Masters Program in Geospatial Technologies

SPATIO-TEMPORAL PATTERN MINING FROM GLOBAL POSITIONING SYSTEMS (GPS) TRAJECTORIES DATASET

TIGABU DAGNE AKAL

Dissertation submitted in partial fulfilment of the requirements
for the Degree of *Master of Science in Geospatial Technologies*

UNIVERSIDADE NOVA ISEGI
UNIVERSITAT JAUME I
WESTFÄLISCHE WILHELMS-UNIVERSITÄT MÜNSTER

SPATIO-TEMPORAL PATTERN MINING FROM GLOBAL POSITIONING SYSTEMS (GPS) TRAJECTORIES DATASET

Dissertation supervised by:

Supervisor:

Prof. Dr. Edzer Pebesma, Institute for Geoinformatics, University of Muenster, Germany

Co-supervisors:

Prof. Dr. Oscar Belmote, Institute of New Imaging Technologies, Universitat Jaume I, Spain

Prof. Dr. Roberto Henriques, Universidade NOVA de Lisboa, Portugal

August 2015

DECLERATION OF ORIGINALITY

I declare that this thesis is my original work and has not been presented for a degree in any other university.

Muenster, August 2015 Tigabu Dagne

ACKNOWLEDGMENTS

First and foremost extraordinary thanks go for my Almighty God and His Mother Saint-Merry.

It is with immense gratitude that I acknowledge the support and help of my supervisor, Prof. Dr. Edzer Pebesma. His patience, advises and support helped me overcome challenging situations and finish this thesis. It gives me great pleasure in acknowledging the support and help of my co-supervisors Prof. Dr. Oscar Belmote and Prof. Dr. Roberto Henriques.

I would like to thank all lecturers, staff members and classmates of Geospatial Technologies for their contribution in the special arena for the success of my study. I have to give a special mention for the program coordinator Dr. Christoph Brox. I have no word to express my feeling for his endless help and support during my good and worst times. I am also thankful to course coordinators Dori A. and Karsten H. who have helped me a lot during my study.

I would like to thank the European Commission for funding my study and gave me a great opportunity to explore lots of cultures in Europe.

Most importantly, none of these would have been possible without the love and patience of my father and my mother. In the meantime, I would like to express my heart-felt gratefulness to my beloved wife, Meazashwork Assefa, who has been a continuous source of love, concern, support and strength all the times.

Finally, I would like to thank all my brothers, sisters and friends who have aided and encouraged me throughout this study.

SPATIO-TEMPORAL PATTERN MINING FROM GLOBAL POSITIONING SYSTEMS (GPS) TRAJECTORIES DATASET

ABSTRACT

The increasing frequency of use location-acquisition technology like the Global Positioning System is leading to the collection of large spatio-temporal datasets. The prospect of discovering usable knowledge about movement behavior, which encourages for the discovery of interesting relationships and characteristics users that may exist implicitly in spatial databases. Therefore spatial data mining is emerging as a novel area of research.

In this study, the experiments were conducted following the Knowledge Discovery in Database process model. The Knowledge Discovery in Database process model starts from selection of the datasets. The GPS trajectory dataset for this research collected from Microsoft Research Asia Geolife project. After taking the data, it has been preprocessed. The major preprocessing activities include:

- ✓ Fill in missed values and remove outliers;
- ✓ Resolve inconsistencies, integration of data that contains both labeled and unlabeled datasets,
- ✓ Dimensionality reduction, size reduction and data transformation activity like discretization tasks were done for this study.

A total of 4,273 trajectory dataset are used for training the models. For validating the performance of the selected model a separate 1,018 records are used as a testing set. For building a spatiotemporal model of this study the K-nearest Neighbors (KNN), decision tree and Bayes algorithms have been tasted as supervised approach.

The model that was created using 10-fold cross validation with K value 11 and other default parameter values showed the best classification accuracy. The model has a prediction accuracy of 98.5% on the training datasets and 93.12% on the test dataset to classify the new instances as bike, bus, car, subway, train and walk classes. The findings of this study have shown that the spatiotemporal data mining methods help to classify user mobility transportation modes. Future research directions are forwarded to come up an applicable system in the area of the study.

TABLE OF CONTENTS

DECLERATION OF ORIGINALITY iii
ACKNOWLEDGMENTS iv
ABSTRACT v
TABLE OF CONTENTS vi
KEYWORDS viii
ACRONYMS ix
INDEX OF TABLES x
INDEX OF FIGURES xi
CHAPTER 1 1
INTRODUCTION 1
1.1. Background 1
1.2. Statement of the Problem 2
1.3. Objective of the Study 2
1.3.1. General Objective 2
1.3.2. Specific Objectives 2
1.4. Methodology of the Study 3
1.4.1. Data Collection 3
1.4.2. System Design and Data Preprocessing 3
1.4.3. Implementation tool 4
1.5. Organization of the thesis 4
CHAPTER 2 5
LITERATURE REVIEW 5
2.1. Spatio-temporal Data Mining and Knowledge Discovery 5
2.2. Spatial Data Mining Models 5
2.2.1. The KDD Process Model 6
2.2.2. The CRISP- DM Process 7
2.2.3. Comparison of Different Spatial Data Mining Models 9
2.3. Spatio-temporal Data Mining Tasks 10
2.3.1. Descriptive Model 11

2.3.2. Predictive Model 13
2.4. Application of Spatial Data Mining 19
2.5. Related Works on Spatial-temporal pattern mining for transportation modes discovery from GPS trajectories 21
CHAPTER 3 23
METHODOLOGY AND EXPERIMENTAL DESIGN 23
3.1. Data Description 23
3.2. Initial Data Selection 24
3.3. Data Cleaning and preprocessing 25
3.4. Evaluation Metrics 26
3.4.1. Error Rate 26
3.4.2. Accuracy 26
3.4.3: Precision and Recall 26
3.5. Experimentation 27
3.5.1. Experimentation Design 27
3.5.2. Train the classifier using KNN Algorithm 28
3.5.3. Train the classifier using J48 decision tree modeling 35
3.5.4. Train the classifier using Naive Bayes modeling 36
3.5.4: Comparison of Supervised Approaches: J48 decision tree, Naive Bayes model and KNN 36
CHAPTER 4 38
RESULTS AND DISCUSSION 38
CHAPTER 5 43
CONCLUSION AND FUTURE WORKS 43
5.1. CONCLUSION 43
5.2. FUTURE WORKS 44
REFERENCES 45
APPENDIXES 50

KEYWORDS

Accuracy

Cross Validation

Data Mining

Geo-life

GPS

K-Nearest-Neighbor

Trajectory

Transportation modes

WEKA

ACRONYMS

AI- Artificial Intelligence

ANN- Artificial Neural Networks

ARFF- Attribute-Relation File Format

BN- Bayesian Network

CRISP-DM- Cross Industry Standard Process for Data Mining

CSV- Comma Separated Value

DM - Data Mining

FN- True Negative

FP - False Positive

GIS- Geographic Information System

GPS- Global Positioning System

KDD- Knowledge discovery in Database

KNN - K-Nearest-Neighbor

LVQ - Learning Vector Quantization

LWL- Locally Weighted Learning

OCA- Overall Classification accuracy

SVM- Support vector machine

TN- True Negative

TP- True Positive

WEKA: Waikato Environment for Knowledge Analysis

INDEX OF TABLES

Table 2.1: Comparison of DM and KD process models and methodologies……10

Table 3.1: Distribution of datasets ……25

Table 3.2: Exemplifier to compute recall and precision accuracy……27

Table 3.3: Classification accuracy using K value 1 ……..28

Table 3.4: Classification accuracy for each modes of transportation with k=1 ……..29

Table 3.5: Classification accuracy using K value 2 ……..30

Table 3.6: Classification accuracy for each modes of transportation with k=2 ……...30

Table 3.7: Classification accuracy using K value 5 ……..30

Table 3.8: Classification accuracy for each modes of transportation with k=5 ……...31

Table 3.9: Classification accuracy using K value 10 ……....31

Table 3.10: Classification accuracy for each modes of transportation with k=10 ……...31

Table 3.11: Classification accuracy using K value 11 ……..32

Table 3.12: Classification accuracy for each modes of transportation with k=10 ……...32

Table 3.13: Classification accuracy using K value 12 ……..32

Table 3.14: Classification accuracy using K value 15 ……..33

Table 3.15: Classification accuracy using K value 20 ……..33

Table 3.16: Classification accuracy using K value 30 ……..34

Table 3.17: Classification accuracy using K value 30 ……..34

Table 3.18: Some of the J48 algorithm parameters and their default values ……...35

Table 3.19: Using J48 algorithm parameters with 10-fold cross validation ……35

Table 3.20: Classification accuracy using Naïve Bayes algorithm ……..36

Table 3.21: Comparison of Supervised Approaches ……36

Table 4.1: Results of classification accuracy with different K values ……..39

Table 4.2: Average TP and FP Rates ……39

Table 4.3: Classification result of unlabeled datasets based on the selected model ……41

Table 4.4: Validating the selected model with real life ……...42

INDEX OF FIGURES

Figure 1.1: An overview of the steps that compose the KDD process ……...4

Figure 2.1: KDD process: “From Knowledge Discovery to Data Mining” ……7

Figure 2.2: CRISP-DM process model ……..8

Figure 2.3: Schematic representation of the neural network model ……16

Figure 2.4: The structure of Bayes network ……..17

Figure 2.5: Simple example of SVM ……...18

Figure: 4.1: Results of classification accuracy with different K-values ……..39

Figure 4.2: True Positive (TP) rate comparison of different K-values……40

Figure 4.3: False Positive (FP) rate comparison of different K-values ……...40

CHAPTER 1

INTRODUCTION

1.1. Background

Geographic and temporal properties are a key aspect of many data analysis problems in business, government, and science. Through the availability of cheap sensor devices we have witnessed an exponential growth of geo-tagged data in the last few years resulting in the availability of fine-grained geographic data at small temporal sampling intervals [1]. Therefore, the actual challenge in geo-temporal analysis is moving from acquiring the right data towards large-scale analysis of the available data.

Many data mining techniques were originally targeted towards simple structured datasets such as relational databases or structured data warehouses. More importantly these algorithms were used to classify data that occur in the same time period or in other words, those datasets that are not expected to change much over time. However, technological improvements and the internet have led to more sophisticated data collection methods thus resulting in more complex data systems such as multimedia, spatial and temporal databases which unstructured or semi-structured [30].

Spatio-temporal data mining is a process of generating new patterns from the existed data based on the spatial and temporal information [1]. As discussed by [1] Spatio-temporal data mining is a subfield of data mining which gained high popularity especially in geographic information sciences due to the pervasiveness of all kinds of location-based or environmental devices that record position, time or/and environmental properties of an object or set of objects in real-time.

The increasing availability of large-scale trajectory data provides us great opportunity to explore them for knowledge discovery in transportation systems using advanced data mining techniques [9]. As a consequence, different types and large amounts of spatio-temporal data became available those introduce new challenges to data analysis and require novel approaches to knowledge discovery [6].

1.2. Statement of the Problem

With the help of various positioning tools, individuals' mobility behaviors are being continuously captured from mobile phones, wireless networking devices and Global positioning system (GPS) appliances [6]. These mobility data serve as an important foundation for understanding individuals' mobility behaviors. As discussed by [6] the dissimilarity in the mobility areas covered by individuals, there is high regularity in the human mobility behaviors, suggesting that most individuals follow a simple and reproducible pattern.

In the current technology users are used to navigate using different GPS technologies. But it is challenging to uncover mobility patterns from GPS datasets. It is important predicting the future moves, detecting modes of transport, mining trajectory patterns and recognizing location based activities.

Using the GPS trajectory dataset it is possible to monitor the behavior of traffic. For example we can use GPS data from cars or other transportation modes to monitor the emergence of unexpected behavior in the given region. Knowing the pattern has the potential to estimate and improve traffic conditions in advance and will reduce emerging anomalous.

This research intends to get answers for the following research questions:

- ✓ What are the patterns to categorize users' transportation modes behavior form GPS logs?
- ✓ What are the ways used for detecting mode of transport for user navigation using GPS?
- ✓ Which machine learning algorithm will achieve a better result in the supervised approach for GPS trajectories?

1.3. Objective of the Study

1.3.1. General Objective

The general objective of this study is to develop Spatio-temporal model for user transportation mode from the GPS trajectories data.

1.3.2. Specific Objectives

The specific objectives of this study are the following:

- ✓ to classify users depending on their transportation mode.

- ✓ to develop supervised model for user mobility modes.
- ✓ to detect modes of transport based on machine learning Algorithms.
- ✓ to select the better machine leaning algorithm for Spatio-temporal trajectory dataset.

