IOSR Journal of Engineering (IOSRJEN) **www.iosrjen.org**
ISSN (e): 2250-3021, ISSN (p): 2278-8719
Vol. 08, Issue 8 (August. 2018), ||V (IV) || 40-49

Location Prediction Model Using Foursquare Check-In Service: A Machine Learning Approach

Dr.G.Anandharaj[1], P.Monisha[2]
[1]*(Assistant Professor, Department of Computer Science and Application, Adhiparasakthi college of arts and science (Autonomous), India) (younganand@gmail.com)*
[2]*(M.Phil (CS) Research Scholar, Department of Computer Science and Application, Adhiparasakthi college of arts and science (Autonomous), India) (yasminmonisha94@gmail.com)*
Corresponding Author: Dr.G.Anandharaj1

Abstract : Location prediction is broadly used to figure out user's next place to visit in light of his/her versatility logs. A stunning accessibility of smart carrying and constantly online capacities, these days' lots of clients is getting used to 'check-in' their portability exercises on numerous prevalent services and applications. , i.e., report their location and audits from their PDAs or other computerized accessories. Delegate check-in applications like Foursquare. In these a basic issue in location information processing, invaluable for reconnaissance, business, and individual applications. It is extremely challenging because of the sparsity issues of check-in information and Heterogeneousness i.e., the location, content and transient information. A regularly overlooked issue in ongoing investigations is the assortment crosswise over various check-in situations, which is becoming more critical because of the increasing accessibility of more location check-in applications. Check-in records profile users mobility activities has been investigated in a different angle, i.e., global feature combination, for better clarification and combination. We propose a spatial Context Feature, Collaboration Feature, Content Feature model, which manages the location prediction of mobile users. Such model is utilized for the arrangement of the clients' directions through Machine Learning (ML) algorithms. Predicting spatial context is dealt with through supervised learning. We assess our show as far as prediction precision as for particular prediction parameters. The proposed model is additionally contrasted and other ML algorithms for location prediction.

Keywords - Location Prediction, Geographical Closeness, Trajectory Data, Check-in Behavior Analysis, machine learning.

Date of Submission: 06-08-2018 Date of acceptance: 23-08-2018

I. Introduction

WE have seen a stunning accessibility of savvy gadgets as of late. With these gadgets' simple carrying and constantly online capacities, these days heaps of clients are getting used to 'check-in' their versatility exercises on numerous prominent administrations and applications, i.e., report their location and audits from their advanced mobile phones or other computerized adornments. Delegate check-in applications include Foursquare, Yelp, and general social stages like Facebook, Twitter and Weibo.

We pick an internet based life check-in dataset and picture these clients' check-in locations. It is clear that there displays an amazing check-in design crosswise over prominent spots like CBD, air terminal and tech parks. Clients' check-in exercises exhibit a one of a kind edge into their life, and the dissemination designs mirror their interest and inclination. As of late, the estimation of check-in information have been exhibited in numerous applications, including however not constrained in versatile advertising, advancement suggestion, activity administration and social reconnaissance. Behind the check-in information processing, location prediction is a key errand. In any case, it is exceptionally challenging because of the check-in information's inherent attributes. To start with, Sparsity: There is an extensive conceivable space which clients can visit, however in actuality they just cover a little arrangement of the spots. Second, Heterogeneousness: Location information comprises of various types of highlights, i.e., the location, content and fleeting information.

The ongoing emerging pattern of more check-in applications makes the prediction issue complex. It prompts the third and normally overlooked yet becoming more troublesome test, which we will center in this article, i.e, Variety. The check-in exercises in various applications are not following similar examples. A growing dataset of various types of clients' directions and portability records wind up accessible. For instance, neighborhood ins in particular App is intense in clients' home and office routines. Interestingly, head out logs can cover a considerably bigger extension and typically exceptionally scanty.

We pick three kinds of check-in situations, i.e., Foursquare as nearby check-in, smaller scale blog as log sharing and Yelp as place reviewing. The time/separate hole of continuous check-ins in a check-in log is accumulated and thought about. The check-ins in miniaturized scale blog and Foursquare take after comparable examples. They display varying yet littler separation traverse, and the time hole is shorter, contrasted and Yelp surveys check-in remove hole, the smaller scale blog check-in has marginally however amazingly longer time hole. Diverse time holes uncover varying portability examples of interest or inclination.

Despite the fact that some ongoing work focus on these check-in conduct contrasts, the inherent assortment between this check-ins and their effect to location prediction is normally missing. Cutting edge location prediction techniques can be classified into three lines, i.e., community oriented component based, spatial element based, and combination include based. Cooperative component based methodologies use the Collaborative Filtering or factorization techniques to adapt to the sparsity challenge however they are touchy to spatial highlights. Interestingly, spatial component based strategies utilize the gravity and area closeness estimations to fit into the location setting, yet are normally hard to sum up the worldly and other related highlights. The component combination approach demonstrates the upside of highlight combination to convey enhanced exactness.

