Thesis for the Degree of Doctor of Philosophy

Novel Semi-Markov Conditional Random Fields Model for Long-term Activity Recognition

La The Vinh

Department of Computer Engineering

Graduate School

Kyung Hee University

Seoul, Korea

August, 2012

Novel Semi-Markov Conditional Random Fields Model for Long-term Activity Recognition

La The Vinh

Department of Computer Engineering

Graduate School

Kyung Hee University

Seoul, Korea

August, 2012

Novel Semi-Markov Conditional Random Fields Model for Long-term Activity Recognition

by

La The Vinh

Advised by

Professor Sungyoung Lee

Submitted to the Department of Computer Engineering
and the Faculty of the Graduate School of
Kyung Hee University in partial fulfillment
of the requirements for the degree of
Doctor of Philosophy

Dissertation Committee:

Professor Young-Koo Lee, Ph.D. .....

Professor Tae-Seong Kim, Ph.D. .....

Professor Brian J. d'Auriol, Ph.D. .....

Professor Guan Dong Hai, Ph.D. .....

Professor Sungyoung Lee, Ph.D. .....

Novel Semi-Markov Conditional Random Fields Model for Long-term Activity Recognition

by

La The Vinh

Submitted to the Department of Computer Engineering
on June 6, 2012, in partial fulfillment of the
requirements for the degree of
Doctor of Philosophy

Abstract

Activity recognition is becoming an important research area, and finding its way to many application domains ranging from daily life services to industrial zones. The majority of research has focused on recognizing some particular simple activities like sitting down, standing up, bending, walking in a realtime or near realtime manner. In some applications, the recognition of those simple activities is not enough. For example, in lifestyle monitoring systems, it is more important and more interesting to detect longer activity routines, which may comprise of several simple activities, like working at an office, having lunch, having dinner, hygiene. Also in such a system, the realtime capability may not be a must-have feature because the sensory data can be processed once a day or even once a week. The focus of this dissertation is to develop a system for long-term activity data analysis.

In any activity recognition system, sensing hardware and inference engine are the two most important components. For sensing devices, accelerometers are preferred because they are low-cost, low-power consumption, and wearable. These characteristics enable long-term logging of daily living activity data. The inference engine can be divided into three main components: signal processing and feature extraction, feature selection, and classification. This work utilizes the several existing algorithms to extract features in the first component, and contributes two novel algorithms for selecting appropriate features and classifying sequential data in the last two steps.

In the area of feature selection research, there has been a great number of proposed algorithms belonging to three main approaches: filter, embedded, and wrapper. The first two approaches require a particular classifier to be repeatedly trained and evaluated. Therefore, they are suitable only for systems with short-duration input and short training time. Wrapper approach is classifier independent and has a low computation cost, which is a very important consideration for long-input systems. Hence, one of the focuses in this thesis is to develop a wrapper-based feature selection method. The proposed method utilizes the mutual information measurement and overcomes the well-known limitation of the existing work in this field (the relevancy-redundancy unbalance).

Regarding the classification step, various algorithms are available such as: k-nearest neighbors (kNN), artificial neural network (ANN), support vector machine (SVM), hidden markov model (HMM), conditional random fields (CRF). Among them only CRF is suitable for observable sequential data, which is actually the input of our long-term activity classification system. However, the existing implementations of CRF suffer from two limitations: the ability to model skip-transitions (also known as long-range relationship), and high training computation cost. Therefore, in this work, a novel semi-Markov conditional random fields model (semiCRF) is proposed to overcome the first limitation, and a smart-caching training algorithm is introduced to lessen the training cost.

The experimental results show that the proposed algorithms achieve better performance in comparison with the existing one regarding classification accuracy as well as execution time.

Thesis Supervisor: Sungyoung Lee
Title: Professor

Acknowledgments

I would have not been able to complete this dissertation without the enthusiastic assistance and cooperation from many people. First and foremost, I gratefully acknowledge my supervisor Professor Sungyoung Lee for his encouragement, guidance and support from my very first day in Korea. I have been strongly impressed by his keen sense of finding research areas, which are theoretically interesting and practically important. He has not only led me to the research area of wearable sensor based activity recognition but also offered me lots of insightful suggestions based on which I have developed and completed my dissertation.

