Multi-view Commercial Hotness Prediction Using Context-aware Neural Network Ensemble

ZHIYUAN HE, Shanghai Key Laboratory of Intelligent Information Processing, School of Computer Science, Fudan University, China
SU YANG*, Shanghai Key Laboratory of Intelligent Information Processing, School of Computer Science, Fudan University, China

Prediction over heterogeneous data attracts much attention in urban computing. Recently, satellite imagery provides a new chance for urban perception but raises the problem of how to fuse visual and non-visual features. So far, the practice is to concatenate the multimodal features into a vector, which may suppress important features. Therefore, we propose a new ensemble learning framework: (1) An estimator is developed for each predictor to score its confidence, which is input adaptive. (2) By applying the output of each predictor to the input of the corresponding estimator as feedback, the estimator learns the performance of the predictor in the input-output space. When a new input is applied to produce a prediction, the similar situations will be recalled by the estimator to score the confidence of the prediction. (3) Using end-to-end training, the estimator learns the weights automatically to minimize the total loss of the neural networks. With the proposed method, data mining based urban computing and computer vision rendered urban perception can be bridged at the task of commercial activeness prediction, where the prediction based on satellite images and social context data are fused to yield better prediction than those based on single view data in the experiments.

CCS Concepts: • **Applied computing** → *Sociology*;

Additional Key Words and Phrases: Urban perception, Ubiquitous Computing, Ensemble Learning

ACM Reference Format:
Zhiyuan He and Su Yang. 2018. Multi-view Commercial Hotness Prediction Using Context-aware Neural Network Ensemble. *Proc. ACM Interact. Mob. Wearable Ubiquitous Technol.* 2, 4, Article 168 (December 2018), 19 pages. https://doi.org/10.1145/3287046

1 INTRODUCTION

Prediction over heterogeneous data has been becoming increasingly important in the context of urban computing since urban big data could be acquired from multiple sources of multimodality. In [6], the heterogeneous data including POIs, population, and human mobility encoded in Foursquare check-in logs are combined to predict regional demands of bikes. The work is further extend to predict spatio-temporal over-demands based on dynamic clustering of bike stations and incorporating opportunistic contextual factors like social and traffic events in addition to the common contextual features such as date, time, weather, and temperature [7]. In [13], user opinions

*Corresponding author. The author is also with School of Computer Science, Xi'an University of Technology.

Authors' addresses: Zhiyuan He, Shanghai Key Laboratory of Intelligent Information Processing, School of Computer Science, Fudan University, Shanghai, China, 16210240032@fudan.eu.cn; Su Yang, Shanghai Key Laboratory of Intelligent Information Processing, School of Computer Science, Fudan University, Shanghai, China, suyang@fudan.eu.cn.

2474-9567/2018/12-ART168 $15.00
https://doi.org/10.1145/3287046

and check-in positions in social media, taxi traces, and bus distribution and running information are incorporated into a statistical model to predict house prices. In [26], spatio-temporal POI demand prediction is solved using Latent Factor Model over the heterogeneous data of taxi trips, POIs, Foursquare check-ins, and Demographic data. In [5], nonnegative tensor factorization is performed on the tensors representing human mobility in terms of both bike trips and Foursquare check-ins for anomaly detection to aware unusual events. In the recent urban computing studies, economic and commercial issues have been attracting increasingly more attention. In [33], mobile communication records, short messages, and Bluetooth scan-enabled social networking are analyzed to form a couple of statistical variables so as to predict the categories of users in terms of money spending. In [2], communication patterns from mobile phones such as social networking, human mobility, and communication frequency as well as targets, along with the population distribution are used to predict richness. In [36], human mobility, transportation, web reviews, locations of restaurants, and road closing noti□cations are collected from FourSquare, Twitter, OpenTable, and Yelp so as to predict the reservation rate of restaurants using a predictive model developed in the context of economics. In [27], the rise and decay of the POIs in association with small business are predicted in a statistical framework referred to as Conditional Random Fields by taking into account the historical performance of such POIs as well as the contextual features extracted from the perspective of socioeconomics, human mobility, demographics, energy consumption, environment-health, and safety.

The aforementioned researches involve prediction over heterogeneous data. So far, however, the most widely adopted scheme is information fusion at the feature level, that is, concatenate the features of multimodality directly to form a single vector, and then, normalization or feature selection is applied [2, 6, 33]. However, due to the di□erent dynamic scales and semantics, feature concatenation may favor some features with big dynamic range while suppress the features with relatively small but important di□erence. In contrast, ensemble at the decision level could be more promising, that is, a predictor is established for the homogeneous data of each modality and the prediction results are fused through a weighted voting strategy to yield the □nal prediction. Although whether model-level fusion or feature-level fusion should be applied is subject to applications due to the variety of the existing scenarios, for the scenario of this study, say, commercial activeness prediction over multimodal data, the experimental demonstration shows that ensemble learning is more rational. It has been demonstrated that the social context data can be □tted into the commercial activeness using a linear predictor [35] while the correlation from satellite images to commercial activeness is much more complex, where the nonlinearity can be captured using Convolutional Neural Network (CNN) followed by Gradient Boosting Decision Tree (GBDT) [15]. Due to the incompatible natures between the visual and non-visual features in terms of □tting into commercial activeness, one identical predictor applied to both features as a whole is demonstrated leading to obviously lower precision in comparison with the ensemble learning scheme in the experiments. Besides, model-level fusion gains advantage over feature-level fusion in that it has a broader spectrum of applications in case heterogeneous representations of data do not allow feature-level fusion. Although there is a strong trend to call for ensemble learning in terms of urban computing, the topic has been rarely visited in the literature.

The state-of-the-art works on prediction over heterogeneous data are mostly focused on spatiotemporal data. Recently, satellite imagery brings in a new chance for urban perception as city infrastructures are visually straightforward in satellite images. For example, land use can be perceived from satellite images using deep learning [1]. Moreover, the night lights reflected in satellite images can be used to predict regional properties based on deep neural networks and transfer learning [20]. In some urban computing tasks such as commercial hotness prediction, in fact, both satellite images and social context data can be applied. In [35], human mobility reflected in taxi trajectories, local population and house prices, POIs, and rating scores of customers are fused to predict regional commercial activeness. In [15], visual features of satellite images extracted by using deep neural networks are used to predict land use categories in the sense of statistics so as to infer regional commercial hotness. Due to the gap between data mining based urban computing and computer vision rendered urban perception, yet, commercial activeness prediction based on both satellite images and social context data has remained a

missing topic so far. Furthermore, due to the lack of knowledge regarding the impact of city infrastructures as well as social contexts on commercial activeness, city planning from the economic point of view has been an open problem for a long time. Moreover, it is crucial for business owners to get recommendations on choosing suitable locations for their business so as to maximize the pro□ts, for which an operational way is to establish a predictor revealing the relation between urban big data and commercial activeness. In the past, the studies on commercial activeness prediction are based on either social context data [35] or satellite images [15], either of which □gures out the urban commercial pro□le from a single view only. The goal of this study is to discover the relation between urban big data and commercial activeness in terms of fusing social context data and visual patterns of satellite images. Once a predictor from social contexts and satellite images to commercial hotness is established by mining the correlation between them, city planning as well as business planning can be conducted in a rational way on the basis of referring to the map of commercial hotness over regions, which is a byproduct turned out from the predictor. Such a map leads to an integral pro□le in terms of commercial activeness over city infrastructures and social contexts throughout the city, which visualizes the city from the economic perspective for city planners. Moreover, for newly developed regions with short history and few data, the predicted commercial hotness over city maps becomes an important clue to foresee the future in terms of business planning.

