CLASSIFICATION APPROACH FOR BIG DATA DRIVEN TRAFFIC FLOW PREDICTION USING APACHE SPARK

Riya Gandewar[1], Anupama Phakatkar[2]

[1]Student, Dept. of Computer Engineering, PICT college, Pune, Maharashtra
[2] Professor, Dept. of Computer Engineering, PICT college, Pune, Maharashtra

---***---

Abstract - Traffic problems are crucial issues in the rapidly developing society. Traffic flow prediction is an important problem in Intelligent Transportation Systems. Over the last few years, traffic data have been exploding, and we have truly entered the era of big data for transportation. In big data driven traffic flow prediction systems, accuracy and timeliness affects the robustness of prediction performance. Existing traffic flow prediction methods uses different classification approaches, dealing with accuracy and processing time problems. To overcome these problems, the system uses K-Nearest Neighbors (KNN) for classification and Convolution Neural Network (CNN) for prediction of traffic flow. The KNN is used to find out the patterns of route, that's how much time should be required from one destination to another. The CNN is used to predict the traffic flow in the particular route. The traffic flow data processing is done in Apache Spark. The output of the proposed system is, routes which have minimum predicted traffic flow. It is observed that, the proposed system for traffic flow prediction has superior performance.

***Key Words*: Big Data, Classification, Prediction, Hadoop, Apache Spark**

1.INTRODUCTION

Accurate and timely traffic flow information are strongly needed for individual travellers, business sectors, and government agencies. Traffic flow prediction has gained more and more attention with the rapid development and deployment of Intelligent Transportation Systems (ITSs). It collects traffic data such as traffic volume and speed on every road and provide statistical summary services, usually on traffic congestion [15]. Traffic congestion effects on Vehicular queuing, travel time, cost, fuel consumption, pollution in the environment. In a big city, the change of traffic flow has a large impact on people's daily life, such as the route selection for drivers.

The big data generated by the Intelligent Transportation Systems are worth further exploring to traffic management. The introduction and development of the Intelligent Transport System have resulted in more reliable traffic information gathering, analyzing, and processing, thereby providing more time-relevant and precise traffic analysis and prediction to users.

Traffic flow disruptions can be categorized as predictable and unpredictable. Predictable disruptions include traffic signals, stop signs, public transit services, scheduled sport events, music concerts, road constructions etc. Unpredictable disruptions include auto mobile accidents, breakdowns, and emergency road closures.

The impact of disruption to traffic flow depends on the location, the duration of the disruption [9]. Due to unpredictable disruptions, long-term forecasting may not be accurate enough for practical use. However, short-term traffic prediction, if properly done, may reach an accuracy level that is useful for several applications, e.g., we may just want to know the traffic flow volume within 10 min to 30 min for deciding our route while driving.

Traffic flow conditions are extremely uncertain in current complex transportation network situation, due to the heterogeneous and dynamic nature of traffic to nonlinear interactions between drivers and environments. Moreover, the traffic state of a specific location is highly influenced by its upstream and downstream traffic conditions besides the traffic conditions in the past and future periods, and thus the spatial and temporal correlations are the inherent features of traffic flow. Most of traditional classification approaches take the traffic flows as the individual and independent instances, which do not consider the correlation information among traffic flows.

In the last several decades, there have been many studies for predicting short term traffic flow. Recently, there have been various traffic flow prediction systems, models and algorithms using statistics-based approaches and computational intelligence-based approaches.

Despite the popularity, existing work tends to suffer from the various problems. Firstly, in a city where traffic jam occurs, the traffic flows at different locations will influence each other. In a complicated traffic network, the influences between different locations are decided by traffic network structure, signal light, commercial layout, etc. Secondly, it requires the storage of the whole training set which would be an excessive amount of storage for large data sets and leads to a large computation time in the classification stage. The existing methods built the model based on time-series data regardless of spatial influences. Most of the existing studies, adopt few useful features, which cannot provide sufficient information for forecasting.

