The Application of Semi-supervised Machine Learning Algorithms in Parkinson's Disease Analysis

Author:

Huichuan LIU

Supervisor:

Dr. Simon HARPER

August 16, 2016

Contents

List of Figures

List of Tables

Abstract

This dissertation represents the feasibility of deploying an automatic tracker of Parkinson's Disease (PD) on smartphones. According to the report of The Royal Free Hospital NHS Trust (2008), Parkinson's Disease has become one of the most prevalent neurodegenerative disorder in the Western Europe. However, most diagnoses of PD progress remain in a conventional way, i.e. regular visiting clinics and diagnosed by physicians, which is rather expensive and inconvenient for patients with motor impairment. Hence, the combination of special equipment and machine learning technologies have been developed to accomplish automatic measuring of PD symptoms. However, the applications are inevitably limited by the cost of extra equipment and the requirement of training labels. A potential solution is to introduce smartphones to replace professional devices. Nevertheless, there are still barriers and challenges in practice scenes. As a pilot research in this field, a series of experiments are set in this project to explore three specific problems concerning PD tremor assessment, namely the strategy of data preprocess, the performance of semi-supervised learning algorithms and the assessments of multiple data features.

In the first part, we propose a strategy of data reduction and some signal processes smartphone data, aiming at reducing the computational cost and improving the data quality. Then three semi-supervised learning algorithms are implemented, evaluated and compared regarding the parameter effectiveness and overall performance. Last but not least, features used in different models are assessed by Principle Component Analysis (PCA) respectively in the final stage to reveal the relevance of features and models.

Chapter 1

Introduction

Parkinson's Disease (PD), also known as idiopathic or primary Parkinsonism, is a progressive neurological disorder caused by cell loss in the substantia nigra. According to the survey by National Collaborating Centre for Chronic Conditions (Great Britain) in 2006, the prevalence of PD was around 100-180 per 100,000 of the population in the UK[1]. Parkinson's Disease have been recognized as an illness that significantly impairs patients' motor ability and arouses sorts of non-motor discomforts[2]. Many previous studies[1, 2, 3, 4, 5] indicate that unnatural tremors is an obvious symptom of Parkinson's Disease and the situation of these tremors are closely relative to the progress of the disease. Hence, having effective and accurate measurement of tremors are important for medical research of PD and its clinical treatments of Parkinson's Disease.

In fact, a lot of methods have been proposed for PD progress measuring and scaling. The most famous ones include Hoen and Yahr[6], the Unique Parkinson's Disease Rating Scale (UPDRS)[7] and Northwestern University Disability Scale (NUDS)[5]. These methods provide questionnaires of PD symptoms and a standard for marking the answers into scores. In the last decades, these conventional ways are proved to be effective and helpful for PD diagnoses. But they are still flawed in terms of objectivity and integrity. Also, these methods literally count on frequently visiting of clinics, hence it is rather inconvenient for patients and quite expensive in human resources.

In order to achieve more effective diagnoses, more advanced instrument and techniques were involved in this area and several exciting achievements have been reported[4, 8, 9]. Motor sensors are widely used to record the motor situation of PD patients and data technology, especially machine learning techniques, were

introduced to implement automatic diagnose systems. In laboratory environment, a number of experiments have been tried to deliver automatic assessment of PD progress. But in practice scenes, there are still problems to be solve.

First of all, data collection is a difficult part for tremor monitoring because extra devices are costly and uncomfortable for some patients[10]. Secondly, the labeling work to produce training data is time-consuming and expensive. What is more, most existing systems provide only PD tremor detecting but few of them are capable of rating the progress.

To overcome these barriers, we proposed a semi-supervised learning system to assess the PD progress from smartphone data. More specifically, we try to apply preprocess to improve the data quality of smartphone sensors, which enables an more natural and economy data collection. Then a semi-supervised learning (SSL) system is implemented to resolve the shortage of labels and predicts the scores/progress of the Parkinson's Disease. For better comprehensiveness, several SSL models will be compared, including regression methods and classification methods.

1 Objectives and Goals

The overall target of this project is to explore the application of semi-supervise learning in distinguishing PD tremors with diverse severities from data collected by smartphones. We are mainly interested in how helpful the unlabelled data is and what factors would effect the semi-supervised system. In our pilot works, certain challenges have been allocated and a set of techniques will be attempted to resolve them. First, sensor signals from smartphones are of low quality but massive and the information they contain are too obscure. Hence, a preprocessing module is built to accomplish sensor signal de-nosing, data reduction and feature extraction. Next, a control experiment is set to conduct semi-supervise machine learning with processed data. Meanwhile, each model will be evaluated in terms of entropy, accuracy and ROC factors.This stage focuses on the performance of models and the impacts of parameter regulation. Finally, by computing the information gain of each feature in different models, a strategy of feature selection in different models is described.

- Data Preprocess:

- Data Cleaning: Deploying Data Smoothers to reduce impacts of interferences.

- Data Reduction: Applying rule-based filtering to capture interesting periods from massive raw data.

- Feature Extraction: Extracting signal features from acceleration data.

- Semi-supervised Machine Learning:

- Implementation: Implementing an semi-logic regression model, an semi-KNN model and a GMM model.

- Parameter Optimization: Varying the model parameters to reveal the effects of parameters.

- Evaluation: Evaluating models by system entropy, accuracy and Receiver Operating Characteristic (ROC) factors.

- Feature Assessment

- Information Gain Analysis: Training models with diverse feature combination and calculating the information gain of each feature respectively.