This study aims to bridge the gap between urban computing and urban perception at the task of commercial hotness prediction based on multimodal data by introducing a new framework of ensemble learning. Here, we develop a new model for neural network ensemble. The mechanism is di□erent from the previous ones in that the weighting of each predictor is subject to the input on the fly and furthermore the output is also applied as feedback to distinguish the context of decision making by recalling the historical performance of each predictor under similar situations, where a bigger weight is generated for the predictor that has better performance in similar contexts. The computing architecture is speci□c in that a neural network based estimator is constructed accompanying each neural network based predictor, the mission of which is learning to score the con□dence of the predictor with regard to each input by recalling its performance on similar samples, where the output of the predictor is applied as feedback to the input of the estimator such that what the estimator learns is not the input only but the input-output pair. It has been experimentally demonstrated that the performance of ensemble learning can be obviously improved by introducing such weight estimator with the feedback mechanism in comparison with the non-feedback ones.

The contribution of this work is summarized as follows: (1) Multi-view commercial hotness prediction based on satellite images and social context data is studied. This bridges the gap between data mining based urban computing and computer vision rendered urban perception at the task of commercial hotness prediction. (2) A novel framework of neural network ensemble to deal with multimodal prediction tasks is established. Except for the two-channel neural network structure, namely, a predictor along with an estimator for weighting its con□dence, the structure of the proposed ensemble framework is unique in that both the input (data example) and the output (prediction result) of each predictor are applied to the corresponding estimator to weight the trust degree of each predictor, where the estimator learns to remember the reactions of the corresponding predictor on given data samples to produce the trustable degrees of such reactions. Moreover, the ensemble learning mechanism is implemented in an end-to-end manner, which learns the parameters of predictors and those of estimators simultaneously. The weight of each predictor is not □xed but subject to the input as well as the reaction of the predictor.

2 RELATED WORKS

For retail stores, the primary issue is location planning. In [21], geographical information is combined with human mobility reflected in FourSquare check-in data to infer the commercial hotness of di□erent categories of business. In [35], commercial activeness prediction from social context data including visitors, local population,

POIs, and user rating is considered by applying sparse representation to reveal the key influential factors as well as the linear relation to render prediction. In [15], commercial hotness perceiving is performed by using deep learning based visual feature extraction and gradient based decision tree for prediction. The goal of this study is to fuse the visual as well as social context information at the decision level to achieve better prediction, which has never been investigated before in the sense of multimodal data based urban computing. Although mobile communication records, population, and satellite images are combined to predict spatiotemporal poverty indices [30], only night lights reflected in satellite images are considered, which do not allow revealing the richer information regarding city infrastructures. In this sense, this paper is the first effort devoted to introducing visual pattern analysis for urban computing.

As for neural network ensemble, the key issue is how to weight the confidence of each predictor. The previous works do not allow dynamic weighting subject to input but this is promised by the proposed framework. To achieve neural network ensemble, expanding the diversity of neural networks is a common practice [23]. The most basic way is to generate different training data using the methods like bagging [3] and boosting [31]. Although boosting is proven effective in increasing the generalization ability of predictors, it degrades when dealing with hard examples [23]. In [11], Adaboosting is introduced to reallocate different weights for training samples, which will enhance the stability of neural network ensemble [32]. Besides bagging and boosting, the work in [22] changes the training data by using cross validation to obtain different proportion of the origin data. The work in [29] utilizes an artificial synthetic method. The work in [4] randomizes the target values to increase the prediction accuracy of neural network ensembles. Additionally, changing the structure of individual neural networks is also effective, such as the number of the hidden neurons [4], as well as the number of the neural networks [19]. For classification, majority vote is usually adopted as the method of combining different neural networks to form the conclusion. For regression, the simple average and weighted average [24, 25] methods are both effective.

In sum, the aforementioned works ensemble neural network predictors based on fixed weights. The reason is that such works do not employ an estimator as proposed in this study to evaluate the performance of each predictor on the fly. By applying the output of the predictor as feedback to the input of the estimator and train the ensemble model in an end-to-end manner as done in this study, the weight of each predictor is in reference to the input-output pair, which learns the contextual situations of varying cases and recalls similar situations to weight each predictor when a new input is fed to produce a prediction. By means of such a scheme, the predictor performing better in similar situations will possess higher weight, vice versa.

3 METHODOLOGY

In terms of urban computing, the data are usually collected from different sensors of different modalities. Correspondingly, they figure out different views of a region of interest. In the case of commercial hotness prediction, on one hand, the satellite imagery promises pervasive perception of city infrastructures [28, 34], to which commercial activities are more or less subject to [15]. On the other hand, taxi trajectories reflect the inflow and outflow into and out of a region of interest to some extent, which characterize somewhat the hotness of a region. In addition, the local population of a region is a notable factor to affect the commercial activeness. Moreover, the statistics resulting from the online comments available on web portals of social media, for example, the average price and rating score of a commercial entity, are obviously correlated to commercial activities. Due to the multimodality of the data, although a great deal of endeavor has been made, how to combine the heterogeneous data for urban computing applications including commercial hotness prediction has remained a missing topic so far. In this paper, we propose a novel ensemble learner in terms of neural networks to fuse the decisions from different views for commercial activeness prediction.

Specifically, we adopt two views of the data. We define view A as the data of satellite imagery. It describes cities in a visual manner and reflects the city infrastructures in a visible way. We define the non-visual data as view B, which is the social context data consisting of population, buying power of local residents, number of visitors, region functions, and rating scores of commercial entities in a given district. In the following, we first introduce the clustering method used to discover the urban commercial districts (UCDs) in a city. Then, we describe the features of view A and view B of a given district in detail. Finally, we present the context-aware neural network ensemble method used to predict commercial hotness, which leads to context-aware weighting in an automatic manner by learning the performance of the neural network predictor on different input-output (IO) spaces of either view. In addition, we introduce the baseline methods for comparison.

3.1 Locating UCDs in a City

Following [35], we use a simple clustering method to find all the UCDs in a city. Given all the commercial entities marked as commercial centers and streets, our goal is to aggregate them into UCDs. First, we initialize some seeds as the original clustering centers. Then, we repeat the following steps until all the clusters have converged:

- For each commercial entity, we assign it to the nearest clustering center if the distance between them does not exceed a predefined threshold. Otherwise, a new clustering center is created.
- If the commercial entities of a cluster have changed, recalculate the position of the center of that cluster.

The obtained UCDs are denoted as $U^{(1)}, U^{(2)}, ..., U^{(n)}$. The motivation behind this algorithm is simple. A UCD is usually a set of some POIs such as shopping centers and commercial streets. If two shopping centers is close to each other, it is likely that they belong to the same UCD. So, we merge the close POIs to be a unified UCD until all UCDs have a relatively far inter distance.

3.2 Constructing View A: Perceiving Land Use in Satellite Images

Satellite imagery can be used to trace urban land use. However, traditional computer vision methods do not work well when applied to satellite images as the images obtained from satellite sensors are in a highly unstructured form. Following [15], we construct view A by applying a deep learning-based model to recognize land use in raw satellite images. The detailed steps are as follows:

- First, we gather satellite images from Google Map. All the images are in the same resolution with the shape of square.
- Second, we use OpenStreetMap to label each image tile. Nodes and ways are two fundamental elements in OpenStreetMap data. A single node is defined as a geographical point consisting of a latitude, longitude and a node id. A way is an ordered list of nodes, which is usually labeled with a corresponding tag. A way can be open or closed. A closed way is a way where the first node and last node are shared to represent a certain area. In this paper, we utilize closed ways to generate labels for satellite images. We calculate the proportion of the area in each satellite tile. If it exceeds a threshold, then, we assign the tag of the way to this image as its label. In total, we make use of 6 classes: Farmlands, water areas, woods, business areas, residential areas, and industrial areas.
- Third, we train Google Inception V3, which is a deep learning-based model usually used to provide a strong baseline, to classify labeled images. The input of the model is a single satellite tile, and the output is the class-belonging probabilities over the 6 categories of land use.
- Finally, we construct view A by generating features for a given district using the trained deep model. To guarantee generality, the images used for training and generating features are in different cities. The feature of view A of a district is the average of the predicted class probabilities of all the satellite images in this district.