1.4. Methodology of the Study

1.4.1. Data Collection

The GPS trajectory dataset for this research collected from Microsoft Research Asia Geolife project by a group of users in a period of over three years (from April 2007 to August 2012) [2] [3] [4] [5]. A GPS trajectory of this dataset is represented by a sequence of time-stamped points, each of which contains the information of latitude, longitude and altitude.

1.4.2. System Design and Data Preprocessing

For this study a combination of data mining tasks and spatial data representation were applied. For identifying users' behavior data mining tasks like supervised algorithms were applied.

Data processing, a critical initial step in data mining work, is often used to improve the quality of training data set. To do so data cleaning and preparation is the core task of data mining which is dependent on software chosen and algorithms used [8].

The data mining model used in this study is the Knowledge discovery in Database (KDD) process. The KDD process refers to the whole process of changing low level data into high level knowledge which is automated discovery of patterns and relationships in large databases and data mining is one of the core steps in the KDD process. The goal of KDD and Data Mining (DM) is to find interesting patterns and/or models that exist in databases but are hidden among the volumes of data [7]. The KDD process as described by [7] consists of five major phases. Data will collect then using appropriate algorithms then mined patterns will be modeled. Figure 1 shows the KDD process model that used in this study.

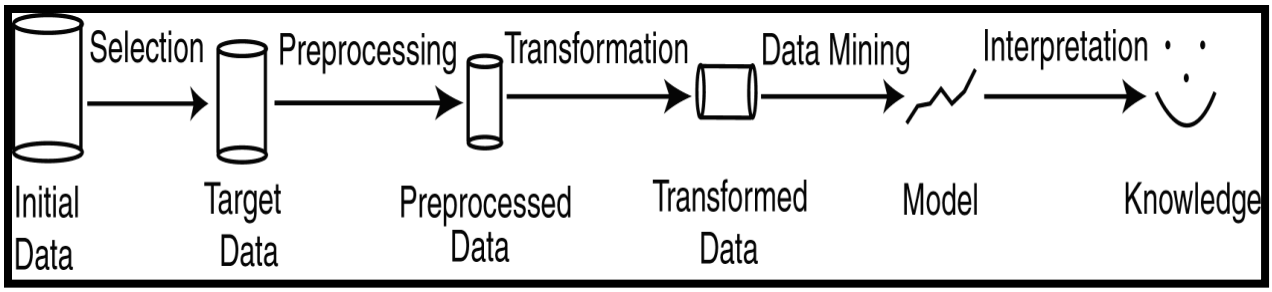

Figure 1.1: An overview of the steps that compose the KDD process [7].

1.4.3. Implementation tool

The research was conducted using DM software packages (Weak tool and Tanagra) and other necessary tools (like Microsoft excel) to identify the transportation modes pattern. Disk Operating System (DOS) and Java Command line interface used for testing the model using unlabeled datasets.

1.5. Organization of the thesis

This thesis is structured into five chapters. The first chapter discusses background to the study problem area, statement of the problem, objective of the study and research methodology.

The second chapter discusses about DM and knowledge discovery technology, DM Processes, DM models, application of DM in general and in particular in the area of Spatio-temporal data mining for transportation modes.

The third chapter deals with understanding the data and dataset preparation, Data transformation and experimentations

The fourth chapter provides a comprehensive results and discussion of the experiential results.

The last chapter presents the conclusions and future works.

CHAPTER 2

LITERATURE REVIEW

The researcher has reviewed different related literatures (books, journal articles, conference proceeding papers, and the Internet) in order to have detailed understanding on the present research.

2.1. Spatio-temporal Data Mining and Knowledge Discovery

Spatial data records information regarding the location, shape and its effect on features (e.g. geographical features). When such a data is time variant, it is called spatiotemporal data. Spatio-temporal data mining is an upcoming research topic which focuses on studying and implementing novel computational techniques for large scale spatiotemporal data [31]. Progress in hardware technologies such as portable display devices, wireless devices has enabled an increase in availability of location-based services. In addition, GPS data is becoming increasingly available and accurate. These developments pave way to a range of new spatio-temporal applications such as distributed systems, location based advertising, disease monitoring, real estate process etc. Having explored temporal data mining, the next step is to extend data mining techniques to spatiotemporal data. This is because in most cases, the spatial and temporal information is implicitly present in the databases; it might be either metric-based (e.g. size) or non-metric based (e.g. terrain, storm path) or both. It is therefore necessary to acknowledge it before carrying out data mining processes that target developing real time models. The spatial and temporal dependencies are inherently present in any spatio-temporal databases. Spatio-temporal data mining can thus be defined as identifying the interesting and non-trivial knowledge from large amounts of spatio-temporal data. Spatio-temporal data mining applications range from transportation, remote sensing, satellite telemetry, monitoring environmental resources and geographic information systems (GISs) [31]

2.2. Spatial Data Mining Models

Temporal data mining is often carried out with one of the following intentions; predictions, classification, clustering, search and pattern identification [34]. Early predictive models assumed a linear combination of the sample values [32] [33]. But later, neural networks [12] and artificial

intelligence (AI) modeling were employed to develop non linear temporal modeling. Clustering in time series data provides an opportunity to understand it at a higher level of abstraction by studying the characteristics of the grouped data. Temporal data clustering has numerous applications ranging from understanding protein structure to learn and characterize financial transactions. The interest for pattern identification in large time series data is comparatively recent and was originated from data mining itself [35]. The sequential pattern analysis was then used to identify the features after which they are input into classifiers (like Nave Bayes Classifier) for data processing [13]. The spatial dimension describes whether the objects considered are associated to a fixed location (e.g., the information collected by sensors fixed to the ground) or they can move, i.e., their location is dynamic and can change in time.

2.2.1. The KDD Process Model

The goal is to provide an overview of the variety of activities in the multidisciplinary field of DM and how they fit together. KDD Process defined by [10], the nontrivial process of identifying valid, novel, potentially useful, and ultimately understandable patterns in data. The term pattern goes beyond its traditional sense to include models or structure in data. In this definition, data comprises a set of facts (e.g., cases in a database), and pattern is an expression in some language describing a subset of the data (or a model applicable to that subset). The term process implies there are many steps involving data preparation; search for patterns, knowledge evaluation, and refinement all repeated in multiple iterations. The process is assumed to be nontrivial in that it goes beyond computing closed-form quantities; that is, it must involve search for structure, models, patterns, or parameters. The discovered patterns should be valid for new data with some degree of certainty.

As showed in figure 2.1 the DM process consists of five steps [10]:

- ✓ **data selection** – having two subcomponents: (a) developing an understanding of the application domain and (b) creating a target dataset from the universe of available data;
- ✓ **preprocessing** – including data cleaning (such as dealing with missing data or errors) and deciding on methods for modeling information, accounting for noise, or dealing with change over time;
- ✓ **transformation** – using methods such as dimensionality reduction to reduce data complexity by reducing the effective number of variables under consideration;

- ✓ **data mining** – having three subcomponents: (a) choosing the data mining task (e.g., classification, clustering, summarization), (b) choosing the algorithms to be used in searching for patterns, (c) and the actual search for patterns (applying the algorithms);
- ✓ **interpretation/evaluation** – having two subcomponents: (a) interpretation of mined patterns (potentially leading to a repeat of earlier steps), and (b) consolidating discovered knowledge, which can include summarization and reporting as well as incorporating the knowledge in a performance system.

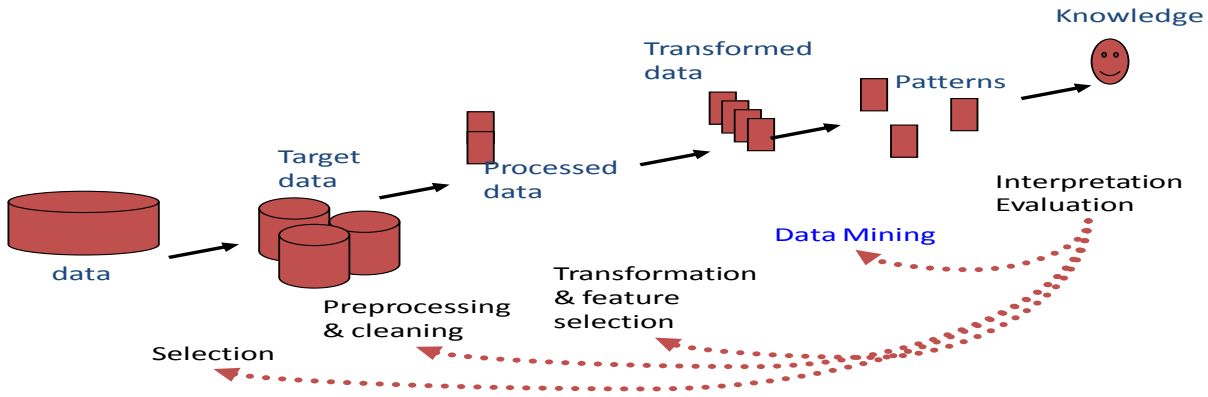

Figure 2.1: KDD process: "From Knowledge Discovery to Data Mining" [10]

2.2.2. The CRISP- DM Process

Cross Industry Standard Process for Data Mining (CRISP-DM) is the most used methodology for developing DM projects [14]. Analyzing the problems of DM and KD projects, a group of prominent enterprises (Teradata, SPSS – ISL, Daimler-Chrysler and OHRA) developing DM projects, proposed a reference guide to develop DM and KD projects. This guide is called CRISP-DM [14]. CRISP-DM is vendor-independent so it can be used with any DM tool and it can be applied to solve any DM problem. CRISP-DM defines the phases to be carried out in a DM project. CRISP-DM also defines for each phase the tasks and the deliverables for each task (Figure 2.2).

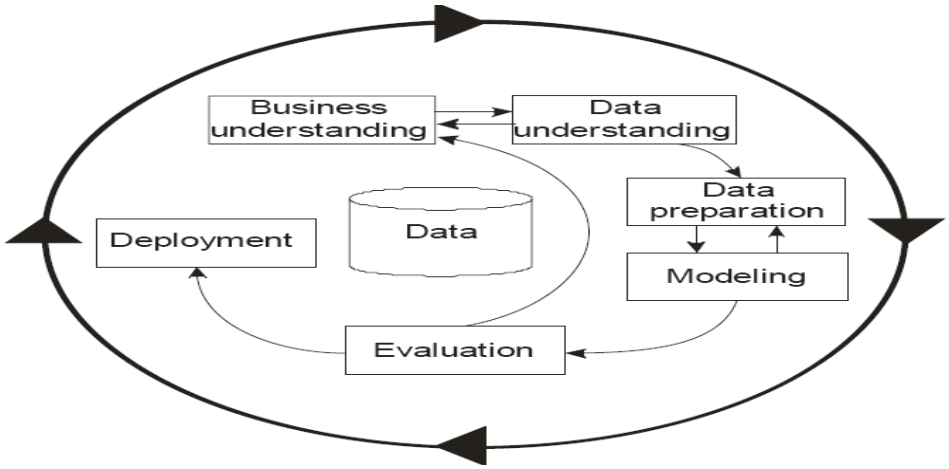

Figure 2.2: CRISP-DM process model [14]

The CRISP-DM is divided into six phases:

- ✓ **Business understanding**: This phase focuses on understanding the project objectives and requirements from a business perspective, then converting this knowledge into a DM problem definition and a preliminary plan designed to achieve the objectives.
- ✓ **Data understanding**: The data understanding phase starts with an initial data collection and proceeds with activities in order to get familiar with the data, to identify data quality problems, to discover first insights into the data or to detect interesting subsets to form hypotheses for hidden information.
- ✓ **Data preparation**: The data preparation phase covers all the activities required to construct the final dataset from the initial raw data. Data preparation tasks are likely to be performed repeatedly and not in any prescribed order.
- ✓ **Modeling**: In this phase, various modeling techniques are selected and applied and their parameters are calibrated to optimal values. Typically, there are several techniques for the same DM problem type. Some techniques have specific requirements on the form of data. Therefore, it is often necessary to step back to the data preparation phase.
- ✓ **Evaluation**: What are, from a data analysis perspective, seemingly high quality models will have been built by this stage. Before proceeding to final model deployment, it is important to evaluate the model more thoroughly and review the steps taken to build it to be certain that it properly achieves the business objectives. At the end of this phase, a decision should be reached on how to use of the DM results.

- ✓ **Deployment**: Model construction is generally not the end of the project. Even if the purpose of the model is to increase knowledge of the data, the knowledge gained will need to be organized and presented in a way that the customer can use it.