I also owe my deepest gratitude to other Professors for their plentiful knowledge and advice. Professor Young-Koo Lee has given me a great sense of dealing with large-scale data throughout his data mining class. Professor Brian J.d'Aurial's advices on my presentation and visualization skill have been going with me not only in the past but also at the present and absolutely in my future research career. Professor Tae-Seong Kim has impressed me by his excellent lectures on pattern recognition area. His wisdom greatly contributed to consolidating and widening my knowledge on that field, which is also the very important background for my dissertation.

I am indebted to many of my colleagues to support me during my stay in Korea. Especially, Ngo Quoc Hung, Phan Tran Ho Truc, and Le Xuan Hung, who are my seniors, have made available their support in a number of ways. Without their suggestions, my life and my research would have been much harder.

I also would like to express my deeply thanks to the dissertation committee members whose helpful comments have helped me to improve and complete this dissertation.

Last but not least, I am grateful to my parents, my parents in law, my grandmother in law, and especially my wife for their boundless love. Needless to say that my little son has been always in my mind and has been my strongest motivation to complete this dissertation.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

At first, this chapter provides an overview of the human activity recognition research including the goal, and the potential applications of the research in this area. Then, different existing approaches are discussed to figure out the advantages as well as disadvantages of each particular method. Based on the discussion, motivations behind the approach proposed in this dissertation are clearly explained. The organization of this dissertation is presented at the end of this chapter.

1.1 Overview

The goal of activity recognition is to recognize the actions and goals of users from a series of sensory data collected by different kinds of sensors. Since the 1980s, this research area has gained the attention of the computer science community because of its potential in providing personalized support for many different applications and its applications in different fields of study such as medicine, human-computer interaction, or sociology.

Many different applications have been studied by researchers in activity recognition. Life style monitoring is a good application area for activity recognition systems [12], [97]. Because the modern life style tends to involve in more sedentary jobs, while there are growing evidences showing the relationship between common health problems such as diabetes, cardiovascular, os-

teoporosis and the level of physical activity [62]. Therefore, the activity monitoring system helps to assess and then alter the life style, this in turn could result in health benefits and reducing the health care costs, which are reportedly increasing in recent years [18], [60], [61].

Besides the life style monitoring applications, activity recognition has been considered to be a potential factor in improving convenience as well as productivity at work places; for example, in smart hospitals [27], [95], in aircraft maintenance [51], or in a workshop [56]. Also, such activity recognition systems can be used to predict abnormal behaviors such as falling down [68] for emergency response in health-care systems. Because of the large application potential, activity recognition has been gaining increasingly interest from research community [48] as well as enterprises [14], [67].

There have been different approaches to activity recognition. Those approaches can be categorized by two important criteria: **sensing technology** and **operation mode**. From a sensing point of view, the three most popular sensing mechanisms used for activity recognition systems are video cameras, object-attached sensors, and wearable sensors. Each sensing method has its own advantages and disadvantages as illustrated in Figure 1.1. The following paragraphs provide more detail about each approach.

Video-based activity recognition systems: These systems [13], [83], [112] utilize cameras for recording the user videos, then the video data is processed to recognize the actual activity. This approach is easy to be applied for the indoor environments, where the camera is installed at fixed positions. Therefore, it is really suitable for applications like surveillance, or in-home entertainment. In such systems, the video-based approach is a better choice than the others because the camera can provide more comprehensive information about the user activities. In addition, the use of video-based systems is more convenient than that of the wearable sensors based systems be-

cause it does not require the user to wear any device. However, camera-based systems are not able to detect on-the-way activities like driving, walking on the street, etc because of the difficulties in camera deployment and the complexity in dealing with changes in scene such as light condition, background picture. In addition, privacy concern can also be considered as another weak point the video-based activity recognition.