ALGORITHM 1: The Clustering Method

Input:
$X_k = (x_k, y_k), k = 1, 2, ..., N$; // The coordinate values (x_k, y_k) of seed X_k
d; // A distance threshold.
Output: C ; // A set containing a couple of clusters
for $k = 1$ *to* N **do**
  $C_k = \{X_k\}$ and $U_k = X_k$; // Initialize a cluster and the center per seed
end
$C = \{C_1, C_2, ..., C_N\}$;
while *true* **do**
  for $i = 1$ *to* $(|C| - 1)$ **do**
    for $j = i + 1$ *to* $|C|$ **do**
      $d_{ij} = ||U_i - U_j||$; // Compute the distance between two cluster centers
    end
  end
  $d_{min} = min_{i,j}\{d_{i,j}\}$;
  $(I, \ ) = ar \ min_{i,j}\{d_{i,j}\}$;
  if $d_{min} < d$; // The distance between cluster centers is below the threshold.
  then
    $C = C - C_I - C_J$; // Remove cluster C_I and C_J from C
    $C_{New} = C_I \cup C_J$; // Merge the two clusters
    $C = C + C_{New}$; // Add the new cluster into C
    $U_{New} = \frac{|C_I|}{|C_I|+|C_J|} U_I + \frac{|C_J|}{|C_I|+|C_J|} U_J$; // Update the center of the cluster
  end
  else
    return C;
  end
end

The main framework to construct view A is illustrated in Fig. 1. It is built by the satellite map of a city. Thus, it reflects the visual aspect of the region of interest. We denote view A of district $U^{(i)}$ as $A^{(i)} = (A_1^{(i)}, A_2^{(i)}, ..., A_6^{(i)})^T$. Note that what is revealed by the visual features is the portion of the area for each category of land use. Here, the portion of the land use for each category is obtained by averaging the class probabilities output by the deep neural network over the image tiles in a UCD. As demonstrated experimentally in [15], land use is somehow correlated to commercial activeness. For instance, farmlands, water, and woods usually dominate the regions in countryside, which correspond to relatively low commercial activeness, while in city centers, the other categories of land use dominate the regions in general, which correspond with relatively high commercial activeness. Thus, the land use probabilities resulting from the CNN features can act as an indicator of commercial activeness. Accordingly, we inherit such features from [15] in this study.

3.3 Constructing View B: The Social Contexts

Intuitively, the social context surrounding a UCD should a□ect the commercial hotness of it. Following [35], we consider 4 factors of the surrounding environment: The local residents and their buying power, the number of visitors, the region functions, and the rating scores from customers. They are combined to build view B, which represents the social context feature of a given district. The details are described as follows.

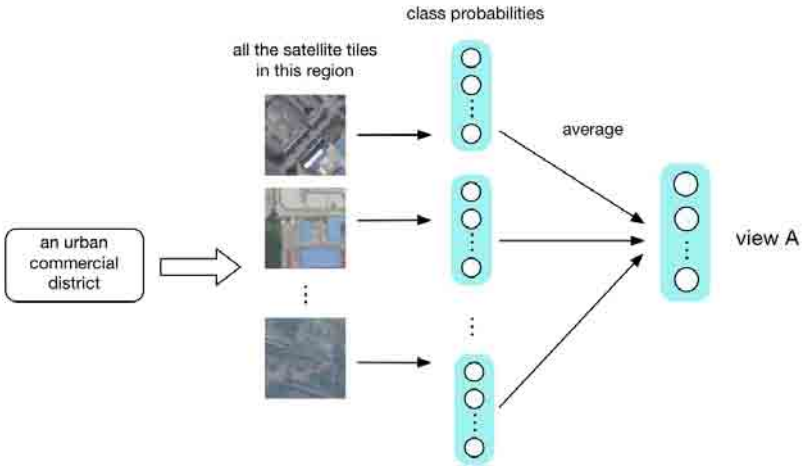


Fig. 1. Constructing view A. The class probabilities of 6 categories of land use are estimated by a deep learning-based model. The view A of a given UCD is the average of all the probability vectors in this district.

3.3.1 Local Population and Buying Power. In this paper, we refer to the index of commercial hotness as the number of people who have interacted with the commercial entities of interest. Intuitively, the number of the local residents should be an important factor to a□ect commercial activeness as they are potential customers. Also, the richness of those residents should a□ect the commercial hotness since it reflects the buying powers targeted by di□erent types of commercial entities. We measure the number of local population and their buying power using the number of and the average price of the houses around this district, respectively. To be speci□c, we count the number of the houses as $NH^{(i)}$, and denote the average price of the houses as $APH^{(i)}$. We only consider the houses within 5 kilometers to the center of the UCD of interest. Here, the number of the houses refers to the number of the units in the buildings of interest, where each unit is for one family, and the units with 2 or 3 rooms dominate the data as the number of the persons in a family is usually 2□3 in China.

3.3.2 Visitors. Despite the local residents, the visitors to the district of interest should also a□ect the commercial hotness. In this paper, we measure the visitors to a given region by calculating the OD (Origin-Destination) flow with regard to the district, which are denoted as $OV^{(i)}$ and $DV^{(i)}$.

3.3.3 Region Functions. To better characterize the functions of the regions surrounding a UCD, we utilize the BOW (Bags of Words) method to quantify the POI distributions. We take into account all the POIs in the range of 5 kilometers to the UCD center and count the number of the POIs of each category. Then, the region functions of district $U^{(i)}$ are formulated as $(RF_1^{(i)}, RF_2^{(i)}, ..., RF_m^{(i)})^T$, where m is the number of the POI categories taken into account. In this paper, we let $m = 13$.

3.3.4 Average Price and Rating of Commercial Entities. The attributes related to commercial entities themselves should also be related issues. Thus, we collect web comments from Dianping.com, which is the largest online service to make reviews on commercial entities in China, to retrieve the average price and the rating score of each district, which is the average of the rating scores regarding all the commercial enities in a UCD. We denote them as $AP^{(i)}$ and $AR^{(i)}$, respectively.

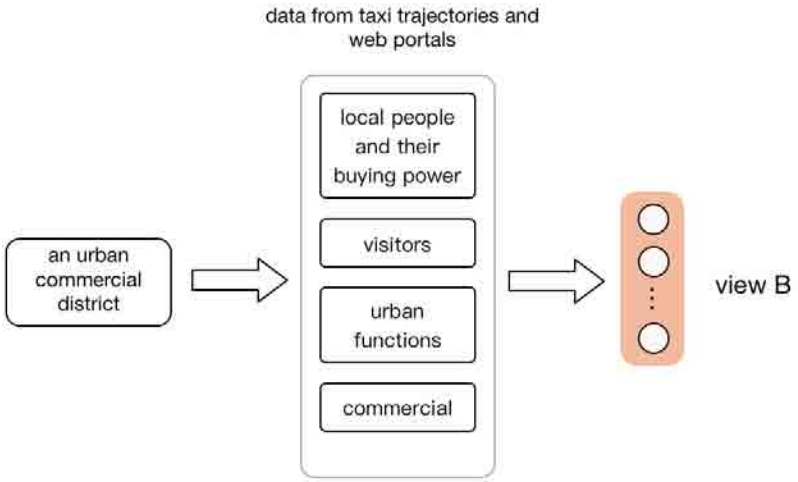


Fig. 2. Constructing view B. We consider 4 factors of the surrounding social contexts: The local residents and their buying power, the number of visitors, the urban functions, and the statistical attributes of the commercial entities.