2   Dissertation Structure

This dissertation is composed of 7 chapters. Chapter 1 gives an overview of the motivation and challenges in PD tremor analysis with semi-supervised machine learning. In Chapter 2, we present some crucial findings of previous research that closely related to our project. Next, we compared several methods to accomplish our targets and justify our choices. Chapter 4 illustrates the system implementation and experiment details. Later in Chapter 5, we demonstrate the experiment results and critical inferences. Then Chapter 6 provides the overall summary and conclusions. In the final chapter, some limitations are pointed out and relevant future work is proposed.

3 Summary

In the first chapter, the challenges and motivation of an SSL-based tremor assessment system is illustrated. Based on that, more detailed objectives, have been listed out. Last but not least, a framework of the whole dissertation is established.

Chapter 2

Background

PD tremor analysis is a comprehensive application crossing multiple domains, including medicine, signal process and data learning. Early research of PD symptoms provided theoretical basis about descriptors of Parkinson's Disease. Based on that, certain methods were proposed and experimentally tested, a part of those delivered crucial inspiration in this field. However, a number of defects still exsist in existing systems. To overcome these defects, we involve semi-supervised learning techniques in our project.

1 Parkinson's Disease and Tremors

In general, patients with PD regularly sustain motor impairments and non-motor discomforts. Non-motor symptoms like cognitive impairment, anxious mood, and fatigue dramatically decrease the life quality of patients, but there are few frames could be used to quantify these symptoms in an analyzable way yet. In contrast, motor symptoms would be more observable and measurable by sensors, hence showing great potentials as indicators of PD progress. With the development of the disease, the motor symptoms of PD includes tremors, bradykinesia, rigidity and postural instability[2]. These motor symptoms are commonly used to scale the progress of PD in clinical diagnoses. Among all motor symptoms, PD tremors are recognized as a typical and apparent descriptor of PD progress. In comparison with other motor symptoms, PD tremors are rather intuitive in observation and quantification because it is relatively independent of other body actions. The vibration changes could be simply tracked and measured by acceleration. On the contrary, many complicate factors would be involved when

analyzing bradykinesia, rigidity and postural instability, such as activity recognition, posture identification, and environment analysis. Therefore, recognition and measurement of tremors could be significant and effective in clinical diagnoses of Parkinson's Disease[11].

However, there are multiple types of tremors appearing on human bodies. Rest tremors are regarded as a cardinal symptom whiles re-emerge tremors are also associated with PD[12]. According to Jankovic, re-emerge tremors, which are proved relevant to Parkinson's Disease, are only a part of posture tremors. A frequency-based analysis has been described in Jankovic's experiment that capable of distinguishing the re-emerge tremors from other posture tremors [12]. As for kinetic tremors, some research successfully discriminated them with the posture tremors, and they were thought to be less prevalent than the posture tremors [13, 14]. As for the difference between posture tremors and rest tremors, a previous experiment held by Koller indicated that posture tremors (92%) are more prevalent than the posture tremors (76%), and the amplitude of posture tremors are 50%higher, 25%the same and 25% less than the rest tremors. As to frequency, there is no significant difference between rest tremors and posture tremors in that experiment[15]. These explorations implied the complicity of analyzing PD tremors through naive observation and rule-base methods. Therefore, more skillful techniques are required in this domain.

In recent years, electronic sensors and machine learning methods are attempted in PD analysis and several relevant characteristics of PD tremors have been presented. The experiment of Robert et al.[16] showed significant diversity in acceleration of hands between PD patients and healthy people. In details, the mean time-averaged acceleration of PD patients was 2.4 times greater than the normal while the coefficient of variation for the PD patients was 4.1 times larger than the normal. Besides, the importances of the range of dominant frequencies and spectral analysis was mentioned.In another project held by Bonato et al.[17], their researchers used 6 groups of attachable sensors and successfully allocated five types of acceleration attributes as informative descriptors in PD tremor assessment, namely the range of amplitude of each channel, the root mean square value (RMS) of each frequency, two cross-correlation-based features, two frequency-based features and the signal entropy[18, 19]. In details, the cross-correlation-based features include the peak of the normalized cross-correlation function and the time lag of the peaks, while the frequency-based features consist

of the dominant frequency component and the corresponding ratio of the energy. However, according to Patel et al.[4], utilizing only three features (RMS, amplitudes, and signal entropy features) resulted in a bottom error rate (2.5%) in terms of tremor recognition and instead of decrease, adding more features slightly raised the error rate (2.7% for 4 features and 2.8% for 5 features).

In summary, tremors assessment plays an important role in PD diagnoses, but the composition of fluctuations are too complex to conduct rule-based methods. Instead, acceleration features have been studied in previous projects and proved of great value as tremor descriptors in the measurement.

2   Conventional Diagnose

Currently, the most common way for PD diagnoses is periodically collecting reports from patients and manually assessing the materials. Questionnaires from the patients play crucial roles in diagnoses and some successful scaling methods have been applied in PD progress tracking and pharmacodynamics proving. The most wildly used ones include the Hoen and Yahr[6], the Unique Parkinson's Disease Rating Scale (UPDRS)[7] and Northwestern University Disability Scale (NUDS)[5]. The primary principle of these methods is similar: collecting relative information, which is determined by professionals, from patients and scaling critical factors according to patients' description and professionals' observation.

As an instance, UPDRS and its revision, Movement Disorder Society-sponsored Unified Parkinson's Disease Rating Scale (MDS-UPDRS), are so far the most widely used assessment criteria in PD diagnoses[5]. In MDS-UPDRS, the marking table consists of four components: Part I spans "non-motor experiences of daily living", Part II concerns the "motor experiences of daily living", Part III and Part IV focus on the motor examination and some complications. For each factor, the severity is marked in 5 levels (0-4). Five types of tremor-relative factors mentioned in Part II and Part IV. In Part II, an overall scaling for all tremors is established while in Part III, factors of tremors are divided into four parts, namely posture tremors, kinetic tremors, rest tremors amplitudes and rest tremor constancy. All items in the table must be answered by patients and the overall marks are believed to roughly present the progress of his/her PD progress.