Object-attached sensor based activity recognition systems: In this approach, sensors are attached on daily living objects such as cups, chopsticks, teeth brushes. The sensors record whether the object is used by the user when performing some activity or not. Then the recognition system decides the activity based on the list of objects, to which the user interacted. This method has proven its potential especially for smart-home environment [72], [86], [100], [107]. Although this approach has the advantages of user-freedom and privacy, it also shares the same limitation with the video-based method, which is the inability to handle on-the-way activities. Furthermore, the systems using object-attached sensors also face with the difficulties of separating the monitored user from other people living in the same space.

Wearable sensor based activity recognition systems: In these activity recognition systems [8], [56], [81] the sensors are worn on the user's body during the normal daily life to continuously record the activity data. Because this approach requires users to wear sensing devices on their body, it is clearly not as comfortable as the others. However, the wearable sensor based approach is better than the others regarding its capability of recoding data for both indoor as well as outdoor environments during a long period of time. In addition, because the sensors are directly attached on the monitored user's body, the wearable sensor based systems are not affected by the interaction of multiple users. Among different kinds of wearable sensors, accelerometer is commonly used because of its low cost, low power consumption and its strong capability in recording both static and dynamic user state.

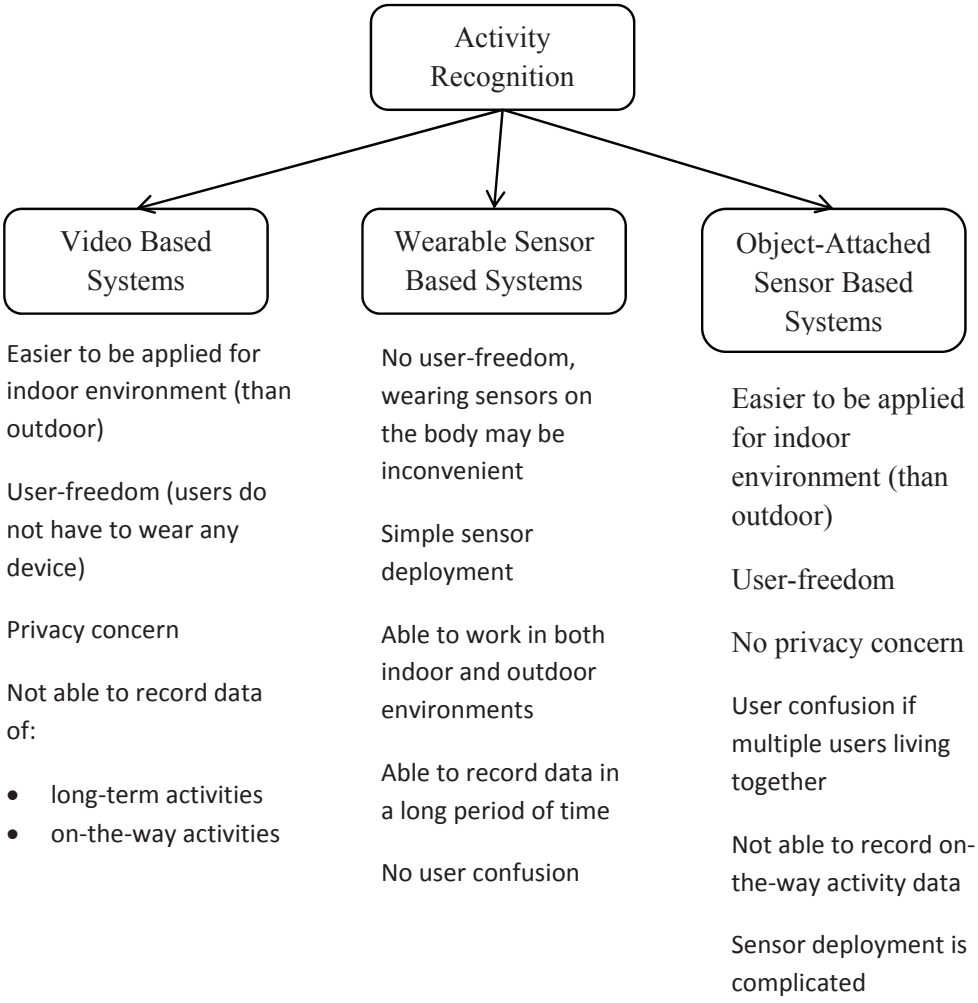


Figure 1.1: Different sensing mechanisms in existing activity recognition systems

From operation point of view, an activity recognition system can be real-time (online) or non real-time (offline) depending on how the input sequence is handled inside the system. Figure 1.2 summarizes the characteristics of those systems.