2.2.3. Comparison of Different Spatial Data Mining Models

In the early 1990s, when the KDD process term was first coined by [15], there was a rush to develop DM algorithms that were capable of solving all problems of searching for knowledge in data. The KDD process [15] [10] has a process model component because it establishes all the steps to be taken to develop a DM project, but it is not a methodology because its definition does not set out how to do each of the proposed tasks. It is also a life cycle. The 5 A's [16] is a process model that proposes the tasks that should be performed to develop a DM project and was one of CRISP-DM forerunners. Therefore, they share the same philosophy: 5 A's proposes the tasks but does not suggest how they should be performed. Its life cycle is similar to the one proposed in CRISP-DM.

Different DM models have different steps for conducting a given study. Table 2.1 compares the phases into which the DM and KD process is decomposed according some of the above scheme. As showed in table 2.1, most of the scheme cover all the tasks in CRISP-DM, although they do not all decompose the KDD process into the same phases or attaches the same importance to the same tasks. However, some steps described above are omitted the study by [17], like 5 A's and DMIE, propose additional phases not covered by CRISP-DM that are potentially very useful in KD and DM projects. 5 A's proposes the "Automate" phase. This phase entails more than just using the model. It focuses on generating a tool to help non-experts in the area to perform DM and KD tasks. On the other hand, DMIE proposes the "On-going support" phase. It is very important to take this phase into account, as DM and KD projects require a support and maintenance phase. This maintenance ranges from creating and maintaining backups of the data used in the project to the regular reconstruction of DM models. The reason is that the behavior of the DM models may change as new data emerge, and they may not work properly. Similarly, if other tools have been used to implement the DM models, the created programs may need maintenance, e.g. to upgrade the behavior of the user application models.

Model	Fayyad et al.	Cabena et al.	Anand & Buchner	CRISP-DM	Cios et al.
No of steps	9	5	8	6	6
Steps	Developing and Understanding of the Application Domain	Business Objectives Determination	Human Resource Identification	Business Understanding	Understanding the Data
			Problem Specification		
	Creating a Target Data Set	Data Preparation	Data Prospecting	Data Understanding	Understanding the Data
	Data Cleaning and Pre-processing		Domain Knowledge Elicitation		
	Data Reduction and Projection		Methodology Identification	Data Preparation	Preparation of the data
	Choosing the DM Task		Data Pre-processing		
	Choosing the DM Algorithm				
	DM	DM	Pattern Discovery	Modeling	DM
	Interpreting Mined Patterns	Domain Knowledge Elicitation	Knowledge Post-processing	Evaluation	Evaluation of the Discovered Knowledge
	Consolidating Discovered Knowledge	Assimilation of Knowledge		Deployment	Using the Discovered Knowledge

Table 2.1: Comparison of DM and KD process models and methodologies [17]

2.3. Spatio-temporal Data Mining Tasks

As discussed by [26] regular structures in space and time, in particular, repeating structures are often called patterns. Patterns that describe changes in space and time are referred to as spatiotemporal patterns. Spatiotemporal data mining tasks are aimed at discovering various kinds of potentially useful and unknown patterns and trends from spatiotemporal databases. These patterns and trends can be used for understanding spatiotemporal phenomena and decision making or preprocessing step for further analysis and mining. The tasks of data mining can be modeled as either Predictive or Descriptive in nature [18]. A Predictive model makes a prediction about values of data using known results found from different data while the Descriptive model identifies patterns or relationships in data. Unlike the predictive model, a descriptive model serves as a way to explore the properties of the data examined, not to predict new properties. Predictive model data mining tasks include classification, prediction, and regression. The Descriptive task encompasses methods such as Clustering, Association Rules, and Sequence analysis.

2.3.1. Descriptive Model

Descriptive data mining is normally used to generate frequency, cross tabulation and correlation. Descriptive method can be defined to discover interesting regularities in the data, to uncover patterns and find interesting subgroups in the bulk of data [19]. In education, studies [19] used descriptive to determine the demographic influence on particular factors. Summarization maps data into subsets with associated simple descriptions [18]. Basic statistics such as Mean, Standard Deviation, Variance, Mode and Median can be used as Summarization approach.

2.3.1.1. Clustering

Clustering is a data mining technique where data points are clustered together based on their feature values and a similarity metric. Researcher [20] breaks clustering techniques into five areas: hierarchical, statistical, exemplar, distance, and conceptual clustering, each of which has different ways of determining cluster membership and representation.

In clustering, a set of data items is partitioned into a set of classes such that items with similar characteristics are grouped together. Clustering is best used for finding groups of items that are similar. For example, given a data set of customers, subgroups of customers that have a similar buying behavior can be identified. Clustering is an unsupervised learning process. Clustering is the process of grouping a set of physical or abstract objects into classes of similar objects, so that objects within the same cluster must be similar to some extent, also they should be dissimilar to those objects in other clusters. In classification which record belongs which class is predefined, while in clustering there is no predefined classes. In clustering, objects are grouped together based on their similarities. Similarities between objects are defined by similarity functions, usually similarities are quantitatively specified as distance or other measures by corresponding domain experts.

Clustering provides some significant advantages over the classification techniques, in that it does not require the use of a labeled data set for training. For example [22] have applied fixed-width and k-nearest neighbor clustering techniques to connection logs looking for outliers, which represent anomalies in the network traffic. [21] also use a similar approach utilizing learning vector quantization (LVQ), which is designed to find the Bayes Optimal boundary between classes, using k-means clustering to determine initial vector positioning.

K-means clustering: is a data mining/machine learning algorithm used to cluster observations into groups of related observations without any prior knowledge of those relationships [40]. The k-means algorithm is one of the simplest clustering techniques and it is commonly used in medical imaging, biometrics and related fields. The k-means algorithm is an evolutionary algorithm that gains its name from its method of operation. The algorithm clusters observations into k groups, where k is provided as an input parameter. It then assigns each observation to clusters based upon the observation's proximity to the mean of the cluster. The cluster's mean is then recomputed and the process begins again. Here's how the algorithm works [40]:

I. The algorithm arbitrarily selects k points as the initial cluster centers ("means").
II. Each point in the dataset is assigned to the closed cluster, based upon the Euclidean distance between each point and each cluster center.
III. Each cluster center is recomputed as the average of the points in that cluster.
IV. Steps II and III repeat until the clusters converge. Convergence may be defined differently depending upon the implementation, but it normally means that either no observations change clusters when steps II and III are repeated or that the changes do not make a material difference in the definition of the clusters.

2.3.1.2. Association Rules

Associations or Link Analysis are used to discover relationships between attributes and items such as the presence of one pattern implies the presence of another pattern. That is to what extent one item is related to another in terms of cause-and-effect. This is common in establishing a form of statistical relationships among different interdependent variables of a model. These relations may be associations between attributes within the same data item like ('Out of the shoppers who bought milk, 64% also purchased bread') or associations between different data items like ('Every time a certain stock drops 5%, it causes a resultant 13% in another stock between 2 and 6 weeks later').

2.3.1.3. Sequence Analysis

The investigation of relationships between items over a period of time is also often referred to as Sequence Analysis. Sequence Analysis is used to determine sequential patterns in data [18]. The patterns in the dataset are based on time sequence of actions, and they are similar to association data, however the relationship is based on time. In Market Basket analysis, the items are to be

purchased at the same time, on the other hand, for Sequence Analysis the items are purchased over time in some order.

2.3.2. Predictive Model

The goal of the predictive models is to construct a model by using the results of the known data and is to predict the results of unknown data sets by using the constructed model. For instance a bank might have the necessary data about the loans given in the previous terms. In this data, independent variables are the characteristics of the loan granted clients and the dependent variable is whether the loan is paid back or not [22]. The model constructed by this data is used in the prediction of whether the loan will be paid back by client in the next loan applications.

2.3.2.1. Classification and regression

Classification and regression are two data analyzing family of methods which determine important data classes or may construct models which can predict future data trends. The classification model predicts the categorical values; the regression is used in the prediction of values showing continuity. For instance while the classification model is constructed to categorize whether the bank loan applications are safe or risky, the regression model may be constructed to predict the spending of clients buying computer products whose income and occupation are given [22][23] .

In the classification models the following techniques are mainly used [22]: Decision Trees, Artificial Neural Networks and Naive-Bayes.

2.3.2.1.1 Decision Trees

A decision tree is a classifier expressed as a recursive partition of the instance space [23]. Classification trees are frequently used in applied fields such as finance, marketing, engineering and medicine. The classification tree is useful as an exploratory technique. A decision tree may incorporate nominal or numeric or even both of attributes types. As discussed by [23] the decision tree consists of nodes that form a rooted tree, meaning it is a directed tree with a node called “root” that has no incoming edges. All other nodes have exactly one incoming edge. Test nodes are those which have outgoing edges and the remaining nodes are the leaf nodes which are also referred to as the decision nodes. For each new sample (i.e., feature vector x), the

classification algorithm will search for the region along a path of nodes of the tree to which the feature vector x will be assigned. Each of the internal nodes splits the instance space into two or more sub-divisions. The split is based on a certain discrete function used as input. In the simplest and most frequent case, each test considers a single attribute, such that the instance space is partitioned according to the attribute's value. There are two possible types of divisions or partitions: Nominal partitions: a nominal attribute may lead to a split with as many branches as values there are for the attribute. Numerical partitions: typically, they allow partitions in ranges like 'greater than' or 'less than' or 'in between'. Each leaf is assigned to one class representing the most appropriate target value. In some case, the leaf nodes may also the probability vector that represents the probability of the target attribute having a certain value. Typically, internal nodes are represented as circles and the leaf node as triangles. Each node is labeled with the attribute it tests, and its branches are labeled with its corresponding values. So depending on the values, the final response can be predicted by iteratively traversing through the appropriate node down the tree and understand the behavioral characteristics of the entire dataset as a whole. The termination/stop criteria or the pruning method determine the complexity of the tree.

Complexity is given as a measure of any one of the following:

- ✓ Total number of nodes in the tree.
- ✓ Total number of leaves in the tree.
- ✓ Height of the tree and the number of attributes used.

Ideally, each non-leaf node should correspond to the most applicable input attribute from the set of all attributes already traversed in the path from the root node to that node i.e.it should be the most descriptive node so that when it comes to prediction, the number divisions made or nodes traversed from the root is kept to a minimum. Decision trees were frequently used in the 90s by artificial intelligence experts because they can be easily implemented and they provide an explanation of the result. A decision tree is a tree with the following properties [19]:

- ✓ Each internal node tests an attribute.
- ✓ Each branch corresponds to the value of the attribute.
- ✓ Each leaf assigns a classification.

ID3 [37] was the first devised decision tree algorithm. It is a greedy algorithm that uses information gain as splitting criteria. The tree is designed to stop when all the instances belong to a single value of a target feature. The tree is first initialized with the original set S as the root

node. The algorithm iteratively computes the entropy of each of the attribute in the set S and selects the one with minimum entropy. Entropy is defined as the degree of disorderliness. When pertaining to the definition of information, the higher the entropy of the data (model), the more is the amount of information required to better describe the data. So when building the decision tree, it is ideally desired to minimize the entropy while reaching the leaf nodes. As upon reaching the leaf nodes there in no more information required (therefore a zero entropy) and all instances have the values assigned to the target label

2.3.2.1.2. Artificial Neural networks

Neural network is a mathematical model developed in an effort to mimic the human brain [38]. The neural network consists of the layered interconnected set of nodes/processors which are analogous to the neurons in the human brain. Each of these nodes has weighted connection to the nodes in the adjacent layers where the information is received from one node, and weighted functions are used to compute the output values. As the model learns, the weights changes while the set of inputs is repeatedly passed through the network. Once the model has completed the learning phase (with the training dataset), the test data is passed through the network and classified according to the values in the output layer. Once trained, an unknown instance passing through the network is classified according to the values seen at the output layer. Artificial neural networks (ANN), being a non-parametric model is very well suited for applications like large databases, remote sensing, weather forecasting etc. Various studies have been carried out to demonstrate the performance improvement over traditional classifier model. Also since it does not rely on any assumptions concerning the underlying density function, it is very well capable of handling multi source data. The perceptron model proposed by [39] is a single layer neural network whose weights and biases are trained to produce a correct target vector when presented with the corresponding input vector.

The training technique used is called the perceptron learning rule. The perceptron was new inits ability to learn with the training data and randomly distributed data. The perceptrons were very well suited for the classification problems where data can be completely separated along a hyperplane.

