Cross-region Traffic Prediction for China on OpenStreetMap

Frank F. Xu[1] Bill Y. Lin[1] Qi Lu[1] Yifei Huang[1] Kenny Q. Zhu[2]
Shanghai Jiao Tong University
Shanghai, China
[1] {frankxu,yuchenlin,icedream,5130309269}@sjtu.edu.cn [2] kzhu@cs.sjtu.edu.cn

ABSTRACT

OpenStreetMap (OSM) is a free, open-source and popular mapping service. However, due to various reasons, it doesn't offer live traffic information or traffic prediction for China. This paper presents an approach and a system to learn a prediction model from graphical traffic condition data provided by Baidu Map, which is a commercial, close-source map provider in China, and apply the model on OSM so that one can predict the traffic conditions with nearly 90% accuracy in various parts of Shanghai, China, even though no traffic data is available for that area from Baidu Map. This novel system can be useful in urban planning, transportation dispatching as well as personal travel planning. [1]

CCS Concepts

•Information systems → Geographic information systems; Data mining;

1. INTRODUCTION

The goal of this work is to develop a framework and a system to predict traffic conditions on any roads given a map formatted in OpenStreetMap data [5]. Traffic prediction can help urban planners optimize roads and arrange land usage. Also, it can assist people to organize their itinerary and routes more reasonable. In addition, it can help with transport distribution. The inputs to the problem include: the topological map data, time of the day, day of the week, location and weather, etc. The output will be classified into four classes: green (good), yellow (slow), red (congested) and deep red (extremely congested).

There are four major challenges in this research. The first one is the data incompleteness. Baidu Map does not directly share their data of POIs and real-time traffic information. Although OpenStreetMap is free, it lacks lots of POIs in China and has no real-time traffic information. Apart from that, integrating these two resources is also difficult because the two maps are using different coordinate systems. The third challenge is feature selection. There are plenty of features that will influence the traffic conditions. We need to select all the possible features without introducing noises. The last one is the data imbalance. This dataset is predominately composed of green (good) than other classes.

Our approach can be divided into three stages.

- First, integrating the map data and live traffic data from Baidu Map into OSM.
- Second, feature engineering. In this stage, we come up with many useful features.
- Third, prediction model training. We use multi-class linear SVM with weight [16] to train our model.

The main contributions of this paper are as follows:

- First, our system can do cross prediction, we employ the transfer learning concept in traffic prediction field so that we can predict traffic in areas where we do not have any historical traffic data, which is one of the first systems that are enable to do this.
- Second, our system combines diverse useful features that can be categorized into two types: geospatial features and implicit features.
- Third, our approach to address the data imbalance issue is effective in this particular multi-classification problem.

2. OUR APPROACH

In this section, we will give preliminaries about OSM and Baidu Map and then present the three stages of our approach in detail. Plus, we showcase a demo at the end.

2.1 OpenStreetMap

The open-source OpenStreetMap data dump is available at http://planet.osm.org/. Elements are the basic components of OpenStreetMap's data. They consist of nodes, ways and relations. All of the above can have several tags, which describe the meaning of a particular element.

A node represents a specific point on the earth's surface defined by its latitude and longitude such as a park. We mainly use nodes to represent local features such as POIs (Points Of Interests).

A way is an ordered list of between 2 and 2,000 nodes that define a polyline. Ways are used to represent roads and

[1] Kenny Q. Zhu is the corresponding author and is supported by NSFC Grant No. 61373031.

IWCTS'16, October 31-November 03 2016, Burlingame, CA, USA
© 2016 ACM. ISBN 978-1-4503-4577-4/16/10. . . $15.00
DOI: http://dx.doi.org/10.1145/3003965.3003972

boundaries of areas such as forests. We mainly focus on the ways with specific tag “highway”, which identifies any kind of road, street or path. For example, “highway=residential” defines a road connecting homes within a residential area.

A relation documents a relationship between two or more data elements (nodes, ways, and/or other relations), which may list the ways that form a highway, a cycle route, or a bus route.

2.2 Baidu Map

While OSM serves as the basis of the map structures, it has not become one of the most popular mapping software in China, and hence its user-generated features such as points of interests are extremely deficient. More importantly, OSM does not provide live traffic conditions in China, which is the key for training a prediction model.