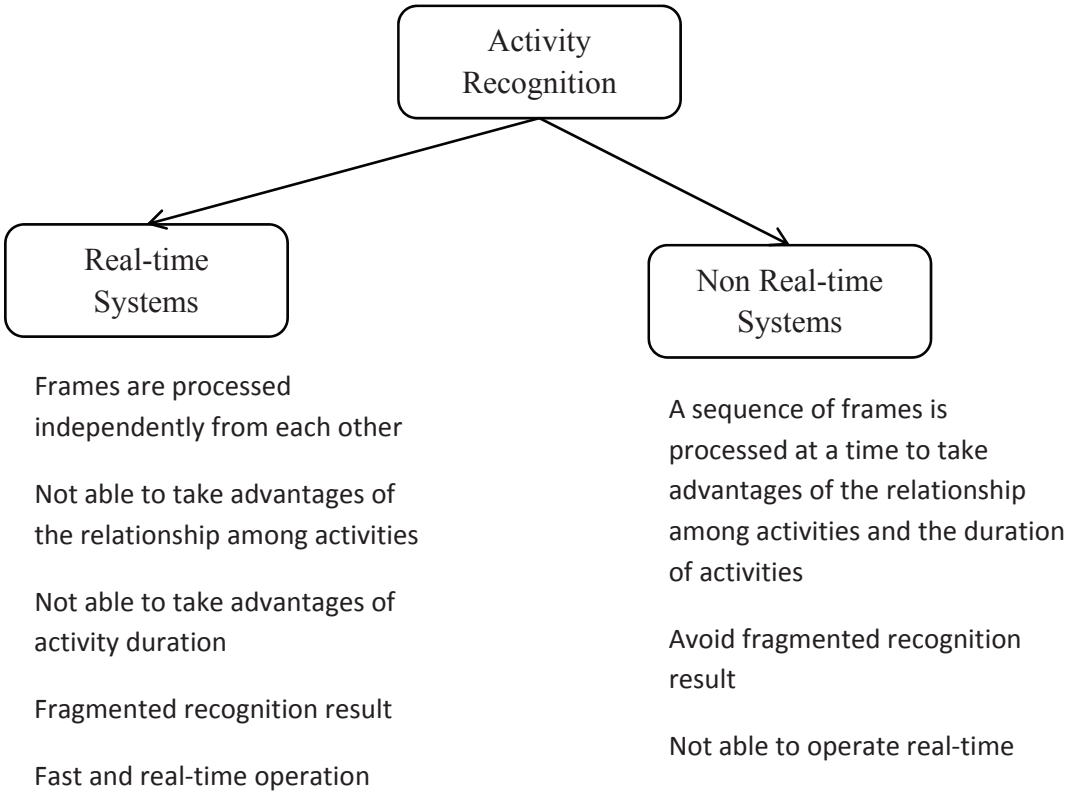


Figure 1.2: Different operation modes in existing activity recognition systems

Real-time activity recognition (online): In these systems [8], [11], [27], [68], [94], the long input signal is divided into short frames and then these frames are processed independently. It means that the classifier is activated to classify every frame, and those frames are recognized separately from each other. Because of that the real-time systems are not able to make use of the relationship among activities and the duration of activities to eliminate fragmented incorrect result.

Non real-time activity recognition (offline, long-term): In these systems, the long input signal is also divided into short frames. However, the whole sequence of frames is processed at a time. Therefore, instead of processing each frame independently, the classifier takes into account the characteristic of the whole sequence such as the transition probability from one activity to another, and the duration of activities to avoid fragmented incorrect results. Hence, the recognition accuracy and smoothness of the result can be improved [98], [107].