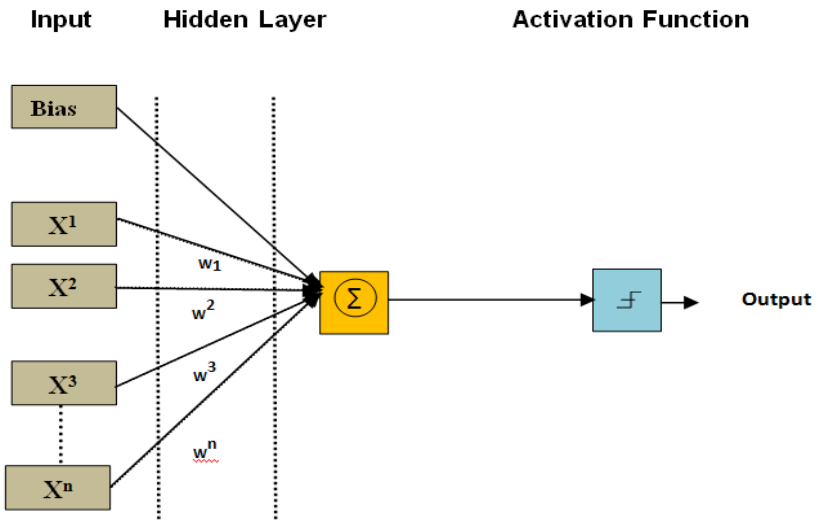

Figure 2.3: Schematic representation of the neural network model [39]

2.3.2.1.3. Bayesian Network (BN)

A popular supervised learning technique is the Bayesian statistical methods which allow taking into account prior knowledge when analyzing data [36]. Its popularity can be attributed to several reasons. It is fairly easy to understand and design the model; it does not employ complicated iterative parameter estimation schemes. It simplicity makes it easier to extend it to large scale datasets. Another reason is that it is also easy to interpret the results. The end users do not require prior expert knowledge in the field which is how it derived the name Na¨ıve Bayes classifier. In the Bayesian approach, the objective is to find the most probable set of class labels given the data (feature) vector and a priori or prior probabilities for each class. It essentially reduces an n-dimensional multivariate problem to n-dimensional univariate estimation.

A BN is a graphical model for probability relationships among a set of variables features as shown in figure 2.4. The BN structure S is a directed acyclic graph (DAG) and the nodes in S are in one-to-one correspondence with the features X. The arcs represent casual influences among the features while the lack of possible arcs in S encodes conditional independencies. Moreover, a feature (node) is conditionally independent from its non-descendants given its parents (X1 is conditionally independent from X2 given X3 if P (X1|X2, X3) =P (X1|X3) for all possible values of X1, X2, X3)

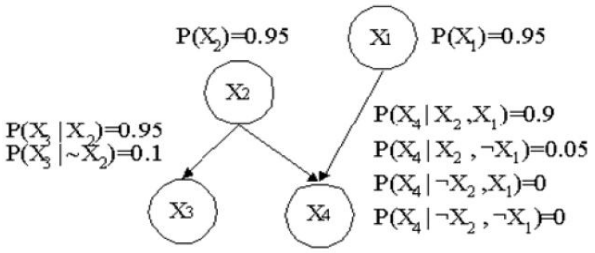

Figure 2.4: The structure of Bayes network [19]

Typically, the task of learning a BN can be divided into two subtasks: initially, the learning of the DAG structure of the network, and then the determination of its parameters. Probabilistic parameters are encoded into a set of tables, one for each variable, in the form of local conditional distributions of a variable given its parents. Given the independences encoded into the network, the joint distribution can be reconstructed by simply multiplying these tables. Within the general framework of inducing Bayesian networks, there are two scenarios: known structure and unknown structure.

2.3.2.1.4. Support vector machine (SVM)

One of the most powerful classification algorithm and it is developed based on statistical learning theory. Statistical learning theory is a system that is derived from statistical and function analysis, used to create a predicative function based on input (training) data. The current form of SVM was initially introduced by Boser, I.M.Guyon and V.N.Vapnik with a paper at the conference workshop on Computational Learning Theory in 1992 [43].

The SVM model is based on two fundamental concepts that are hyperplane classifiers for linearly separable patterns and kernel function for non- linearly separable patterns (Noble, 2006). As indicated earlier, the SVM is based on mathematical function; however, the idea can be expressed without any mathematical equation, and is illustrated below.

The first major concept of SVM is the Hyperplane classifier, where it is the process of drawing a separation line in between different objects in order to distinguish by their features. These separation lines are decision planes to delineate decision boundaries between the objects. A descriptive example using a Figure 2.5 contains two different classes where one is red and the other is green dot points. These dot points are separated by a line according to their classes, where the red spots are located on the left hand side and the green spots are located on the right

hand side. Thus, the separating line defines a boundary between the two different classes. This example is a typical example for linear classifier, where a line separates objects into different classes.

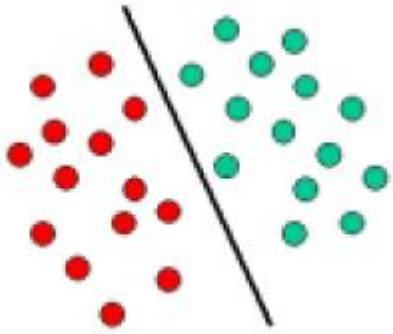

Figure 2.5: Simple example of SVM [44]

2.3.2.1.5. Lazy classifiers

Lazy learners store the training instances and do no real work until classification time. IB1 is a basic instance-based learner which finds the training instance closest in Euclidean distance to the given test instance and predicts the same class as this training instance (Aha et al., 1991). IBk is a k-nearest-neighbor classifier that uses the same distance metric. The number of nearest neighbors (default k¼1) can be specified explicitly in the object editor or determined automatically using leave-one-out cross-validation, subject to an upper limit given by the specified value. KStar is a nearest neighbor algorithm with a generalized distance function based on transformations [49]. LWL is a general algorithm for Locally Weighted Learning, and it assigns weights using an instance-based method and builds a classifier from the weighted instances.

KNN (*k*-nearest-neighbor) [29] has been widely used in classification problems. KNN is based on a distance function that measures the difference or similarity between two instances. The standard Euclidean distance $d(x, y)$ between two instance x and y is often used as the distance function, defined as follows.

$$d(x,y) = \sqrt{\sum_{i=1}^{n}(a_i(x) - a_i(y))^2}$$

As discussed by [29] the KNN algorthim is a method for classifying objects based on closest training examples in the feature space. It is a type of instance-based learning, or lazy learning where the function is only approximated locally and all computation is delayed until classification. A majority vote of an object's neighbors is used for classification, with the object being assigned to the class most common amongst its k (positive integer, typically small) nearest neighbors. If k is set to 1, then the object is simply assigned to the class of its nearest neighbor. The kNN algorithm can also be applied for regression in the same way by simply assigning the property value for the object to be the average of the values of its k nearest neighbors. It can be useful to weight the contributions of the neighbors, so that the nearer neighbors contribute more to the average than the more distant ones. No explicit training step is required since training consists of just storing training instance feature vectors and corresponding class labels. In order to identify neighbors, the objects are represented by position vectors in a multidimensional feature space. It is usual to use the Euclidean distance, but also further distance measures, such as the Manhattan distance could be used instead. In the classification/testing phase, the test sample is represented as a vector in the feature space. Distances from this vector to all stored vectors are computed and the k closest samples are selected to determine the class/real- value of the test instance.

2.4. Application of Spatial Data Mining

As discussed by [26] the significance of spatiotemporal data analysis and mining is growing with the increasing availability and awareness of huge amount of geographic and spatiotemporal datasets in many important application domains like:

- ✓ Meteorology: all kinds of weather data, moving storms, tornados, developments of high pressure areas, movement of precipitation areas, changes in freezing level, droughts.
- ✓ Biology: animal movements, mating behavior, species relocation and extinction.
- ✓ Crop sciences: harvesting, soil quality changes, land usage management, seasonal grasshopper infestation.
- ✓ Forestry: forest growth, forest fires, hydrology patterns, canopy development, planning tree cutting, planning tree planting.
- ✓ Medicine: patients' cancer developments, supervising developments in embryology.
- ✓ Geophysics: earthquake histories, volcanic activities and prediction.
- ✓ Ecology: causal relationships in environmental changes, tracking down pollution incidents.

- ✓ Transportation: traffic monitoring, control, tracking vehicle movement, traffic planning, vehicle navigation, fuel efficient routes.

Databases today can range in size into the terabytes — more than 1,000,000,000,000 bytes of data [24]. Within these masses of data lies hidden information of strategic importance. But when there are so many trees, how do you draw meaningful conclusions about the exact location of each kind of forest? According to [24] the newest answer is spatial data mining, which is being used both to increase revenues and to reduce costs.

As discussed by [10] in business, main KDD application areas includes marketing, finance (especially investment), fraud detection, and manufacturing, telecommunications, and Internet agents.

Marketing: In marketing, the primary application is database marketing systems, which analyze customer databases to identify different customer groups and forecast their behavior.

Fraud detection: HNC Falcon and Nestor PRISM systems are used for monitoring credit card fraud, watching over millions of accounts. The FAIS system [25], from the U.S. Treasury Financial Crimes Enforcement Network, is used to identify financial transactions that might indicate money laundering activity.

Manufacturing: The CASSIOPEE troubleshooting system, developed as part of a joint venture between General Electric and SNECMA, was applied by three major European airlines to diagnose and predict problems for the Boeing 737. To derive families of faults, clustering methods are used.

Telecommunications: The telecommunications alarm-sequence analyzer (TASA) was built in cooperation with a manufacturer of telecommunications equipment and three telephone networks [25]. The system uses a novel framework for locating frequently occurring alarm episodes from the alarm stream and presenting them as rules. Large sets of discovered rules can be explored with flexible information- retrieval tools supporting interactivity and iteration. In this way, TASA offers pruning, grouping, and ordering tools to refine the results of a basic brute-force search for rules.

2.5. Related Works on Spatial-temporal pattern mining for transportation modes discovery from GPS trajectories

Data mining technique, which is a process of extracting useful hidden knowledge from large volumes of data, can be used to discover spatio-temporal regularities in trajectories [3]. The existing travel mode inference procedure shares the following general principle. At first, the classification models developed based on historical data of mobility patterns upon either supervised or unsupervised learning method. These methods are from machine learning methods that use algorithms to classify the dataset accordingly. Supervised learning method creates a general hypothesis from known dataset which then used to predict the unknown dataset. On the contrary, unsupervised learning uses the input dataset to develop particular patterns or models by using statistical mechanisms; this learning method builds models from dataset without predefined classes.

Different methods have been developed and tested, which derive travel modes as reliably as possible by combining the x and y coordinates and timestamps in the GPS logs without respondent involvement [27]. Of course, average speed and maximum speed can be determined from the location and time data in the GPS logs. However, none of the travel modes can be distinguished with full certainty without additional information. For example, a train and a car trip may deliver the same average and maximum speed as can occur with a car trip in a jammed city and a cycling trip. To be able to distinguish between different modes with the same speed, methods have been developed that combine GPS data with IS maps.

As described by [41] probability matrix uses a method for determining travel mode. Trip characteristics such as the average speed, the maximum speed and the speed recorded most often and bicycle ownership define whether a trip is assigned as on foot, by bicycle or by motorized vehicle. Subsequently, street and public transportation maps are used in GIS for specifying the type of the motorized vehicles.

As addressed by [42] a hierarchical inference model to infer a user's destination and mode of transport through traveler's daily activities is developed. The research has demonstrated an efficient way to infer traveler's activities based on unsupervised learning algorithms. The model

developed in the form of hierarchical predicative approach. This model uses GPS raw data and map information like road networks and bus stops.

This study is divided into two steps, the first step is the learning procedure and the other one is the inference of the travel mode. The first step is to automatically considering the complete labeled GPS data; it is provided in some parts of the Geo-life datasets. Then classification features are derived from the labeled data to be used as a training data. This classification training data is used as input data for the algorithms to form classification systems for the second step based on the features. This classification system predicts the travel modes in the same travel modes. The main contribution of this study is to consider both labeled and unlabeled records which solved the problem of both supervised and unsupervised approaches.

CHAPTER 3

METHODOLOGY AND EXPERIMENTAL DESIGN

This chapter showed the data preparation and experimental setups. As showed in chapter one, under the methodology section of this thesis, the spatial data mining process model selected is KDD. KDD model is selected because it is the best method for the classification of data where their business is clearly mentioned. In KDD there is no need of understanding the business. The researcher has taken a readymade dataset from Microsoft Research Asia Geolife project by a group of users in a period of over three years (from April 2007 to August 2012) [2] [3] [4] [5].

3.1. Data Description

This GPS trajectory dataset was collected from (Microsoft Research Asia) Geolife project by 182 users in a period of over five years (from April 2007 to August 2012) [2] [3] [4] [5]. A GPS trajectory of this dataset is represented by a sequence of time-stamped points, each of which contains the information of latitude, longitude and altitude. This dataset recoded a broad range of users' outdoor movements, including not only life routines like go home and go to work but also some entertainments and sports activities, such as shopping, sightseeing, dining, hiking, and cycling. Although this dataset is wildly distributed in over 30 cities of China and even in some cities located in the USA and Europe, the majority of the data was created in Beijing, China. The dataset has two parts: the GPS points and the labeled transport mode. The GPS points are collected by means of different GPS loggers and mobile phones. This data contains coordinates information for each GPS points in the form of latitude, longitude and altitude where the coordinates are in the form of decimal degrees. Information of time and date is also available for each point in the data set.