The proposed Machine Learning based GALLOP prediction approach isn't just broad over various check-in situations yet in addition far reaching of various highlights. In the context highlight, we plan a various granularity model to profile the topographical closeness. We select the anticipated applicants in view of the combination of neighborhood area, nearby city and state scales. The weights of each scale are found out from training information. This approach is like late introduced thickness display; however we enhance it with new indexing and granularity association techniques, guaranteeing its adaptability. In the collective component, we depend on the true factorization strategies to demonstrate the other clients' check-in logs and supplement it with a weighted rendition. In the substance highlight, we change clients' continuous check-in succession, i.e., progress examples of the check-in grouping into a diagram portrayal, and concentrate these traits' closeness with a Irregular Walk with restart display.

II. LITERATURE SURVEY

2.1 D. Lian, C. Zhao, X. Xie, G. Sun, E. Chen, and Y. Rui, "GeoMF: Joint geographical modeling and matrix factorization for point-ofinterest recommendation"

Point-of-Interest (POI) recommendation has become an important means to help people discover attractive locations. However, extreme sparsity of user-POI matrices creates a severe challenge. To cope with this challenge, viewing mobility records on location-based social networks (LBSNs) as implicit feedback for POI recommendation, we first propose to exploit weighted matrix factorization for this task since it usually serves collaborative filtering with implicit feedback better. Besides, researchers have recently discovered a spatial clustering phenomenon in human mobility behaviour on the LBSNs, i.e., individual visiting locations tend to cluster together, and also demonstrated its effectiveness in POI recommendation, thus we incorporate it into the factorization model. Particularly, we augment users' and POIs' latent factors in the factorization model with activity area vectors of users and influence area vectors of POIs, respectively. Based on such an augmented model, we not only capture the spatial clustering phenomenon in terms of two-dimensional kernel density estimation, but we also explain why the introduction of such a phenomenon into matrix factorization helps to deal with the challenge from matrix sparsity. We then evaluate the proposed algorithm on a large-scale LBSN dataset. The results indicate that weighted matrix factorization is superior to other forms of factorization models and that incorporating the spatial clustering phenomenon into matrix factorization improves recommendation performance.

2.2 M. Lichman and P. Smyth, "Modelling human location data with mixtures of kernel densities,"

Location-based data is increasingly prevalent with the rapid increase and adoption of mobile devices. In this paper we address the problem of learning spatial density models, focusing specifically on individual-level data. Modeling and predicting a spatial distribution for an individual is a challenging problem given both (a) the typical sparsity of data at the individual level and (b) the heterogeneity of spatial mobility patterns across individuals. We investigate the application of kernel density estimation (KDE) to this problem using a mixture model approach that can interpolate between an individual's data and broader patterns in the population as a whole. The mixture-KDE approach is evaluated on two large geolocation/check-in data sets, from Twitter and Gowalla, with comparisons to non-KDE baselines, using both log-likelihood and detection of simulated identity theft as evaluation metrics. Our experimental results indicate that the mixture-KDE method provides a useful and accurate methodology for capturing and predicting individual-level spatial patterns in the presence of noisy and sparse data.

2.3 J.-D. Zhang and C.-Y. Chow, "GeoSoCa: Exploiting geographical, social and categorical correlations for point-of-interest recommendations,"

Recommending users with their preferred points-of-interest (POIs), e.g., museums and restaurants, has become an important feature for location-based social networks (LBSNs), which benefits people to explore new places and businesses to discover potential customers. However, because users only check in a few POIs in an LBSN, the user-POI check-in interaction is highly sparse, which renders a big challenge for POI recommendations. To tackle this challenge, in this study we propose a new POI recommendation approach called GeoSoCa through exploiting geographical correlations, social correlations and categorical correlations among users and POIs. The geographical, social and categorical correlations can be learned from the historical check-in data of users on POIs and utilized to predict the relevance score of a user to an unvisited POI so as to make recommendations for users. First, in GeoSoCa we propose a kernel estimation method with an adaptive bandwidth to determine a personalized check-in distribution of POIs for each user that naturally models the geographical correlations between POIs. Then, GeoSoCa aggregates the check-in frequency or rating of a user's friends on a POI and models the social check-in frequency or rating as a power-law distribution to employ the social correlations between users. Further, GeoSoCa applies the bias of a user on a POI category to weigh the popularity of a POI in the corresponding category and models the weighed popularity as a power-law distribution to leverage the categorical correlations between POIs.

2.4 W. Wang, H. Yin, S. Sadiq, L. Chen, M. Xie, and X. Zhou, "SPORE: A sequential personalized spatial item recommender system,"

With the rapid development of location-based social networks (LBSNs), spatial item recommendation has become an important way of helping users discover interesting locations to increase their engagement with location-based services. Although human movement exhibits sequential patterns in LBSNs, most current studies on spatial item recommendations do not consider the sequential influence of locations. Leveraging sequential patterns in spatial item recommendation is, however, very challenging, considering 1) users' check-in data in LBSNs has a low sampling rate in both space and time, which renders existing prediction techniques on GPS trajectories ineffective; 2) the prediction space is extremely large, with millions of distinct locations as the next prediction target, which impedes the application of classical Markov chain models; and 3) there is no existing framework that unifies users' personal interests and the sequential influence in a principled manner. In light of the above challenges, we propose a sequential personalized spatial item recommendation framework (SPORE) which introduces a novel latent variable topic-region to model and fuse sequential influence with personal interests in the latent and exponential space. The advantages of modeling the sequential effect at the topic-region level include a significantly reduced prediction space, an effective alleviation of data sparsity and a direct expression of the semantic meaning of users' spatial activities.