In recent years, deep learning has drawn a lot of attention on the tasks of classification, natural language processing, dimensionality reduction, object detection, motion modeling, etc. The method first uses the deep network to rebuild input features, and then train neural network for predicting the traffic flow [12]. The proposed system uses a convolution neural network approach under Apache Spark on Hadoop platform. Convolution Neural Network (CNN) is used to forecast traffic flow and KNN is used to find the spatial influences that exist in locations. The input to the method is historical traffic flow data from target location and historical data about other nearby locations. These data are grouped together to build a feature matrix for prediction. This proposed approach for real-time traffic flow prediction has superior performance by speeding up the classification process and reducing the storage requirements and processing time. More importantly, with reasonable execution time, the classification performance can be significantly improved by flow correlation information, even under the extremely difficult circumstance of very large training samples.

2. RELATED WORK

Researchers have been trying to make traffic flow forecasting more accurate since last few years. Based on the length of time interval, there are long-term forecasting and short-term forecasting.

Long-term forecasting indicates making a prediction by month or by year. The volume of traffic flow is large and relatively stable, and it is slightly affected by the daily accident. For example, Zhong sheng Hou et al. [6] proposed a method based on ARIMA to forecast traffic flow in a month after they revealed that their time series data had the long term trend and the fluctuations at the 12-hour time scale.

For short term forecasting, auto regressive integrated moving average (ARIMA) models, and artificial neural network (ANN) [1] models are widely exploited. Xiangjie Kong et al. [11] proposed a plan moving average algorithm for utilizing previous days historical data. In addition, some researchers have compared the seasonal ARIMA (SARIMA) model, which is a TSA model, and showed its excellent prediction performance. Fuying Yu et al.[9] compared SARIMA and nonparametric (data-driven regression) models. The authors in this research concluded that the SARIMA model has better performance than the nonparametric regression models.

H. Hu et al.[14] additionally compared time-series analysis methods and SVR models and concluded that SARIMA showed the best performance. In addition, there are some integration models for this task. For example, Su Yang et al. [2] proposed a 3-stage model which integrates ARIMA and ANN, which uses the ARIMA forecasting data as a part of the input of ANN. X. Chen et al. [16] studied an ensemble model which consists of a statistical method and a neural network bagging model.

Yuqi Wang and Wengen Li proposed method for predicting traffic congestion correlation between road segments on GPS trajectories [3]. Method extract various features on each pair of road segments from road network. The result of this is input to the several classifiers to predict congestion correlation. It use classifiers like decision tree, Logistic regression, Random forest and Support vector machines.

Besides, researchers also consider the use of integrated data. Hao-Fan Yang et al. [4] proposed a Gaussian mixture model clustering (GMM) method to partition the data set for training ANN. Deep learning methods have also gained a lot of attention recently. Jinyoung Ahn et al. [5] used a stacked autoencoder (SAE) to learn generic traffic flow features. Y.Lv et al. [12] applied a deep belief networks model in traffic flow prediction, which adopts multitask learning to reduce the error.

To sum up, all the above-mentioned methods have many desirable properties in different disciplines, and thus it is hard to conclude that which one of these is significantly superior to other methods in any situation. One of the best essential reasons is that the accuracy of prediction models which are developed with small scale separate specific traffic data depends on the traffic flow features embedded in the collected traffic data. Furthermore, most of the existing models are performed in stand-alone models, and thus the computational effort is expensive [7] and the capability of data processing and storage is restricted. However, researchers also developed a general architecture of distributed modeling in a MapReduce framework for traffic flow forecasting[8], to efficiently process large-scale traffic data on a Hadoop platform.

As of now, there is diversified research in all the techniques and platform. The proposed work suggests, Traffic flow prediction of large volume traffic data using a classification approach in Apache Spark.

