn all-level features and capture spatiotemporal correlations in all-time stages with a modified densely connected convolutional network as well as current ingredients and future expectations. Next, we devise a novel JGC based fusion method to better fuse the heterogeneous data of multi-stage models by considering their interactions in temporal dimension. Finally, extensive empirical studies on real datasets demonstrate the effectiveness of our proposal, and this fine-grained housing price forecasting has the potential to support a broad scope of applications, ranging from urban planning to housing market recommendations.

Index Terms—Subregion house price prediction, multi-modal networks, heterogeneous data fusion

1 Introduction

Housing price forecasting plays a vital role in macroeconomic and financial decision supporting. In the past decade, a global financial crisis has been witnessed, due to inaccurate housing price forecasting and unconscionable financial policymaking [1]. In terms of spatial granularities of predictions, existing studies on housing price forecasting models are mostly on city levels, for supporting macroeconomic analysis and policymaking. The city-level forecasting, however, can not capture the fact of imbalanced development between a city's mile-level subregions. For instance, in Xi'an, the average real estate prices of three districts in Sept. 2018 increased more than 10 percent while the average prices of the other six districts decreased about 5 percent during the same period [2].

Thus, in this work, we study another type of housing price forecasting, that supports fine-grained and micro-level analysis. Specifically, the analysis of mile-level subregion is more fine-grained than urban regions. Such a type of forecasting depicts the potential fluctuations, the distribution of housing prices among different urban subregions as well as the relevant static impact factors. With the help of that, we can find broad applications in urban planning, such as community service support and transportation facilities optimization.

There exist many studies, however, on city-level housing price forecasting, which can be categorized as machine learning-based methods [3], [4], [5], [6], [7], [8], [9] and deep learning-based methods [10], [11], [12], [13]. Most machine learning-based methods only capture the temporal dependencies. Some newly proposed methods involve extra spatial dependencies by learning low-level spatial features. Deep learning methods aim at predicting housing prices by capturing temporal correlations. Nevertheless, all solutions cannot be extended to the subregion scenarios because of the unavailability of involving the all-level spatiotemporal influences in the learning process and the unavoidable overfitting in small and sparse subregion-level datasets. Also, there exist works [7], [14], [15], [16] on predicting the price of individual houses. However, such methods have a high dependency on data quality, i.e., requiring detailed property-transaction records and features, which are unavailable for most publicly released data.

Challenges. However, challenges arise for accomplishing fine-grained urban subregion housing price forecasting and analysis. The influence factors of housing price forecasting are known to be complex. Existing works mostly take the long-term, short-term, and low-level spatial correlations of house prices into account. Moreover, as mentioned in [17], the current policies also affect the tendency of housing prices. And the trend of macroeconomic or future price-growth expectation also has a great influence on the current housing price [17], e.g., the economic growth of Japan in the 1980s and China's megalopolis in the last decade. Besides, some additional static features such as house properties, transportation conditions, school districts, surrounding environments, and facilities can also significantly affect individual housing price. So far, how to design an integrated framework to incorporate various impact factors with considering their different characteristics remains challenging.

• *The authors are with the University of Science and Technology of China, Hefei, China. E-mail: {pengkun, gcc810, zzy0929, wx309, liyuantao}@mail.ustc.edu.cn, angyan@ustc.edu.cn.*

Manuscript received 15 Apr. 2020; revised 15 May 2021; accepted 24 June 2021. Date of publication 0 . 0000; date of current version 0 . 0000.
(Corresponding author: Yang Wang.)
Recommended for acceptance by L. Xiong.
Digital Object Identifier no. 10.1109/TKDE.2021.3093881

Another challenge is the sparsity of property-transaction data. Sparse data limits the sample size and incurs selectivity sample bias for building efficient and accurate forecasting models [18], letting alone the limited availability of publicly released data with heterogenous transaction properties. Such data might be dense enough for city-level forecasting but is shown to be sparse when the urban region is decomposed into mile-level subregions. In particular, the sparse data can result in insufficient house price features which are critical in model training.

Contributions. In summary, previous works on housing price forecasting never set foot on the issues of mile-level subregion housing price predictions.

To our best knowledge, our JGC_MMN is the first work on effective mile-level subregion housing price forecasting, which has profound effects on trading recommendations for housing markets and urban planning for public facility and optimization. Our main contributions are as follows.

i) We propose to use densely connected networks to capture the all-level features in order to overcome the sparsity challenge and alleviate the corresponding overfitting. Besides, we consider more well-selected factors, including current ingredients and future price-growth expectations, as the submodules of prediction. ii) We propose a novel multi-modal framework by fusing multiple learners on the different temporal characteristics (i.e., long-term periodicity, recent tendency, current, and future periods) for depicting spatiotemporal dependencies. To achieve that, we improve the original DenseNet structure, combine the Kalman Filter, and adjust the diverse structures to further improve the accuracy. iii) To fully fuse these numerous factors with four learners, we design a new method, JGC, to learn the correlations between them automatically by generating joint attention flows within various modalities and filtrating noises of multiple similar modalities with the gated function. iv) We evaluate our proposed JGC_MMN with real-world house price datasets from NYC and Beijing. Extensive cross-validation experiments demonstrate that our model can improve the accuracy significantly compared to the start-of-the-art solutions.

The rest of this paper is organized as follows. After discussing related works in Section 2, Section 3 introduces preliminaries and formalizes the problem. Section 4 investigates our technical proposals. Section 5 presents empirical studies, Section 6 discusses some practical issues of this paper, and Section 7 concludes the paper.

2 Related Works

Many efforts have been paid in housing price forecasting, including city-level housing price forecasting [3], [4], [5], [6], [7], [8], [9], [10], [11], [12], [13], [19], [20], [21], [22], [23], [24], [25], [26], [27], [28], [29], [30], [31], [32] and predicting housing price of individual real estates [7], [14], [15], [16]. The individual housing price prediction methods rely heavily on the level of detailed property-transaction records and features, whereas most publicly released data is not adequate to support that. So in this paper, we focus on how to achieve fine-grained housing price forecasting with such publicly released data.

2.1 City-Level Housing Price Forecasting

Existing results on housing price forecasting models are mostly on city-level, and researches in this field can also be divided into three categories, geostatistical methods, machine learning-based methods, and deep learning-based methods.

- *Geostatistical methods:* Geostatistical approaches mainly include Geographically Characteristically Temporally Weighted Regression (GCTWR) models [19], [20], [21], [22], [23] and Eigenvector Spatial Filter Regression (ESFR) models [13], [24], [25]. The GCTWR models, which are a typical category of geostatistical models, integrate both temporal and spatial information in weighted matrices to capture both spatial and temporal heterogeneity in house price predictions. In particular, [19] first proposes a geographically weighted regression (GWR) model for spatiotemporal analysis and modeling. [20] proposes a geographical and temporal weighted regression (GTWR), which is an extension of geographically weighted regression, to account for spatiotemporal local effects, and [22] employs the GTWR approach to estimate house prices by using travel time distance as the metric. Based on the previous GWR and GTWR models, [21] develops a geographically and temporally weighted autoregressive model (GTWAR) to account for both non-stationary and auto-correlated effects simultaneously, and formulates a two-stage least squares framework to estimate this model. To further improve house price estimations, [23] proposes a Kernel-Based GTWAR (KBGTWAR) model by incorporating the basic principles of support vector machine regression. The ESFR models incorporate spatial influences with the traditional Ordinary Least Square (OLS) model to achieve better performances. An early study conducted by [24] analyzes two different spatial filtering approaches to create spatial predictors which can be easily incorporated with conventional regression models. Considering the interactions in spatial perspective, [25] introduces an ESF model to the predictions of house prices to achieve a comprehensive understanding of complex spatial dependencies and autocorrelations. [13] verifies the spatial distributions of house prices with an eigenvector spatial filtering (ESF) procedure and then analyzes the local variations and spatial heterogeneity of house prices.
- *Machine learning-based methods:* Most machine learning-based methods can only capture the single temporal dependencies. In particular, [3] first uses the VAR (Vector Auto Regression) to forecast city-level average housing price with time series analysis. By taking temporal dependencies into account, [5] forecasts housing prices with the STAR (Smooth Transition AutoRegression) model. And [6] proposes a hybrid algorithm for the housing price prediction by combining the SVR and particle swarm optimization together.[7] makes the predictions by building a univariate model with the ARIMA (Auto regression Integrated Moving Average) model, which is applied for the short-term prediction of time series. [8] uses a decision tree-based method for summarizing possible influence factors of housing prices. [9] uses a SOM (Self Organizing Map) and LVQ (Learning Vector Quantization) combined complex network for the real estate forecasting. In addition, some methods such as the SPVAR, CAR, SAR, and

Lasso and Ridge regression models, can involve extra spatial dependencies by learning low-level spatial features. On the basis of VAR, [4] forecasts housing price for districts in city-level with the SPVAR (Space Vector AutoRegressive) model by taking both spatial (low-level) and temporal dependencies into account. The CAR models are also a typical category of spatial econometric models. For modeling and statistical analysis of spatiotemporal economic data, [33] proposes a spatial temporal conditional autoregressive model. Similarly, [26] proposes a poisson conditional autoregressive model to analyze spatiotemporal economic data. Given the fact that traditional CAR neighborhood selections are based on distances or boundaries between regions, [27] proposes a Stochastic Neighborhood CAR (SNCAR) model where the neighborhood selection depends on unknown parameters. The SAR models are another typical category of spatial econometric models, and are similar to the spatial lag model. As a pioneer of SAR methods, [28] proposes a two-stage least squares spatial estimator to improve the spatial lag model. Based on [28], [29] proposes the best spatial two-stage least squares estimators, which are asymptotically optimal instrumental variable estimators, to further improve the spatial lag model. To achieve cross-sectional spatial autoregressive, [30] combines the spatial lag model with nested random effects to propose new estimators based on the instrumental variable approaches, and the proposed approach is used to analyze the house price variation in England. The lasso and ridge regression models are the variants of standard linear regression by adding L1 and L2 regularization, respectively. Specifically, [31] employs both the lasso and ridge regression models on house price predictions, and demonstrates that both these two regressions can deal with multi-collinearity. [32] involves multiple boosting tricks into ridge regression to achieve better performances.