III. PRELIMINARIES

This section first presents the data characteristics of different check-in datasets and then lists the Models used in this paper.

3.1 CHECK-IN DATASET

Three large real check-in datasets are chosen in this paper, they are

• Foursquare: the online check-in records collected within the areas of New York. Foursquare is a unique check-in service and used to report users' position with comments.

• Yelp: the online business review records, supporting user-contributed reviews about local businesses.

• Micro-blog: Here we extract check-in logs from *Sina Weibo*, which is one of the most popular micro-blog platforms. Of its hundreds of millions of users' life records, lots are check-in logs.

Dataset is a collection of data or a single statistical data where every attribute of data represents variable and each instance has its own description. For prediction of disease we used data set for prediction and classification of algorithms in order to compare their accuracy using weka's three interfaces: Explorer, Experimenter and Knowledge Flow.

3.2 MODELS

The check-in records profile users' mobility activities and have several different classes of features. Though this has been investigated in recent studies, here we treat and divide them in a different angle, *i.e.,* global feature combination, for better clarification and combination. We list the features used in this work.

• Context Feature: refers to the spatial dimension. Users' check-in activities are distributed in a spatial scope. Nearby places can contribute to the representation of users' check-in records, especially when the users visit a focused set of places.

• Collaboration Feature: refers to the collaborative dimension. It is contributed from a large group of users' records. Similar to the collaborative filtering formalization in recommendation systems, here the users' check-in activities can be reduced into a latent dimension, representing their collaboration preferences.
• Content Feature: refers to the place dimension. Places are not merely visited by users. These places have inherent attributes, *i.e.*, categories, text descriptions and other kinds of annotations. We discuss how the transition between places reveals users' interest/preference over time. In further, the transition patterns benefit the closeness extraction of places for better prediction. In this paper, we use the categories of the location as its attribute description.

3.3 Problem Statement [Location Prediction]

The location prediction problem is to predict the preference probability $p(l|u)$, *i.e.*, a user u to an unvisited place l. The prediction problem studied in this paper is more challenging than successive POI recommendation. While general POI recommendation focuses on estimating users' preference over POI, successive POI recommendation pays more attention on most recent checked-in locations to deliver satisfied recommendations. The location prediction problem is related to sequential patterns of check-in behaviors and aims to identify the next place to be visited, given the target users.

IV. CLASSIFICATION IN MACHINE LEARNING

Classification is the task of learning to categorize (predict) an unseen example to a discrete class value. An *example* is a $(m+1)$ - dimensional vector e of m attributes e_i, $i = 1, \ldots, m$ and a class attribute e_{m+1}, that is $e = [e_1 \ldots e_m, e_{m+1}]$. The e_i attributes determine the value of the class attribute e_{m+1} of e. An e_i attribute assumes values from the corresponding domain $Dom(e_i)$. The input of a classification algorithm is a set E of s example vectors, $E = \{e_i, i = 1 \ldots s\}$, and the output is a classification model M(E). Such model is capable of predicting / estimating the value of the class attribute of an unseen and yet unlabeled $(m+1)$ - dimensional example vector q based only on the values of its q_i attributes with $Dom(e_i) = Dom(q_i), \forall i$

The performance of a classification algorithm is estimated on the basis of the quality of predictions delivered by the trained models. In order to estimate the classification performance, a test phase is required. In the test phase we employ a test set V that was not used in the training phase. A classification model, M(E), is established through the training set, E, as a result of the learning phase. The produced model M(E) is then applied on the test set V and the correctness of its predictions is assessed and quantified. Prediction accuracy ε is a quantitative measure for the classification performance. It refers to the proportion of the correctly predicted examples $C \subset V$ out of V, i.e., the fraction $\varepsilon = |C| / |V|$, where $|C|$ denotes the cardinality of C. One method to estimate the prediction accuracy of a classification algorithm is a re-sampling method called *crossvalidation* [1, 2]. In n-fold cross-validation, the training set E is divided into n subsets of equal size. A different model M is trained n times, each time setting aside one of the subsets which will be used as the test set on which the predictive accuracy will be computed. The final accuracy estimation is the averaged accuracy over the n different repetitions of the training and testing phases.

A number of different learning paradigms have been developed over the years for classification. Since there is no single classification algorithm that is better than all the others irrespective of the application domain, each time face a new classification problem we have to assess anew the suitability of the algorithms. We have experimented with several classification algorithms trying to cover a range, as broad as possible, of different learning paradigms. We found a voting ensemble-learner to be the best for our application problem according to the estimated predictive performances. In the next paragraphs we give a brief description of the different types of learning paradigms that we are going to consider in this paper.

4.1 Bayesian learning

Bayesian classification algorithms are statistical learning algorithms based on the Bayes theorem. The Bayes Net algorithm (the simplest Bayesian classifier) assumes that, the effect of the value of an attribute on the class attribute is independent of the values of the other attributes given the value of the class attribute (conditional independence).