73 users have labeled their trajectories with transportation mode (walk, bike, bus, car, train, and other) and this stored individually.

All users' GPS log files are available in the PLT format. The following are the fields

Field 1: Latitude in decimal degrees.

Field 2: Longitude in decimal degrees.

Field 3: Altitude in feet (-777 if not valid).

Field 4: Date - number of days.

Field 5: Date as a string.

Field 6: Time as a string.

Field 7: Transportation mode.

Depending on the above list of features, the researcher has formulated other features that would be important for the experiment. Depending on the latitude and longitude it is possible to calculate the distance between two points.

Calculations of distances based on latitude/longitude points are on the basis of a spherical earth (ignoring ellipsoidal effects) – which is accurate enough for most purpose, in fact, the earth is very slightly ellipsoidal; using a spherical model gives errors typically up to 0.3% [48].

This uses the 'haversine' formula to calculate the great-circle distance between two points – that is, the shortest distance over the earth's surface – giving an 'as-the-crow-flies' distance between the points (ignoring any hills they fly over, of course). The Euclidean distance used as a metric for distance measurement.

$$a = \sin^2(\Delta\varphi/2) + \cos\varphi_1 \cdot \cos\varphi_2 \cdot \sin^2(\Delta\lambda/2)$$

Haversine formula:

$$c = 2 \cdot \operatorname{atan2}(\sqrt{a}, \sqrt{(1-a)})$$

$$d = R \cdot c$$

where φ is latitude, λ is longitude, R is earth's radius (mean radius = 6,371km);

Therefore, the distance between two points can be calculated using the following formula [48]:

```
=a*COS( SIN(lat1)*SIN(lat2) + COS(lat1)*COS(lat2)*COS(lon2-lon1) ) * 6371
                        Or
=  a*COS( SIN(lat1*PI()/180)*SIN(lat2*PI()/180) + COS(lat1*PI()/180) *COS(lat2*PI()/180)
* COS(lon2*PI()/180-lon1*PI()/180) ) * 6371
```

3.2. Initial Data Selection

Before generating a pattern the data should be ready in the way that is easily understandable. Using random selection method the researcher has taken 5,291 records. The researcher has taken only 5,291 records because of the process selecting records from the collection of trajectories were time taken plus those 5,291 records have similar characteristics and were taken from near Beijing square.

Some of the datasets transportation mode is labeled and some of them not. From these, 5,291 data sets 4,273 records were labeled and 1,018 records were not labeled. Those unlabeled records (1,018) used as validation records. The model would be validated using those records. Those records were selected based on the distribution of unlabeled records from five users. The distribution of labeled records and unlabeled records for 5 years almost ration of ¼. From every 5 records 4 of them are labeled and 1 is unlabeled record. The datasets have both quantitative and qualitative features which are in total 6 features. The distributions of datasets which include both labeled and unlabeled are shown in Table 3.1:

Labeled dataset-Transportation-mode	Number of Records Collected
Bike	463
Bus	437
Car	727
Subway	232
Train	928
Walk	1468
Unlabeled records	1018

Table 3.1: Distribution of datasets

3.3. Data Cleaning and preprocessing

Before data is fed into a spatial data mining algorithm, it must be collected, inspected, cleaned and selected. Since even the best predictor will fail on bad data, data quality and preparation is crucial. Also, since a predictor can exploit only certain data features, it is important to detect which data preprocessing works best [28]. For this study preprocessing of the KDD dataset contains the following processes:

- ✓ Assigning transportation mode names in to one of the six classes- bike, bus, car, subway, train and walk
- ✓ To identify and label each transportation mode depending on user documentation of the GeoLife datasets and
- ✓ Microsoft Excel helps to filter and name easily using fill handle.
- ✓ There are records which don't have attributes and these are removed from the dataset.

- ✓ The Geolife dataset is available in text format; so to be read by the spatial data mining too it has to be changed into CSV (comma separated value) or Attribute-Relation File Format (ARFF).

3.4. Evaluation Metrics

General performance of spatial data mining systems is measured in terms of numbers of selected features and the classification accuracies of the machine learning algorithms giving the best classification results.

3.4.1. Error Rate

The error rate, which is only an estimate of the true error rate and is expressed to be a good estimate, if the number of test data is large and representative of the population, is defined as [46]:

Error Rate= [(Total Test samples- Total Correctly Classified Samples)*100%] /
Total Test Samples ……. 3.3

3.4.2. Accuracy

Overall Classification accuracy (OCA) is the most essential measure of the performance of a classifier. It determines the proportion of correctly classified examples in relation to the total number of examples of the test set i.e. the ratio of true positives and true negatives to the total number of examples. From the confusion matrix, we can say that accuracy is the percentage of correctly classified instances over the total number of instances in total test dataset, namely the situation TP and TN, thus accuracy can be defined as follows [47]:

Accuracy = ((TP+TN)*100%)/(TP+TN+FP+FN) …….3.4

3.4.3: Precision and Recall

The parameters are evaluated based on recall accuracy; the definition of precision accuracy and recall are defined below.

Precision

Precision is defined as the proportion of correctly classified of the specific travel mode (like car, bus, walk…) in the set to that of all similar travel mode returned by the classification model. Thus, to determine the precision prediction for travel mode A as shown in the table 3.2, the

precision is the proportion of true positive to that of the total number of true and false positive prediction for travel mode A. It is computed as shown in equation 3.5;

$$Pr_A = \frac{tp_A}{fp_A + tp_A} \quad \text{---3.5}$$

		Predicted Class	
Unknown Class	Travel Mode	A	B
	A	tp	fn
	B	fp	tn

Table 3.2: Exemplifier to compute recall and precision accuracy

Where;

tp_A- true positive prediction for class A;

fn_A- false negative prediction for class A;

fp_A- false positive prediction for class A;

tn_A- true negative prediction for class A.

Recall

Recall is defined as the ability of a prediction model to predict instances travel mode correctly (like walk, car, bus, bike…) Thus, to determine the recall prediction for travel mode A as shown in the Table 3.2, the recall is the proportion of true positive to that of the total number of true positive and false negative prediction for travel mode A. It is computed as shown in equation 3.6;

$$Rc_A = \frac{tp_A}{fn_A + tp_A} \quad \text{---3.6}$$

3.5. Experimentation

This section describes experimental study of the algorithms and procedures, which are described in the previous chapters. In this research both labeled and unlabeled records are used. Data mining tool used is Weka and Microsoft Excel used for labeling the datasets and for other mathematical operations. .

3.5.1. Experimentation Design

The following are the steps used for the experimentation approach:

I. In the beginning, in order to build the experiment, the researcher selected WEKA as the spatial data mining software used for developing the model; Java command line interface and DOS interface used for validation the model. Both labeled and unlabeled records were chosen. All the records considered for this study were taken from five users. All these five users were navigating in the same region, near Beijing square. At the same time all the records are consecutives.

II. The selected records are changed from text format into Microsoft excel format. The Microsoft excel portion of the records contain both labeled and unlabeled records.

III. To come up with cleaned datasets preprocessing tasks are undertaken for underling missing values, outliers and other issues.

IV. Using the latitude and longitude of every trajectories distance has been calculated from one point to another point. After calculating the distance between each points, the speed of travelers from one point to another point can be calculated since time of travelers at each trajectory point is given.

V. Construct the model using the spatial data mining software packages. The KNN, decision tree and Navi bayes supervised approach have been selected since the aim of the research is to classify similar trajectories according to their modes of transportation.

VI. The last step is validating the model using previously unseen records. Speed used as a major variable for classifying users' navigation with respect to their transportation mode. The speed is calculated using the distance between consecutive points and the time taken travel from one point to the next point. Distance is the result of latitude and longitude as discussed in the beginning of this chapter. And speed is the major variable for the learning algorithm.

3.5.2. Train the classifier using KNN Algorithm

To train the classifier the researcher applied the classification technique of KNN algorithm. In the KNN classification algorithm the default value for the k is 1. In the following section, the researcher has tried to show different results based on the value of K and using speed as classification criteria. The K value was selected based on the KNN algorithm principle of which the minimum value of K is 1 and the maximum k value is positive infinitive number. Then

algorithm was tested using different k values. K values from smallest to increasing order were considered for the sake of comparisons. After comparing the result for consecutive k-values then k values with random numbers were tested to select the best classifier with given k-value.

3.5.2.1. Experimentation 1: with K value 1

In k-NN classification, the output is a class membership. An object is classified by a majority vote of its neighbors, with the object being assigned to the class most common among its k nearest neighbors (k is a positive integer typically small). If k = 1, then the object is simply assigned to the class of that single nearest neighbor. The first experimentation is performed with the default parameter that is the value of k is 1. Also the 10-fold cross validation test option is employed for training the classification model. Table 3.3 depicts the resulting confusion matrix of this model.

Total number of instances (training sets)	Correctly classified Instances	Incorrectly Classified Instances
4,273	3,882 (90.85 %)	391 (9.15%)

Table 3.3: Classification accuracy using K value 1

The detailed classification accuracy for each transportation mode is showed in the table 3.4

	TP Rate	FP Rate	Precision	Recall	F-Measure	ROC Area	Class
	0.915	0.016	0.922	0.915	0.919	0.951	Car
	0.854	0.017	0.85	0.854	0.852	0.924	Bus
	0.863	0.04	0.857	0.863	0.86	0.918	Train
	0.708	0.033	0.721	0.708	0.715	0.844	Bike
	1	0	1	1	1	1	Walk
	0.987	0.002	0.966	0.987	0.977	0.991	Subway
Weighted Average	0.908	0.017	0.908	0.908	0.908	0.949	

Table 3.4: Classification accuracy for each modes of transportation with k=1

As shown in the resulting confusion matrix, the KNN classification algorithm with value of K=1 scored an accuracy of 90.85 %. This result shows that out of the total training datasets 4,273 (90.85 %) records are correctly classified, while 391 (9.15%) of the records are incorrectly classified. Resulting confusion matrix is presented in Appendix A.

3.5.2.2. Experimentation 2: with K value 2

The Second experimentation is performed with value of k is 2. Also the 10-fold cross validation test option is employed for training the classification model. Table 3.5 depicts the resulting confusion matrix of this model.

Total number of instances (training sets)	Correctly classified Instances	Incorrectly Classified Instances
4,273	3,933(92.04%)	340(7.96%)

Table 3.5: Classification accuracy using K value 2

The detailed classification accuracy for each transportation mode is showed in the table 3.6

	TP Rate	FP Rate	Precision	Recall	F-Measure	ROC Area	Class
	0.966	0.025	0.889	0.966	0.926	0.979	Car
	0.785	0.009	0.905	0.785	0.841	0.969	Bus
	0.97	0.054	0.832	0.97	0.896	0.973	Train
	0.605	0.007	0.909	0.605	0.726	0.932	Bike
	1	0	1	1	1	1	Walk
	0.957	0.001	0.974	0.957	0.965	0.997	Subway
Weighted Average	0.92	0.018	0.924	0.92	0.917	0.98	

Table 3.6: Classification accuracy for each modes of transportation with k=2

As shown in the resulting confusion matrix, the KNN classification algorithm with value of K=2 scored an accuracy of 92.04 %. This result shows that out of the total training datasets 4,273 (92.04 %) records are correctly classified, while 340 (7.96 %) of the records are incorrectly classified. Resulting confusion matrix is presented in Appendix B.

3.5.2.3. Experimentation 3: with K value 5

The third experimentation is performed with the default parameter that is the value of k is 5. Also the 10-fold cross validation test option is employed for training the classification model. Table 3.7 depicts the resulting confusion matrix of this model.

Total number of instances (training sets)	Correctly classified Instances	Incorrectly Classified Instances
4,273	4,150(97.12%)	123(2.88%)

Table 3.7: Classification accuracy using K value 5

The detailed classification accuracy for each transportation mode is showed in the table 3.8

	TP Rate	FP Rate	Precision	Recall	F-Measure	ROC Area	Class
	0.975	0.005	0.975	0.975	0.975	0.996	Car
	0.945	0.006	0.949	0.945	0.947	0.989	Bus
	0.968	0.014	0.95	0.968	0.959	0.997	Train
	0.898	0.008	0.933	0.898	0.915	0.994	Bike
	1	0	1	1	1	1	Walk
	0.983	0.001	0.974	0.983	0.979	1	Subway
Weighted Average	0.971	0.005	0.971	0.971	0.971	0.971	

Table 3.8: Classification accuracy for each modes of transportation with k=5

As shown in the resulting confusion matrix, the KNN classification algorithm with value of K=5 scored an accuracy of 97.12%. This result shows that out of the total training datasets 4,150(97.12 %) records are correctly classified, while 123(2.88%) of the records are incorrectly classified. Resulting confusion matrix is presented in Appendix C.