Certain advantages and advantages of these methods have been identified.

First, little special training is required and in these methods. These questionnaires are designed to transfer the interested points of professionals into straightforward questions in order to help patients describe quickly and clearly. Also, they are relatively economical for patients. Patients only need to answer the questions or fill the questionnaires, no extra equipment are demanded. However, these approaches are also costly from another perspective. Frequent visiting clinics and communication would be inconvenient for patients and time-consuming for medical professionals, which means less accessibility and higher expense on human resources. Moreover, responses from patients would suffer from subjectivities and information loss. Subjectivity and cognition diversities are unavoidable when patients try to measure and describe their symptoms. As a result, inconsistency would depress the reliability of human-based diagnoses. Meanwhile, due to the limitation of human memory, it would be difficult for patients to give a integral description of their symptoms. More or less, daily details could be ignored or forgotten without any conscious. Last but not least, most people could not take precise measurement of physical phenomena, such as frequency and amplitude of tremors, time and duration of action freezing.

Overall, conventional diagnoses of PD provides very detailed and scientific illustration about how to measure the PD progress, but they are human-resource expensive and lack of objectivity and precision.

3 Diagnose with Advanced Technologies

To overcome defects aroused by manual works, certain technologies been developed during last decades. An art-of-the-state is the combination of portable devices and machine learning technologies.

3.1 Automatic Monitoring

Recently, with the development of wearable devices, some experiments successfully achieve nonintrusive and longitudinal monitoring of PD symptoms in portable devices. In contrast with direct communication, these methods shows superior qualities in objectivity, accessibility and precision. Portable devices, especially wearable or attachable equipment, could keep monitoring for very long time[20] and provide statistic records with excellent precision.

In a project named SPARK[21], a multi-layer system combining smartphones and smartwatches were implemented to support remote monitoring of PD symptoms. The pilot work of SPARK suggested promising procedures in remote monitoring. A part of physical variables are collected by smartwatches and then periodically transported to the smartphone via wireless communication. Then the data were upload to the computing cloud, where the analysis algorithms are taken, and personalized interventions would be performed in smartphones in the end. Similar experiments are also carried out by Robert LeMoyne et al. [16]. Differently, an iPhone was mounted in the back of a glove and collects acceleration data by the embedded accelerometer sensor in this experiment. What is more, a research of Giuffrida et al. [10] complete tremor detection with a sensor-embedded ring. Nevertheless, although all 40 participants in their experiment agreed that the wearable devices (a ring and a wrist strap) were comfortable and nonintrusive, only 55% of them would wear them in public. It suggested that more natural monitors are expected. Herein, we introduce smartphones as a potential PD monitor.

In contrast with specially-made devices, smartphones seem like proper alternatives. By the end of 2015, the number of smartphones reached 1860 million[22], so in some ways, smartphones have become a sort of essential goods for the public. In comparison with special devices, smartphones arouse little interference on patients' routine. They are also more accessible and economical. In our pilot study[20], we investigated 17 types of common smartphones, priced from 110 (Blue Studio Energy) to 509 (Samsung Galaxy Note 4), and 5 data collection Android applications. It was proved that most smartphones are capable of carrying out whole-day-monitoring (except sleeping time) with only one charge. As for data collection platform, four of five applications could collect more than 15 raw features without extra equipment and all of five are free to use. Therefore, we believe that smartphones obtain potentials in long-term and continuous monitoring.

According to the former research, it has been ascertained that using acceleration sensors as tremor monitors is promising in PD diagnoses. Nevertheless, special-made equipment could be costly and the extra devices might cause discomforts and inconvenience for patients. And in our pilot work, smartphones were proved to be an ideal monitor without little extra expense or interference of patients' routines. Hence, smartphones seem like the ideal substitution of

scientific instruments. However, the sensors of smartphones can not be as accurate as scientific specialties, which results in low data quality. But data cleaning techniques could be set to improve the data quality.

3.2 Data Technology on Assessment

Machine learning techniques have become a hot point in recent years, they grant computer systems human-like learning ability when proper data are provided. More technically, machine learning techniques are a collection of algorithms designed for capturing information from data. According to specific targets, machine learning algorithms could be classified into three classes: classification algorithms, regression algorithms and clustering algorithms. The nature of our project is to reveal the relationship between tremor data and PD progress, which could be seen as a typical regression problem. However, in another way, it could also be viewd as a classification problem, i.e. distinguishing PD tremors of different progresses. In this area, some machine-learning-based systems have been proposed.

In the experiment taken by Patel et al.[4], an Support Vector Machine (SVM) with a third-order polynomial kernel was developed to recognizing PD tremors. With data collected in a full-controlled laboratory experiment, this system reached a 97% accuracy. Similarly, Song et al. [8] reported an experiment aiming at comparing neural networks and SVMs in PD tremor recognition and the result indicated that SVMs produced an better overall classification rate than Multiple Layer Perceptron and Radial Basis Function Network. Another experiment taken by David et al. [9] yet combined SVM and Artificial Neural Networks in a PD diagnose system and resulted in higher accuracy around 90%. The best results came from the experiment held by Daphne et al.[23], which applied a decision tree in PD motor symptom classification and resulted in an overall 99% accuracy.