- *Deep learning-based methods:* Deep learning-based methods refer to using models such as LSTM (Long Short Term Memory), or ANN (Artificial Neural Network), to capture low-level temporal correlations to predict housing prices. Specifically, [10] utilizes ANN to predict house price on city-level by considering low-level spatial and additional static features. It investigates the predictive power of both the hedonic model and the ANN model. [4] proposes a memristors-based ANN model to learn a multi-variable regression model from housing price labeled samples. [7] adopts massive deep learning functions such as Adam optimizer and Relu function to capture the price trends, and then fed them into the ARIMA model to predict housing price on city-level. [11] first uses LSTM to build housing price prediction model by exploiting temporal correlations, and further employs stateful LSTM and stack LSTM to improve the accuracy.

2.2 Analysis of City-Level Works

The reasons that these city-level solutions cannot be extended to solve the problem of subregion-level housing price forecasting are as follows:

- *Unavailability of involving all-level spatiotemporal dependencies:* Most previous models can only capture temporal dependencies with time series methods. Given that some newly proposed methods can utilize low-level spatial dependencies to improve housing price prediction model, the all-level spatial dependencies in near and far neighborhoods, which are essential for region-level housing price forecasting, have never been effectively captured by proposed methods due to the lack of massive convolutions in the structure of these methods.
- *Inefficiency of fusions:* Feature-based fusion is widely used in previous housing price prediction methods to improve the accuracy of prediction by involving more realistic features. However, owing to the fact that the increasing number of features for fusing may lead to the curse of dimensionality and spatial-temporal asynchronism, the effectiveness of feature-based fusion remains limited. To enhance the effectiveness of feature-based fusion, model-based fusion is employed in some recent methods. However, instead of effectively capturing the correlations among multiple modalities, traditional model-based fusion tends to lean to some special models, hence affects the accuracy of proposed models.
- *No non-linear capability of classic geostatistical and machine learning methods:* Due to the lacking of non-linear capability, geostatistical approaches and most machine learning-based methods cannot extract the complicated interactions among multiple influential modalities which are essential for house price learning in both spatial and temporal perspectives, therefore the performances of these methods can be significantly limited.
- *Overfitting of advanced deep learning approaches:* In recent years, tons of advanced deep learning methods including DNN-based Deep-ST [34] , ST-ResNet [35] and ST-InceptionV4 [36] have been devised to address the spatiotemporal prediction issues. Actually, due to the network connectivity, these methods are mainly dependent on high-level features instead of all-level features, and this eventually leads to their weakness in generalization and unavoidable overfitting in small and sparse subregion datasets.

3 Preliminaries

3.1 Problem Definition

In this paper, we formally define basic concepts as well as the problem studied in the work.

Definition 1 (City Region). Given a city, its urban region can be divided into small square-shaped subregions with the side-length of d_0[1] kilometers. So, the urban region can be represented by a set of equal-sized grids, with m_r rows and m_c columns. A grid at ith row and jth column can be denoted as $r_{i,j}$, where $i \in \{1, \ldots, m_r\}$ and $j \in \{1, \ldots, m_c\}$.

1. The setting of d_0 should balance the trade-off between the fineness of urban region house price predictions and the densities of historical data. To eliminate the influences of some adventive abnormal transactions, we request that there should be no less than 10 transactions in one single area during one whole month. To this end, in our implementation, we divide cities into small square-shaped areas with the length of 2 kilometers.

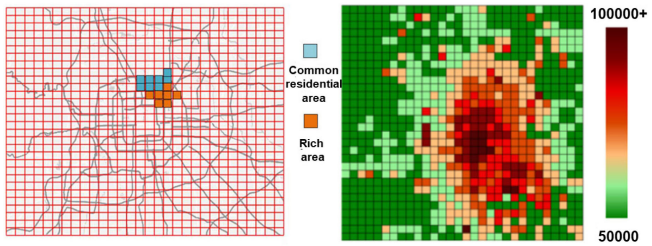


Fig. 1. An Example of Beijing. (a) Regions; (b) Housing Price Distributions

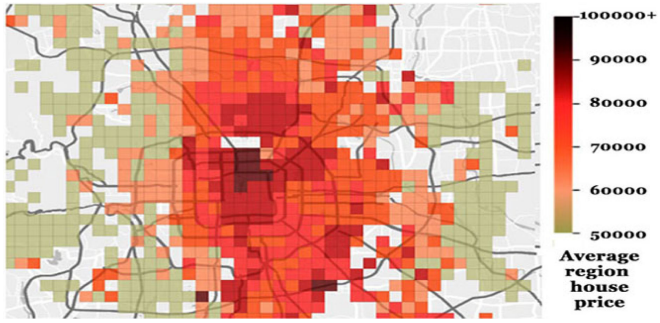


Fig. 2. Subregion Housing Prices of Beijing during 2017.

Fig. 1a illustrates the urban regions of Beijing, which has been divided into a 30×30 subregions.

Definition 2 (Housing Price Set). Given a month T and a city, we define the housing transaction price set of the entire city during this month as $\mathbb{S}_T$. We have $\mathbb{S}_T = \mathbb{S}^T_{r_{1,1}} \cup \ldots \cup \mathbb{S}^T_{r_{m_r,m_c}}$, where $\mathbb{S}^T_{r_{i,j}}$ $(i \in \{1 \ldots m_r\},\ j \in \{1 \ldots m_c\})$ denotes the housing price set of an urban subregion $r_{i,j}$ during month T. Within each set $\mathbb{S}^T_{r_{i,j}}$, a transaction can be uniquely identified by the transaction timestamp t_k together with the subregion IDs, so that the transaction can be represented by $\mathbb{S}^T_{r_{i,j}} = \{s^{t_1}_{r_{i,j}}, s^{t_2}_{r_{i,j}} \ldots\}$ $(t_1, t_2, \ldots \in T)$, where $s^{t_k}_{r_{i,j}}$ indicates the price of the transaction in region $r_{i,j}$ and time t_k.

Definition 3 (Housing Price of a Subregion). Given a month T and a subregion $r_{i,j}$, the housing price of this subregion can be calculated by:

$$p^T_{r_{i,j}} = \frac{1}{\left|\mathbb{S}^T_{r_{i,j}}\right|} \sum_{k=1}^{\left|\mathbb{S}^T_{r_{i,j}}\right|} s^{t_k}_{r_{i,j}}. \tag{1}$$

The subregion housing prices of all $m_r \times m_c$ subregions of month T can be denoted as a tensor $p^T_{r_{i,j}} \in \mathbb{R}^{m_r \times m_c \times 1}$. The image-like housing price matrix is shown in Fig. 1b.

Definition 4 (Subregion Housing Price Forecasting). Given a historical housing price dataset $\{\mathbb{S}_T | t = 0, \ldots, n\}$ of a city, our purpose is to design a method such that the housing price $p^{n+1}_{r_{i,j}}$ can be predicted for any subregion $r_{i,j}$.

The quality of the prediction can be measured by RMSE (Root Mean Square Error) as shown by Equation 2, where $\widehat{p^{n+1}_{r_{i,j}}}$ denotes the predicted housing price of subregion $r_{i,j}$.

$$RMSE = \sqrt{\frac{1}{m_r \times m_c} \sum_{i=1}^{m_r} \sum_{j=1}^{m_c} \left(\widehat{p^{n+1}_{r_{i,j}}} - p^{n+1}_{r_{i,j}}\right)^2}. \tag{2}$$

4 Subregion Housing Price Forecasting Model

In this section, we first analyze the parameters which can influence the subregion housing price and then introduce the forecasting model for the subregion housing price problem.

4.1 Influence Factors of Subregion Housing Price

In most previous works, the housing price prediction is modeled in the form of temporal dependency analysis . Some recent research suggests that the de facto influence factors are more complicated [37]. To this end, we analyze the ingredients that influence the future housing price, systematically.

Spatial Correlations. To formulate the problem in Section 2, we have divided an entire city into small subregions. Intuitively, the housing prices of two neighboring subregions have strong correlations. For instance, a more developed subregion tends to be more commercially bustling, more convenient in transportation, and safer in securities. Such ingredients have a radiative effect on their neighboring subregions as shown in Fig. 2.

Long-Term Periodicity and Short-Term Tendency. It is widely accepted that the future housing price is greatly affected by long-term periodicity and short-term tendency. In [17], the influences of long-term periodicity[2] are discussed. The impacts of short-term tendency[3] on future housing price are evaluated in [38].

Current Ingredients. It has been concluded that the future housing price is greatly affected by many current economic and social elements, such as down-payment ratios, mortgage rates, house property tax policy, GDP (Gross Domestic Product) growth, and demographic factors, and some other static features.

The Future Price-Growth Expectations. Theoretically, from an economic perspective, the future price-growth expectations would give feedbacks on the tendency of housing price, once the public shows cognitions on the housing market [37]. Such a type of influence has been observed in Tokyo before 1991 and in China in the past decade.