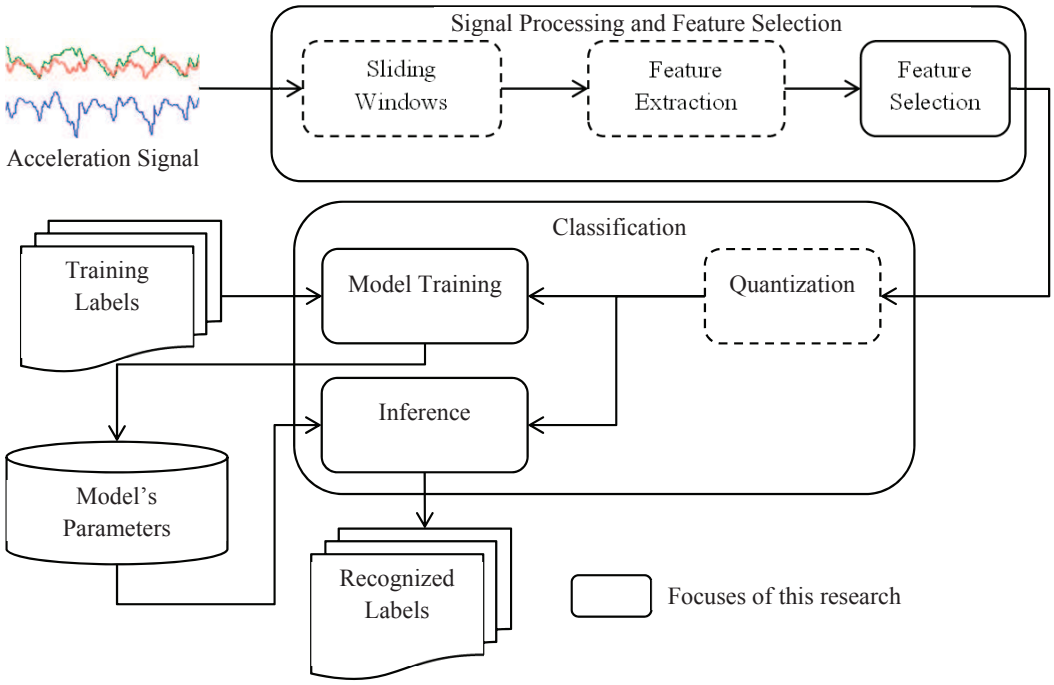


Figure 1.3: The proposed activity recognition system

1.2 Motivation

From the above analysis, it can be seen that long-term activity recognition plays an important role in the whole activity recognition area, especially for the applications which prefer high accuracy to fast response time such as daily living activity monitoring [3], or calorie consumption monitoring [101]. Although real-time approach can still be used in long-term activity recognition systems through the sliding windows mechanism, it is not able to take the advantages of the long frame sequences to eliminate fragmented incorrect results by using the transition and duration information. Therefore, our motivation of this work is to propose a method for long-term activity recognition that overcomes the limitations of the existing work in two important areas: feature selection, and classification. The proposed algorithms help to select good features from the input data, and take the advantages of long-term sequential data (the transitions among activities and the duration of activities) to improve the accuracy. Figure 1.3 illustrates the proposed system's block diagram.

1.3 Problem Statement

There has been a great number of proposed algorithms belonging to three main approaches: wrapper, embedded, and filter. The first two approaches are classifier-dependent and have high computational cost. Therefore, they are suitable only for systems with short-duration input and short training time. Filter approach is classifier-independent and has a low computational cost, which is a very important consideration for long-input systems. However, the existing filter based feature selection methods face with the problem of imbalance between the feature's classification power (relevancy) and feature's redundancy. Such an imbalance may cause the selection algorithm biased toward one particular kind of feature (the algorithm tends to favor strongly relevant or weakly redundant features). As a result, the accuracy may be affected negatively because of the inappropriate features. Therefore, a feature selection method that takes the advantages of the filter approach while overcomes the imbalance problem is one of the main focus of this dissertation.