Generally speaking, limited experiments could be found about applying machine learning algorithms in tremor analysis but several inspiring outcomes have been published. These facts suggested the potential of using machine learning techniques in PD tremor measurement. However, all these projects faces limitation in two aspects. First, all these experiments only applied supervised machine learning, which requires heavy labeling work. However, the labeled data in our project is very limited, (17 days). which is barely enough to drive supervised

models. Hence, semi-supervised models would be explored. Secondly, these studies mainly focuses on classification of PD patients and healthy subjects instead of predicting and scaling the PD progress. In technical words, most past experiments solved a classification or discriminative problem, not a regression problem. These two methods share a similar distinction that both of them are literally studying certain relationship between inputs and outputs during training. But the main difference here is that classification systems only produced closed outcomes, i.e. predictions must be seen at least once in system training, but a regression system is open and it should be able to produced unseen labels. From another point of view, it is known that classification algorithms usually provide discrete classes while regression algorithms could acquire continuous scores.

4   Summary

To sum up, many studies have been taken to explore the characteristics of Parkinson's Disease and tremors. Based on that, many traditional methods are proposed to manually evaluate the PD progress. Some advanced systems successfully introduced portable devices and machine learning techniques so as to resolve problems of manual ways. Although some of them acquired considerable accuracy indeed, they are not fully functional and due to the requirement of labels, the implementation could be expensive and time-consuming.

Chapter 3

System Design

In background studies, a number of issues and valuable experience have been revealed. In order to overcome these problems, we investigated multiple approaches to build up a workflow consisting of data preprocess, machine learning and evaluation. In this chapter, we list out candidate options we learned from literature and comment on their advantages and disadvantages in our project. This is to justify our strategy in workflow design and more detailed methods and functions will be given in Chapter 4.

1   Data Description

All raw data in this project was collected by smartphone accelerators in our pilot work. Three people (aged 58.5, 72.4 and 70.9) participated in our project and all of three were diagnosed with clinically idiopathic PD. Each participate received a smartphone with AWARE, an application collecting smartphone status. The data collection work lasted from 21/06/2015 to 29/9/2015 and finally we got data from two patients, Patient 02 and Patient 04. However, collected data indicated that Patient 02 rarely used the smartphone to send message and make calls. This fact implies that this patient hardly held his smartphone in hand, which means the accelerations would not present the limb tremor situation. So we choose the acceleration records of Patient 04 as our raw data. The sampling rate of smartphone sensors could be regulated but still varied in a wide range. According to the timestamps, the sampling rate in Patient 04's smartphone was around 80 Hz to 100 HZ, so is the gravity, while the sampling rate of linear acceleration, the acceleration without gravity, was much lower, about 50Hz. For higher integrity,

Table 3.1: Data Structure of Acceleration

Data Name	Type	Description
_id	INTEGER	primary key, auto incremented
timestamp	REAL	unixtime milliseconds since 1970
device_id	TEXT	AWARE device UUID
double_values_0	REAL	value of X axis
double_values_0	REAL	value of Y axis
double_values_0	REAL	value of Z axis

we used the acceleration data rectified by gravity, i.e. each acceleration record is minus by the gravity with the closest timestamp. In the end, we obtain 116908982 records containing three axises acceleration and timestamps. Besides acceleration and gravity, the screen status and communication history was also monitored and stored in an SQLite database. Table 3.1 to 3.3 illustrated the details of the all data structures.

As regards labels, an overall well-being scores of the same patients were estimated and recorded from 17/06/2015 to 17/07/2015. There is an overlapping between these scores and our data collection work (from 21/06/2015 to 17/07/2015). Therefore, we use the well-being scores to partially represent the severity of PD symptoms.

Table 3.2: Data Structure of Communication

Data Name	Type	Description
_id	INTEGER	primary key, auto incremented
timestamp	REAL	unixtime milliseconds since 1970
device_id	TEXT	AWARE device UUID
message_type	INTEGER	message type (1=received, 2=sent)
trace	TEXT	SHA-1 one-way source/target of the message

Table 3.3: Data Structure of Screen

Data Name	Type	Description
_id	INTEGER	primary key, auto incremented
timestamp	REAL	unixtime milliseconds since 1970
device_id	TEXT	AWARE device UUID
screen_status	INTEGER	screen status, one of the following: 0=off, 1=on, 2=locked, 3=unlocked

2 Data Preprocess

The aim of data preprocess is to convert continuous acceleration to segments where features are extracted and machine learning algorithms could be applied. Besides, three rule-based filters are involved to remove redundant data with the purpose of data reduction. The following sections explain insights of different filters and provide rationales for the feature extraction.

2.1 Rule-based Filter

In the data collection stage, 116908982 records were stored in an SQLite database, which could be too much to be analyzed. More importantly, a huge part of this data could be identified as useless in tremor analysis. Hence, certain filtering must be set to filter out redundant data. The purpose of rule-based filters are to find out valid segments of the original time series.

First of all, it is obvious that data without tremor information should be abandoned. In fact, smartphones literally collected fluctuations appearing on themselves but only a part of them passed from patients' bodies. In some occasions, smartphones are placed in pockets, on tables or somewhere else that not directly touches human bodies. However, previous experiments revealed that PD tremors are mainly recognized on fingers and toes[2, 3]. As a result, the collected data must not contain acceleration changes relevant to PD tremors. Therefore, the first condition of acceleration is that the smartphone must directly touch the body when the data was. In other words, we look for the periods when the smartphone was held in hand, i.g. when users open the screen or send messages or make calls . Herein, we introduce a status-based filter, which positions valid data according the screen status and the communication status.