In summary, the influence factors can be generalized into four parts: *Long-term* spatiotemporal correlations, *short-term* spatiotemporal correlations, *current* economic and social ingredients, and *future* price-growth expectations. All these factors are integrated for predicting subregion housing prices.

4.2 Major Components of the Forecasting Model

Hereby, we propose the solution framework for the problem of subregion housing price forecasting. The architecture overview is shown in Fig. 3, which consists of five major

2. The regression period of long-term periodic influences on housing price prediction l_{long} can be set to 5 years. [17].

3. In previous studies [38], the regression period of short-term tendency l_{short} is set to 12 months (1 year).

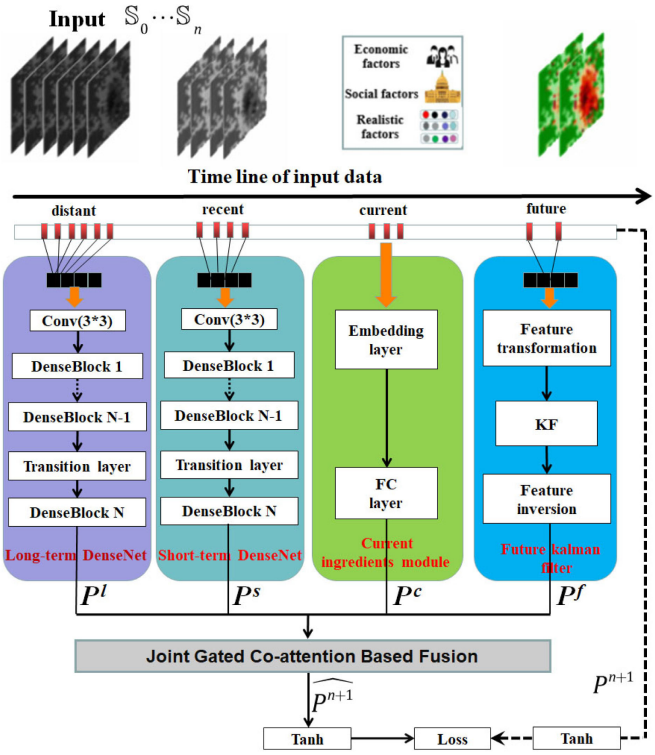


Fig. 3. Architecture.

components: i) Long-term spatiotemporal DenseNet; ii) Short-term spatiotemporal DenseNet; iii) Current ingredient module; iv) Kalman Filter for future price-growth expectations; v) Joint gated co-attention based fusion. The previous four components are in correspondence to the above-mentioned influence factors. We organize them as four types of inputs in accordance with their time dimensional attributes, i.e., distant periodicity, recent tendency, current, and future factors, as shown in the upper half of Fig. 3. And finally, the outputs of the previous components are fused with the last fusion component by considering the correlations between the previous components.

Given historical transactions of a city, we transform them into tensors $P^l \in \mathbb{R}^{m_r \times m_c \times 5}$ and $P^s \in \mathbb{R}^{m_r \times m_c \times 12}$, whereas each tensor refers to the monthly aggregated housing price values.

The long-term and short-term DenseNet components share the same network structure with a modified DenseNet [39]. Such a structure captures the spatial correlations of housing prices between neighboring subregions and the temporal dependencies during different time periods. For the current ingredient component, we manually extract features from economic, social, and static ingredients, then feed them into the embedding layer and the FC (Fully Connected) layer. The last component simulates the effects of future price-growth expectations. In our implementation, we use the Kalman Filter to model the subjective expectations from the public.

The outputs of the previous four components, P^l, P^s, P^c, and P^f, are fed into the joint gated co-attention based fusion as the input respectively. The integrated result is further mapped by a $Tanh$ function to interval $[-1, 1]$. Compared with standard logistic functions, $Tanh$ function offers a faster convergence in processing back-propagation learning [40].

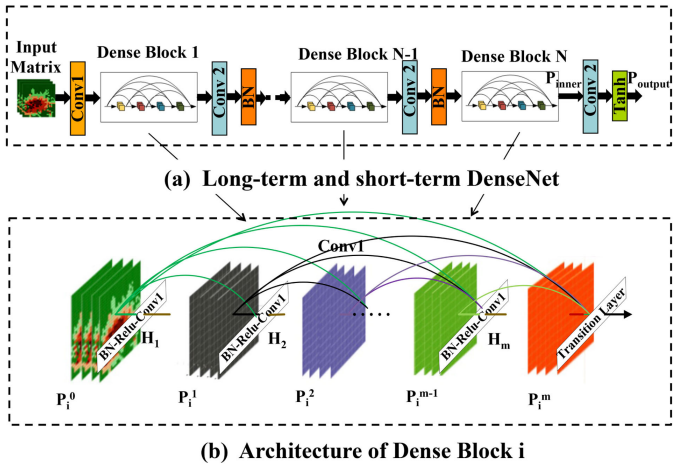


Fig. 4. Architecture of long-term and short-term DesNet.

4.2.1 *Long-Term and Short-Term Spatial-Temporal DesNet*

The long-term and short-term components share the same network structure consisting of three sub-components: convolution, dense block, and a transition layer. Based on the particular characteristics of housing price predictions, we modify DenseNet as illustrated in Fig. 4.

Convolution. As described in Section 3.1, the housing prices of neighboring subregions have obvious spatial correlations. Such a type of correlations can be effectively captured by adopting CNN (Convolution Neural Network), which has shown its efficiency on extracting spatial structural information [40]. Also, as shown in Fig. 2, this kind of correlations has radiative effects, not only affecting direct neighboring subregions, but also rather distant neighboring subregions. To perceive more neighboring urban subregions, we adopt a multi-layer CNN. [35]. For example, there are m convolutions (i.e., $Conv1$[4]), as shown in Fig. 4a, in one dense block. The total number of all $Conv1$s in the componet is $N \times m + 1$, so that the stack of convolutions is capable of capturing subregion housing price correlations city-wide.

Dense Block. With the increased number of layers, the issues of gradient vanishing and overfitting become more and more serious. To handle these issues, [39] proposes a densely connectivity mode *Dense Connectivity*, as illustrated in Fig. 4b. For dense block i, the input of layer m is:

$$I_m = H_m(P_i^0, P_i^1 \ldots, P_i^{m-1}),\ m = 1, 2, 3, \ldots \quad (3)$$

The function H_m is a nonlinear function consisting of one convolution $Conv1$, one $Relu$ function, and one BN (Batch Normalization) [41] function. The BN is to convert the output of each layer into a normal distribution, which can also avoid the overfitting significantly. Compared to P_i^0, the P_i^{m-1} is transformed into high-level features from low-level after several nonlinear functions. Notice that the connection mode between feature maps P_i^m and $P_i^0, P_i^1 \ldots, P_i^{m-1}$ is an channel-wise addition. We have:

$$CN(P_i^m) = sum\{CN(P_i^0), CN(P_i^1) \ldots, CN(P_i^{m-1})\}. \quad (4)$$

4. The kernel size of convolution $Conv1$ is fixed to 3×3.

Here, function CN denotes the channel number. And we can observe that the last layer's input comes from all the front layers output, which indicates that our model can learn the low-level and high-level features. When making decisions for prediction, our model depends on both the high-level and the low-level futures, which can help model remit overfitting in some extent. Besides, the features of sparse housing price datasets are insufficient, which makes it hard to capture the spatiotemporal characteristics completely. Thus this kind of densely connectivity can make sure that our model learns all levels of spatiotemporal features, which proves that we can have superior performance on small and sparse datasets.

Transition Layer. In the original DenseNet, an $average\ pooling$ function is used in the transition layer to reduce the features while they are lacking, which is unfit for sparse datasets. Besides, the original FC layer after Dense Block N is initially used for image classification, which can not be adopted for the task of forecasting subregion housing prices. Therefore, in the improved version of DenseNet of our work, we replace the $average\ pooling$ function by a BN function to further remit the overfitting.

To aggregate the multi-channel feature maps into the final prices better, we try a $Conv2$ to replace the original FC layer. As shown in Fig. 4a, P^{inner} has multiple channels after N dense blocks. The final output feature map is then calculated by the following equation,

$$P^{output} = f(W^{in} * P^{inner} + b^{in}). \tag{5}$$

Here, function f is the activation function $Tanh$. $*$ denotes the convolution $Conv2$[5]. W^{in} and b^{in} are the learnable parameters in the modified DenseNet.

4.2.2 Current Ingredient Module

As analyzed in Section 4.1, subregion housing prices can be influenced by many complex economic, social, and political factors as well as some additional static factors. All these factors at the current time point can be collectively known as current ingredients, and they can be divided into five categories, economic and social factors, transportation conditions, school district, house properties, and surrounding environment and facilities, as shown in Table 1.

Economic and Social Factors. As shown in Table 1, economic and social factors including the growth of GDP, mortgage rates, average incoming of the public, unemployment rate and the number of permanent residents. Figs. 5 and 6 illustrate the historical trends of the ingredients of mortgage rates, average incoming of the public and the number of permanent residents in both NYC and Beijing. As observed, these ingredients changed hugely, constantly and randomly in both cities. Due to the inherent correlations between economic and social ingredients and housing prices [17], we select the five economic and social ingredients as the main current ingredients.