4.2 Decision Tree learning

A decision tree consists of decision nodes and leaves. A leave is usually associated with a single class; the majority class of the training examples that arrive to that leave. Splits are introduced in the building of the tree according to the outcome of a function f (information gain ratio). When examples are classified a function is used in each split to determine the downstream path to be followed. An indicative decision tree-based algorithm is the C4.5 classifier.

4.3 Rule-Induction learning

Rule-induction performs a depth-first search in a graph G (V, E) generating one path. V is a set of attributes and E is the set of edges denoting dependencies among attributes. Such path represented as a classification rule is a conjunction of conditions with discrete or numeric attributes. A rule is said to cover an example if the later fulfills all the conditions of that rule. A representative rule-induction algorithm is the RIPPER (Repeated Incremental Pruning to Produce Error Reduction).

4.4 Instance-based learning

An instance-based algorithm uses a distance function $\|\boldsymbol{e} - \boldsymbol{q}\|$ in order to determine which example vector $\boldsymbol{e}$ of the training set is closest to an un-*classified* example $\boldsymbol{q}$ (or *instance*). Once the nearest example vector has been determined, its class label is selected as the class label for $\boldsymbol{q}$. A representative instance-based algorithm is the *k* nearest neighbor classifier.

4.5 Ensemble-Learning Algorithms

Ensemble-learning algorithms combine a number of base classification models, classifiers, produced by different learning algorithms or by different training sets, in order to achieve better classification performance than their constituents. Base models can be combined in different ways in order to generate ensemble-learning algorithms. Popular ensemble-learning algorithms are the following:

4.5.1 Voting

Each base classifier predicts, *votes*, a class. The final class is that, which assumes the greatest number of votes

4.5.2 Bagging

Several training sub-sets *Ei* are formed from the initial training set *E* by random re-sampling with replacement. A base classifier is learned from each training sub-set. The final class is determined by voting of the base classifiers.

4.5.3 Boosting

In boosting also the diversity of the base classifiers is a result of different training sets. The method works in an iterative manner re-sampling the current training dataset by giving higher resample weights to instances that are hard to classify. The final class is determined by a weighted voting of the base classifiers where the weights are determined on the basis predictive performance of the base classifiers. A typical example of a boosting algorithm is AdaBoost M1 boosting.

4.5.4 ARFF format files

You do not need to know about ARFF format unless you wish to convert data from other formats. However, it is useful to see the information that such files provide to WEKA.

```
@relation TRAINING
@attribute Userid numeric
@attribute 'Venue ID'
@attribute Latitude numeric
@attribute Longitude numeric
@attribute TIME
@attribute 'Venue category name '
 @data
50756,3fd66200f964a52000e81ee3,40.758102,-73.975734,18:00:06,Gym
190571,3fd66200f964a52000ea1ee3,40.732456,-74.003755,18:00:07,'Indian Restaurant'
50756,3fd66200f964a52000ec1ee3,42.345907,-71.087001,18:00:08,'Indian Restaurant'
66981,3fd66200f964a52000ee1ee3,39.933178,-75.159262,18:00:08,'Sandwich Place'
```

Fig 1 Opening a data set.

In the Explorer window, click on “Open file” and then use the browser to navigate to the ‘data’ folder within the WEKA-3.8.1 folder