Baidu Map, the biggest map provider in China, however, has all the bells and whistles. It provides lots of web APIs. We use the following APIs in this work. One is coordinate conversion API used for converting other global coordinate systems (e.g., WGS-84 and GCJ-02) to Baidu lon-lat coordinates BD-09. Another API converts BD-09 coordinates to Baidu’s internal (x, y) point coordinates. These internal coordinates can be used to locate image pixels on Baidu map. The third API is bounded POIs search, which can return all POIs of a certain category within a certain area bound.

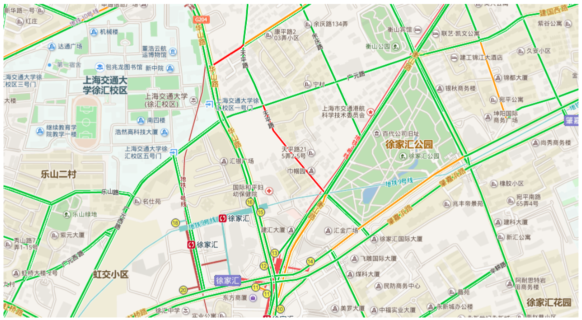


Figure 1: Part of Shanghai on Baidu Map with live traffic

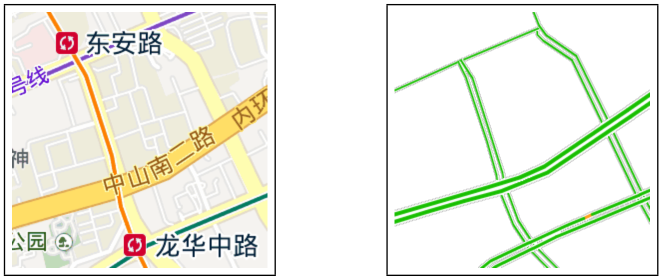
Baidu also provides near real-time traffic condition information. For example in Fig. 1, green, yellow, red and deep red lines represent good, slow, congested and extremely congested traffic respectively. Programmatically, the traffic information can be obtained by two image APIs:

- Map image API. [2] See Fig. 2a for returned image.
- Traffic image API. [3] See Fig. 2b for returned image.

Each of the APIs returns a 256 × 256 pixels image tile given a pair of (x, y) coordinates and a zoom level. With the same x and y and zoom level, the two APIs return tiles of the same physical size and in the same location, such as shown in Fig. 2.

2.3 Integration of Baidu Map into OSM

Having collected the POIs and traffic condition image tiles from Baidu, our major challenges are i) that Baidu and OSM use different coordinate systems; ii) the traffic condition needs to be read from the image tiles and assigned to individual road segments on OSM. We next explain our solution to these challenges.



(a) map tile (b) traffic tile

Figure 2: Image tiles (256 × 256 pix) at same location

2.3.1 *Transformation of coordinates*

There are three types of coordinate Systems involved in this work:

- WGS-84, real coordinates from satellites, adopted by OSM and Google map;
- GCJ-02, encrypted from WGS-84 in for use in China;
- BD-09, encrypted from GCJ-02 for use in Baidu.

The mapping from WGS-84 to GCJ-02 and then to BD-09 can be done by transformation functions provided by [9]. However, the inverse mapping, which is what we need to merge Baidu’s POIs into OSM, is not available. The transformation between GCJ-02 and BD-09 are analytically solvable, so the challenge lies in the GCJ-02 and WGS-84 transformation. We use an iterative numerical analysis to estimate the inverse function with an error within 0.5 meters.

To match the road segments in OSM to Baidu and hence obtain the traffic condition, we need to first convert OSM coordinates to BD-09 by f and then from BD-09 to Baidu’s (x, y) *point coordinates* by the Baidu API. With the (x, y) point coordinates in Baidu, we can compute the exact pixel in the traffic tile image downloaded from Baidu.

2.3.2 *Parsing Traffic Information*

Despite all above efforts, the translation from WGS-84 into Baidu’s pixel coordinates is not perfect. Furthermore, the nodes along an OSM road are often mapped to the center of the a two-way road (shown as the black dots in Fig. 3), where the traffic information (the colored lines) are painted on the sides of the road (see Fig. 3). Therefore, in this section, we introduce our approach to fuzzy match the nodes translated from OSM to traffic condition.