Static Feature Selection. Also as demonstrated in Table 1, housing prices can also be influenced by other categories of static factors. However, the trade-off between costs and benefits of involving static factors should be carefully balanced in practice. To this end, we evaluate the detailed contribution of each factor to the final accuracy of subregion housing price prediction, and select the ones with higher contributions. Specifically, we employ the widely-used IGI (Information Gain Index) [42] as the metric to evaluate the validity and importance of each feature. The validity and importance v_i of feature i can be defined as follows:

$$v_i = \frac{sum(\mathcal{I}_i)}{sum(\mathcal{F}_i)}. \tag{6}$$

Where $\mathcal{I}_i$ denotes the information gain index of the i_th feature ($0 <= i <= 20$), and $\mathcal{F}_i$ denotes the frequency that this feature occurs. The validities of all additional static features are then evaluated and illustrated in Fig. 7. Intuitively, the features of total units and schools have the two maximum validities, while the factors of tourist spot and museum are with the two minimum importances. Next, we combine the top k valid static features with the five main current ingredients as the final current ingredients. We will discuss the impacts of the value of k in experiments.

Feature Processing. Notice that all static factors are confined to individual houses. To calculate a selected static factor of a given subregion, we use the mean value of the corresponding factors of all houses within this subregion. For the economic and social ingredients, we let each subregion be with the same values of them. Next, we integrate all features as a $m_r \times m_c \times (k+5)$ tensor and then feed it into the embedding layer to map the data fields into a structural and dense input space. Finally, we use the FC layer to transform low-dimensional values into the high dimensions in order to get ready for the final fusion.

4.2.3 Future Price-Growth Expectations

In this subelement, we simulate the subjective expectations of the public. [17] proposes a Kalman Filter (KF) based method to predict the influences of future price-growth expectations. We hereby adopt the KF-based filter into our integrated network. The solution of combining KF into the integrated networks is borrowed from the idea of bagging in machine learning. With the integrated KF-based methods and neural networks, we construct a novel stronger learner, which effectively enhances the prediction accuracy.

Given a historical housing price dataset $\{\mathbb{S}_T|_{T=1,\ldots,n}\}$, for time $n+1$ and subregion $r_{i,j}$, we define the housing demands of all residents of the subregion as $\mathscr{D}^{n+1}_{r_{i,j}}$. The growth rate of the housing demands in this region is defined as $\mathscr{G}^{n+1}_{r_{i,j}}$. The average trading price of the subregion is defined as $\mathscr{P}^{n+1}_{r_{i,j}}$.

By using the proposed KF-based method in [17], we first predict the housing demands $\mathscr{D}^{n+1}_{r_{i,j}}$ and the growth rate of the housing demands $\mathscr{G}^{n+1}_{r_{i,j}}$, based on the historical housing price set $\{\mathbb{S}_{n-1}, \mathbb{S}_n\}$. After predicting the housing demand and growth rate of the housing demand of future time $n+1$, we can calculate the expected housing price of region by:

$$\mathrm{E}\left(\mathscr{P}^{n+1}_{r_{i,j}} \middle| \{\mathscr{D}^{n+1}_{r_{i,j}}, \mathscr{G}^{n+1}_{r_{i,j}}\}\right) = \frac{\mathscr{D}^{n+1}_{r_{i,j}}}{r} + \frac{\mathscr{G}^{n+1}_{r_{i,j}}}{r(r+\nu)}. \tag{7}$$

Here, r and ν indicate the discount rate and the demand growth revision [17], respectively. With the method, we can calculate the price-growth expectation for each subregion in

5. The kernel size of $Conv2$ should be 1×1.

TABLE 1
Current House Pricing Features in NYC

Category		Attributes	Min	Max	Description
Economic and social factors		Growth of GDP	-5.4%	8.8%	The growth of GDP
		Mortgage rates	3.2%	7.1%	Mortgage rates of bank
		Average incoming of the public($)	40000	62000	The average incoming
		Unemployment rate	4.2%	8.9%	Unemployment rate of people
		Permanent residents(Million)	8.05	8.4	Number of residents
Static factors	Transportation conditions	Bus	0	34	Bus station
		Subway	3	20	Subway station
	School district	School	0	7	Organization of education
	House properties	Building class category	1	41	The category of house
		Block	1	16350	Encoding of address
		Residential units	1	13	Number of residential units
		Commercial units	1	12	Number of commercial units
		Total units	1	17	Number of total units
		Land square feet	104	4000	Number of land square feet
		Gross square feet	122	3000	Number of gross square feet
	Surrounding environment and facilities	Hospital	0	4	Number of hospitals
		Market	0	10	Number of Markets
		Museum	0	2	Number of museums
		Park	0	3	Number of parks
		Restaurant	0	36	Number of restaurants
		Tourist attractions	0	8	Number of tourist spots

the city, and generate the future price-growth expectation tensor sized $m_r \times m_c \times 1$.

4.2.4 Joint Gated Co-Attention Based Fusion

So far, we have discussed the learners in different temporal dimensions, the outputs of which should be fused. Theoretically, the impacts of each learner on the final prediction result should be differential, and the outputs of P^l, P^s, P^c, and P^f should be significantly interactional with each other. For instance, long-term, short-term tendencies and future expectations of housing prices can definitely be influenced by those current ingredients. This kind of correlations between the outputs of different components is not considered in previous multi-modality housing price prediction. Besides, the outputs of four components are quite heterogeneous. As demonstrated in Table 2, the properties, units and structures of data in these four modalities are disparate in some extent. Hence, how to fuse them reasonably is a challenging task.

To address the fusion issue of multi-modality learnings with considering the correlations between different learners, co-attention based fusion has been widely used in the fields of question answering [43], healthcare prediction [44], entity recognition [45], and commodity recommendation [46] in recent years. The co-attention mechanism aims at capturing the relationships among various modalities. However, co-attention based fusion can only capture partial correlations among different components due to their linear-combination-based conditional fusion. To this end, we propose a novel fusion method, JGC, as illustrated in Fig. 8, for completely fusing multi-modality components. The JGC based fusion includes three submodules: joint co-attention submodule, filtration gate submodule, and joint representation submodule. We will introduce the detailed design of this fusion module subsequently.

Joint Co-Attention Submodule. In this submodule, we first incorporate all linear combinations in the original co-attention based method as the conditional affinity matrices for components pairs. To learn the correlations between paired components completely, we further compute the corresponding joint affinity matrices between components defined as:

$$\begin{cases} A^l = softmax\left(\begin{bmatrix} P^l \oplus P^s \\ P^l \oplus P^c \\ P^l \oplus P^f \end{bmatrix}\right) \oplus softmax\left(\begin{bmatrix} P^l \otimes P^s \\ P^l \otimes P^c \\ P^l \otimes P^f \end{bmatrix}\right) \\ A^s = softmax\left(\begin{bmatrix} P^s \oplus P^l \\ P^s \oplus P^c \\ P^s \oplus P^f \end{bmatrix}\right) \oplus softmax\left(\begin{bmatrix} P^s \otimes P^l \\ P^s \otimes P^c \\ P^s \otimes P^f \end{bmatrix}\right) \\ A^f = softmax\left(\begin{bmatrix} P^f \oplus P^l \\ P^f \oplus P^s \\ P^f \oplus P^c \end{bmatrix}\right) \oplus softmax\left(\begin{bmatrix} P^f \otimes P^l \\ P^f \otimes P^s \\ P^f \otimes P^c \end{bmatrix}\right) \end{cases}. \tag{8}$$

Here, $\oplus$ and $\otimes$ are the element-wise addition and element-wise multiplication. Notice we only compute the conditional

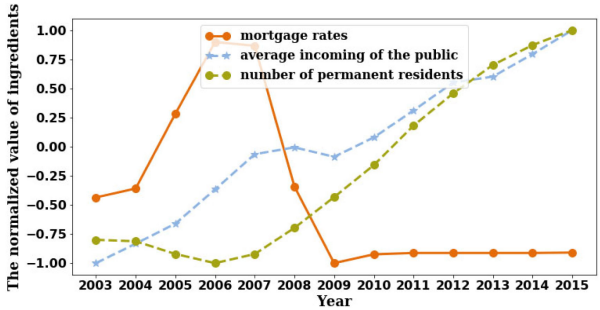


Fig. 5. Historical influence ingredients in NYC.

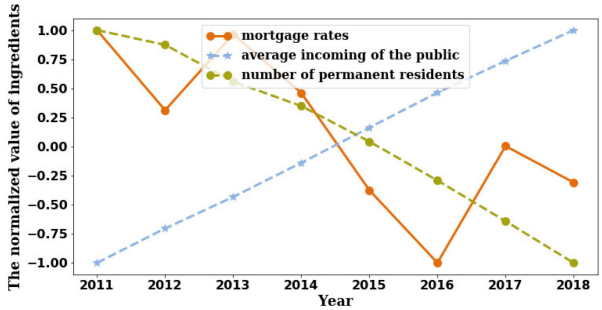


Fig. 6. Historical influence ingredients in Beijing.

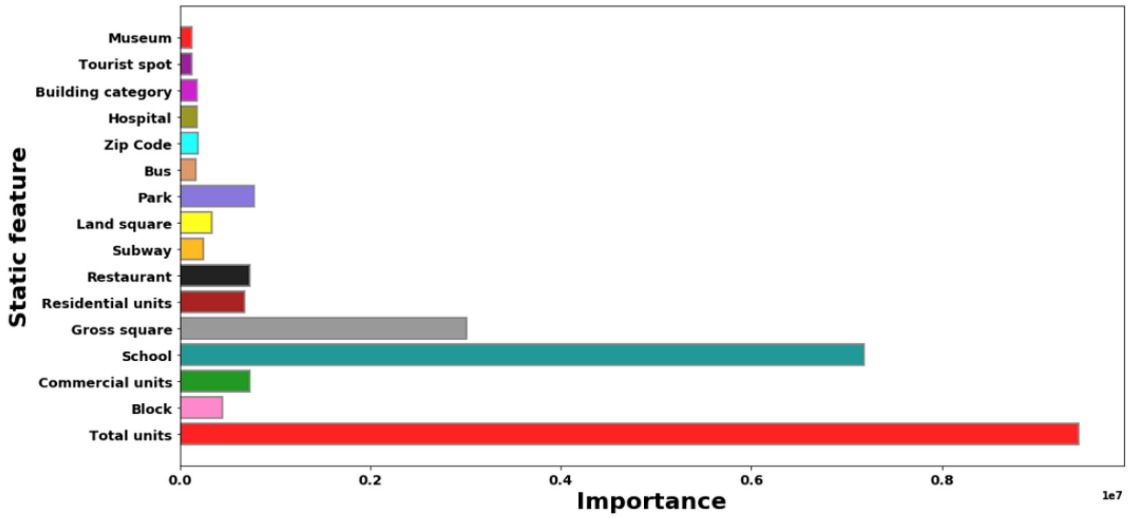


Fig. 7. The validity evaluation of static features in NYC.