V. RESULTS

5.1 Screenshots

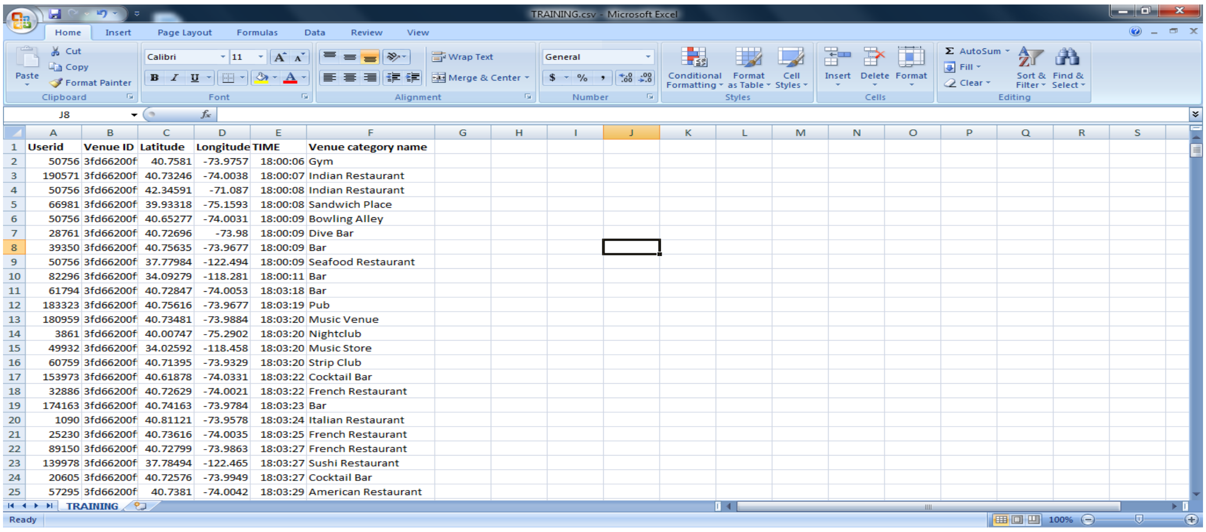

Fig 2 Training data

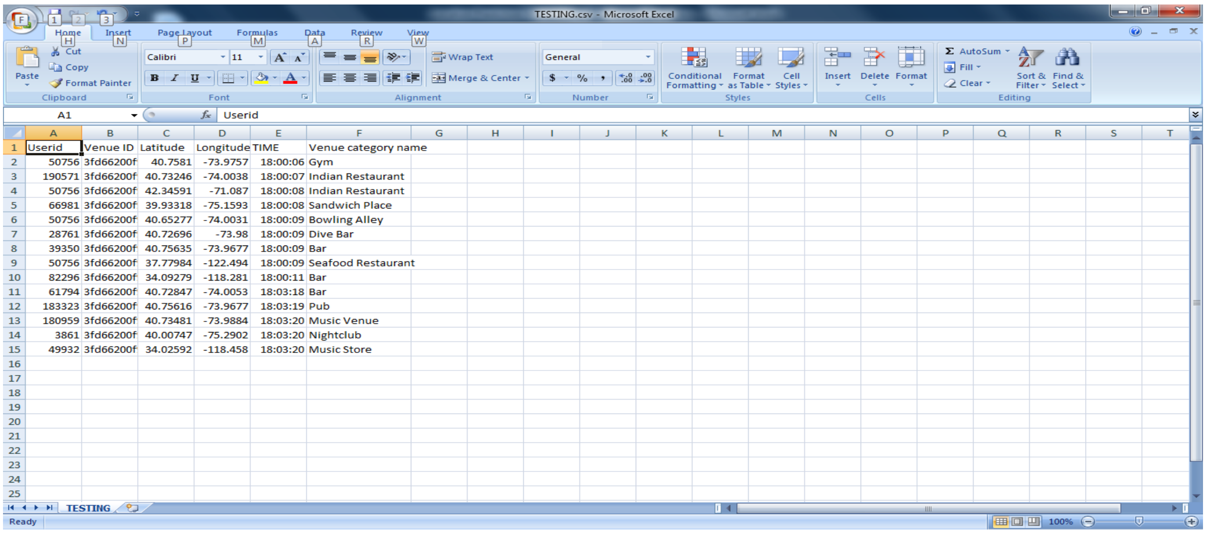

Fig 3 Testing data

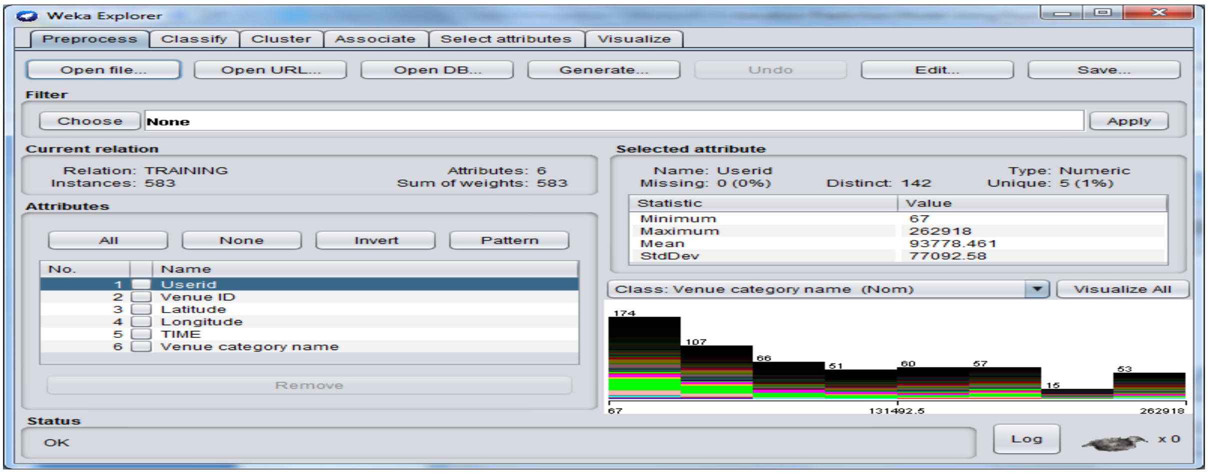

Fig 4 Screenshot view of csv Location history Data set File open in Explorer Interface.