The classification algorithm plays an indispensable role in the whole system. There are quite many algorithms for pattern recognition such as k-Nearest-Neighbor (kNN), artificial neural network (ANN), support vector machine (SVM), hidden Markov model (HMM), topic model (TM), etc. Those algorithms can be used to recognize activities from a long sequence of input data through the sliding window mechanism [8], [40], [81]. However, one obvious limitation is that, they process frames of data independently in both training and inferring phase. Because of that, they are not aware of some natural characteristics of activities, for example, a long activity can occupy several successive frames or the current activity may affect the appearance probability of the following activity. Considering those characteristics clearly helps eliminating the incorrect decisions, hence results in improving the recognition accuracy. Although there are existing works dealing with the problem of modeling sequential activity data [98], [107], they are still limited by the Markov assumption which prevents the classifier from modeling the long-range relationships and the duration of activities. That is the reason why it is a strong motivation of this dissertation to develop an algorithm for long-term activity recognition. The proposed algorithm is expected to improve the accuracy by taking into account the activity duration as well as the long-range transition from one activity to another.

1.4 Contributions

As mentioned above, this work focuses on proposing solutions to solve the two important issues: feature selection and classification algorithm for long-term activity recognition.

The feature selection method proposed in this work is derived from filter approach to take the advantages of the low calculation cost, and the classifier-independence; it also exploits a normalization mechanism to guarantee that the feature's classification power and the feature's redundancy are comparable to each other. Therefore, it is able to overcome the limitations of the existing work in the area of feature selection: classifier dependence, high computational cost, biased selection

because of the imbalance between the feature's classification power and the feature's redundancy. Hence, the recognition accuracy is improved, this improvement is supported by comprehensive experimental results.

For classification algorithm, the proposed method in this dissertation is based on the conditional random fields (CRF) model, a very well-known method for sequential data modeling. However, the existing CRF implementations are limited in modeling "skip-transition" and "duration", they also have a very high computational complexity preventing them from being utilized in large-scale applications. In this work, a novel implementation of the CRF model, called semi-Markov CRF, is proposed together with a novel fast computing method to solve both the above-mentioned problems of the existing work.

1.5 Thesis Organization

The dissertation is organized as follows. The chapter organization is also presented in Figure 1.4.

- **Chapter 1 Introduction**. In this chapter a brief introduction to the activity recognition research area is presented. The definition, importance, and existing approaches are clearly addressed. After that the dissertation focuses and contributions are also made clear.

- **Chapter 2 Related Work**. This chapter first shows the state of the art of the activity recognition research. Then a comprehensive survey of the existing work relating to the recognition process (including the three main algorithms feature extraction, feature selection and classification) is presented. The limitations of existing work in the feature selection and sequential data modeling areas are also clearly addressed, because these are the focuses of this dissertation.

- **Chapter 3 Signal Processing and Feature Selection**. All the details of the sliding windows, feature extraction and feature selection modules are described in this chapter. At the

end of this chapter, the proposed feature selection algorithm is validated individually using different datasets and classifiers.

- **Chapter 4 Classification**. This chapter first explains the detail of the classification module including the quantization and Semi-CRF algorithms for modeling sequential data. After that experiments are conducted to validate the proposed Semi-CRF model individually as well as together with the proposed feature selection in an integrated system. Finally, computational cost of the proposed Semi-CRF algorithm is analyzed to highlight the contribution of this work in reducing the complexity of the Semi-CRF.

- **Chapter 5 Conclusion and future work**. In this chapter a conclusion is given. Besides, some limitations of the work are also pointed out with potential solutions, which may need further research effort to be completed.