For calling scenes, the communication records contained the time and duration of each called on this smartphone. So valid acceleration could be firstly identified via the timestamps and the corresponding communication status. As for texting scenes, communication data also records the time of each text, but different from calling, only those with message type SENT are counted because there is no guarantee that people could receive them immediately. As for screen status, as introduced before, a record would be generated when the smartphone screens were turned on, off, lock and unlocked. It should be mentioned that there would be overlaps between these screen-based filter and communication-based filter. This

is because the screen will also be turned on in the beginning of calling, but it would be turned off when users put it near the ears. In short, the screen check finds out most time the smartphone is being used for any purpose and the call and text check fills the screen-off part in calling time.

Apart from redundant data, mis-recording would also cause serious problems. In our early development, several unreasonable results have been found. For example, the filter functions returned three calling periods longer than 5 hours, which was much longer then the usual duration. By manual inspection, it is confirmed that the smartphone records two continued lock-on status, without any lock-off status in between. Specific reasons were unknown but we speculated that it could be caused by sudden power-off or some simply bug of AWARE. Moreover, some very short segments, less than 1 s, were also marked as valid by the status-based filter. Obviously, they seemed to be unusual according to people's habits. We decided to remove these results because they are suspicious but even if they were correct, they provide very few examples after feature extraction. To avoid the above mistakes, a time-based filter must be deployed to identify records with too long or too short time.

In short, rule-based filters use screen status and communication history to position time segments relevant to PD tremors. Sequentially, a time-based filter is used to eliminate abnormal segments. After this stage, the original time series will be split into separated segments. Each segment represents one using period, which could also viewed as one actual action of the user.

2.2   Data Splitting

The data splitting stage is applied so as to normalized the segments. After rule-based filtering, valid acceleration will be captured from continued time series and form up a number of segments. However, the length, i.e. the number of acceleration points, of these segments will be very different. Diversity of segment length would lead to biased information in machine learning procedures. Specifically, segments with longer duration, i.e. more acceleration points, should impact more on the training systems. For example, a segment covering 23 hours literally offer much more information than a 1 minute segment. Thus a normalization of segments is required to split segments into the same length. Consequentially, segments with longer length will be divided into more fragments and deliver higher influence in the flowing training. As for the proper size, Patel et al. highlighted

that they obtained the best accuracy by using a 3 s fragment in PD fluctuation analysis[4]. Similarly, in the experiment of Daphne et al. [23], 4 s splitting turned out to be the best normalization.

2.3 Feature Extraction

In literature reading, pervious

3 Supervised Machine Learning Algorithms

3.1 Logistic Regression

3.2 K-NN Model and K-d Tree

4 Semi-supervised Machine Learning

Semi-supervised machine learning is a sort of algorithms that capable of training models with partial labels. Traditional machine learning algorithms demand full-paired data as training materials. However, labels of data are expensive, as it always required massive manual works. On the contrary, unlabeled data could be collected via sensors and machines, so it is relatively easy to obtain. In our pilot study, the acceleration data within 84 days are collected by smartphones, but only 18 of them could be labeled from manual scores. By applying SSL models, the unlabeled data could probably support an more accurate system.

Currently, SSL algorithms have been widely used in various domains. The most often-used methods include self-training models, generative models, semi-supervised support vector machines (S3VMs), co-training models and graph-based models. Implementing all models would be of low efficiency, it is critical to look out for the most proper models for our system according to their strength and weakness.

4.1 Generative Models

As the oldest semi-supervised machine learning algorithm[24], the generative model is fairly intuitive and lots of advanced variants have been proposed. The whole algorithm is motivated by building a model to predict $p(x, y) = p(y)p(x|y)$,

where $p(x|y)$ presents the possibility of example x belonging to label groups y. In practice, we assume this distribution is composed of certain known components, such as Gaussian Mixture Models (GMM)[24], Mixture of Experts[25] and Naive Bayes[25]. With unlabeled data, we can approach these component models. In practice, this algorithm could be embedded into an expectation-maximization (EM) framework, in which the parameters of component models could be iteratively rectified by conducting the maximum likelihood estimation and finally result in an approximation of the ground truth. This method was applied in text classification in 2000 and proved of greater performance than traditional ways[26]. Advantages of generative models include: easy to implement with existing GMM functions and better performance with very few labels. Meanwhile, generative model obtains considerable flexibility and adaptability because affluent distribution models could be used as a component model. On the other hand, the usage of unlabeled data will lower its generalization ability when the assumed distribution is improper[27].

4.2 Self-training Models

In comparison with generative models, self-training models are more intuitive and similar to supervised learning models. In essential, self-training algorithms are an extension of traditional supervised learning methods. First, the training data are divided into two parts, labeled data and unlabeled data. Then certain supervised classifiers are picked up as cores of a self-training model. In each training iteration, the supervised machine learning algorithms will be trained with labeled data and prediction of unlabeled data will be given. Next, the most probable predictions of unlabeled data are accepted as labels and corresponding examples are added to labeled data set. When all unlabeled examples finally obtain the 'possible' label, the iteration will be stopped and the classification is accomplished. For better performance, there is some limitation for the supervised classifiers. First, it must provide the measurable descriptor (possibility) of predictions to conduct the label selection. Moreover, several supervised algorithms require parameter optimization, which would be computationally expensive if the iteration amount is huge, but without optimization, some supervised classifiers might work inefficiently and misclassification will occur frequently. As for merits, similar to generative models, self-training models benefit from the multiformity of model types. Besides, for applications which already have supervised learning versions,

the self-training model version could be built easily.

Many successful trials of self-training models could be found in previous experiments. For example, Riloff and Wiebe reported a highly precise bootstrapping process for sentence subjectivity classifying in 2003[28]. Later in 2005, C Rosenberg et al. proposed a semi-supervised object detection system based on their previous supervised systems[29]. An extended application from D McClosky et al. revealed the effects of applying bootstrapping systems as a parser-reranker[30].