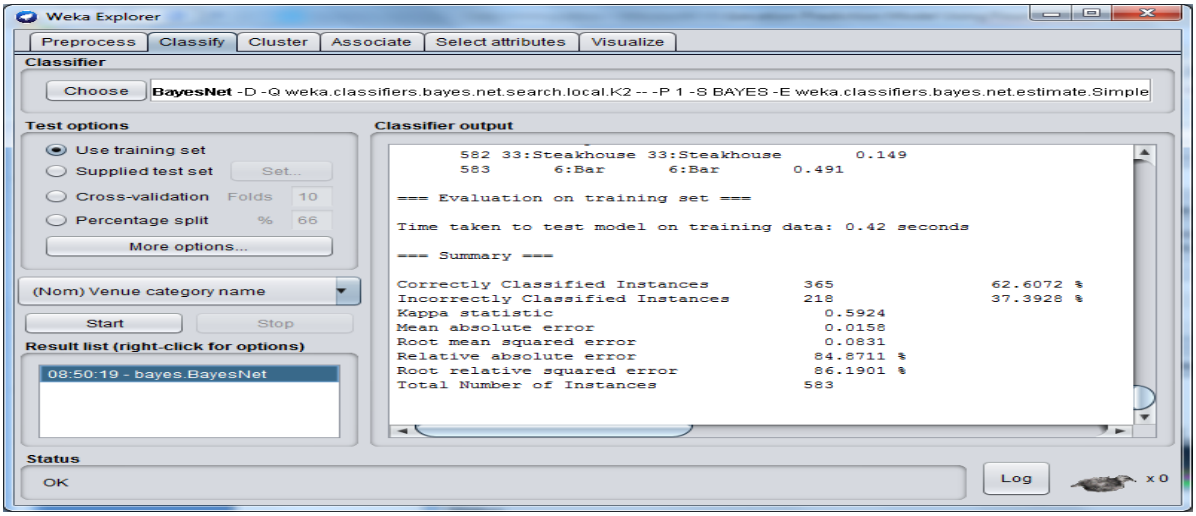

Fig 5 Bayesnet Algorithm

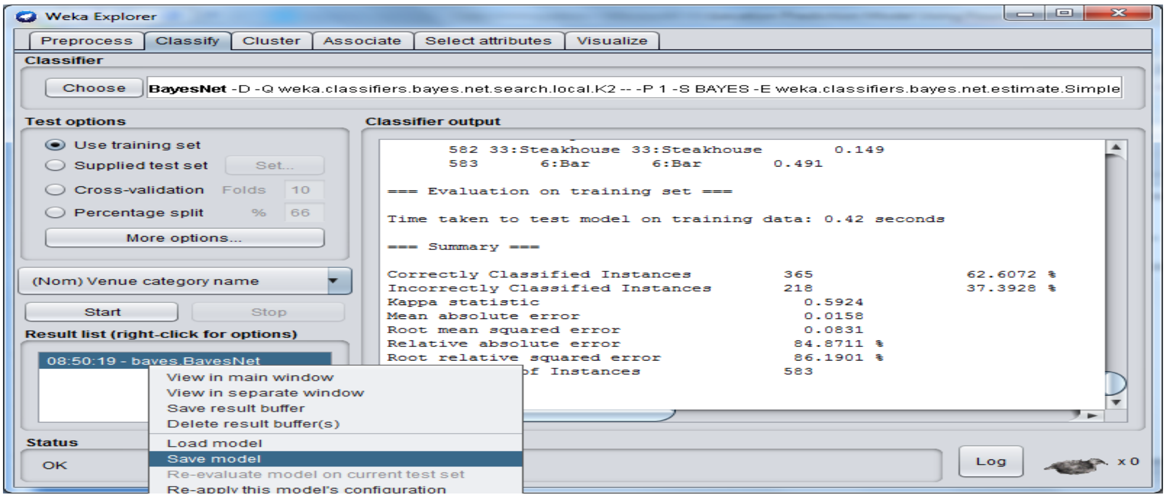

Fig 6 SAVE Model

5.2 Load Model

In the “Test Options”, we have to select “Supplied test set”, and once the file is loaded we select “No class” from the list of attributes.

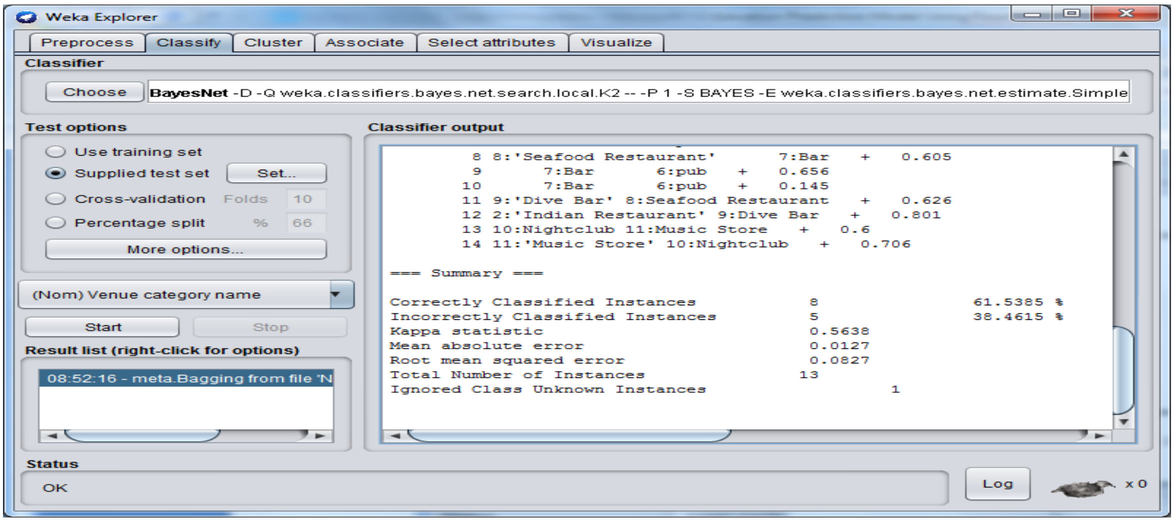

Fig 7 Load Model

5.3 Experimenter Interface

Experimenter Interface has been used in this paper to analyze data by experimenting through algorithms such as Naïve Bayes, J48, IBK Tree and Random Tree to classify the data using train and test sets.