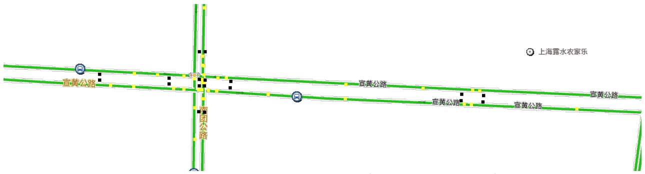


Figure 3: Fuzzy matching example

We assume the road segment between two consecutive nodes n_1 and n_2 (shown as black dots in the figure) is a

[2] `http://online1.map.bdimg.com/onlinelabel/?qt=tile&x=13204&y=3550&z=16`

[3] `http://its.map.baidu.com:8002/traffic/TrafficTileService?level=16&x=13204&y=3550&time=1464020567811`

straight line and compute n sample points on this line. The size of n is proportional to the distance between n_1 and n_2. If this is a one-way road, we scan the pixels along the line perpendicular to the road segment to look for nearest colored pixels on the image within threshold distance. The four colors have standard RGB values for Baidu map. If any of these colors is located, we have found a *traffic sample*, and this is marked as a yellow dot in Fig. 3. By majority votes, the traffic samples for each road segment together decide the traffic condition of that segment.

For two-way roads, traffic conditions can be found on two directions and this needs to be handled by looking for two parallel colored lines nearby the sample points, indicated as red and blue dots in Fig. 4. Almost all traffic conditions are correctly identified in this very complex scenario.

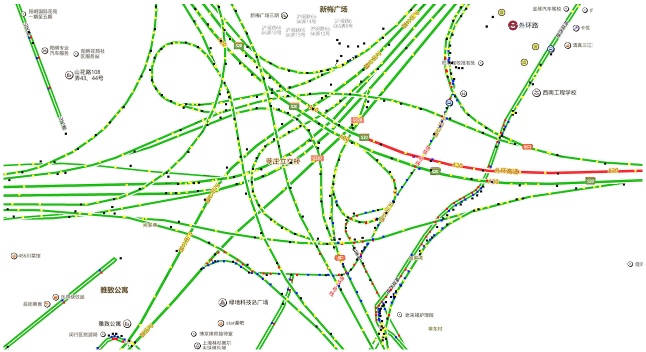


Figure 4: Massive junction of elevated roads

2.4 Feature Engineering

Our system makes use of two types of features: geospatial features and implicit features. Each feature is calculated and assigned to every individual road segment on the road map. Next we explain these features in more details.

2.4.1 Geospatial Features

The main intuition behind these geospatial features is that the traffic condition depends on the the density of the local road network and also the type and number of POIs near by. If there are too many traffic lights in a short stretch of road, chances are the traffic will be slow. On the other hand, if there is a school by the road, traffic will be affected in the morning and afternoon due to the delivery and picking up of school children by their parents.

To measure the density of the road network, we calculate the average distance between two adjacent traffic lights [4] in both directions of the road segment. We divide all POIs on Baidu map into 37 types. This classification is obtained by merging some of the finer categories which were defined by Baidu's 2-level POI type hierarchy. Our second type of local features are the distribution of these 37 types of POI in the vicinity of the target road segment, computed in both directions of travel (see Fig. 5).

2.4.2 Implicit Features

The following features are not properties of the map, but have implicit impact on the population density of an area, drivers' behaviors and the travel patterns. We proposed 8 features in this category:

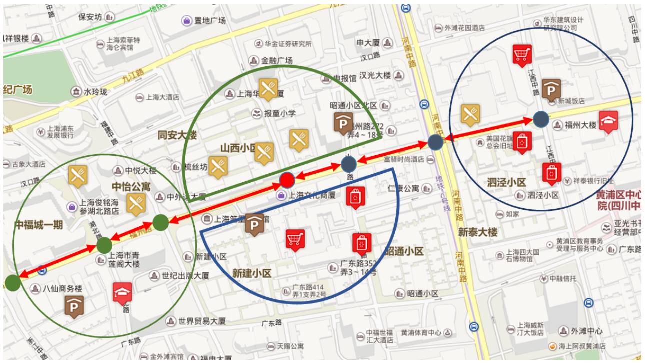


Figure 5: Local Geospatial Features

- the time of the day (in hours);
- weekday or weekend;
- the 1-, 2- and 3-bedroom apartment rental price in the area;
- the temperature in the day and night; and
- the weather condition of the time (i.e., sunny, cloudy, shower or rainy, etc.).