TABLE 2
Heterogeneous Data of Four Components

Modality	Property and unit	Structure
P^l, P^s	Property: housing price; Unit: \$;	Hierarchical convolutional characteristics;
P^f	Property: housing price; Unit: \$;	Non-linear simulation of public expectation;
P^c	Property: rate, count, price and encoding; Unit: square foot, \$;	Linear concatenation of $(k+5)$ separate factors;

and joint affinity matrices for P^l, P^s, and P^f, it is because P^c including complex static features is different from other outputs in the value and property of data and is not appropriate to be directly concatenated with them. Therefore, we only fuse P^l, P^s, and P^f by considering their correlations. Next, as defined in Equation (8), the conditional and joint affinity matrices are all normalized to produce the attention weights by the $softmax$ function, and finally, as illustrated in Fig. 8, combined into the final balanced weights.

Filtration Gate Submodule. Even if there exist significant correlations among multiple modalities, the final fusion result may still be dominated by some individual submodule [47]. Furthermore, the calculation of joint representations may bring noises if two modalities are too similar. Therefore, we then introduce the mechanism of Filtration Gate (FG) to dynamically adjust the weight of each modality. The definition of FG can be formulated by:

$$\begin{bmatrix} \beta_s^l \\ \beta_c^l \\ \beta_f^l \\ \beta_l^s \\ \beta_c^s \\ \beta_f^s \\ \beta_l^f \\ \beta_s^f \\ \beta_c^f \end{bmatrix} = \sigma \left(\begin{bmatrix} W_s^l (||h^l - h^s||_F)^{-1} \\ W_c^l (||h^l - h^c||_F)^{-1} \\ W_f^l (||h^l - h^f||_F)^{-1} \\ W_l^s (||h^s - h^l||_F)^{-1} \\ W_c^s (||h^s - h^c||_F)^{-1} \\ W_f^s (||h^s - h^f||_F)^{-1} \\ W_l^f (||h^f - h^l||_F)^{-1} \\ W_s^f (||h^f - h^s||_F)^{-1} \\ W_c^f (||h^f - h^c||_F)^{-1} \end{bmatrix} \right) \tag{9}$$

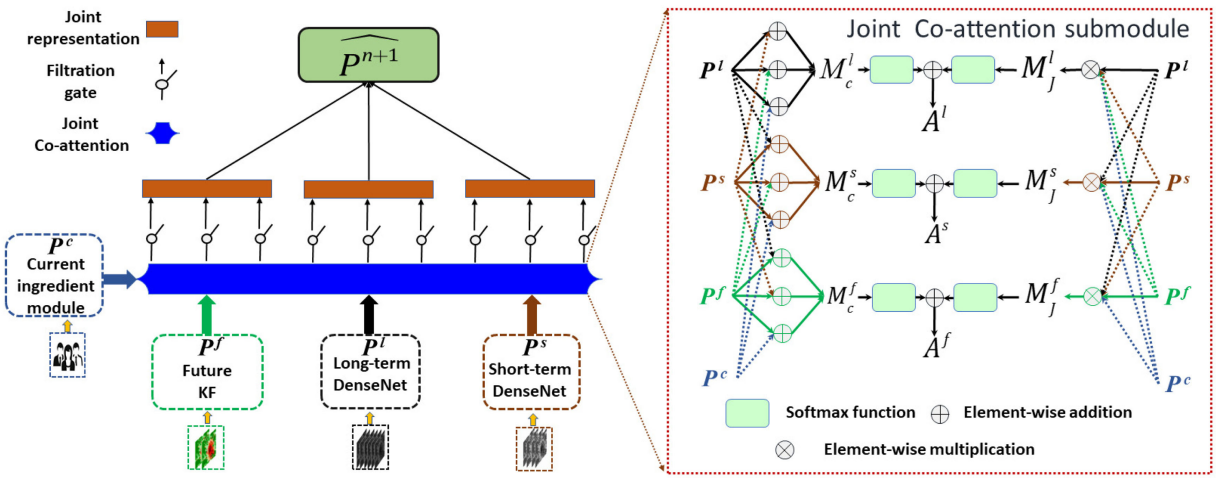


Fig. 8. Architecture of joint gated co-attention based fusion and detailed procedure of joint co-attention submodule.

where

$$\begin{bmatrix} h^l \\ h^s \\ h^c \\ h^f \end{bmatrix} = \mathrm{T}anh\left(\begin{bmatrix} W^l P^l \\ W^s P^s \\ W^c P^c \\ W^f P^f \end{bmatrix} + \begin{bmatrix} b^l \\ b^s \\ b^c \\ b^f \end{bmatrix}\right). \tag{10}$$

Here the series of β refer to the filtration gates, and the value of an individual β depends on the similarity between the two corresponding modalities. The series of W and b are the learnable parameters in the FG, and σ is the logistic sigmoid activation function. Besides, $||_F$ refers to the Frobenius Norm [48] which is widely adopted to measure the distance of matrices, and it is helpful to simplify the computational process and reduce the number of parameters compared to the design of traditional filtration gate [49].

Joint Representation Submodule. Based on the balanced weights obtained by the joint co-attention submodule, we first compute the temporary joint representations C_t^l, C_t^s, and C_t^f for different components by:

$$\begin{cases} C_t^l = [\beta_s^l P^s, \beta_c^l P^c, \beta_f^l P^f] \otimes A^l \\ C_t^s = [\beta_l^s P^l, \beta_c^s P^c, \beta_f^s P^f] \otimes A^s \\ C_t^f = [\beta_l^f P^l, \beta_s^f P^s, \beta_c^f P^c] \otimes A^f \end{cases}. \tag{11}$$

Notice that, to address the noises caused by the similarities between paired modalities, we filter the output of the previous four components by multiplying them with the corresponding filtration gate respectively. Next, to capture further correlations between modalities, we consider the joint representations for components in Equation (11) by:

$$\begin{cases} C^l = C_t^l \otimes softmax(P^l \otimes P^a) \\ C^s = C_t^s \otimes softmax(P^s \otimes P^a) \\ C^f = C_t^f \otimes softmax(P^f \otimes P^a). \end{cases}, P^a = P^l \oplus P^s \oplus P^f \tag{12}$$

By fusing these three formal joint representations, we then generate an intermediate joint representation C.

$$C = C^l \oplus C^s \oplus C^f. \tag{13}$$

Finally, to consider the all-stage joint representations, we calculate the final output of our JGC based fusion by,

$$\widehat{P^{n+1}} = \begin{pmatrix} W^l \otimes (P^l \oplus C_t^l) + W^s \otimes (P^s \oplus C_t^s) \\ +W^f \otimes (P^f \oplus C_t^f) + W^c \otimes C \end{pmatrix}. \tag{14}$$

Here W^l, W^s, W^f, and W^c are the learnable parameters for adjusting the influence weights of various components respectively.

5 Experiments

5.1 Setup and Data Analysis

In this subsection, we introduce the datasets of NYC and Beijing, as well as some settings of experiments.

NYC and Beijing Datasets. The house transaction price dataset of NYC is provided on the public platform of NYC Open Data.[6] The current ingredients can be taken from the Federal Reserve Economic Data.[7] The NYC house transaction dataset, which starts from Jan. 2003 to Feb. 2015, has a time span of 13 years.

The house transaction dataset of Beijing is taken from the Lianjia dataset[8] in Kaggle public datasets and the Zhugezhaofang.[9] In addition, the current ingredients are provided on the website of the State Statistics Bureau.[10] The house transaction dataset of Beijing, which starts from Jan. 2011 to June. 2018, has the time span of 7 years.

TABLE 3
Datasets Description

DataSets	NYC house	Beijing house
Time Span	1/2003-12/2015	1/2011-12/2017
Time Span of Training Set	1/2003-8/2014	1/2011-3/2017
Time Span of Testing Set	9/2014-12/2015	4/2017-12/2017
Time Interval Size	one month	one month
Range of House Prices	95053-81262300 USD	500000-19980000 CNY
Number of Subregions	(12*12)	(30*30)
AVT of a subregion	15+	10+
Number of Time Intervals	156	84
Number of Ingredients	156*5	84*5

Data Sparsity. For the datasets of NYC and Beijing, the average number of transaction records (AVT) per subregion is no more than 30, as shown in Table 3. The work of VAR model applies for datasets with AVT greater than 100. For works of SVR and ANN models, the AVT value is above 50. Hence, our housing price data is sparse.