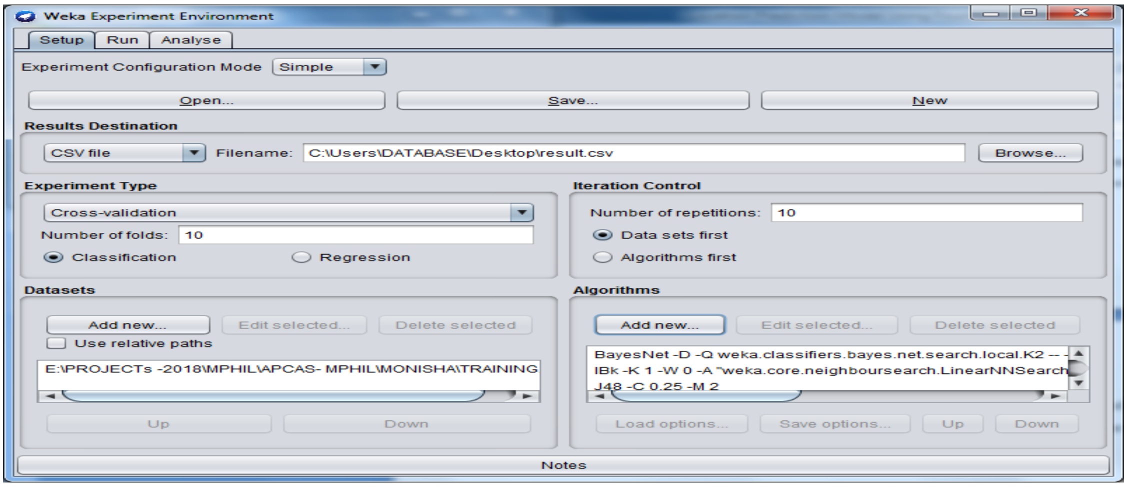

Fig 8 Experimenter Interface

Then, clicking "More Options", a new window opens and we choose PlainText from 'Output predictions'. Finally, we need to right click in the model and run "Re-evaluate model on current test set"

.

Algorithm	Time Taken to Build Model (seconds)	Correctly Classified Instances %Accuracy	Incorrectly Classified Instances %Accuracy	Mean Absolute Error	ROC Area
J48	0.02	94.75	5.25	0.0498	1
Naïve Bayes	0.02	99.5	0.5	0.0196	1
SMO	0.11	98.75	3.25	0.0125	0.990
IBK	0.01	96.75	3.25	0.0918	0.984

Table 1: Classification algorithm accuracy

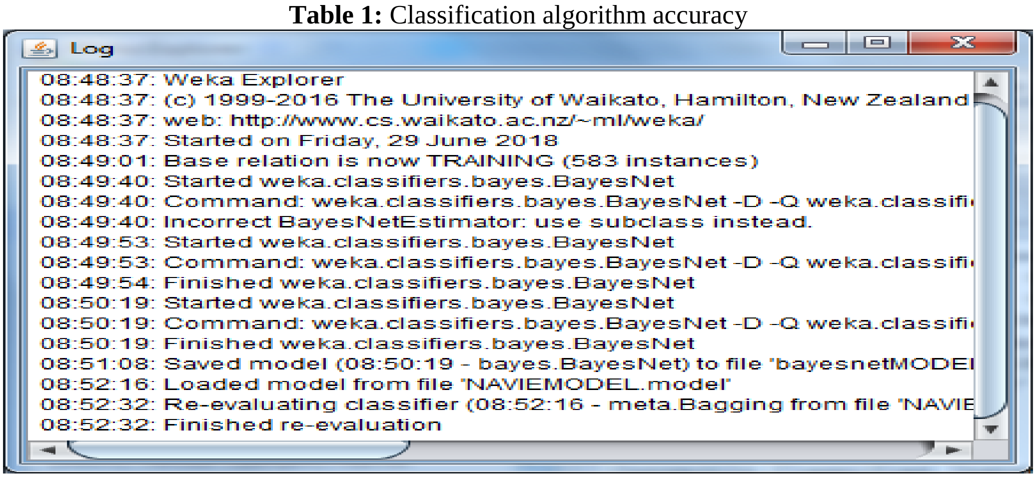

Fig 9 Log file

VI. CONCLUSION

Finally after all analysis we obtained the result for the corresponding dataset. We analysis that SMO is the best classification algorithm analysed till now; it's then followed by naive bayes. But at some point both SMO and naive shows same level of accuracy. Both of these are followed by J48 and IBK.. J48 is being the third least influenced algorithm. IBk is being the worst one.

FUTURE ENHANCEMENT

This research has several interesting future directions. For example, better ways to improve the feature pre-processing stage and design the compact structure to maintain the extracted features. It is also valuable to exploit the evolving factors in the location prediction. Additionally, the feature extraction methods we proposed in this work can be extended to enable incremental updating. And new comprehensive location prediction and update setting can be utilized.