In all, we combine all the above factors to construct view B. We denote view B of district $U^{(i)}$ as

$$B^{(i)} = (B_1^{(i)}, B_2^{(i)}, ..., B_{19}^{(i)})^T = (NH^{(i)}, APH^{(i)}, OV^{(i)}, DV^{(i)}, RF_1^{(i)}, RF_2^{(i)}, ..., RF_{13}^{(i)}, AP^{(i)}, AR^{(i)})^T \quad (1)$$

which reflects the social context information near a UCD in a non-visual manner. Fig. 2 shows all the related factors in view B.

3.4 Proxy of Commercial Hotness

In terms of commercial activeness, since the true transaction data are private for commercial entities, which are mostly unavailable for public, a couple of measures can be applied as proxy such as human mobility in the form of origin-destination (OD) flows, and the number of web comments on commercial entities. Here, we prefer the later one, since the users who dropped comments on a commercial entity are true customers. Therefore, such a variable reflects how much attention has been attracted with regard to each commercial entity or UCD, and can be counted explicitly. In contrast, OD flows correspond with potential customers. Although higher values of OD flows correspond with greater chances of transactions, as occasional behaviors, it is not easy to obtain the transfer rate from potential customers to true customers. On account of the straightforward perspective in counting how each commercial entity or UCD attracts true customers attention, we prefer to use the number of web comments on commercial entities as the proxy of commercial activeness. In this study, we treat the total number of the comments made by customers on a UCD as the proxy of commercial hotness. The data are also collected from Dianping.com. The reason to use this proxy is that Dianping.com is the primary web service to leave comments on commercial entities in China. The number reflects how many customers have participated in commercial activities to some extent. Thus, it is a sound measure for commercial hotness.

3.5 Preliminaries

For the aforementioned terms and de nitions, we present them in Table 1.

Table 1. Terms and Definitions

Terms	De nition
P_i	The ith point of interest (POI).
$U^{(i)}$	The ith urban commercial district (UCD). It usually consists of some POIs.
$A^{(i)}$	The feature of view A for the ith UCD. $A^{(i)} = (A_1^{(i)}, A_2^{(i)}, ..., A_6^{(i)})^T$
$NH^{(i)}$	The number of the houses in the ith UCD.
$APH^{(i)}$	The average price of the houses in the ith UCD.
$OV^{(i)}$	The outflow from the ith UCD.
$DV^{(i)}$	The inflow to the ith UCD.
$RF_j^{(i)}$	The jth region functions of the ith UCD.
$AP^{(i)}$	The average price of the commercial services in the ith UCD.
$AR^{(i)}$	The average rating score of the commercial services in the ith UCD.
$B^{(i)}$	The feature of view B for the ith UCD. $B^{(i)} = (B_1^{(i)}, B_2^{(i)}, ..., B_{19}^{(i)})^T = (NH^{(i)}, APH^{(i)}, OV^{(i)}, DV^{(i)}, RF_1^{(i)}, RF_2^{(i)}, ..., RF_{13}^{(i)}, AP^{(i)}, AR^{(i)})^T$

3.6 Context-aware Neural Network Ensemble

We propose a novel context-aware neural network ensemble method (CNNE for short) to fuse decisions based on di erent views , which promises better prediction than using the data of a single view. A neural network [14] is usually a feed forward system, which takes the raw features as input, and produces the prediction directly. The weights in a neural network are learned by backpropagation [16]. It has been demonstrated that a multilayer neural network is a universal approximator [18]. When considering multi-view data, a single neural network could be insufficient to deal with the di erent dynamic scales as well as the varying semantics of multimodality. In such a case, neural network ensemble is a natural choice, where a couple of neural network models are combined to make nal decision and each model deals with a certain view of the data. Regarding neural network ensemble, the challenge lies in how to determine the weight of each individual neural processor in the nal decision, especially for the regression problem. The existing methods assign xed weights to all the predictors, which does not comply with the practical situations. In practice, each predictor performs di erently on di erent samples in di erent contexts. In this study, we propose a new neural network ensemble model, where the weighting of each predictor is adaptive to the input sample, learnt from the historical performance of each predictor over di erent samples by using another neural network to compute the weight, where the weight decision component is referred to as estimator. By applying the output of the predictor on each sample as a feedback to the input, the estimator functions to remember the performance of the corresponding predictor on each sample as well as the resulting prediction, say, context. By aggregating the predictor and the estimator of each view into a neural network ensemble system, the weight of each predictor can be determined in an automatic manner by conducting end-to-end training. When each neural processor performs to predict, its weight varies with the input as well as the output since the estimator computes the weight by recalling the performance of the corresponding predictor on similar situations, obtained by applying the output as the feedback to the input of the estimator. For the commercial hotness prediction problem, since the predictive power of di erent views of the data may vary from region to region, we apply the estimator to produce a "con dence score" to weight the prediction. Thus, we can refer to the weighting scores to see how di erent views contribute to the prediction in each case. For example, view A is more suitable for predicting commercial hotness than view B in some areas in the experiments. Correspondingly, the output of the CNNE should produce a larger weighting score when applying the data of

view A than those of view B, and the final result will be biased to view A, resulting in more accurate prediction. In our model, the weighting scores are learned automatically given the supervised data of different views and the targeted prediction.

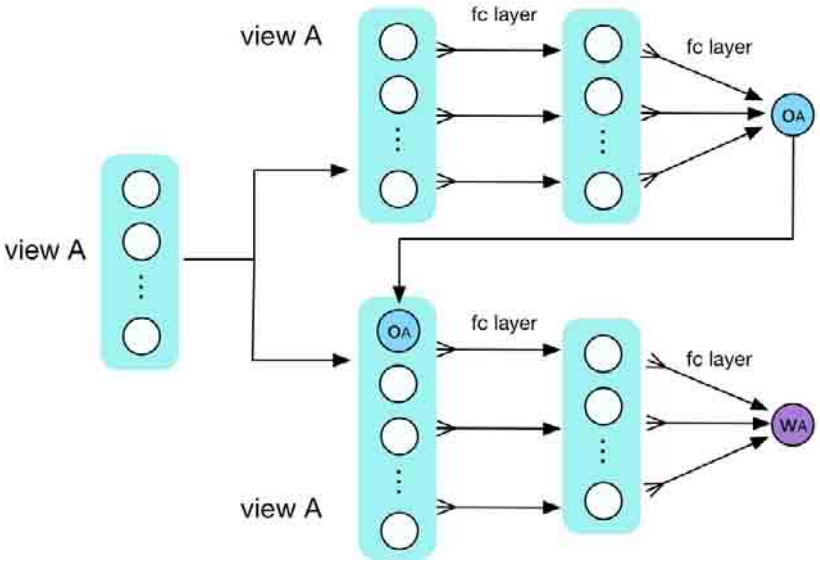


Fig. 3. The 2-channel structure used for generating prediction and the corresponding weighting score for a single view, where fc layer means fully connected layer in the neural network.

The network structure used for generating the prediction and the weighting score for single view is shown in Fig. 3, taking view A as an example. The network has a 2-channel structure:

- We perform traditional neural transformation to obtain prediction in the upper channel. In this paper, we use two fully-connected layers to produce the predicted value o_A from the input feature $A = (A_1, A_2, ..., A_6)^T$. Here, we omit the notation i for simplicity. The hidden layer is composed of a linear transform with an element-wise activation function, which can be formulated as $H_A = f(W_1 A + b_1)$, where W_1 is a matrix of $k_1 \times 6$, k_1 the size of this hidden layer, and f the activation function. In this paper, we use ReLU as the activation function if not otherwise specified. The final prediction of view A is produced as $o_A = f(W_2 H_A + b_2)$, where W_2 is a $1 \times k_1$ matrix.
- The lower channel in Fig. 3 is used for calculating the weighting score of view A. We not only utilize the features of view A, but also take the prediction o_A into consideration when computing the weighting score. Here, information from the raw input and the prediction resulting from it are combined to generate the confidence score. First, we concatenate the feature of view A and the resulting prediction to form a new input vector $A' = (o_A, A_1, A_2, ..., A_6)^T$. The hidden layer is calculated similarly and we denote it as $I = f(U_1 A' + c_1)$. The weighting score w_A is formulated as $w_A = \ (U_2 I + c_2)$, where g is a special activation function:

$$(x) = \frac{tanh(x) + 1}{2} \tag{2}$$

We use this activation function because it guarantees that the weight is in the range of $(0, 1)$.