Settings. The model we proposed is implemented based on Keras.[11] In our JGC_MMN, we use Min-Max normalization to scale the input data into the range [-1,1] before feeding them into the network. There are two kind of convolutions ($Conv1$ and $Conv2$) in our model with the filter sizes of 3×3 and 1×1 respectively. The filter number of $Conv1$ appeared in each DenseBlock is named growth rate, which has been set to a fixed value in our model. The filter number of the last $Conv2$ after DenseBlock N is 1, and the filter number of other $Conv2$ is set to different values (e.g., 32, 64) according to the experimental results. We use 90 percent of the original data for training and 10 percent for validation. For every model, we adopt the same learning rate and epochs. The parameter of KF-based submodule follows the setting of [37].

Housing Price Data Analysis. For both NYC and Beijing, we analyze the long-term and short-term housing price trends in different regions. We first randomly select 3 regions from two cities, and illustrate the long-term housing prices of selected regions in Figs. 9 and 10. As observed, for long-term housing prices, there exist significant differences between the long-term housing prices of different regions. For instance, from 2012 to 2013, the housing price of region 1 in NYC increases rapidly while the prices of the other two regions decrease moderately. The same scene happens in Beijing from 2017 to 2018. For short-term housing prices, from the selected regions in the year 2015 in Figs. 11 and 12,

6. https://opendata.cityofnewyork.us

7. https://fred.stlouisfed.org
8. https://www.kaggle.com/ruiqurm/lianjia
9. http://su.zhuge.com
10. http://www.stats.gov.cn/
11. https://keras.io/

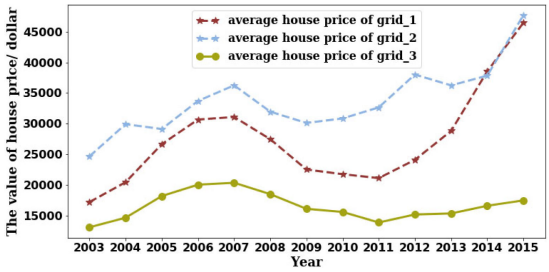


Fig. 9. Long-term prices of different regions in NYC.

we find that the prices of different regions in NYC vary independently. These phenomena confirm the necessity of carrying out the research, and also reflect the challenge of the problem.

5.2 Baselines

The baseline solutions are as follows:

GTWR. Geographical and temporal weighted regression is an extension of geographically weighted regression, to account for spatiotemporal local effects, and we utilize additional software to model it.

SAR. Spatial auto-regression model is a typical category of spatial econometric models and is similar to the spatial lag model. We also use the same features as our models to calculate its result.

Lasso. Lasso regression is a traditional regression method wildly used in economic areas. We feed it with the same feature as our models in each region.

Ridge. Like the lasso regression, ridge regression is another classic regression method in economic issues. We feed this model as what we do in Lasso regression.

SVR. We feed the same features as our models including historical transaction records, current ingredients and future expectations into SVR for training. Further, to involve the spatial correlations, we also feed the SVR with the housing prices of 8 neighboring regions.

VAR. Vector AutoRegressive is a widely used method in economic issues including forecasting housing price in city-level. For each subregion, we feed it with the same features as our models.

ST-ANN. The ST-ANN is fed with the spatial (8 neighboring regions' prices) and temporal (12 previous months' prices) features of each subregion. Besides, the same current ingredients and future expection are also considered.

Deep_ST. The DNN-based model has been widely used for spatial-temporal prediction issues. We adopt it by the method suggested in [34]. And other factors are also considered in the process of training.

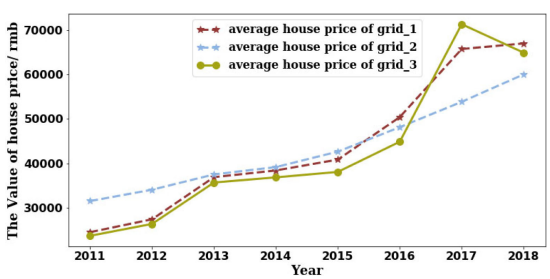


Fig. 10. Long-term prices of different regions in Beijing.

Fig. 11. Short-term prices of different regions in NYC in 2015.

ST-InceptionV4. InceptionV4 [36] has the same excellent performance as other deep networks on abundant datasets in image classification. We adjust the layers of network structure and feed it with the same features, to select optimal results to make comparisons.

ST-ResNet. ST-ResNet is first proposed for spatiotemporal crowd flows predictions [35]. Similar to InceptionV4, the popular model is compared with ours in the same condition.

P-D6-L9.* P-D6-L9* are a set of ablative variants based on the model of the previous version.

Further, to investigate the effectiveness of our joint gated co-attention based fusion, we use two additional baselines.

Conditional Co-Attention. We here simplify our JGC_fusion by only employing the linear combination to exploit partially relationships among multiple modalities, and name this variant as conditional co-attention.

Model-Based Fusion. In model-based fusion, we just fuse all the outputs of the long-term and short-term DesNets, current ingredient module and future KF by multiplying them with a series of learnable weights.

5.3 Evaluation

5.3.1 Comparison With Baseline Solutions in Both NYC and Beijing

We show the comparison results on NYC and Beijing datasets with baseline solutions in Tables 4 and 5, respectively. Also, we consider 3 other variants of JGC_MMN, by varying the number of layers and dense blocks and the inclusion of different features (i.e., current ingredients, or future expectations). We use C and F to represent the inclusion of current ingredients and future expectations, respectively. We use D and L to represent the number of dense blocks and the number of layers of a dense block, respectively. For example, D6-L9-F refers to a variant of JGC_MMN, which has 6 dense

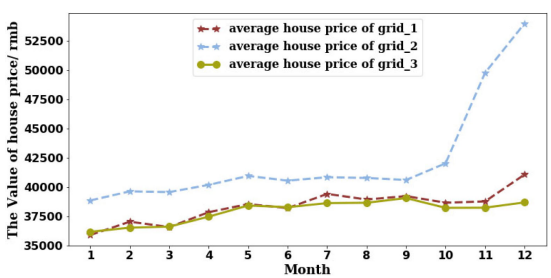


Fig. 12. Short-term prices of different regions in Beijing in 2015.

TABLE 4
Comparison With Different Baselines in NYC, and Different Variants of Our Models

Model		RMSE	MAPE
GTWR		27.29	18.98
SAR		28.16	20.35
Lasso		28.34	20.76
Ridge		29.13	21.43
VAR		29.27	13.69
SVR		28.65	13.76
ST-ANN		28.08	20.68
Deep-ST+C+F		27.81	12.69
ST-InceptionV4+C+F		27.03	11.11
ST-ResNet+C+F		26.04	11.74
Previous	**FTD_DenseNet**		
P-D6-L9	Long-term+short-term DenseNet	25.45	14.51
P-D6-L9-F	Long-term+short-term DenseNet+F	23.47	10.46
P-D6-L9-C	Long-term+short-term DenseNet+C	24.26	12.03
P-D6-L9-C-F-no Fusion	Long-term+short-term DenseNet+C+F+without Fusion	24.50	10.63
P-D6-L9-C-F (short-term)	Short-term DenseNet+C+F	26.31	10.97
P-D6-L9-C-F (long-term)	Long-term DenseNet+C+F	25.67	10.69
P-D6-L9-C-F	Long-term+short-term DenseNet+C+F	22.81	9.98
Ours	**JGC_MMN**		
D6-L9	Long-term+short-term DenseNet	24.32	14.35
D6-L9-F	Long-term+short-term DenseNet+F	23.18	9.76
D6-L9-C	Long-term+short-term DenseNet+C	23.93	10.65
D6-L9-C-F (short-term)	Short-term DenseNet+C+F	24.58	10.23
D6-L9-C-F (long-term)	Long-term DenseNet+C+F	24.77	10.41
D6-L9-C-F	Long-term+short-term DenseNet+C+F	**21.43**	**9.16**

TABLE 5
Comparison With Different Baselines in Beijing, and Different Variants of Our Models

Model		RMSE	MAPE
GTWR		74.10	16.63
SAR		75.05	17.66
Lasso		75.48	18.24
Ridge		77.12	19.25
VAR		83.09	**8.25**
SVR		76.23	10.07
ST-ANN		75.18	17.55
Deep-ST+C+F		74.62	10.78
ST-InceptionV4+C+F		73.79	10.48
ST-ResNet+C+F		73.11	10.63
Previous	**FTD_DenseNet**		
P-D6-L9	Long-term+short-term DenseNet	69.01	12.48
P-D6-L9-F	Long-term+short-term DenseNet+F	65.72	10.34
P-D6-L9-C	Long-term+short-term DenseNet+C	67.69	10.31
P-D6-L9-C-F-no Fusion	Long-term+short-term DenseNet+C+F+without Fusion	68.34	10.26
P-D6-L9-C-F (short-term)	Short-term DenseNet+C+F	70.93	9.93
P-D6-L9-C-F (long-term)	Long-term DenseNet+C+F	72.45	10.49
P-D6-L9-C-F	Long-term+short-term DenseNet+C+F	64.83	9.42
Ours	**JGC_MMN**		
D6-L9	Long-term+short-term DenseNet	67.12	12.11
D6-L9-F	Long-term+short-term DenseNet+F	63.76	9.82
D6-L9-C	Long-term+short-term DenseNet+C	62.38	9.68
D6-L9-C-F (short-term)	Short-term DenseNet+C+F	66.85	9.35
D6-L9-C-F (long-term)	Long-term DenseNet+C+F	68.14	10.04
D6-L9-C-F	Long-term+short-term DenseNet+C+F	**60.19**	9.04

blocks of 9 layers and is associated with future expectations as features. It can be observed that our method can outperform most alternative solutions in terms of RMSE and MAPE. Particularly, in Tables 4 and 5, we find that D6-L9-C-F gets the best forecasting accuracy, which has a 23.12 percent lower RMSE value and a 38.55 percent lower MAPE value than the SVR method averagely.