2.5 Model Training

We adopt a supervised training approach here where the training data are feature vectors extracted for road segments and the labels are the traffic conditions (green, yellow, etc.). One obvious challenge for our problem is that the classes are extremely imbalanced. That is, there's probably an order of magnitude more instances of green labels than the yellow labels, and similarly between yellow and red labels, because over a long period of time, normal traffic should dominate the the road network.

We train our classifier on LIBLINEAR [7]. Our multi-class SVM is actually an ensemble of 4 one-versus-rest binary SVMs. We convert categorical attributes such as weather into numerical values. We use a 1-hot m-dimensional binary vector to represent an m-category attribute. Further, we linearly scale each attribute in both training and test data to the range [0, 1].

To counter the data imbalance issue, we add a penalty for mis-classification to each class [12], which minimizes the following:

$$\min \left(\frac{1}{2} w \cdot w + C^{+} \sum_{i|y_i=+1}^{l} \xi_i + C^{-} \sum_{i|y_i=-1}^{l} \right)$$
$$s.t.\ y_i(w \cdot \Phi(x_i) + b) \geq 1 - \xi_i \qquad (1)$$
$$\xi_i \geq 0, \quad i = 1 \ldots l$$

where C^+ and C^- are the weights for the positive and the negative classes, respectively. In this method, the SVM soft margin objective function is modified to assign two misclassification costs, such that C^+ is the misclassification cost for postive class example, while C^- is the misclassification cost for negative class examples, as given in the following formula. here we also assume positive class to be the minority class and negtive class to be the majority class.

The misclassification penalty for the minority class is chosen to be larger than that of the majority class. Essentially this is equivalent to oversampling the minority class.

3. EVALUATION

[4] This is calculated by averaging from three pairs of consecutive crossroads starting from the road segment in question.

This section first introduces the datasets for the evaluation and then present four experiments. The first one compares the performance of two models specifically targeting data imbalance on Huangpu district. The second examines the effectiveness of our features. The third experiment evaluates the same area traffic predication by comparing our model with Baidu Map prediction. The last experiment evaluates the cross-region prediction.

3.1 Datasets

Our training and test data are the map features and live traffic data of four districts of Shanghai, China: Huangpu (HP), Changning (CN), Baoshan (BS) and Minhang (MH). The first two districts are smaller but populous downtown districts with more business and commercial destinations; the rest are suburban districts with larger area and contain more industrial and residential locations.

In this work, we consolidate Baidu's POI classification into 37 types, and use bounded POI search API to query all the POIs of all types. Table 1 shows the number of POIs of some types in four areas. As Minhang and Baoshan district are larger than Huangpu and Changning district, the total numbers of POIs are higher in those areas. As a urban center district, Huangpu has lots of shopping places considering the area. Also, there are significantly more factories in these district. It is shown that our selection of the datasets is sufficiently diverse.

Table 1: An overview statistics of POI data

	HP	CN	MH	BS
Residential	13173	14451	17884	20559
Business	12965	11465	21753	13716
Shopping	7959	4734	11574	7103
Factory	44	44	565	671
Other	16718	15006	30699	20399
Total	50859	45700	82475	62448

The traffic information and other non-geospatial features such as weather and rental prices were crawled from May 19 to May 25 and from May 28 to June 1, 2016. The first 7 days of data is used for training while the remaining 5 days are used for testing.

Our dataset is intrinsically imbalanced. Table 2 shows the class distribution in the ground truth data of all four districts.

Table 2: Imbalanced distribution of classes

	Green	Yellow	Red	Deep Red
Support	3,481,345	174,106	41,912	3808

Due to the imbalance nature of this data, we consider F1-score for each class label as well, rather than only overall accuracy for predicting.

3.2 Model Evaluation

We compare SVM+w with vanilla SVM model for predicting traffic of Shanghai Huangpu district.

Table 3 shows the F1 for predicting green, yellow, red and deep red classes.

SVM+w can be trained much more efficiently, the prediction F1-score, except for the green class, is always higher. Therefore, we eventually choose the SVM+w as our primary model for its outstanding performance and its fast training process.