3.5.2.4. Experimentation 4: with K value 10

The fourth experimentation is performed with the default parameter that is the value of k is 10. Also the 10-fold cross validation test option is employed for training the classification model. Table 3.9 depicts the resulting confusion matrix of this model.

Total number of instances (training sets)	Correctly classified Instances	Incorrectly Classified Instances
4,273	4,206 (98.43%)	67 (1.57%)

Table 3.9: Classification accuracy using K value 10

The detailed classification accuracy for each transportation mode is showed in the table 3.10

	TP Rate	FP Rate	Precision	Recall	F-Measure	ROC Area	Class
	0.988	0.003	0.986	0.988	0.987	0.998	Car
	0.961	0.005	0.959	0.961	0.96	0.996	Bus
	0.994	0.008	0.973	0.994	0.983	0.999	Train
	0.944	0.002	0.986	0.944	0.965	0.998	Bike
	1	0	1	1	1	1	Walk
	0.961	0.002	0.97	0.961	0.965	1	Subway
Weighted Average	0.984	0.003	0.984	0.984	0.984	0.999	

Table 3.10: Classification accuracy for each modes of transportation with k=10

As shown in the resulting confusion matrix, the KNN classification algorithm with value of K=10 scored an accuracy of 98.43%. This result shows that out of the total training datasets

4,206(98.43%) records are correctly classified, while 67(1.57%) of the records are incorrectly classified. Resulting confusion matrix is presented in Appendix D.

3.5.2.5. Experimentation 5: with K value 11

The fifth experimentation is performed with the default parameter that is the value of k is 11. Also the 10-fold cross validation test option is employed for training the classification model. Table 3.11 depicts the resulting confusion matrix of this model.

Total number of instances (training sets)	Correctly classified Instances	Incorrectly Classified Instances
4,273	4,209 (98.5%)	64 (1.5%)

Table 3.11: Classification accuracy using K value 11

The detailed classification accuracy for each transportation mode is showed in the table 3.12

	TP Rate	FP Rate	Precision	Recall	F-Measure	ROC Area	Class
	0.985	0.002	0.99	0.985	0.988	0.998	Car
	0.959	0.005	0.957	0.959	0.958	0.997	Bus
	0.992	0.006	0.979	0.992	0.986	1	Train
	0.957	0.002	0.984	0.957	0.97	0.999	Bike
	1	0	1	1	1	1	Walk
	0.966	0.003	0.953	0.966	0.959	1	Subway
Weighted Average	0.985	0.002	0.985	0.985	0.985	0.999	

Table 3.12: Classification accuracy for each modes of transportation with k=10

As shown in the resulting confusion matrix, the KNN classification algorithm with value of K=11 scored an accuracy of 98.5%. This result shows that out of the total training datasets 4,209(98.5%) records are correctly classified, while 64(1.5%) of the records are incorrectly classified. Resulting confusion matrix is presented in Appendix E.

3.5.2.6. Experimentation 6: with K value 12

The sixth experimentation is performed with the default parameter that is the value of k is 12. Also the 10-fold cross validation test option is employed for training the classification model. Table 3.13 depicts the resulting confusion matrix of this model.

Total number of instances (training sets)	Correctly classified Instances	Incorrectly Classified Instances
4,273	4,201 (98.32%)	72 (1.68%)

Table 3.13: Classification accuracy using K value 12

As shown in the resulting confusion matrix, the KNN classification algorithm with value of K=12 scored an accuracy of 98.32%. This result shows that out of the total training datasets 4,201(98.32%) records are correctly classified, while 72(1.68%) of the records are incorrectly classified. The detailed classification accuracy is shown at appendix F.

3.5.2.7. Experimentation 7: with K value 15

The seventh experimentation is performed with the default parameter that is the value of k is 15. Also the 10-fold cross validation test option is employed for training the classification model. Table 3.14 depicts the resulting confusion matrix of this model.

Total number of instances (training sets)	Correctly classified Instances	Incorrectly Classified Instances
4,273	4,195 (98.17%)	78 (1.83%)

Table 3.14: Classification accuracy using K value 15

As shown in the resulting confusion matrix, the KNN classification algorithm with value of K=15 scored an accuracy of 98.17%. This result shows that out of the total training datasets 4,195(98.17%) records are correctly classified, while 78(1.83%) of the records are incorrectly classified. The detailed classification accuracy is shown at appendix G.

3.5.2.8. Experimentation 8: with K value 20

The eighth experimentation is performed with the default parameter that is the value of k is 20. Also the 10-fold cross validation test option is employed for training the classification model. Table 3.15 depicts the resulting confusion matrix of this model.

Total number of instances (training sets)	Correctly classified Instances	Incorrectly Classified Instances
4,273	4,187 (97.99%)	86 (1.01%)

Table 3.15: Classification accuracy using K value 20

As shown in the resulting confusion matrix, the KNN classification algorithm with value of K=20 scored an accuracy of 97.99%. This result shows that out of the total training datasets 4,187(97.99%) records are correctly classified, while 86(1.01%) of the records are incorrectly classified. The detailed classification accuracy is shown at appendix H.

3.5.2.9. Experimentation 9: with K value 30

The ninth experimentation is performed with the value of k is 30. Also the 10-fold cross validation test option is employed for training the classification model. Table 3.16 depicts the resulting confusion matrix of this model.

Total number of instances (training sets)	Correctly classified Instances	Incorrectly Classified Instances
4,273	4,185 (97.94%)	88 (2.06%)

Table 3.16: Classification accuracy using K value 30

As shown in the resulting confusion matrix, the KNN classification algorithm with value of K=30 scored an accuracy of 97.94%. This result shows that out of the total training datasets 4,185(97.94%) records are correctly classified, while 88(2.06%) of the records are incorrectly classified.

3.5.2.10. Experimentation 10: with K value 60, 100, 150, 200

Other experimentations are performed with the value of k= 60, 100, 150, 200. Also the 10-fold cross validation test option is employed for training the classification model. Table 3.17 depicts the resulting confusion matrix of this model.

K Value	Total number of instances (training sets)	Correctly classified Instances	Incorrectly Classified Instances
60	4,273	4,131 (96.68%)	142(3.32%)
100	4,273	4,031 (94.34%)	242(5.66%)
150	4,273	3978(93.1%)	295(6.9%)
200	4,273	3,744(87.62%)	529(12.38%)

Table 3.17: Classification accuracy using K value 30

As shown in the resulting confusion matrix, the KNN classification algorithm with value of K=60 scored an accuracy of 96.68%, with k=200 scored an accuracy of 87.62%. The detailed classification accuracy for K value 100 is shown at appendix I.

3.5.3. Train the classifier using J48 decision tree modeling

As described earlier, the J48 algorithm is used for building the decision tree model. The training of the decision tree classification models of the experimentation is done by employing the 10-fold cross validation and the percentage split classification models.

J48 algorithm contains some parameters that can be changed to further improve classification accuracy. Initially the classification model is built with the default parameter values of the J48 algorithm. Table 3.18 summarizes the default parameters with their values for the J48 decision tree algorithm.

parameter	Description	Default Value
ConfidenecFactor	The confidence factor used for pruning (smaller values incur more pruning)	0.25
minNumObj	The minimum number of instances per leaf	2
Unpruned	Weather pruning is performed	False
Subtreeraising	Weather sub tree information is hidden or expanded	True

Table 3.18: Some of the J48 algorithm parameters and their default values

As described before, the J48 algorithm is used for building the decision tree model. The training of the decision tree classification models of the experimentation is done by employing the 10-fold cross validation. Table 3.19 depicts the resulting confusion matrix of this model.

Total number of instances (training sets)	Correctly classified Instances	Incorrectly Classified Instances
4,273	4,124 (96.5%)	149 (3.5%)

Table 3.19: Using J48 algorithm parameters with 10-fold cross validation

As shown in the resulting confusion matrix, the J48 learning algorithm scored an accuracy of 96.5%. This result shows that out of the total training datasets 4,124 (96.5%). records are correctly classified, while 149 (3.5%) of the records are incorrectly classified.

3.5.4. Train the classifier using Naive Bayes modeling

The other classification data mining technique employed for the classification sub phase of this research is the Naïve Bayes. To build the Naïve Bayes model, WEKA software package is used and it employs the Naïve Bayes Simple algorithm in developing the model. The 10-fold cross validation, which is set by default for training and testing the model test options are employed.

Naive Bayes is one of the Probabilistic learning algorithm that Calculate explicit probabilities for hypothesis. In Naïve Bayes learning algorithm prior knowledge and observed data can be combined. It is also an incremental learning algorithm that each training example can incrementally increase/decrease the probability that a hypothesis is correct. It predicts multiple hypotheses, weighted by their probabilities.

Table 3.20 showed the resulting confusion matrix of the model developed using the Naive Bayes Simple algorithm with the default 10-fold cross validation test option.

Total number of instances (training sets)	Correctly classified Instances	Incorrectly Classified Instances
4,273	3,518 (82.3%)	755 (17.3%)

Table 3.20: Classification accuracy using Naïve Bayes algorithm

As shown in the above confusion matrix the Naïve Bayes Simple Algorithm scored an accuracy of 82.3%. This means out of the total 4,273 records 3,518 (82.3%) of the records are correctly classified, while 755(17.3 %) of the records are misclassified.

3.5.4: Comparison of Supervised Approaches: J48 decision tree, Naive Bayes model and KNN

Comparing different classification techniques and selecting the best model for predicting the transportation mode is one of the aims of this study. Summary of experimental result for the three two classification algorithms is presented in table 3.21 below:

Classifier/ Model	**Correctly Classified**	**Incorrectly Classified**
J-48 decision tree	96.5%	3.5%
Navi-Bayes	82.3 %	17.3%

KNN- with K value 11	98.5%	1.5%

Table 3.21: Comparison of Supervised Approaches

In this thesis the three algorithms performed different prediction accuracy. Accordingly the KNN algorithm with k value 11 scored better classification accuracy showed in table 3.11.

The reason for the KNN performing better than others is because of absence of linearity nature of the dataset. This means there is no a comprehensible segregation point that can be defined by the algorithm to predict the class of a particular transportation modes.

CHAPTER 4

RESULTS AND DISCUSSION

Comparing different results based on the value of K and selecting the best model for predicting the transportation mode is the major aims of this study. The total numbers of instances considered for the classification were 4,273. Summary of experimental result for some selected K values is presented in table 4.1 below

K-Value	Correctly classified Instances (Accuracy)	Incorrectly Classified Instances
1	3,882 (90.85 %)	391 (9.15%)
2	3,933(92.04%)	340(7.96%)
5	4,150(97.12%)	123(2.88%)
10	4,206 (98.43%)	67 (1.57%)
11	4,209 (98.5%)	64 (1.5%)
12	4,201 (98.32%)	72 (1.68%)
15	4,195 (98.17%)	78 (1.83%)
20	4,187 (97.99%)	86 (1.01%)
30	4,185 (97.94%)	88 (2.06%)
60	4,131 (96.68%)	142(3.32%)
100	4,031 (94.34%)	242(5.66%)
150	3978(93.1%)	295(6.9%)
200	3,744(87.62%)	529(12.38%)

Table 4.1: Results of classification accuracy with different K values

Accordingly the K-nearest neighbor classification with k value 11 scores the best classification accuracy. As shown in the table with smallest value of K the classification result is very low and as K increases the classification Accuracy increases. The problem happens when the value of K is getting larger. Therefore the highest classification result is at some point in this case at K= 11. Graphically is looks like the following:

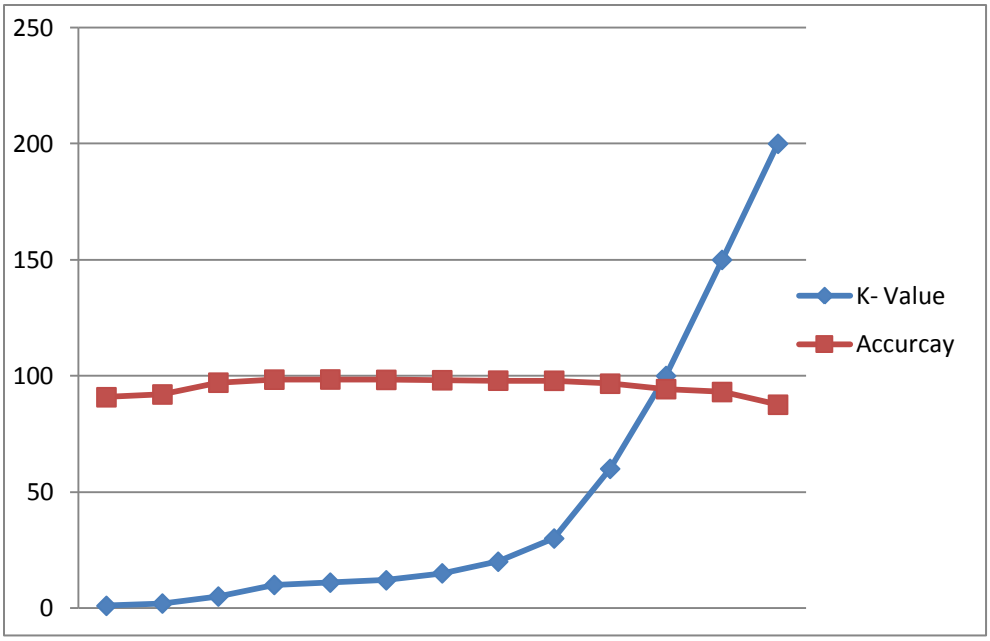

Fig: 4.1: Results of classification accuracy with different K values

As showed in figure 4.1, when k-value (blue line) increases the accuracy value (red line) decreases.