3. PROPOSED SYSTEM

3.1 Proposed Architecture

Fig. 1 shows architectural design of proposed system. Following are important modules in the system :

Storage System : Apache HDFS is used as an underFS storage system. User's CSV files, trained models, testing and training data set stored in HDFS. Data processing is done through Apache Spark.

Computation Framework : Apache Spark is used for computation. Data classifier and predictor is built in Spark. It access traffic flow data in CSV format. Spark's Mlib is used for training KNN and CNN model.

Preprocessing : The module fetches data according to user's query, to analyze the data and to find the minimum average flow, average speed and average journey time.

Data Classifier and predictor : The input of K-Nearest Neighbor model is user's current location, user's destination location and timestamp. It gets the current time and other related values. It collects all parameters such as traffic flow, average speed. Distance is calculated from source to destination considering different routes. The top k routes with their respective parameters are fetched from this. The input to the CNN is traffic flow, average speed, average journey time. All routes are fetched from KNN and calculate weight for each route. Generate a hash map and store each link and traffic flow in it. From polling layer, more accurate values will be calculated. Store all list into result.

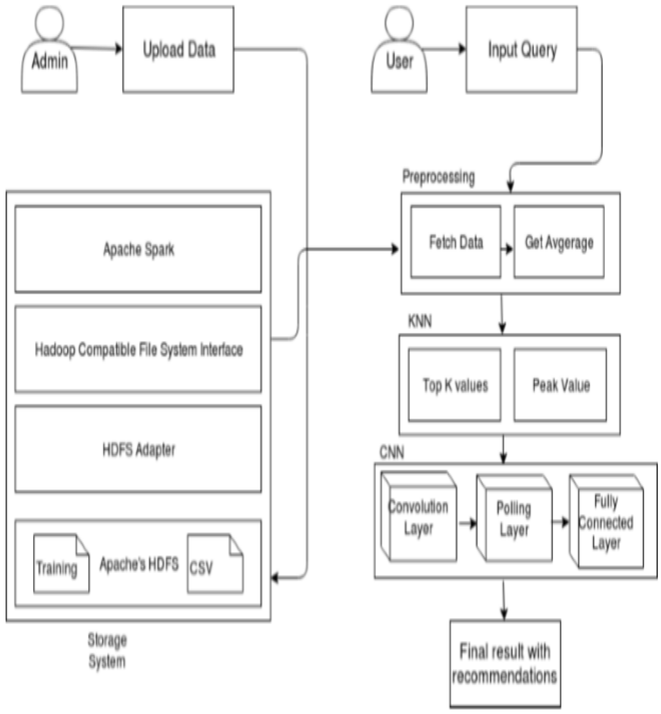

Fig 1 : Traffic Flow Prediction System architecture

Web Application : User registers for traffic flow prediction system using a web application, providing their own details. After successfully login to site, user can search for places where they want to go at a particular time, to know the traffic flow at that particular place. After analyzing data, the system gives output as different routes considering the input and traffic flow of that place.

3.1 System Algorithm

1. KNN CLASSIFICATION ALGORITHM

```
Input: User's current location, User's destination location, Timestamp
Output: Link description collection with path details KNN [].
Step 1: Get current time and other related values.
Step 2: Collect all average total values and traffic flow.
Step 3: Calculate distance using below formula.
$L=W_{i1},W_{i2},W_{i3},\ldots\ldots,W_{in}$ Weight of each list
$C = c_1,c_2,\ldots\ldots,c_n$ clusters of each list
Step 4: UrlList = URL1(w), URL2(w).. URLn(w)
Step 5: Add each UL to KNN [].
Step 6: Return KNN [].
```

2. CNN PREDICTION ALGORITHM

```
Input: Output list of KNN [].
Output: Single or double Link description with path details
Step 1: For each (Link k to KNN)
Step 2: Get each object KNN[k]
Step 3: Calculate each object weight
Step 4: Generate hashmap <double, string>
Store each link into Map <x, object>
end for
Step 5: Polling layer : SortMap(Map)
Step 6: Find first 3 objects from list
Step 7: Store all list into Res[].
Step 8: Return Res[].
```