Table 3: F1-scores for predicting Huangpu district

Model/Class	Green	Yellow	Red	Deep Red
SVM	**0.9582**	0.0065	0.0000	0.0000
SVM+w	0.9367	**0.1587**	**0.1191**	**0.0049**

3.3 Feature Assessment

To illustrate the effectiveness of our features, we train four models with each of our features(road density, local POI distribution and rental price). And then we individually turn off one of them, and see the change in F1-score of the 4 classes. The results are shown in Fig. 6, in which we can obviously conclude that each of our features are effective, especially POIs.

The reason why POIs is a very important feature to describe the traffic situation is that it represents the popularity of certain road segments. Apart from that, we can find that other local features are also very helpful in training models, for the fact that local features like `roadType` and `roadDensity` are good indicators of traffic status.

3.4 Same-area Prediction

Baidu Map also has a traffic prediction system but it can predict traffic for a certain area only after gathering enough amount of historical data for the area. We sampled the Baidu's prediction at a historical time point (at one-hour interval) and use this as a kind of baseline.

Using the Huangpu district as an example, we first show the effect of date and time on the average prediction accuracy over 5 days for all methods.

Fig. 7 shows the accuracy results which compares our weighted SVM model with the Baidu Map's system. We can see that prediction accuracy at rush hours (9:00 and 17:00) is generally lower than other times of the day in both two results. This is because rush hour traffic is more volatile and exceptional events such as accidents may happen at different locations, which makes it hard to predict.

As a result, we reached an accuracy of 87.425% while Baidu Map has an accuracy of 82.798%.

In Table 4, we could see another evaluation result of our model compared Baidu's system. Although the F1-score of our system is a little bit lower than Baidu's, but our recall value is quite higher than Baidu's. Considering the application of the traffic prediction system, the recall value is a more important indicator: It can cause severe problems on user experience when the system tells users a certain road is clear, but actually the traffic is very slow.

3.5 Cross-area Prediction

In this experiment, we train our model from training data of four different districts and use them to test the other districts. Table 5 documents the F1-score on four different traffic classes. As a comparison, we also show the results of

Table 4: Comparison of scores on red and deep red class of Baidu Map prediction with our system

	Baidu Map			Our System		
	P	R	F1	P	R	F1
Red	0.255	0.110	0.154	0.096	0.157	0.119
Deep Red	0.000	0.000	0.000	0.003	0.233	0.005

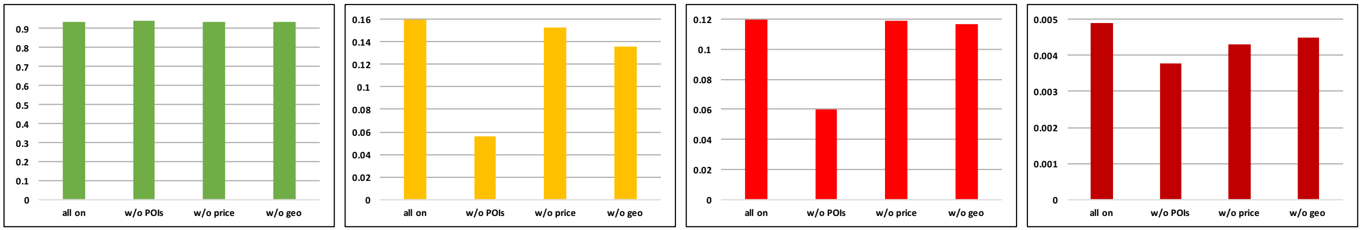


Figure 6: F1-score with some features turned off for traffic conditions green, yellow, red and deep red

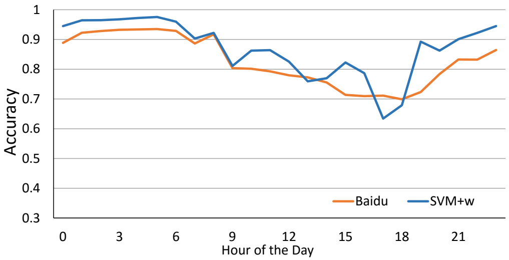


Figure 7: Prediction accuracy in a day for Huangpu District

same-district prediction as well. From the table, we could see that in most cases our model yield good prediction results on cross-region prediction.

In summary, we considered our cross-region prediction to be effective at predicting other geographically or characteristically similar areas without historical traffic data.