5.3.2 *Impacts of Components and Features*

From Tables 4 and 5, we then analyze the impacts of the proposed components. Compared to the model-based fusion, we here have considered the relationships among submodules. First, as can be discovered, the long-term and short-term DenseNets decrease the RMSE by 11.36 and 12.57 percent and the MAPE by 7.54 and 12.36 percent respectively and independently. Also, the results show the importance of incorporation of different features. For example, in Table 4, D6-L9-C-F has a lower RMSE value than D6-L9-C, which demonstrates the necessity of considering future expectations. Similar results can be observed for the effect of current ingredients. The current ingredients and KF components decrease the mean RMSE by 6.57 and 7.0 percent respectively. For D6-L9, the RMSE values and MAPE values of NYC and Beijing are 24.32 and 67.12 respectively, and it still outperforms other baselines.

5.3.3 *Impacts of JGC Based Fusion*

In this subsection, we investigate the effectiveness of our JGC based fusion by comparing it with conditional co-attention and model-based fusion based on the network structures of D6-L9-C-F, D5-L9-C-F, and D5-L9-C-F in Table 6. In both NYC and Beijing, our approach with JGC based fusion can outperform the other two alternative approaches with different network structures, and this verifies the effectiveness of our JGC based fusion in terms of the accuracy of forecasting. Specifically, with the network structure of D6-L9-C-F, our JGC based fusion can reduce the RMSE by 5.39 percent averagely compared to the model-based fusion. Further, the conditional co-attention can always outperform the model-based fusion with all three different network structures. This indicates that the co-attention mechanism works for the fusion optimization, even a small part of correlations among different temporal components are obtained. It also verifies the rationality of the idea of enhancing the fusion methods by fully and deeply capturing the correlations between different components.

5.3.4 *Impacts of Parameters*

Furthermore, we test the effect of other parameters, such as the number of dense blocks and its layers. The result is shown in Fig. 14. X-axis refers to the total number of layers in the network, including the head and tail convolutional

TABLE 6
Comparison With Different Fusion Methods in NYC and Beijing

Fusion methods	NYC/RMSE	Beijing/RMSE
Joint gated co-attention(D6-L9-C-F)	**21.43**	**60.19**
Conditional co-attention(D6-L9-C-F)	22.38	63.22
Model based Fusion(D6-L9-C-F)	22.53	63.97
Joint gated co-attention(D5-L9-C-F)	**21.71**	**61.85**
Conditional co-attention(D5-L9-C-F)	22.51	64.89
Model based Fusion(D5-L9-C-F)	22.93	65.88
Joint gated co-attention(D4-L9-C-F)	**22.54**	**62.03**
Conditional co-attention(D4-L9-C-F)	23.79	65.08
Model based Fusion(D4-L9-C-F)	24.02	66.85

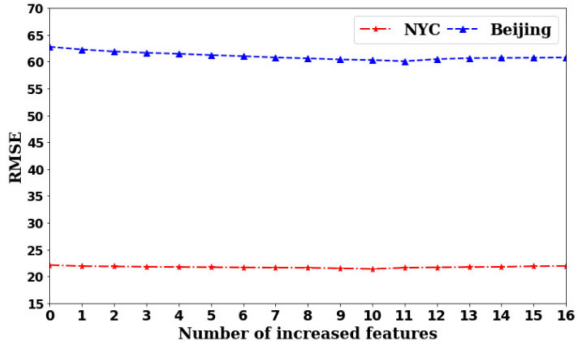


Fig. 13. RMSE of different number of static features.

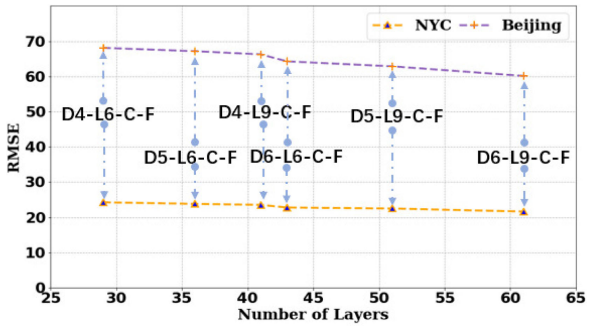


Fig. 14. RMSE of different structures in our models.

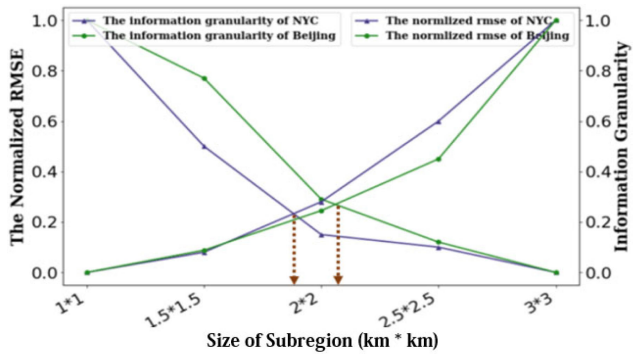


Fig. 15. RMSE and information granularity of different subregions.

layers (i.e., $Conv1$ and $Conv2$) in the JGC_MMN, and the transition layers between dense blocks. We can see with a larger number of dense blocks and layers, our model can learn more all-level features and thus better capture the spatiotemporal dependencies. The performance converges when the number of layers is greater than 61. But the computation overheads increase sharply with a large number of layers. In our work, we find the D6-L9 setting best captures the tradeoff between accuracy and efficiency and hence is used as our default setting.

To evaluate the impact of k in the static feature selection, we first sort the 16 static features by their validities in the descending order. The result is shown in Fig. 13, where X-axis refers to the number of increased static features. It can be observed that our model gets the best RMSE results when adding top 10 and 11 features for NYC and Beijing. If more (≥ 10) features are incorporated, overfitting may occur due to the information redundancy. Hence we select top 10 and 11 features, for NYC and Beijing, which are combined with the other five ingredients to form the final ingredients.

The effect of subregion size is studied in Fig. 15. We find the balance point between the information granularity[12] and the best normalized RMSE is achieved when subregion size equals $2km * 2km$, which is thus selected as the default subregion size for our model.

6 Conclusion

In this paper, we propose a fine-grained forecasting model, JGC_MMN, for subregion spatiotemporal housing price prediction. In particular, we modify the structure of DenseNet and adopt the method of bagging by fusing the KF-based method to improve the accuracy. For better fusion, we design a novel method to fuse the heterogeneous data of multi-stage models by fully and deeply capturing the correlations between them. Experiments on two different real-world datasets have demonstrated that our proposed model outperforms state-of-the-art solutions. In the future, we will apply our model which includes an all-time period(i.e., distant , recent, current, and future time) and fuse their correlations to other similar domains, such as air quality prediction and power demand prediction.

12. We define it with the normalized number of average transaction records per square kilometer for measuring the data sparsity.

Acknowledgments

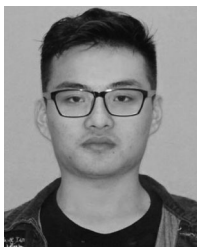
This work was supported in part by the Anhui Science Foundation for Distinguished Young Scholars under Grant 1908085J24, in part by NSFC under Grants 62072427, 61672487, and 61772492, in part by the Jiangsu Natural Science Foundation under Grant BK20191193, and in part by Zhejiang Lab's International Talent Fund for Young Professionals.

References

[1] H. Zhu *et al.*, "Days on market: Measuring liquidity in real estate markets," in *Proc. 22nd ACM SIGKDD Int. Conf. Knowl. Discov. Data Mining*, 2016, pp. 393–402.
[2] CRIC, "The real estate price map of Xi'an," 2018. [Online]. Available: http://www.yidianzixun.com/article/0KIs62Eb.
[3] R. K. Pace, R. Barry, J. M. Clapp, and M. Rodriquez, "Spatiotemporal autoregressive models of neighborhood effects," *J. Real Estate Finance Econ.*, vol. 17, no. 1, pp. 15–33, 1998.
[4] T. H. Kuethe and V. O. Pede, "Regional housing price cycles: A spatio-temporal analysis using us state-level data," *Regional Stud.*, vol. 45, no. 5, pp. 563–574, 2011.
[5] C. P. Barros, L. A. Gil-Alana, and J. E. Payne, "Comovements among US state housing prices: Evidence from fractional cointegration," *Econ. Model.*, vol. 29, no. 3, pp. 936–942, 2012.
[6] X. Wang, J. Wen, Y. Zhang, and Y. Wang, "Real estate price forecasting based on SVM optimized by PSO," *Optik-Int. J. Light Electron Opt.*, vol. 125, no. 3, pp. 1439–1443, 2014.
[7] A. S. Fotheringham, R. Crespo, and J. Yao, "Exploring, modeling and predicting spatiotemporal variations in house prices," *Ann. Regional Sci.*, vol. 54, no. 2, pp. 417–436, 2015.
[8] G.-Z. Fan, S. E. Ong, and H. C. Koh, "Determinants of house price: A decision tree approach," *Urban Stud.*, vol. 43, no. 12, pp. 2301–2315, 2006.
[9] T. Kauko, P. Hooimeijer, and J. Hakfoort, "Capturing housing market segmentation: An alternative approach based on neural network modeling," *Housing Stud.*, vol. 17, no. 6, pp. 875–894, 2002.
[10] V. Limsombunchai, "House price prediction: Hedonic price model vs. artificial neural network," in *Proc. New Zealand Agricultural Resour. Econ. Soc. Conf.*, 2004, pp. 25–26.
[11] X. Chen, L. Wei, and J. Xu, "House price prediction using LSTM," 2017, *arXiv:1709.08432*.
[12] J. Wang *et al.*, "Predicting house price with a memristor-based artificial neural network," *IEEE Access*, vol. 6, pp. 16523–16528, 2018.
[13] F. Wang, Y. Zou, H. Zhang, and H. Shi, "House price prediction approach based on deep learning and arima model," in *Proc. IEEE 7th Int. Conf. Comput. Sci. Netw. Technol.*, 2019, pp. 303–307.
[14] R. Meese and N. Wallace, "House price dynamics and market fundamentals: The parisian housing market," *Urban Stud.*, vol. 40, no. 5–6, pp. 1027–1045, 2003.
[15] O. Bin, "A prediction comparison of housing sales prices by parametric versus semi-parametric regressions," *J. Housing Econ.*, vol. 13, no. 1, pp. 68–84, 2004.
[16] W. T. Lim, L. Wang, Y. Wang, and Q. Chang, "Housing price prediction using neural networks," in *Proc. 12th Int. Conf. Nat. Comput., Fuzzy Syst. Knowl. Discov.*, 2016, pp. 518–522.
[17] E. L. Glaeser, J. Gyourko, E. Morales, and C. G. Nathanson, "Housing dynamics: An urban approach," *J. Urban Econ.*, vol. 81, pp. 45–56, 2014.