4. RESULTS

4.1 Performance Evaluation

The performance of the algorithm is measured on following parameters:

1. Accuracy
2. System Response Time

We use two widely employed evaluation measures to assess the forecast performance:

The Root Mean Squared Error (RMSE) is a way to measure the average error of the forecasting results and is calculated by

$$RMSE = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(x_t - x_p)^2}$$

Here, xt and xp are the actual and the forecast values. n is the total number of locations.

The Mean Relative Error (MRE) is a way to measure the proportional error of the forecasting results and is calculated by

$$MRE = \frac{1}{n}\sum_{i=1}^{n}\frac{(x_t - x_p)}{x_t} * 100\%$$

4.2 Analysis and Discussion

Lower the RMSE higher the system accuracy, Fig. 2 shows Root Mean Squared Error of Single CNN, DGCNN and Proposed System from which we can conclude that our proposed system have higher accuracy than the existing system.

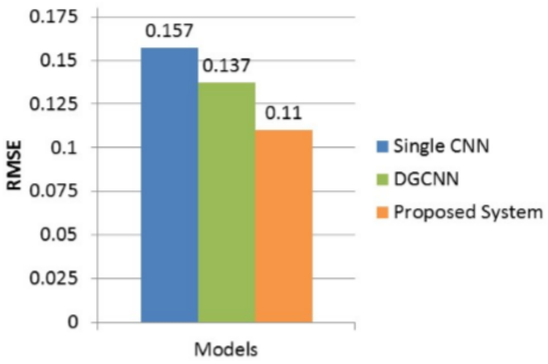

Fig 2 : RMSE for Single CNN, DGCNN and Proposed System

Lower the MRE higher the system accuracy, Fig. 3 shows Mean Relative Error of Single CNN, DGCNN and Proposed System from which we can conclude that our proposed system have higher accuracy than the existing system.

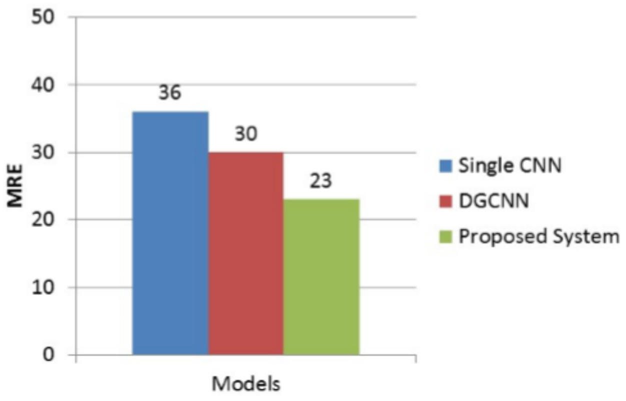

Fig 3 : MRE for Single CNN, DGCNN and Proposed System

5. CONCLUSION

A system is proposed to predict the traffic flow from historical traffic flow data. Whenever user search for a route, then only that data will be extracted from historical traffic flow data. The K-Nearest Neighbor algorithm is used for finding K-Nearest Neighbors from source to destination. The Convolution Neural Network algorithm is used to predict traffic flow from those nearest neighbors. The system gives output as predicted traffic flow with their respective routes. Proposed system improves the accuracy of traffic flow prediction system and reduces the prediction time by a factor of 2.5x .

REFERENCES

[1] Fuying Yu , Zhijie Song, “The Short-Term Traffic Flow Prediction Method Based on Detectors PSO Algorithm”, Sixth International Conference on Intelligent Systems Design and Engineering Applications, 2015, pp. 890-893.