4.3   S3VM Models

Semi-supervised Support Vector Machine, also known as Transductive Support Vector Machine Learning (TSVM), was firstly brought up by Joachims et al. in 1999[31]. As well as standard SVMs, S3VMs make an assumption that boundary of clusters is supposed to locate in sparse regions. Hence, SVMs attempt different boundaries and the one with maximum margin in the Reproducing Kernel Hilbert Space is picked as the predicted boundary. Comparing to traditional discriminant classifiers, SVMs consider both $p(y|x)$ and $p(x)$, i.e. the likelihood of instance x having label y and the likelihood of instance x in the model. According to Williams and Seeger, $p(x)$ and $p(y|x)$ must contain shared parameters in the learning models of semi-supervised learning[32]. Hence, SVMs are capable of avoiding traps caused by ignoring the connection of models and instances in comparison with traditional discriminative methods[33]. In S3VMs, predicted labels are estimated when the margin of a boundary is being determined. More specifically, S3VM trains the model with labeled data and give preliminary predictions for unlabeled data, then iteratively switch the labels of unlabeled data to obtain the maximum margin and update the model. As a result, S3VMs find the discriminative boundary that obtains the maximum margin from both labeled data and unlabeled data. However, the loss function of S3VM could be non-convex, resulting in local minimum point in training. This makes the model vulnerable in generation ability. To overcome that, several variants have been proposed to remit the impacts of non-convex loss function in model optimization procedures, such as Deterministic Annealing[34] and ConCave-Convex Procedure (CCCP)[35].

In practice, J Weston et al. described an experiment based on the S3VM framework proposed by Vapnik et al.[36] and the results showed obvious accuracy improvements in comparison to supervised SVM[37]. Experiments took by K

Bennett and A Demiriz[38] also supported the improvements of S3VM regarding generalization performance.

4.4   Graph-based Models

Graph-based models are posed to organize labeled and unlabeled data in a geometry structure, where the edges between nodes refer to the similarity of corresponding examples. The nature of classification graph-based models is label propagation from labeled data to the most specious unlabeled data in geometry views. In essential, graph-based models could be viewed as a compromise between constructing graphs closely reflecting the node labels, i.e. the cluster assumption, and keeping this graph geometric smooth, i.e. the manifold assumption.In general, there are two types of Graph-based semi-supervised classifiers. The most intuitive and natural methods explicitly propagate labels from labeled nodes to unlabeled nodes when certain conditions are satisfied. For example, Zhou X et al. presented an algorithm based on Gaussian Random Field and Harmonic Function[39]. Another method was reported by Zhu D et al. in 2003 which measures the consistency in local and global as the propagation cndition[40]. The implicit methods are implemented by applying regularization in classification to produce similar outcomes for geometrically close examples. The most common functions used in regularization include Mincut Regularizer[41], Tikhonov Regularizer[42] and Manifold Regularizer[43]. Apart from regularizer functions and loss functions, geometry structure is another critical component that significantly affects the model performance. In past applications, strong understanding of data structure was demanded in model construction. For instance, Balcan et al. introduced a video surveillance system using time series, colors and face features as edges, which is understood by researchers before the experiment. Another limitation of graph-based models, which is also shared in most geometry-based methods, is the computational problem. The computation of edges results in the limitation of nodes amount. According to W Liu, most graph-based models in semi-supervised learning have a time complexity of $O(n^3)$, much higher than that of S3VMs $(O(n^2))$[44]. So graph-based models would not be a wise choice for applications involving massive data.

4.5 Co-training Models

The co-training model was firstly proposed by Blum and Mitchell in 1998 and some boosted versions were proposed in last decades. Co-training models define two (or more in later versions) weak classifiers in the training stage. In standard co-training algorithm, it makes the assumption that i)the training data must obtain at least 2 sufficient and redundant views, i.e. feature sets capable of training a strong classifier, and ii) those views must be conditionally independent, i.e. features are irrelative to each other. Then two classifiers are set to be trained by labeled data with different views (feature sets). In each iteration, the most confident results will be added to the labeled data set and re-train both classifiers in next iteration. These procedures repeat until all data is labeled. Essentially, the whole model makes a compromise of 2 views during the training because each classifier is trained by some instances labeled by another in the previous iteration. Hence, the nature of co-training is mutual teaching between different classifiers. In fact, co-training could be viewed as a combination of self-training and multi-view learning. In other words, if the training views are completely overlapping (the same feature sets), the system will degenerate to a self-training model. The importance of disagreement in weak classifiers have been proved and described by Wang et al[45]. However, conditionally independent views would not always exist in practice and it is also difficult to determine the feature set. Herein, some boosted versions of co-training are introduced to relax this constraint. Zhou et al. described a tri-training algorithm using three weak classifiers in training stage[46], which uses bootstrap, instead of nature independence, to guarantee the disparities of classifiers and a naive voting process in labeled data extension. Li et al. improved the robust of tri-training by replacing the bootstrap algorithm with Random Forest implemented by bagging algorithm[47]. As a young style, advantages and disadvantages of co-training are not yet explicitly explored. But some experiments have been reported to support them. For example, Xu et al. applied co-training in protein subcellular localization and acquired 10% higher accuracy over the supervised system[48] and Li et al. also successfully implemented a semi-supervised document retrieval system based on co-training learning[49].