As shown in the graph as K increases, the classification accuracy goes up, then stabilizes, and then down again. The average TP and FP rates for all experiments conducted in this study showed in table 4.2.

K-Values	TP	FP
1	0.908	0.017
2	0.92	0.018
5	0.971	0.005
10	0.984	0.003
11	0.985	0.002
12	0.983	0.003
15	0.982	0.003
20	0.98	0.004
30	0.979	0.003
60	0.967	0.005
100	0.943	0.011
150	0.931	0.015
200	0.876	0.029

Table 4.2: Average TP and FP Rates

As shown in the above table for the best classification accuracy there should be the highest the True Positive value. In this case for K= 11, the TP value is the highest one. The following figure shows that the variation of the TP at different K values.

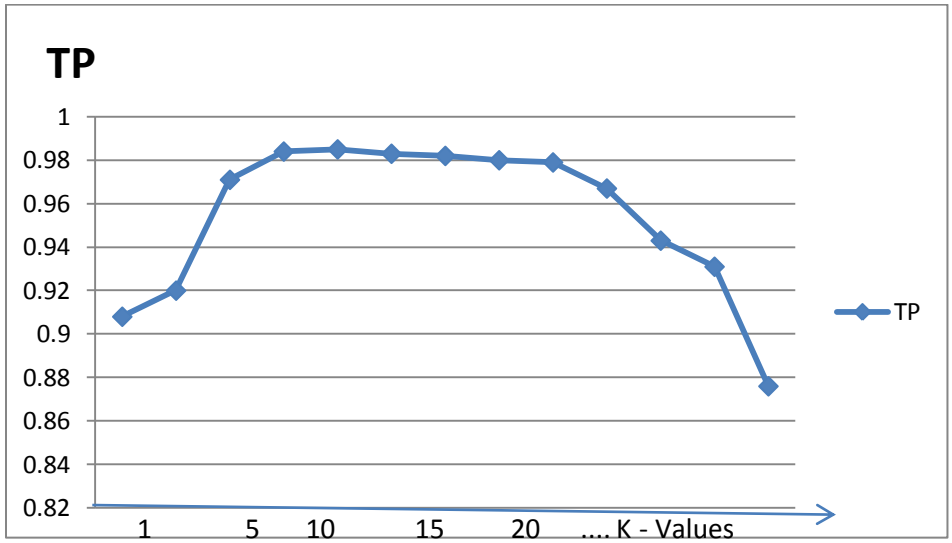

Figure 4.2: True Positive (TP) rate comparison of different K values

As shown in figure 4.2 at the lower k –values the TP values increasing but after a while the result is decreasing. This indicates the highest accuracy values means the TP value is highest too.

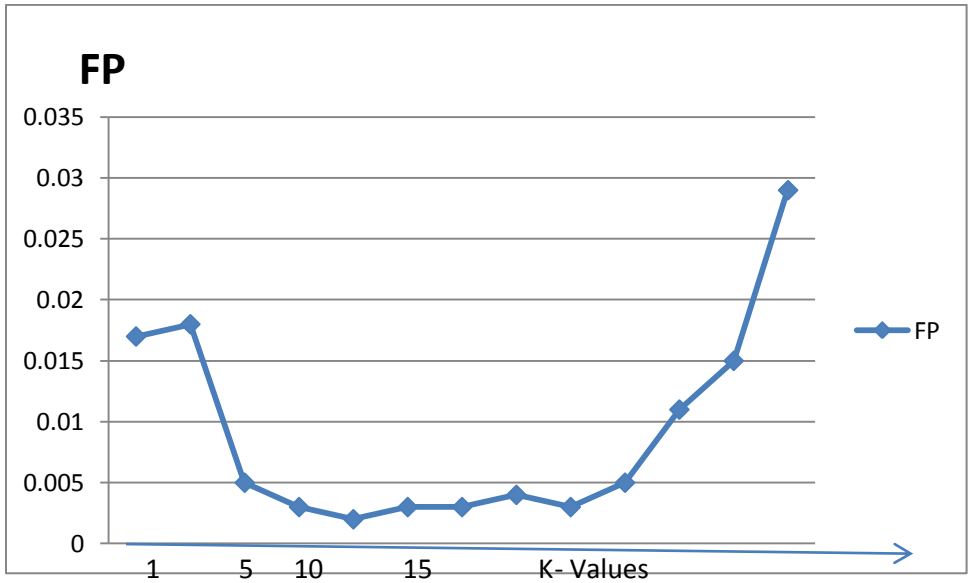

Figure 4.3: False Positive (FP) rate comparison of different K – Values

For a good a good classification algorithm the FP rate should be low. As shown in figure 4.3 the FP rate of the KNN classification result is lowest is at K=11

The selected model for this study is validated by unlabeled datasets which means records without having transportation modes. The prediction performance of this model is tested using a java code by using either Disk Operating System (DOS) or Simple Command line Interface (SCLI) on WEKA.

The command used for training purpose of the KNN algorithm is the following:

>Java weka.classifiers.lazy.IBk –[ConditionsForK][PATH]\Training_data_Name.arff -d [destination path]\Model_Name.model

>weka.classifiers.lazy.IBk -K 11 -W 0 -A C:\Users\Mr.Tigabu\Desktop \UnlabeledDatasets.arff –d C: \Users\Mr.Tigabu\Desktop\selectedmodel.model

The command used for testing purpose of the KNN algorithm is the following:

Java weka.classifiers.lazy.IBk –p [path]\Model_Name.model -T [destination path] \Testing_data_Name.csv

>Java weka.classifiers.lazy.IBk –p:\Users\Mr.Tigabu\Desktop \selectedmodel.model -T C: \Users\Mr. Tigabu\Desktop\ real_life_data.csv

Using the selected model the unlabeled 1,018 datasets were classified as shown in the table 4.3. The validation records were taken from the same area near Beijing city square.

Labeled dataset-Transportation-mode	Number of Records Collected
Bike	187
Bus	171
Car	174
Subway	98
Train	119
Walk	269

Table 4.3: Classification result of unlabeled datasets based on the selected model

The performance of the selected model is tested with the separately prepared 1,018 validating datasets as shown in the table 4.4:

Total number of instances (Validating tests)	Correctly classified Instances	Incorrectly Classified Instances
1,018	948 (93.12%)	70 (6.88%)

Table 4.4: Validating the selected model with real life

The performance of the model on the test set was 93.12% to classify the new instances as bike, bus, car, subway, train and walk classes. This result showed that out of the 1.018 testing datasets, the developed KNN classification model predicted around 948 (93.12%) records correctly. The model has a prediction accuracy of 98.5% on the training datasets when the value of K is 11. There may different reasons why the performance of the model was lower during the testing phases. According to [51] the errors committed by a classification model are generally divided into two types: training errors and generalization errors. Training error is the number of misclassification error committed on training records, whereas generalization error is the expected error of the model on previously unseen records. Here, the training error which is the FP rate was high for the training phase of the model in this study. The lower performance registered also due to lack of representative Samples in the model training.

CHAPTER 5

CONCLUSION AND FUTURE WORKS

5.1. CONCLUSION

Nowadays, a vast amount of spatio-temporal data are being generated by devices like cell phones, GPS and remote sensing devices and therefore discovering interesting patterns in such data became an interesting topics for researchers. One of the research area is discovering the transportation modes from GPS generated information.

The increasing frequency of location-acquisition technology like the Global Positioning System is leading to the collection of large spatio-temporal datasets and to the prospect of discovering usable knowledge about movement behavior, which encourages for the discovery of interesting relationships and characteristics users that may exist implicitly in spatial databases. Therefore spatial data mining is emerging as a novel area of research.

In this study, attempts have been made to use spatial data mining technology for identifying users travel mode depending on their travel behaviors. The KDD process model designed by [10] has been followed during undertaking the experimentation and discussion. A series of chronologically ordered points, e.g., p1→p2→…→ pn, where each point consists of a geospatial coordinate set and a timestamp such as p = (x;y; t) were considered for the study. Using serious points and timestamp, speed was calculated. And speed used as classification criteria.

The KDD process model designed by [7] has been followed during undertaking the experimentation and discussion. The dataset used in this study has been taken from Microsoft Research Asia Geolife project. After taking the data, it has been preprocessed and prepared in a format suitable for the DM tasks.

A total of 4,273 trajectory dataset are used for training the models. For validating the performance of the selected model a separate 1,018 records are used as a testing set. For building a spatiotemporal model of this study the K-nearest Neighbors (KNN) algorithm has been tasted as supervised approach. The k-nearest neighbor determines the predicted label by asking the *k*-nearest neighbor points in the training set to “vote” for the label.

The model that was created using 10-fold cross validation with K value 11 and other default parameter values showed the best classification accuracy. The model has a prediction accuracy of 98.5% on the training datasets and 93.12% on the test dataset to classify the new instances as bike, bus, car, subway, train and walk classes.

5.2. FUTURE WORKS

The result of the study has shown that the KNN supervised with cross-validation test mode and K-value 11 is appropriate for predicting transportation mode. Hence, based on the findings of this study, the following are recommended as future research directions:

- ✓ This study was carried out using KNN supervised algorithm. So further investigation needs to be done using other supervised algorithms such as Decision Tree, Neural Networks and Support Vector Machine.
- ✓ This study was conducted on a limited datasets. So applying the KNN algorithm with the whole records and compare the results how it could be.
- ✓ This study considers consecutive points with the given time interval. Taking a common time interval for the whole datasets and testing the algorithm.
- ✓ Reclassifying the classified results and comparing the result.
- ✓ Cross validation used as a validation techniques for this study. Considering other validation mechanism and comparing the result is also other future study area.

REFERENCES

[1] Slava Kisilevich, Florian Mansmann, Mirco Nanni, Salvatore Rinzivillo, Spatio-Temporal Clustering: a Survey Report, ISTI-CNR, Data Mining and Knowledge Discovery Handbook, Springer, Italy, 2011

[2] http://research.microsoft.com/en-us/downloads/b16d359d-d164-469e-9fd4-daa38f2b2e13/, accessed date July 19, 2014

[3] Yu Zheng, Lizhu Zhang, Xing Xie, Wei-Ying Ma. Mining interesting locations and travel sequences from GPS trajectories, in Proceedings of International conference on World Wild Web (WWW 2009), Madrid Spain, ACM Press: 791-800.

[4] Yu Zheng, Quannan Li, Yukun Chen, Xing Xie, Wei-Ying Ma. Understanding Mobility Based on GPS Data. In Proceedings of ACM conference on Ubiquitous Computing (UbiComp 2008), Seoul, Korea. ACM Press: 312-321.

[5] Yu Zheng, Xing Xie, Wei-Ying Ma, GeoLife: A Collaborative Social Networking Service among User, location and trajectory. Invited paper, in IEEE Data Engineering Bulletin. 33, 2, 2010, pp. 32-40.

[6] Miao Lin , Wen-Jing Hsu " Mining GPS data for mobility patterns: A survey", Elsevier, 2013

[7]Fayyad U., Piatetsky G., and Smyth P.," The KDD process for Extracting Useful Knowledge from Volumes of Data", Communications of the ACM, Vol. 39, PP. 27-34, 1996.

[8] Mahbod T., Ebrahim B., Wei L., and Ali A., "A detailed analysis of the KDD CPU 99 Dataset", proceedings of 2009 of the IEEE Symposium on computational Intelligence in Security and Defense Applications, National Research Council, PP.1-6, Canada, 2009

[9]Linsey Xiaolin Pang , Sanjay Chawla , Wei Liu and Yu Zheng , "On detection of emerging anomalous traffic patterns using GPS data", Elsevier B.V., 2013

[10] Fayyad U., Piatetsky G., and Smyth P.," The KDD process for Extracting Useful Knowledge from Volumes of Data", Communications of the ACM, Vol. 39, PP. 27-34, 1996.