[2] Su Yang, Shixiong Shi, Xiaobing Hu, Minjie Wang , “Discovering Spatial Contexts for Traffic Flow Prediction with Sparse Representation based Variable Selection”,IEEE international conference on transportation, 2015, pp. 364-367.

[3] YuqiWang , Jiannong Cao ,Wengen Li and Tao Gu ,“Mining Traffic Congestion Correlation between Road Segments on GPS Trajectories”, IEEE International Conference on Smart Computing, 2016, pp. 1-8.

[4] Hao-Fan Yang, Tharam S. Dillon, Life Fellow, and Yi-Ping Phoebe Chen, “Optimized Structure of the Traffic Flow Forecasting Model With a Deep Learning Approach”, IEEE transactions on neural networks and learning systems, vol.89, 2016, pp. 1-11.

[5] Jinyoung Ahn, Eunjeong Ko, Eun Yi Kim, “Highway Traffic Flow Prediction using Support Vector Regression and Bayesian Classifier”, IEEE International Conference Big Data on Smart Computing, 2016, pp. 239-244.

[6] Zhongsheng Hou, Senior Member, IEEE, and Xingyi Li, “Repeatability and Similarity of Freeway Traffic Flow and Long-Term Prediction Under Big Data”, IEEE transactions on intelligent transpiration system, vol. 17, 2016, pp. 1786-1796.

[7] Jiwan Lee , Bonghee Hong , Kyungmin Lee and Yang-Ja Jang , “A Prediction Model of Traffic Congestion Using Weather Data”, IEEE International Conference on Data Science and Data Intensive Systems, 2015, pp. 81-88.

[8] Zhiyuan Ma and Guangchun Luo, “Short Term Traffic Flow Prediction Based on On-line Sequential Extreme Learning Machine”, 8th International Conference on Advanced Computational Intelligence, 2016, pp. 143-149.

[9] Fuying Yu , Zhijie Song, "A MapReduce-Based Nearest Neighbor Approach for Big-Data-Driven Traffic Flow Prediction", Sixth International Conference PICT, Department of Computer Engineering

on Intelligent Systems Design and Engineering Applications, vol.4, 2015, pp. 890-893.

[10] Hong-jun Yang , Xu Hu, "Wavelet neural network with improved genetic algorithm for traffic flow time series prediction", Elsevier International Journal for Light and Electron Optics, 2016, pp. 8103-8110.

[11] Xiangjie Kong , Zhenzhen Xu , Guojiang Shen , Jinzhong Wang , Qiuyuan Yang, Benshi Zhang, "Urban traffic congestion estimation and prediction based on floating car trajectory data", Elsevier future generation computer system, vol. 6, 2016, pp. 97-107.

[12] Y. Lv, Y. Duan, W. Kang, Z. Li, and F.-Y. Wang, "Traffic flow prediction with big data: A deep learning approach", IEEE Transaction Intelligent Transportation System, vol. 16, 2015, pp. 865-873.

[13] Dawen Xiaa, Binfeng Wanga, Huaqing Lic, Yantao Lia, Zili Zhang, "A distributed spatial temporal weighted model on MapReduce for short-term traffic flow forecasting", Neurocomputing, vol. 179, 2016, pp. 246-263.

[14] H. Hu, Y. Wen, T.S. Chua, and X. Li, "Toward scalable systems for big data analytics: A technology tutorial", vol.2, IEEE Access, 2014, pp. 652-687.

[15] J. Zhang, F.Y. Wang, K. Wang, W.-H. Lin, X. Xu, and C. Chen, "Data- driven intelligent transportation systems: A survey", vol. 12, IEEE Transaction Intelligent Transportation System, 2011, pp. 1624-1639.

[16] X. Chen and X. Lin, "Big data deep learning: Challenges and perspectives", vol. 2, IEEE Access, 2014, pp. 514-525.

[17] Highways England network journey time and traffic flow data[Online]. Available: https://data.gov.uk/dataset/highways-england-network-journey-time-andtraffic-flow-data