5   Receiver Operating Characteristic

6   Cross-Validation

7   Information Gain

Chapter 4

Experiment Implementation

1 Data Description

2 System Workflow

3 Data Preprocess

3.1 Data Clean

3.2 Rule Based Filter

3.3 Feature Extraction

Also, the window length of time fragments dramatically affected the accuracy in these experiments and 4s (Patel) and 3s (Daphne G. M. Zwartjes) time window were thought to be suitable for tremor analysis

4 Model Implementation

4.1 Self-training Models

4.2 Gaussian Mixture Models

5 System Evaluation

5.1 Model Evaluation

5.2 Feature Evaluation

Chapter 5

Experiment Results

1 Model Performance

2 Feature Assessments

Chapter 6

Conclusion

Chapter 7

Limitation and Future Work

Bibliography

[1] National Collaborating Centre for Chronic Conditions (Great Britain). Parkinson's disease: national clinical guideline for diagnosis and management in primary and secondary care. Royal College of Physicians, 2006.

[2] Charles Anthony Davie. A review of parkinson's disease. *British medical bulletin*, 86(1):109–127, 2008.

[3] Andrzej Izworski, Marcin Michałek, Ryszard Tadeusiewicz, MONIKA Rudzińska, Jarosław Bulka, and Ireneusz Wochlik. Acquisition and interpretation of upper limbs tremor signal in parkinsonian disease. In *Proceedings of the 4th WSEAS international conference on electronics, signal processing and control (ESPOCO 2005), Rio de Janeiro*, pages 25–27, 2005.

[4] Shyamal Patel, Konrad Lorincz, Richard Hughes, Nancy Huggins, John Growdon, David Standaert, Metin Akay, Jennifer Dy, Matt Welsh, and Paolo Bonato. Monitoring motor fluctuations in patients with parkinson's disease using wearable sensors. *Information Technology in Biomedicine, IEEE Transactions on*, 13(6):864–873, 2009.

[5] Christopher G Goetz, Barbara C Tilley, Stephanie R Shaftman, Glenn T Stebbins, Stanley Fahn, Pablo Martinez-Martin, Werner Poewe, Cristina Sampaio, Matthew B Stern, Richard Dodel, et al. Movement disorder society-sponsored revision of the unified parkinson's disease rating scale (mds-updrs): Scale presentation and clinimetric testing results. *Movement disorders*, 23(15):2129–2170, 2008.

[6] Margaret M Hoehn, Melvin D Yahr, et al. Parkinsonism: onset, progression, and mortality. *Neurology*, 50(2):318–318, 1998.

[7] Movement Disorder Society Task Force on Rating Scales for Parkinson's Disease et al. The unified parkinson's disease rating scale (updrs): status and

recommendations. *Movement disorders: official journal of the Movement Disorder Society*, 18(7):738, 2003.

[8] Song Pan, Serdar Iplikci, Kevin Warwick, and Tipu Z Aziz. Parkinson?s disease tremor classification–a comparison between support vector machines and neural networks. *Expert Systems with Applications*, 39(12):10764–10771, 2012.

[9] David Gil and Magnus Johnson. Diagnosing parkinson by using artificial neural networks and support vector machines. *Global Journal of Computer Science and Technology*, 9(4):63–71, 2009.

[10] Joseph P Giuffrida, David E Riley, Brian N Maddux, and Dustin A Heldman. Clinically deployable kinesia? technology for automated tremor assessment. *Movement Disorders*, 24(5):723–730, 2009.

[11] Suzanne Lesage and Alexis Brice. Parkinson's disease: from monogenic forms to genetic susceptibility factors. *Human molecular genetics*, 18(R1):R48–R59, 2009.

[12] Joseph Jankovic, Kenneth S Schwartz, and William Ondo. Re-emergent tremor of parkinson's disease. *Journal of Neurology, Neurosurgery & Psychiatry*, 67(5):646–650, 1999.

[13] Anthony E Lang and Andres M Lozano. Parkinson's disease. *New England Journal of Medicine*, 339(15):1044–1053, 1998.

[14] Kevin C Brennan, Eva C Jurewicz, Blair Ford, Seth L Pullman, and Elan D Louis. Is essential tremor predominantly a kinetic or a postural tremor? a clinical and electrophysiological study. *Movement disorders*, 17(2):313–316, 2002.

[15] William C Koller, Bridget Vetere-Overfield, and Ruth Barter. Tremors in early parkinson's disease. *Clinical neuropharmacology*, 12(4):293–297, 1989.

[16] Robert LeMoyne, Timothy Mastroianni, Michael Cozza, Cristian Coroian, and Warren Grundfest. Implementation of an iphone for characterizing parkinson's disease tremor through a wireless accelerometer application. In *Engineering in Medicine and Biology Society (EMBC), 2010 Annual International Conference of the IEEE*, pages 4954–4958. IEEE, 2010.

[17] Paolo Bonato. Advances in wearable technology and applications in physical medicine and rehabilitation. *Journal of NeuroEngineering and Rehabilitation*, 2(1):2, 2005.

[18] Paolo Bonato, Delsey M Sherrill, David G Standaert, Sara S Salles, and Metin Akay. Data mining techniques to detect motor fluctuations in parkinson's disease. In *Engineering in Medicine and Biology Society, 2004. IEMBS'04. 26th Annual International Conference of the IEEE*, volume 2, pages 4766–4769. IEEE, 2004.

[19] Rudy Moddemeijer. On estimation of entropy and mutual information of continuous distributions. *Signal processing*, 16(3):233–248, 1989.

[20] Julio Vega. Parkinson?s disease progression assessment using behavioural inferences and smartphones (pandas).

[21] Vinod Sharma, Kunal Mankodiya, Fernando De La Torre, Ada Zhang, Neal Ryan, Thanh GN Ton, Rajeev Gandhi, and Samay Jain. Spark: personalized parkinson disease interventions through synergy between a smartphone and a smartwatch. In *Design, User Experience, and Usability. User Experience Design for Everyday Life Applications and Services*, pages 103–114. Springer, 2014.