REFERENCES

[1]. J. Sang, T. Mei, J.-T. Sun, C. Xu, and S. Li, "Probabilistic sequential pois recommendation via check-in data," in Proceedings of the 20th International Conference on Advances in Geographic Information Systems, 2012, pp. 402–405.
[2]. X. Liu, Y. Liu, K. Aberer, and C. Miao, "Personalized point-of interest recommendation by mining users' preference transition," in Proc. of CIKM, 2013, pp. 733–738.
[3]. J.-D. Zhang, C.-Y. Chow, and Y. Li, "Lore: exploiting sequential influence for location recommendations," in Proceedings of the 22nd ACM SIGSPATIAL International Conference on Advances in Geographic Information Systems, 2014, pp. 103–112.
[4]. W. Wang, H. Yin, S. Sadiq, L. Chen, M. Xie, and X. Zhou, "SPORE: A sequential personalized spatial item recommender system," in Proc. of ICDE, 2016, pp. 954–965.
[5]. D. Lian, X. Xie, V. W. Zheng, N. J. Yuan, F. Zhang, and E. Chen, "CEPR: A collaborative exploration and periodically returning model for location prediction," TIST, vol. 6, p. 8, 2015.
[6]. A. Noulas, S. Scellato, N. Lathia, and C. Mascolo, "Mining user mobility features for next place prediction in location-based services," in Proc. of ICDM. IEEE, 2012, pp. 1038–1043.
[7]. J. Ye, Z. Zhu, and H. Cheng, "Whats your next move: User activity prediction in location-based social networks," in Proc. of SDM, 2013.
[8]. A. Noulas, S. Scellato, C. Mascolo, and M. Pontil, "An empirical study of geographic user activity patterns in Foursquare," in Proc. of ICWSM, 2011, pp. 70–573.
[9]. E. Cho, S. A. Myers, and J. Leskovec, "Friendship and mobility: user movement in location based social networks," in Proc. Of SIGKDD, 2011, pp. 1082–1090.
[10]. J. Bao, Y. Zheng, D. Wilkie, and M. Mokbel, "Recommendations in location-based social networks: A survey," Geoinformatica, vol. 19, no. 3, pp. 525–565, Jul. 2015.
[11]. D. Lian, X. Xie, F. Zhang, N. J. Yuan, T. Zhou, and Y. Rui, "Mining location-based social networks: A predictive perspective," IEEE Data Engineering bulletin, vol. 38, no. 2, pp. 35–46, 2015.
[12]. D. Brockmann, L. Hufnagel, and T. Geisel, "The scaling laws of human travel," Nature, vol. 439, no. 7075, pp. 462–465, 2006.
[13]. M. C. Gonzalez, C. A. Hidalgo, and A.-L. Barabasi, "Understanding individual human mobility patterns," Nature, vol. 453, no. 7196, pp. 779–782, 2008.
[14]. D. Lian, C. Zhao, X. Xie, G. Sun, E. Chen, and Y. Rui, "GeoMF: Joint geographical modeling and matrix factorization for point-ofinterest recommendation," in Proc. of SIGKDD, 2014, pp. 831–840.
[15]. M. Lichman and P. Smyth, "Modeling human location data with mixtures of kernel densities," in Proc. of SIGKDD, 2014, pp. 35–44.
[16]. J.-D. Zhang and C.-Y. Chow, "GeoSoCa: Exploiting geographical, social and categorical correlations for point-of-interest recommendations," in Proc. of SIGIR, 2015, pp. 443–452.
[17]. Y. Wang, N. J. Yuan, D. Lian, L. Xu, X. Xie, E. Chen, and Y. Rui, "Regularity and conformity: Location prediction using heterogeneous mobility data," in Proc. of SIGKDD, 2015, pp. 1275–1284.
[18]. B. W. Silverman, Density estimation for statistics and data analysis. CRC press, 1986.
[19]. R. A. Finkel and J. L. Bentley, "Quad trees a data structure for retrieval on composite keys," Acta informatica, vol. 4, no. 1, pp. 1–9, 1974.
[20]. Y. Koren, R. Bell, and C. Volinsky, "Matrix factorization techniques for recommender systems," Computer, vol. 42, no. 8, pp. 30–37, Aug. 2009.
[21]. D. Saad, "Online algorithms and stochastic approximations," Online Learning, 1998.
[22]. R. Motwani and P. Raghavan, Randomized algorithms. Chapman & Hall/CRC, 2010.

[23]. T. H. Haveliwala, "Topic-sensitive PageRank," in Proc. of WWW, 2002, pp. 517–526.
[24]. A. Monreale, F. Pinelli, R. Trasarti, and F. Giannotti, "Wherenext: a location predictor on trajectory pattern mining," in Proc. Of SIGKDD, 2009, pp. 637–646.
[25]. M. Li, A. Ahmed, and A. J. Smola, "Inferring movement trajectories from GPS snippets," in Proc. of WSDM, 2015, pp. 325–334

Dr.G.Anandharaj1" Location Prediction Model Using Foursquare Check-In Service: A Machine Learning Approach." IOSR Journal of Engineering (IOSRJEN), vol. 08, no. 8, 2018, pp. 40-49