Table 5: F1-score of cross-area prediction for 4 classes. T:Test data, M:Training data

Green

M T	HP	CN	MH	BS
HP	0.9367	0.8166	0.7029	0.6854
CN	**0.9443**	0.9454	0.8803	0.8017
MH	**0.9763**	0.9613	0.9691	0.9559
BS	**0.9689**	0.9529	0.9515	0.9662

Yellow

M T	HP	CN	MH	BS
HP	0.1587	**0.1705**	0.1124	0.1248
CN	0.0509	0.1841	0.0970	0.0825
MH	0.0007	0.0549	0.0798	**0.0721**
BS	0.0040	0.0530	0.0458	0.0876

Red

M T	HP	CN	MH	BS
HP	0.1191	0.0440	0.0301	0.0277
CN	0.0426	0.1096	**0.0831**	0.0429
MH	0.0100	0.0387	0.0841	0.0406
BS	0.0217	0.0414	0.0581	0.1124

Deep Red

M T	HP	CN	MH	BS
HP	0.0049	0.0002	0.0011	0.0002
CN	0.0027	0.0287	0.0013	0.0015
MH	0.0000	0.0043	0.0241	0.0022
BS	0.0000	0.0005	0.0290	0.1002

4. RELATED WORK

In this section, we'll review the related work on traffic prediction, data imbalance problem, transfer learning.

4.1 Traffic Prediction

Much work has been done on predicting traffic conditions. Most commercial mapping services such as Google Map and Baidu Map offer the functionality to predict traffic status of major roads in future, often based on the historical traffic data in the past. Prediction of brand new area with no historical data is rare.

Research work has been done to tackle this problem from different aspects. The previous traffic prediction methods can be divided into two categories mainly: Simulation Models and Machine Learning Techniques.

Simulation Models

Based on observed traffic data, Clark et al.[4] proposes a non-parametric regression model to predict traffic. Bierlaire [1], described a real time dynamic traffic assignment system that provides traffic prediction and other services like travel guidance. Their microscopic simulation models are both based on trajectories of individual vehicles to simulate overall traffic data and further prediction. In some other papers [17], researchers use traffic data from GPS-equipped taxicabs to estimate the traffic flow of certain road segments.

The main limitation of such studies is that they all rely on very sporadic observations so that they are often restricted to synthetic and simplified data for simulation. Also, these simulation models cannot be applied to other road segments where there isn't any historical traffic data.

Machine Learning Techniques

A very common idea is to consider changes to traffic condition on any stretch of road as a simple time series. Much valuable work was done along this line. Since early 1980s, univariate time series models, such as Auto-Regressive Integrated Moving Average(ARIMA)[2] and Exponential Smoothing(ES) models [11], have been widely used in traffic prediction. In the last decade, more researchers turned to Neural Network(NN) models in forecasting of various traffic parameters such as speed[8], estimated travel time[10] and traffic flow.

More recently, researchers have attempted to solve similar problems by proposing complex networks [6, 15], machine learning [14, 18] and even deep learning methods [3].

However, there are two main shortcomings in most of these models: The first one is that most of them are still based on the naive idea. Many of them still treat traffic flow as univariate time-series data and ignore all the other important information. In our model, we included many local features like weather, house pricing, POI and so on.

4.2 Data Imbalance Problem

Real-world datasets are predominately composed of normal examples with only a small percentage of abnormal or important examples, which will cause misclassified problems. As our problem is based on real-world traffic information, the class imbalance is a serious issue. For our case, there would be more data representing the class of clear traffic than other classes, often to large ratio.

Previous researchers have done a lot in this field. One approach is to assign distinct costs to training examples. The other is to re-sample the original dataset by under-sampling and up-sampling. Under-sampling of the majority class is a good way to increase the sensitivity of a classifier to the minority class. But it doesn't improve minority class recognition. And the general idea of the cost function based approaches is that we think one false negative is worse than one positive, in other words, we give more weights to the false negative than false positive, so the machine learning algorithm will try to make fewer false negatives compared to false positives. In case of SVM, different classes can have different weights on them, resulting desired loss penalty.

4.3 Transfer Learning

Many machine learning are based on the assumption that training and future data are in the same future space and have the same distribution. However, in real-world applications, there are many cases that we have a classification task in one domain while the training data in another domain. One of the best ways to solve this problem is transfer learning.