It is worth mentioning that the structure to produce the prediction and the weight here can be replaced by any other networks. This feature gives our model enough freedom that can be expanded in the future.

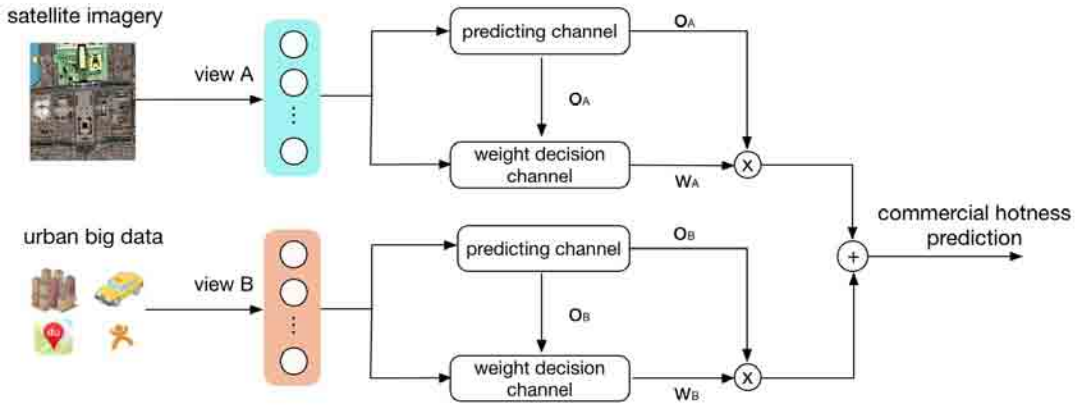


Fig. 4. The context-aware neural network ensemble system.

The network that fuses different views to form the final prediction is shown in Fig. 4. We use the 2-channel networks shown in Fig. 3 to generate o_A and w_A for view A, and o_B and w_B for view B with the same structure. The final prediction o is defined as:

$$o = \frac{w_A o_A + w_B o_B}{w_A + w_B} \tag{3}$$

We define the $L2$ loss function of each view to guide the training of the networks, denoted as $Loss_A$ and $Loss_B$, respectively. To learn the weight of each view, we also give an $L2$ loss for the output o, which is marked as $Loss_{combined}$. The final loss to be optimized is the sum of these losses:

$$Loss = Loss_A + Loss_B + Loss_{combined} \tag{4}$$

The model promises adaptive weighting since it can not only produce prediction based on the data of each view, but also gives the weighting score of the prediction. The weighting scores reflect the confidence of the predicting model on each view and is learned automatically in an end-to-end manner. By combining the decision from multiple views, more reasonable prediction can be promised.

Provided the input and the corresponding output of predictor F_A is (A, O_A), what the estimator (weight decision component) learns is the confidence when A produces O_A through F_A, namely, $W_A = P\{O_A = F_A(A)\}$, where P denotes the probability. By learning the performance of F_A over $\{(A, O_A)\}$ via the weight estimator, the ensemble learning scheme gains advantage over the existing ones focused on either $P\{O_A\}$ or $P\{A\}$ only. Besides, $W_A = P\{O_A = F_A(A)\}$ and $W_B = P\{O_B = F_B(B)\}$ result directly from the end-to-end learning in terms of minimize the total $Loss$ as defined in Eq. 4 in the neural network ensemble framework.

3.7 Models for Comparison

To validate the proposed model, we compare it with the following baseline methods:

- Linear Regression: Linear regression is the most basic regression model in the field of machine learning. Given the input features x_i and the output scalar y_i, linear regression is formulated as $y'_i = W^T x_i + b$. The loss function is $Loss = \sum_i ||y_i - y'_i||_2^2$.
- Ridge Regression: Ridge regression [17] is a traditional regression model. It introduces a Gaussian prior of the parameter matrix W in standard linear regression, and the loss function becomes $Loss = \sum_i ||y_i - y'_i||_2^2 + ||W||_2^2$, where is a hyperparameter to be adjusted.
- Support Vector Regression (SVR): SVR [10] is a non-linear regression method developed from the support vector machine (SVM) [9]. It is designed to minimize

$$\frac{1}{2}||w||_2^2, \; s.t. \; y_i - \quad \leq w^T \quad (x_i) + b \leq y_i + \tag{5}$$

  where stands for a kernel function, a predefined hyperparameter for thresholding, and $w^T \quad (x_i) + b$ the model prediction.
- Gradient Boosting Decision Tree (GBDT): GBDT takes advantage of the technique of gradient boosting [12]. It combines multiple weak regressors such as decision trees to form a more powerful and stable predictor. In this paper, we use XGBoost [8], a more complex implementation, as the experimental tool for GBDT.
- Neural Network (NN): Our method can be regarded as the model-level fusion of two neural networks. It produces the prediction on view A and view B, respectively, and then leverages two dynamic weights to obtain the final prediction. Here, we compare it with the feature-level fusion neural network: We concatenate the features of view A and view B directly, and then employ a single neural network for the final prediction. Since our model-level fusion method has only one hidden layer for each view, we use one hidden layer for feature-level fusion network too. Besides, the NN model is also applied to single view based prediction, namely, view A and view B, respectively.

4 EXPERIMENTS

4.1 The Data

We download satellite images of Beijing and Shanghai with Google Map API [1]. Only a rectangle area that covers the main urban area and contains the most commercial districts in either city is taken into account. [2]. We randomly sample 48,000 images in Beijing to train the deep learning model for visual feature extraction. The training set contains 6 classes and 8,000 images for each class. Then, we use the trained model to generate the features of view A with the satellite images in Shanghai for testing to guarantee generality of the model.

Using the clustering method of Algorithm 1, we find 385 UCDs in Shanghai. The distribution of the found UCDs are shown in Fig. 5. We construct view A using the satellite images of Shanghai, where the deep model learned from Beijing is utilized to compute the features of Shanghai UCDs. To construct view B, we use POI data and house data obtained from http://map.baidu.com and http://www1.fang.com. Besides, we collect the comments and ratings from customers at http://www.dianping.com, which is the most widely used web service to make reviews on commercial entities in China. Additionally, we gather the GPS trajectories of about 30,000 taxis in Shanghai to calculate the visiting flows to the UCDs. The GPS position is recorded 1-2 times per minute for a taxi. We treat the total number of the online reviews on the entities in a UCD as the target to be predicted, say, the proxy of commercial hotness, as this reflects how many users have interacted with the commercial entities inside the UCD of interest. Details of the data are presented in Table 2.

[1] https://developers.google.com/maps/

[2] For Beijing, it is 40° 09′24″N, 116° 09′27″E to 39° 44′01″N, 116° 40′18″E. For Shanghai, the area is 31° 25′45″N, 121° 07′57″E to 30° 49′21″N, 121° 59′22″E.

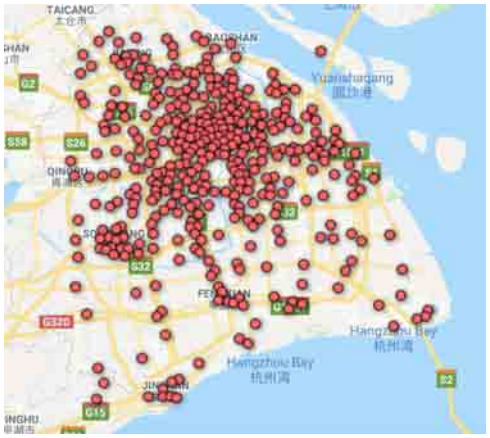


Fig. 5. The found 385 UCDs in Shanghai using the clustering method. Each UCD is marked as a red solid circle.