[11] Nanda A, "Data Mining and Knowledge Discovery in Database: An AI perspective", Proceedings of national Seminar on Future Trends in Data Mining, 2010

[12] Michael C. Mozer. Neural net architectures for temporal sequence processing. In A. Weigend and N. Gershenfeld, editors, Predicting the Future and Understanding the Past. Addison-Wesley, Reading, MA, 2007.

[13] Gideon Berger and Alexander Tuzhilin. Discovering unexpected patterns in temporal data using temporal logic. In Temporal Databases, Dagstuhl, pages 281–309, 1997.

[14] Chapman P., Clinton J., Kerber R., Khabaza T., Reinartz T., Shearer C., and Wirth R., "CRISPDM 1.0 step-by-step data mining guide", Technical report, CRISP-DM, 2003

[15] Piatetsky S. and Frawley W., "Knowledge Discovery in Databases", AAAI/ MIT Press, MA, 1991

[16] Martínez d., "Optimización Mediante Técnicas de Minería de Datos Del Ciclo de Recocido de Una Línea de Galvanizado, PhD thesis, Universidad de LaRioja, 2003

[17] Yang Q., and Wu X., "10 Challenging Problems in Data Mining Research", International Journal of Information Technology and Decision Making, World Scientific Publishing Company , Vol. 5, No. 4, PP.597–604., 2006

[18] Dunham H., "Data mining introductory and advanced topics", Upper Saddle River, NJ: Pearson Education, Inc, 2003

[19] Charles E., "The foundations of cost-sensitive learning". In Proceedings of the Seventeenth International Joint Conference of Artificial Intelligence, Morgan Kaufmann, Seattle, Washington, PP. 973–978, 2001

[20] Frank J., "Artificial intelligence: Current and future directions", In Proc. of the 17th National Computer Security Conference, Baltimore, MD. National Institute of Standards and Technology (NIST), 1994

[21] Marin J., Ragsdale D. and Surdu J., "A hybrid approach to profile creation and intrusion detection", In Proc. of DARPA Information Survivability Conference and Exposition, Anaheim, CA. IEEE Computer Society, 2001

[22] Eskin E., Arnold A., Preraua M., Portnoy L., and Stolfo S., "A geometric framework for unsupervised anomaly detection: Detecting intrusions in unlabeled data", Data Mining for Security Applications, Kluwer Academic Publishers, Boston, 2002

[22] Berson A., Smith S. and Thearling K., "Building Data Mining Applications for CRM", McGraw-Hill Professional Publishing, New York, USA, 2000

[23] Chaudhuri S. "Data Mining and Database Systems: Where is the Intersection?" IEEE Bulletin of the Technical Committee on Data Engineering, Vol. 21, No.1, PP. 4-8, 1998

[24] Two Crows Corporation, "Introduction to Data Mining and Knowledge Discover", 3rd edition, MD 20854(USA), 2005

[25] Mannila, Toivoenn and Verkamo, "Discovering Frequent Episodes in Sequences", In proceedings of the First International Conference on Knowledge Discovery and Data Mining", American Association for Artificial Intelligence Press, PP. 201-215, 1995

[26] Venkateswara Rao, Govardhan A. and Chalapati Rao "SPATIOTEMPORAL DATA MINING: ISSUES, TASKS AND APPLICATIONS", International Journal of Computer Science & Engineering Survey (IJCSES) Vol.3, No.1, February 2012

[27] Bohte, W. and K. Maat (2009). Deriving and validating trip purposes and travel modes for multi-day GPS-based travel surveys: A large-scale application in the Netherlands. Transportation Research Part C 17, 285–297

[28] Meera G., Gandhi and Srivatsa S, "Adaptive Machine Learning Algorithm (AMLA) Using J48 Classifier", Advances in Computational Sciences and Technology, Vol. 3, PP. 291–304, 2010

[29] D. Aha, D. Kibler, "Instance-based learning algorithms". Machine Learning , Vol 6, 37-66, 1991

[30] Claudia M Antunes and Arlindo L Oliveira. Temporal data mining: An overview. KDD Workshop on Temporal Data Mining, 2001

[31] John F. Roddick and Myra Spiliopoulou. A survey of temporal knowledge discovery paradigms and methods. IEEE Trans. on Knowl. and Data Eng., 14(4):750–767, ISSN 1041-4347, July 2002.

[32] George Edward Pelham Box and Gwilym Jenkins. Time Series Analysis, Forecasting and Control. Holden-Day, Incorporated, 1990. ISBN 0816211043

[33] Chris Chatfield. The analysis of time series: an introduction. CRC Press, Florida, US, 6th edition, 2004.

[34] Laks V. S. Lakshmanan, Raymond T. Ng, Jiawei Han, and Alex Pang. Exploratory mining and pruning optimizations of constrained association rules. In SIGMOD Conference, pages 13–24. ACM Press, ISBN 0-89791-995-5, 1998.

[35] Sankar K. Pal, Sushmita Mitra, C. A. Murthy, P. S. Sastry, and Santanu Chaudhury, editors. Pattern Recognition and Machine Intelligence, Third International Conference, PReMI 2009, New Delhi, India, December 16-20, 2009 Proceedings, volume 5909 of Lecture Notes in Computer Science,. Springer. ISBN 978-3-642-11163-1, 2009.

[36] D. J. Hand and K. Yu. Idiots bayes not so stupid after all?., 2001.

[37] Ross J. Quinlan. C4.5: Programs for Machine Learning. Morgan Kaufmann Publishers Inc., San Francisco, CA, USA, 1993.

[38] Antony Browne and Shuang Yang. Neural network ensembles: combining multiple models for enhanced performance using a multistage approach. Expert Systems, 21(5):279–288, 2004. ISSN 1468-0394. doi: 10.1111/j.1468-0394.2004.00285.x.
URL: http://dx.doi.org/10.1111/j.1468-0394.2004.00285.x , accessed date April 24, 2015

[39] Frank Rosenblatt. The perceptron: A probabilistic model for information storage and organization in the brain. Psychological Review, 65(6):386–408, 1958.

[40] Berkhin P., “Survey of clustering data mining techniques”, Tech. rep., Accrue Software, San Jose, CA, 2002

[41] Stopher, P. R, "Collecting and processing data from mobile technologies", in the 8th international conference on survey methods in transport, Annecy, France, 2008.

[42] Kautz, L. L, "Learning and Inferring Transportation Routines". ARTIFICIAL INTELLIGENCE, 311-331, 2007

[43] Vladimir N. Vapnik, B. E, "A Training Algorithm for Optimal Margin Classifiers", 1992

[44] Statsoft, I, " Statsoft Electronic Statistics Textbook", 2007. Accessed date April 25, 2015, from Naive Bayes: http://www.statsoft.com/textbook/naive-bayes-classifier/

[45] Noble, W. S, "What is a support vector machine?" NATURE BIOTECHNOLOGY, 1565-1567, 2006

[46] Mrutyunjaya P. "Evaluating Machine Learning Algorithms", International Journal of Recent Trends in Engineering, Vol. 1, No 1, PP. 472-477, 2009

[47] Farhan A., Zulkhairi M., Dahalin and Shaidah J., "Distributed and Cooperative Hierarchical Intrusion Detection on MANETs", International Journal of Computer Applications, Vol. 12, No.5, PP. 33-40, 2010

[48] http://www.movable-type.co.uk/scripts/latlong.html, accessed date may 17, 2015

[49] Witten, I.H. and Frank, E, "Data Mining: Practical Machine Learning Tools and Techniques" , 2nd ed., Morgan Kaufmann. San Francisco, 2005

[50] Aha, D.W., Kibler, D. and Albert, M.K, "Instance-based learning algorithms", Mach. Learn., Vol. 6, pp.37–66, 1991

[51] Shilendra K. and Preeti J., "Effective movement detection using Rough Set Theory and Support Vector Machine", Vol. 18, No. 3, 2011

APPENDIXES

Appendix A: Resulting confusion matrix

Classified as	Car	Bus	Train	Bike	Walk	Subway
Car	665	62	0	0	0	0
Bus	56	373	0	0	0	8
Train	0	0	801	127	0	0
Bike	0	1	134	328	0	0
Walk	0	0	0	0	1486	0
Subway	0	3	0	0	0	229

Table A-1: Confusion Matrix using k value of 1

Appendix B: Resulting confusion matrix

Classified as	Car	Bus	Train	Bike	Walk	Subway
Car	702	25	0	0	0	0
Bus	88	343	0	0	0	6
Train	0	0	900	28	0	0
Bike	0	1	182	280	0	0
Walk	0	0	0	0	1486	0
Subway	0	10	0	0	0	222

Table B-1: Confusion Matrix using k value of 2

Appendix C: Resulting confusion matrix

Classified as	Car	Bus	Train	Bike	Walk	Subway
Car	709	18	0	0	0	0
Bus	18	413	0	0	0	6
Train	0	0	898	30	0	0
Bike	0	0	47	416	0	0
Walk	0	0	0	0	1486	0
Subway	0	4	0	0	0	228

Table C-1: Confusion Matrix using k value of 5

Appendix D: Resulting confusion matrix

Classified as	Car	Bus	Train	Bike	Walk	Subway
Car	718	9	0	0	0	0
Bus	10	420	0	0	0	7
Train	0	0	922	6	0	0
Bike	0	0	26	437	0	0
Walk	0	0	0	0	1468	0
Subway	0	9	0	0	0	223

Table D-1: Confusion Matrix using k value of 10

Appendix E: Resulting confusion matrix

Classified as	Car	Bus	Train	Bike	Walk	Subway
Car	716	11	0	0	0	0
Bus	7	419	0	0	0	11
Train	0	0	921	7	0	0
Bike	0	0	20	443	0	0
Walk	0	0	0	0	1486	0
Subway	0	8	0	0	0	224

Table E-1: Confusion Matrix using k value of 11

Appendix F: Detailed classification accuracy for K value 12

	TP Rate	FP Rate	Precision	Recall	F-Measure	ROC Area	Class
	0.985	0.002	0.99	0.985	0.988	0.998	Car
	0.959	0.006	0.95	0.959	0.954	0.998	Bus
	0.994	0.008	0.973	0.994	0.983	1	Train
	0.944	0.002	0.986	0.994	0.965	0.999	Bike
	1	0	1	1	1	1	Walk
	0.953	0.003	0.953	0.953	0.953	1	Subway
Weighted Average	0.983	0.003	0.983	0.983	0.983	0.999	

Table F.1: Classification accuracy for each modes of transportation with k=12

Appendix G: Detailed classification accuracy for K value 15

	TP Rate	FP Rate	Precision	Recall	F-Measure	ROC Area	Class
	0.982	0.002	0.992	0.982	0.987	0.998	Car
	0.934	0.005	0.958	0.934	0.946	0.998	Bus
	0.992	0.007	0.975	0.992	0.983	1	Train
	0.948	0.002	0.984	0.948	0.966	0.999	Bike
	1	0	1	1	1	1	Walk
	0.978	0.006	0.908	0.978	0.942	1	Subway
Weighted Average	0.982	0.003	0.982	0.982	0.982	0.999	

Table G.1: Classification accuracy for each modes of transportation with k=15

Appendix H: Detailed classification accuracy for K value 20

	TP Rate	FP Rate	Precision	Recall	F-Measure	ROC Area	Class
	0.978	0.002	0.992	0.978	0.985	0.999	Car
	0.936	0.004	0.962	0.936	0.949	0.999	Bus
	0.998	0.012	0.959	0.998	0.978	0.999	Train
	0.914	0.002	0.995	0.914	0.953	0.998	Bike
	1	0	1	1	1	1	Walk
	1	0.005	0.913	1	0.955	0.999	Subway
Weighted Average	0.98	0.004	0.981	0.98	0.98	0.999	

Table H.1: Classification accuracy for each modes of transportation with k=20

Appendix I: Detailed classification accuracy for K value 200

	TP Rate	FP Rate	Precision	Recall	F-Measure	ROC Area	Class
	0.816	0	1	0.816	0.898	0.998	Car
	0.744	0.035	0.708	0.744	0.725	0.982	Bus
	0.939	0.068	0.794	0.939	0.86	0.985	Train
	0.512	0.015	0.806	0.512	0.626	0.973	Bike
	1	0.023	0.958	1	0.979	1	Walk
	1	0.012	0.832	1	0.908	0.998	Subway
Weighted Average	0.876	0.029	0.881	0.876	0.871	0.992	

Table I.1: Classification accuracy for each modes of transportation with k=200