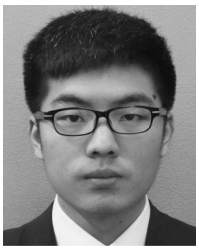
[18] Y. M. Goh, G. Costello, and G. Schwann, "Accuracy and robustness of house price index methods," *Housing Stud.*, vol. 27, no. 5, pp. 643–666, 2012.
[19] D. C. Wheeler and A. PÃąez, *Geographically Weighted Regression*. Berlin, Germany: Springer, 2010.
[20] A. S. Fotheringham, R. Crespo, and J. Yao, "Geographical and temporal weighted regression (GTWR)," *Geograph. Anal.*, vol. 47, pp. 431–452, 2015.
[21] B. Wu, R. Li, and B. Huang, "A geographically and temporally weighted autoregressive model with application to housing prices," *Int. J. Geograph. Inform. Sci.*, vol. 28, no. 5, pp. 1186–1204, 2014.
[22] J. Liu, Y. Yang, S. Xu, Y. Zhao, W. Yong, and F. Zhang, "A geographically temporal weighted regression approach with travel distance for house price estimation," *Entropy*, vol. 18, no. 8, 2016, Art. no. 303.
[23] J. Shim, C. Hwang, and F. Wen, "Kernel-based geographically and temporally weighted autoregressive model for house price estimation," *PLoS ONE*, vol. 13, no. 10, 2018, Art. no. e0205063.
[24] G. Peres-Neto , "Spatial modeling in ecology: The flexibility of eigenfunction spatial analyses," *Ecology*, vol. 87, no. 10, pp. 2603–2613, 2006.
[25] M. Helbich and D. A. Griffith, "Spatially varying coefficient models in real estate: Eigenvector spatial filtering and alternative approaches," *Comput. Environ. Urban Syst.*, vol. 57, pp. 1–11, 2016.
[26] Y. Kharin and M. Zhurak, "Statistical analysis of spatio-temporal data based on poisson conditional autoregressive model," *Informatica*, vol. 26, no. 1, pp. 67–87, 2015.
[27] G. White and S. K. Ghosh, "A stochastic neighborhood conditional autoregressive model for spatial data," *Comput. Stat. Data Anal.*, vol. 53, pp. 3033–3046, 2009.
[28] H. H. Kelejian and I. R. Prucha, "A generalized spatial two-stage least squares procedure for estimating a spatial autoregressive model with autoregressive disturbances," *J. Real Estate Finance Econ.*, vol. 17, no. 1, pp. 99–121, 1998.
[29] L. F. Lee, "Best spatial two-stage least squares estimators for a spatial autoregressive model with autoregressive disturbances," *Econometric Rev.*, vol. 22, no. 4, pp. 307–335, 2003.
[30] B. H. Baltagi, B. Fingleton, and A. Pirotte, "Spatial lag models with nested random effects: An instrumental variable procedure with an application to english house prices," *J. Urban Econ.*, vol. 80, no. 1, pp. 76–86, 2014.
[31] S. J. Xin and K. Khalid, "Modeling house price using ridge regression and lasso regression," *Int. J. Eng. Technol.*, vol. 7, no. 4.30, p. 498, 2018.
[32] G. Tutz and H. Binder, "Boosting ridge regression," *Comput. Stat. Data Anal.*, vol. 51, no. 12, pp. 6044–6059, 2007.
[33] L. Mariella and M. Tarantino, "Spatial temporal conditional autoregressive model: A new autoregressive matrix," *Austrian J. Stat.*, vol. 39, no. 3, 2010, pp. 223–244.
[34] J. Zhang, Y. Zheng, D. Qi, R. Li, and X. Yi, "DNN-based prediction model for spatio-temporal data," in *Proc. 24th ACM SIGSPATIAL Int. Conf. Adv. Geograph. Inf. Syst.*, 2016, pp. 1–4.
[35] J. Zhang, Y. Zheng, and D. Qi, "Deep spatio-temporal residual networks for citywide crowd flows prediction," in *Proc. AAAI Conf. Artif. Intell.*, 2017, pp. 1655–1661.
[36] C. Szegedy, S. Ioffe, V. Vanhoucke, and A. A. Alemi, "Inception-v4, inception-ResNet and the impact of residual connections on learning," in *Proc. 31st AAAI Con. Artif. Intell.*, 2017, pp. 4278–4284.
[37] E. L. Glaeser and C. G. Nathanson, "An extrapolative model of house price dynamics," *J. Financial Econ.*, vol. 126, no. 1, pp. 147–170, 2017.
[38] A. Caplin and J. Leahy, "Trading frictions and house price dynamics," *J. Money, Credit Banking*, vol. 43, pp. 283–303, 2011.
[39] G. Huang, Z. Liu, L. Van Der Maaten, and K. Q. Weinberger, "Densely connected convolutional networks," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2017, pp. 2261–2269.
[40] Y. LeCun, L. Bottou, Y. Bengio, and P. Haffner, "Gradient-based learning applied to document recognition," *Proc. IEEE*, vol. 86, no. 11, pp. 2278–2324, Nov. 1998.
[41] S. Ioffe and C. Szegedy, "Batch normalization: Accelerating deep network training by reducing internal covariate shift," 2015, *arXiv:1502.03167*.
[42] B. Azhagusundari and A. S. Thanamani, "Feature selection based on information gain," *Int. J. Innov. Technol. Exploring Eng.*, vol. 2, no. 2, pp. 18–21, 2013.
[43] J. Lu, J. Yang, D. Batra, and D. Parikh, "Hierarchical question-image co-attention for visual question answering," in *Proc. Adv. Neural Inf. Process. Syst.*, 2016, pp. 289–297.
[44] J. Gao *et al.*, "CAMP: Co-attention memory networks for diagnosis prediction in healthcare," in *Proc. IEEE Int. Conf. Data Mining*, 2019, pp. 1036–1041.
[45] Q. Zhang, J. Fu, X. Liu, and X. Huang, "Adaptive co-attention network for named entity recognition in tweets," in *Proc. Thirty-Second AAAI Conf. Artif. Intell.*, 2018, pp. 5674–5681.
[46] B. Hu, C. Shi, W. X. Zhao, and P. S. Yu, "Leveraging meta-path based context for top-N recommendation with a neural co-attention model," in *Proc. 24th ACM SIGKDD Int. Conf. Knowl. Discov. Data Mining*, 2018, pp. 1531–1540.
[47] H. Driessen and Y. Boers, "Map estimation in particle filter tracking," in *Proc. IET Seminar Target Tracking Data Fusion: Algorithms Appl.*, 2008, pp. 41–45.
[48] X. Peng, C. Lu, Z. Yi, and H. Tang, "Connections between nuclear-norm and frobenius-norm-based representations," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 29, no. 1, pp. 218–224, Jan. 2018.
[49] S. Chen and Q. Jin, "Multi-modal conditional attention fusion for dimensional emotion prediction," in *Proc. 24th ACM Int. Conf. Multimedia*, 2016, pp. 571–575.

Pengkun Wang received the bachelor's degree in automation from Jilin University in 2017. He is currently working toward the doctoral degree with the School of Data Science, University of Science and Technology of China. His research interests mainly include data mining, multimodal fusion, and computer vision.



Chuancai Ge received the bachelor's degree in software engineering from Xidian University in 2017. He is currently working toward the graduation degree with the School of Computer Science and Technology, University of Science and Technology of China. His research interests include deep learning, machine learing, and urban computing.

Zhengyang Zhou (Student Member, IEEE) is currently working toward the doctoral degree with the School of Computer Science and Technology, University of Science and Technology of China. His research interests include machine learning, spatiotemporal data mining, and artificial intelligence in traffic applications. He is a student member of AAAI .



Xu Wang received the bachelor's degree in automation from North Eastern University in 2017, He is currently working toward the doctoral degree with the School of Data Science, University of Science and Technology of China. His research interests mainly include data mining, machine learning, and computer vision.

Yuantao Li received the bachelor's degree in automation from Zhejiang University in 2020. He is currently working toward the graduation degree with the School of Software Engineering, University of Science and Technology of China. His research interests mainly include data mining and machine learning.

Yang Wang (Senior Member, IEEE) received the PhD degree from the University of Science and Technology of China (USTC) in 2007, under supervision of professor Liusheng Huang. He is currently an associate professor with USTC. His research interests mainly include wireless networks, distributed systems, data mining, and machine learning.