Table 2. Description of the data

GPS trajectories	GPS trajectories of 50,000 taxis in Shanghai with 1-2 times sampling per minute collected from Dec. 28, 2014 to Jan.10, 2015, for each of which taxi ID, longitude, latitude, time stamp of sampling, and the state of taking passenger or not is continuously recorded.
POIs	1,340,000 POIs, each of which includes name, longitude, latitude, and annotation of one of the 13 categories: Restaurant, Transportation Facility, Scenic Spot, Corporation & Business, Shopping Mall & Commercial Street, Financial Service, Education & Training, Motor Service, Life Service, Fitness Center, Hospital, Government & Organization, Residence & Hotel.
Local Population and House Prices	Prices and household information of 14,000 houses in Shanghai from Fang.com , where the name, address, longitude, latitude, average house price, and number of householders are recorded for each residential district.
Rating Scores of Customers	Number of comments, rating scores, and average prices of 110,000 commercial entities in Shanghai collected at Dianping.com from Dec. 9, 2014 to Feb. 11, 2015.
Satellite Images	48,000 satellite images of Beijing to train the deep learning-based model. 191,020 satellite images whose centers fall within the UCDs of Shanghai obtained from the clustering algorithm.

4.2 Evaluation and Parameter Settings

We randomly divide the data into a training set, an evaluation set, and a test set. The proportion for such data partition is 8:1:1. We train each model on the training set and use the evaluation set to select the best parameter. Finally, the performance on the test set is evaluated. To alleviate the influence of data partition and other random factors, we repeat the above process 50 times, and report the average performance for each model. In this paper, we use the coefficient of determination, denoted as R^2, as the metric for performance evaluation on the regression models. In a general form, given the observed data $y_1, y_2, ..., y_n$ and the predicted values $f_1, f_2, ..., f_n$, R^2 is:

$$R^2 = 1 - \frac{\sum_{i=1}^{n} (f_i - y_i)^2}{\sum_{i=1}^{n} (\bar{y} - y_i)^2} \tag{6}$$

where $\bar{y}$ stands for the average value of the observed data:

$$\bar{y} = \frac{\sum_{i=1}^{n} y_i}{n} \tag{7}$$

There is no parameters to be decided in the model of linear regression. For the other models, since the prediction of commercial hotness is a brand new problem, we do not know the exact range of the parameters in different models. Therefore, we adopt the grid search method to find the best parameters, and widen the search range to a relatively large scope. For ridge regression, is selected from $\{0, 0.5, 1, 2, 5, 10, 15, 20\}$. We apply a linear kernel and an RBF kernel for SVR model. The penalty parameter C in SVR is chosen from $\{1, 5, 10\}$. For the GBDT model, the max tree depth is selected from $\{3, 4, 5\}$ and the number of trees from $\{10, 50, 100, 150\}$. For the the neural networks for feature-level fusion, we choose the hidden layer size from $\{10, 20, 30, 40\}$ for single-view evaluation and $\{20, 40, 60, 80\}$ for multi-view scenario.

For the CNNE, we use the batch size of 32, and the initial learning rate is 0.1. We set the max number of epochs to be 100. Thereafter, an early stop mechanism is adopted: We evaluate the model after the training of every epoch. If the performance is not improved compared with that of the last time, the learning rate will be set to be one quarter of the present value. If the performance is not improved for 5 epochs, we will stop the training. The model that has the best performance on the evaluation set will be tested on the test set. We set the hidden size of the weighting channel to be 10.

4.3 Model Performance on Single Views

Before conducting multi-view learning, we try to predict commercial hotness from single view with traditional regression methods. The results are shown in Table. 3. It can be concluded that view B has better predictive performance than view A in terms of commercial hotness prediction, since the best model on view B achieves the accuracy of 62.45%, while the value for view A is only 50.87%. The best model is not the same on view A and view B because of the different statistical natures of the data. The results suggest that such models on single view do not perform well. In the following, we will evaluate these models on multiple views.

4.4 Model Performance on Multiple Views

We predict commercial hotness from multi-view data using both the traditional methods and the proposed approach. The results are shown in Table 3. For the traditional methods, namely, linear regression, ridge regression, SVR, GBDT, and feature-level fusion neural networks, we concentrate the features of the two views to form the input for these regressors. Our approach is denoted as CNNE(n_a, n_B), where n_a is the hidden size of the network to perform the prediction based on view A, and n_b is that for view B.

We can find in the table that CNNE outperforms all the traditional methods. The best result of the traditional methods is 61.90%, while CNNE achieves 70.39%. Generally, the performances become better when n_a and n_b become larger. We also provide the performance for each view in Table 4, which is the output based on either independent view as a byproduct of the overall prediction using the proposed method, say, O_A and O_B in Fig. 4. It is notable that the performance based on multi-view prediction is much better than that based on single-view prediction, which verifies that our model can combine multi-view predictions appropriately to render better prediction. Moreover, it is notable that the performance based on feature-level fusion improves little or even gets worse compared with that of the corresponding single view based prediction, in accordance with the results in Table 3.

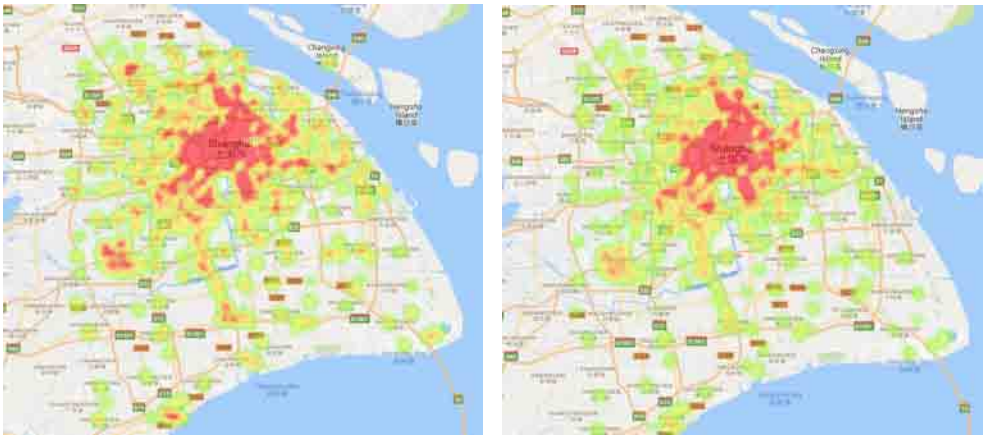
Table 3. First, we evaluate the traditional models on view A and B, separately. Then, prediction over feature-level fusion is conducted using Linear Regression, Ridge Regression, SVR, GBDT, and NN, respectively, where NN stands for the feature-level fusion neural networks. The best model for view A is Ridge Regression, which has an R^2 value of 50.87%. The best model on view B is GBDT. Its R^2 score is 62.45%. Then, the traditional methods and the proposed approach are both evaluated on the multi-view data combining view A and view B. We denote the proposed approach as CNNE_nf(n_a, n_B) and CNNE(n_a, n_B), where we do not feedback the prediction result to the weighting channel in CNNE_nf, while CNNE is the standard model. n_a stands for the hidden size of the network to perform prediction based on view A, and n_b is that for view B.