[22] eMarketer. Number of smartphone users worldwide from 2014 to 2019 (in millions), 2016.

[23] Daphne GM Zwartjes, Tjitske Heida, Jeroen PP van Vugt, Jan AG Geelen, and Peter H Veltink. Ambulatory monitoring of activities and motor symptoms in parkinson's disease. *IEEE transactions on biomedical engineering*, 57(11):2778–2786, 2010.

[24] Behzad M Shahshahani and David A Landgrebe. The effect of unlabeled samples in reducing the small sample size problem and mitigating the hughes phenomenon. *IEEE Transactions on Geoscience and remote sensing*, 32(5):1087–1095, 1994.

[25] David J Miller and Hasan S Uyar. A mixture of experts classifier with learning based on both labelled and unlabelled data. *Advances in neural information processing systems*, pages 571–577, 1997.

[26] Kamal Nigam, Andrew Kachites McCallum, Sebastian Thrun, and Tom Mitchell. Text classification from labeled and unlabeled documents using em. *Machine learning*, 39(2-3):103–134, 2000.

[27] Fabio Gagliardi Cozman, Ira Cohen, and M Cirelo. Unlabeled data can degrade classification performance of generative classifiers. In *FLAIRS Conference*, pages 327–331, 2002.

[28] Ellen Riloff and Janyce Wiebe. Learning extraction patterns for subjective expressions. In *Proceedings of the 2003 conference on Empirical methods in natural language processing*, pages 105–112. Association for Computational Linguistics, 2003.

[29] Chuck Rosenberg, Martial Hebert, and Henry Schneiderman. Semi-supervised self-training of object detection models. 2005.

[30] David McClosky, Eugene Charniak, and Mark Johnson. Effective self-training for parsing. In *Proceedings of the main conference on human language technology conference of the North American Chapter of the Association of Computational Linguistics*, pages 152–159. Association for Computational Linguistics, 2006.

[31] Thorsten Joachims. Transductive inference for text classification using support vector machines. In *ICML*, volume 99, pages 200–209, 1999.

[32] Shai Fine and Katya Scheinberg. Efficient svm training using low-rank kernel representations. *Journal of Machine Learning Research*, 2(Dec):243–264, 2001.

[33] Xiaojin Zhu. Semi-supervised learning literature survey. 2005.

[34] Vikas Sindhwani, S Sathiya Keerthi, and Olivier Chapelle. Deterministic annealing for semi-supervised kernel machines. In *Proceedings of the 23rd international conference on Machine learning*, pages 841–848. ACM, 2006.

[35] Ronan Collobert, Fabian Sinz, Jason Weston, and Léon Bottou. Large scale transductive svms. *Journal of Machine Learning Research*, 7(Aug):1687–1712, 2006.

[36] Olivier Chapelle, Bernhard Schölkopf, and Alexander Zien. Transductive inference and semi-supervised learning.

[37] Jason Weston, Ronan Collobert, Fabian Sinz, Léon Bottou, and Vladimir Vapnik. Inference with the universum. In *Proceedings of the 23rd international conference on Machine learning*, pages 1009–1016. ACM, 2006.

[38] Kristin Bennett, Ayhan Demiriz, et al. Semi-supervised support vector machines. *Advances in Neural Information processing systems*, pages 368–374, 1999.

[39] Xiaojin Zhu, Zoubin Ghahramani, John Lafferty, et al. Semi-supervised learning using gaussian fields and harmonic functions. In *ICML*, volume 3, pages 912–919, 2003.

[40] Dengyong Zhou, Olivier Bousquet, Thomas Navin Lal, Jason Weston, and Bernhard Schölkopf. Learning with local and global consistency. *Advances in neural information processing systems*, 16(16):321–328, 2004.

[41] Avrim Blum and Shuchi Chawla. Learning from labeled and unlabeled data using graph mincuts. 2001.

[42] Mikhail Belkin and Partha Niyogi. Semi-supervised learning on riemannian manifolds. *Machine learning*, 56(1-3):209–239, 2004.

[43] Mikhail Belkin, Partha Niyogi, and Vikas Sindhwani. Manifold regularization: A geometric framework for learning from labeled and unlabeled examples. *Journal of machine learning research*, 7(Nov):2399–2434, 2006.

[44] Wei Liu, Junfeng He, and Shih-Fu Chang. Large graph construction for scalable semi-supervised learning. In *Proceedings of the 27th international conference on machine learning (ICML-10)*, pages 679–686, 2010.

[45] Ani Nenkova, Sameer Maskey, and Yang Liu. Automatic summarization. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Tutorial Abstracts of ACL 2011*, page 3. Association for Computational Linguistics, 2011.

[46] Zhi-Hua Zhou and Ming Li. Tri-training: Exploiting unlabeled data using three classifiers. *IEEE Transactions on knowledge and Data Engineering*, 17(11):1529–1541, 2005.

[47] Ming Li and Zhi-Hua Zhou. Improve computer-aided diagnosis with machine learning techniques using undiagnosed samples. *IEEE Transactions on Systems, Man, and Cybernetics-Part A: Systems and Humans*, 37(6):1088–1098, 2007.

[48] Qian Xu, Derek Hao Hu, Hong Xue, Weichuan Yu, and Qiang Yang. Semi-supervised protein subcellular localization. *BMC bioinformatics*, 10(1):1, 2009.

[49] Ming Li, Hang Li, and Zhi-Hua Zhou. Semi-supervised document retrieval. *Information Processing & Management*, 45(3):341–355, 2009.