Transfer learning has been studied by researches for a long time to solve this problem. Semi-supervised classification [19] addresses the problem that the labeled data may be too few to build a good classifier, by making use of large amount of unlabeled data together with a small amount of labeled data. Variations of supervised and semi-supervised learning for imperfect data sets have also been studied on how to deal with the noise problems. Pan et al [13] categorize transfer learning into three sub-settings, inductive transfer learning, transductive transfer learning and unsupervised transfer learning.

In our task, we get the training data from Baidu Map. Because there are some rural places where Baidu Map cannot provide their traffic conditions, we lack the training data of these regions. So we need to do transfer learning. We use data that those areas have similar features as these to be the training data to do transfer learning.

5. CONCLUSION

This paper shares our experience of developing a system capable of learning from both geospatial and non-geospatial features from one part of a city and predicting the traffic condition at any time for another part in China. The system is trained from crawled live traffic data and POIs from Baidu map and shows the prediction results on a web demo based on OpenStreetMap platform. This type of cross-area prediction is very useful when no historical traffic data for a place is available, such as in urban planning.

We address the multi-class imbalance issue by using a weighted linear SVM as our primary model, and achieves prediction accuracy on par with the prediction given by Baidu Map itself, which possesses larger and finer grained data such as traffic speed.

6. REFERENCES

[1] M. Bierlaire. DynaMIT: a simulation-based system for traffic prediction and guidance generation. In *TRISTAN III*, 1998.

[2] G. Box. *Box and Jenkins: Time Series Analysis, Forecasting and Control*. Palgrave Macmillan UK, 2013.

[3] K. Y. Chan, T. S. Dillon, J. Singh, and E. Chang. Neural-network-based models for short-term traffic flow forecasting using a hybrid exponential smoothing and levenberg-marquardt algorithm. *IEEE TITS*, 13(2):644–654, 2012.

[4] S. Clark. Traffic prediction using multivariate nonparametric regression. *J. of Trans. Eng.*, 129(2):161–168, 2003.

[5] S. Coast. Openstreetmap, 2004.

[6] S. Çolak, A. Lima, and M. C. González. Understanding congested travel in urban areas. *Nature communications*, 7:10793, 2016.

[7] R.-E. Fan, K.-W. Chang, C.-J. Hsieh, X.-R. Wang, and C.-J. Lin. LIBLINEAR: A library for large linear classification. *JMLR*, 9:1871–1874, 2008.

[8] S. Ishak, C. Alecsandru, and G. Student. Optimizing traffic prediction performance of neural networks under various topological, input, and traffic condition settings. *J. of Trans. Eng.*, 130(4):452–465, 2004.

[9] G. Lee. Eviltransform. https://github.com/googollee/eviltransform/, 2015.

[10] J. W. C. V. Lint, S. P. Hoogendoorn, and H. J. V. Zuylen. Accurate freeway travel time prediction with state-space neural networks under missing data. *Transportation Research Part C Emerging Technologies*, 13(5-6):347–369, 2005.

[11] R. S. Marshment, R. C. Dauffenbach, and D. A. Penn. Short-range intercity traffic forecasting using econometric techniques. *ITE Journal*, 66(2):37,40–43, 1996.

[12] E. Osuna, R. Freund, and F. Girosi. Support vector machines: Training and applications. 1997.

[13] S. J. Pan and Q. Yang. A survey on transfer learning. *IEEE TKDE*, 22(10):1345–1359, 2010.

[14] Y. Qi and S. Ishak. A Hidden Markov Model for short term prediction of traffic conditions on freeways. *Transportation Research Part C: Emerging Technologies*, 43:95–111, 2014.

[15] Y. Ren, M. Ercsey-Ravasz, P. Wang, M. C. González, and Z. Toroczkai. Predicting commuter flows in spatial networks using a radiation model based on temporal ranges. *Nature communications*, 5:5347, 2014.

[16] Y. Tang and L. I. Ying-Zhen. Geometric construction method of linear svm multi-class classifier. *Computer Engineering*, 38(04):152–154, 2012.

[17] J. Yuan, Y. Zheng, X. Xie, and G. Sun. Driving with knowledge from the physical world. In *SIGKDD*, pages 316–324, 2011.

[18] R. Zhang, Y. Shu, Z. Yang, and P. Cheng. Hybrid traffic speed modeling and prediction using real-world data. In *IEEE International Congress on Big Data*, 2015.

[19] X. Zhu. Semi-supervised learning literature survey. *Computer Science*, 37(1):63–77, 2008.