Model	R^2 based on View A	R^2 based on View B
Linear Regression	50.53%	46.15%
Ridge Regression	50.87%	46.28%
SVR	48.16%	60.47%
GBDT	47.73%	62.45%
NN	49.12%	61.15%
	R^2 based on View A and view B	
Linear Regression	51.71%	
Ridge Regression	53.07%	
SVR	59.89%	
GBDT	61.90%	
NN	61.23%	
CNNE_nf(10, 10)	60.86%	
CNNE(10, 10)	62.56%	
CNNE_nf(20, 20)	65.83%	
CNNE(20, 20)	66.78%	
CNNE_nf(30, 30)	67.06%	
CNNE(30, 30)	69.51%	
CNNE_nf(40, 40)	68.10%	
CNNE(40, 40)	70.39%	

Notably, to evaluate our contribution in network structure design, we remove the feedback denoted as o_A in the network model shown in Fig. 3, which is denoted as CNNE_nf in Table 3 to check the effect. Here, we compare the modified model with the original model in Fig. 3 and from Table 3, we can see that the performance drops down obviously once the feedback from the output of the predictor to the input of the weight estimator is removed. We attribute this to the mechanism of the proposed neural network ensemble model, that is, what the weight estimator learns is the performance of the predictor on different contexts, which is encoded by the IO values. The weight estimation component remembers the contextual situation in terms of not only the input sample but also its effect, say, the resulting output.

Table 4. To verify the generalization ability on multi-view data of our model, we provide the results for each view and the overall performance. It is notable that, although the performance on single view is not good enough, our model is able to combine these two views appropriately to perform much better.

Data	Model	R^2 based on View A	R^2 based on View B	R^2
View A and View B	CNNE(10, 10)	51.63%	61.97%	62.56%
	CNNE(20, 20)	52.37%	62.21%	66.78%
	CNNE(30, 30)	53.98%	63.06%	69.51%
	CNNE(40, 40)	54.32%	65.50%	70.39%



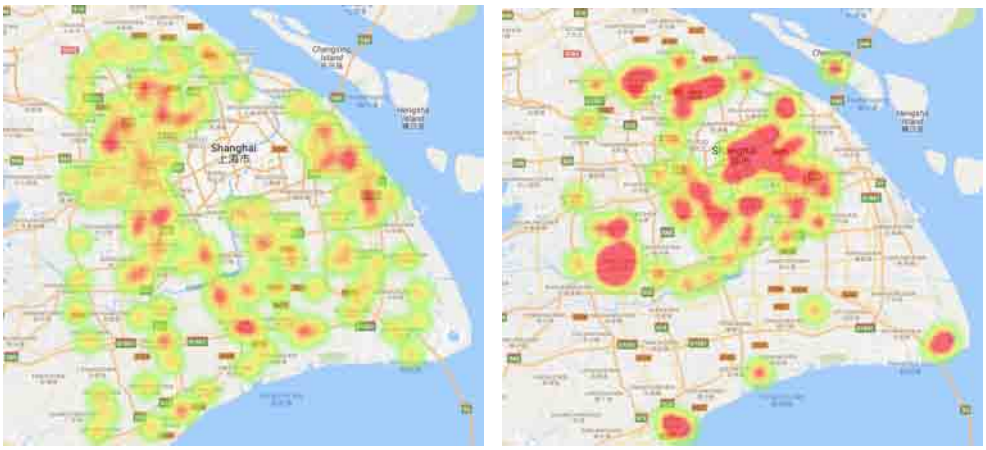
(a) The ground truth (b) The predicted commercial hotness

Fig. 6. Visualization of the commercial activeness in Shanghai. Figure (a) represents the ground truth commercial hotness counted as the total number of the reviews on the commercial entities. Figure (b) results from the predicted commercial hotness using the best model.

One of the interesting things is that the performances of SVR and GBDT are worse than their performances on a single view. The reason might be that these models can not treat the multi-view data well, and the additional dimension of features only make the models behave worse, due to the nonlinear nature of such models, which is subject to over□tting. In contrast, the simple models such as linear regression and ridge regression perform a little bit better when using the multi-view data.

4.5 Visualization of the Prediction and the Weights Over the City

Finally, we visualize the predicted commercial hotness in the form of a heatmap for Shanghai. To visualize the heatmap at city scale, we divide the whole dataset into 10 folds. Every time, we employ 8 folds for training and 1 fold for validation. The best model on the validation fold will be used to produce prediction on the remaining fold. We repeat the process for 10 times to get the model prediction for the whole city, as shown in Fig. 6. It is clear that the predicted commercial hotness map is similar to the ground truth. Such a map is very useful to trace the development of the whole city, and provides a valuable reference for city regulation and business planning.



(a) The learned weighting scores of view A (b) The learned weighting scores of view B

Fig. 7. Visualization of the weighting scores of the learned model in Shanghai. Figure (a) represents the scores of view A and Figure (b) is the case for view B. Generally, view B is more important than view A. View A is useful when used for UCDs in the countryside, while the model prefers view B when predicting commercial hotness in the city center.

In addition to visualizing the predicted commercial hotness map, we can also visualize the learned weights of different views to analyze the confidence scores over different views and different districts quantitatively. In the best model of CNNE, the average weight score of view A is 0.340, while the value of view B is 0.660. So, it suggests that the features of view B are more powerful than those of view A. The visualization of the weights of view A and view B are shown in Fig. 7, respectively. We can conclude from the figure that the model prefers view B when predicting commercial hotness in the city center, while view A is more important for UCDs at the edge of the city. The main reason might be that the city center is filled with too many objects to be classified in satellite images, so the model favors the data of view B in predicting the commercial hotness. At the edge of the city, satellite imagery becomes useful to distinguish man-made constructions from natural landscapes, where the model can perform to improve its accuracy.

5 CONCLUSION

In this paper, we have investigated predicting commercial hotness of a whole city with multi-view heterogeneous data. We develop a novel context-aware neural network ensemble method to fuse the decisions from different views, which achieves better prediction performance.

We use two views of urban data. One is from the satellite images, which describe the visual patterns of the city. The other view is formed by the social contexts surrounding the UCDs such as taxi trajectories and online comments from social media. Then, we compare our model with 4 traditional regression methods. The experimental results show that our model outperforms all the traditional methods with the accuracy of 70.39% against 62.45%, the best one of the traditional models.

The contribution lies in the following aspects: (1) This research aims to bridge the gap between data mining based urban computing and computer vision rendered urban perception in the context of commercial activeness prediction across a city. Here, the visual features resulting from deep learning over satellite images and the social context features are combined at the decision level to promise better prediction. To the best of our knowledge, this should be the first endeavor in this direction. (2) Prediction from heterogeneous data is becoming an increasingly

important issue for urban computing as urban big data are in general from di erent sources with multimodality. So far, there are not e ective means to fuse heterogeneous data to promise better decision. Here, we propose a novel neural network ensemble model to tackle such challenging problem. The novelty is that an estimator is developed to evaluate the con dence of the corresponding predictor with the output applied to the input of the estimator as feedback such that the weight of each predictor varies with the input on the fly, which is recalled from its historical performance on similar situations.

Yet, we are at the very beginning and future e ort to improve the model is really needed. As this study is focused on the commercial activeness of a region, we take into account all the commercial entities in a region as a whole and omit the categorical di erence between such commercial entities. For example, we use the average price of all the services in a region as one variable to characterize such region. In future works, investigation into category-sensitive commercial activeness prediction merits further endeavors.

ACKNOWLEDGMENTS

This work is supported by NSFC (grant No. 61472087) and Shanghai Science and Technology Commission (grant No. 17511104203).

REFERENCES

[1] Adrian Albert, Jasleen Kaur, and Marta C Gonzalez. 2017. Using convolutional networks and satellite imagery to identify patterns in urban environments at a large scale. In *Proceedings of the 23rd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*. ACM, 1357 1366.

[2] Joshua Blumenstock, Gabriel Cadamuro, and Robert On. 2015. Predicting poverty and wealth from mobile phone metadata. *Science* 350, 6264 (2015), 1073 1076.

[3] Leo Breiman. 1996. Bagging predictors. *Machine learning* 24, 2 (1996), 123 140.

[4] Leo Breiman. 2000. Randomizing outputs to increase prediction accuracy. *Machine Learning* 40, 3 (2000), 229 242.

[5] Longbiao Chen, Jérémie Jakubowicz, Dingqi Yang, Daqing Zhang, and Gang Pan. 2017. Fine-Grained Urban Event Detection and Characterization Based on Tensor Cofactorization. *IEEE Trans. Human-Machine Systems* 47, 3 (2017), 380 391.

[6] Longbiao Chen, Daqing Zhang, Gang Pan, Xiaojuan Ma, Dingqi Yang, Kostadin Kushlev, Wangsheng Zhang, and Shijian Li. 2015. Bike sharing station placement leveraging heterogeneous urban open data. In *Proceedings of the 2015 ACM International Joint Conference on Pervasive and Ubiquitous Computing*. ACM, 571 575.

[7] Longbiao Chen, Daqing Zhang, Leye Wang, Dingqi Yang, Xiaojuan Ma, Shijian Li, Zhaohui Wu, Gang Pan, Thi-Mai-Trang Nguyen, and Jérémie Jakubowicz. 2016. Dynamic cluster-based over-demand prediction in bike sharing systems. In *Proceedings of the 2016 ACM International Joint Conference on Pervasive and Ubiquitous Computing*. ACM, 841 852.

[8] Tianqi Chen and Carlos Guestrin. 2016. Xgboost: A scalable tree boosting system. In *Proceedings of the 22nd acm sigkdd international conference on knowledge discovery and data mining*. ACM, 785 794.

[9] Corinna Cortes and Vladimir Vapnik. 1995. Support-vector networks. *Machine learning* 20, 3 (1995), 273 297.

[10] Harris Drucker, Christopher JC Burges, Linda Kaufman, Alex J Smola, and Vladimir Vapnik. 1997. Support vector regression machines. In *Advances in neural information processing systems*. 155 161.

[11] Yoav Freund, Robert E Schapire, et al. 1996. Experiments with a new boosting algorithm. In *Icml*, Vol. 96. Bari, Italy, 148 156.

[12] Jerome H Friedman. 2001. Greedy function approximation: a gradient boosting machine. *Annals of statistics* (2001), 1189 1232.

[13] Yanjie Fu, Yong Ge, Yu Zheng, Zijun Yao, Yanchi Liu, Hui Xiong, and Jing Yuan. 2014. Sparse real estate ranking with online user reviews and o ine moving behaviors. In *Data Mining (ICDM), 2014 IEEE International Conference on*. IEEE, 120 129.

[14] Martin T Hagan, Howard B Demuth, Mark H Beale, et al. 1996. *Neural network design*. Vol. 20. Pws Pub. Boston.

[15] Zhiyuan He, Su Yang, Weishan Zhang, and Jiulong Zhang. 2018. Perceiving Commerial Activeness Over Satellite Images. In *Companion of the The Web Conference 2018 on The Web Conference 2018*. International World Wide Web Conferences Steering Committee, 387 394.

[16] Robert Hecht-Nielsen. 1992. Theory of the backpropagation neural network. In *Neural networks for perception*. Elsevier, 65 93.

[17] Arthur E Hoerl and Robert W Kennard. 1970. Ridge regression: Biased estimation for nonorthogonal problems. *Technometrics* 12, 1 (1970), 55 67.

[18] Kurt Hornik, Maxwell Stinchcombe, and Halbert White. 1989. Multilayer feedforward networks are universal approximators. *Neural networks* 2, 5 (1989), 359 366.

[19] Md M Islam, Xin Yao, and Kazuyuki Murase. 2003. A constructive algorithm for training cooperative neural network ensembles. *IEEE Transactions on neural networks* 14, 4 (2003), 820 834.

[20] Neal Jean, Marshall Burke, Michael Xie, W Matthew Davis, David B Lobell, and Stefano Ermon. 2016. Combining satellite imagery and machine learning to predict poverty. *Science* 353, 6301 (2016), 790–794.
[21] Dmytro Karamshuk, Anastasios Noulas, Salvatore Scellato, Vincenzo Nicosia, and Cecilia Mascolo. 2013. Geo-spotting: mining online location-based services for optimal retail store placement. In *Proceedings of the 19th ACM SIGKDD international conference on Knowledge discovery and data mining*. ACM, 793–801.
[22] Anders Krogh and Jesper Vedelsby. 1995. Neural network ensembles, cross validation, and active learning. In *Advances in neural information processing systems*. 231–238.
[23] Hui Li, Xuesong Wang, and Shifei Ding. 2018. Research and development of neural network ensembles: a survey. *Artificial Intelligence Review* 49, 4 (2018), 455–479.
[24] Ling-ling Li, Xi-yu Liu, and Shu-qiang Lu. 2007. Constructive methods for parallel learning neural network ensemble based on particle swarm optimization. *ShanDong Sci* 20, 4 (2007), 16–20.
[25] J Lin and BZ Zhu. 2006. Neural network ensemble based on forecasting e-ective measure and its application. *Journal of Computational Information Systems* 6 (2006), 781–787.
[26] Yanchi Liu, Chuanren Liu, Xinjiang Lu, Mingfei Teng, Hengshu Zhu, and Hui Xiong. 2017. Point-of-Interest Demand Modeling with Human Mobility Patterns. In *Proceedings of the 23rd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*. ACM, 947–955.
[27] Xinjiang Lu, Zhiwen Yu, Chuanren Liu, Yanchi Liu, Hui Xiong, and Bin Guo. 2017. Forecasting the rise and fall of volatile point-of-interests. In *Big Data (Big Data), 2017 IEEE International Conference on*. IEEE, 1307–1312.
[28] Yueliang Ma and Ruisong Xu. 2010. Remote sensing monitoring and driving force analysis of urban expansion in Guangzhou City, China. *Habitat International* 34, 2 (2010), 228–235.
[29] Prem Melville and Raymond J Mooney. 2005. Creating diversity in ensembles using arti-cial data. *Information Fusion* 6, 1 (2005), 99–111.
[30] Christopher Njuguna and Patrick McSharry. 2017. Constructing spatiotemporal poverty indices from big data. *Journal of Business Research* 70 (2017), 318–327.
[31] Robert E Schapire. 1990. The strength of weak learnability. *Machine learning* 5, 2 (1990), 197–227.
[32] Holger Schwenk and Yoshua Bengio. 1997. Adaboosting neural networks: Application to on-line character recognition. In *International Conference on Artificial Neural Networks*. Springer, 967–972.
[33] Vivek K Singh, Laura Freeman, Bruno Lepri, and Alex Sandy Pentland. 2013. Predicting spending behavior using socio-mobile features. In *Social Computing (SocialCom), 2013 International Conference on*. IEEE, 174–179.
[34] William L Stefanov, Michael S Ramsey, and Philip R Christensen. 2001. Monitoring urban land cover change: An expert system approach to land cover classi-cation of semiarid to arid urban centers. *Remote sensing of Environment* 77, 2 (2001), 173–185.
[35] Su Yang, Minjie Wang, Wenshan Wang, Yi Sun, Jun Gao, Weishan Zhang, and Jiulong Zhang. 2017. Predicting Commercial Activeness over Urban Big Data. *Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies* 1, 3 (2017), 119.
[36] Yingjie Zhang, Beibei Li, and Jason Hong. 2016. Understanding user economic behavior in the city using large-scale geotagged and crowdsourced data. In *Proceedings of the 25th International Conference on World Wide Web*. International World Wide Web Conferences Steering Committee, 205–214.

Received May 2018; revised August 2018; accepted October 2018