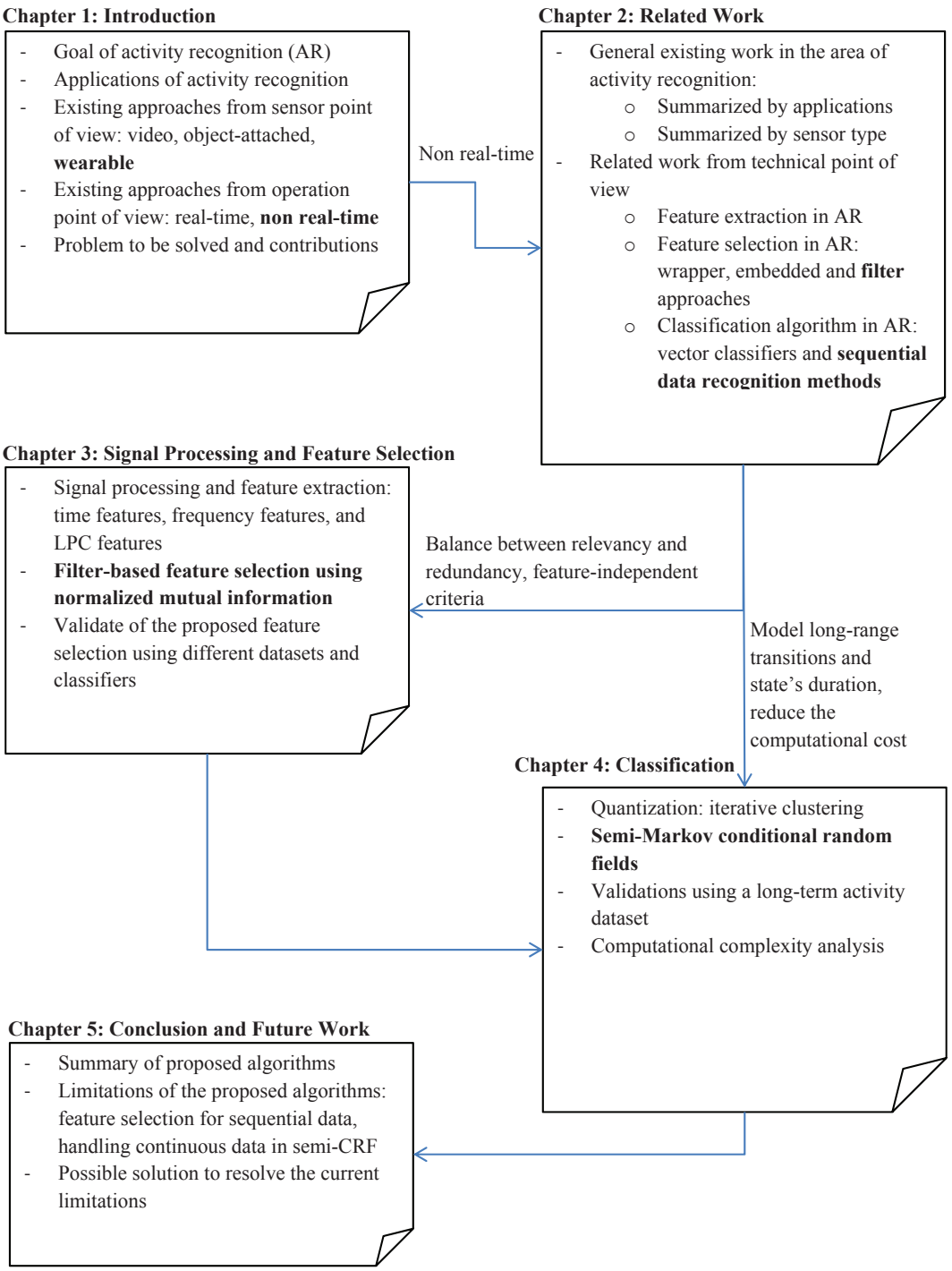


Figure 1.4: Chapter organization of the dissertation

Chapter 2

Related Works

This chapter first presents the state of the art of the general activity recognition research from the application and the sensor type points of view. After that the existing methods relating to the implementation of an activity recognition system are presented. Regarding the methodology, feature extraction, feature selection, and classification are among the most important components in any activity recognition system. Therefore, typical research works that relate to the development of each particular component are analyzed. However, stronger emphasis is given to the last two modules because they are the main focuses of this dissertation.

2.1 Applications of Activity Recognition

In the following paragraphs, existing activity recognition systems are summarized from the application point of view to highlight the typical application domains of the activity recognition including healthcare and assited living, industrial area, entertainment.

Healthcare and Assisted Living: The modern life style tends to involve in more sedentary jobs, while there are growing evidences showing the relationship between common health prob-

lems such as diabetes, cardiovascular, osteoporosis and the level of physical activity [62]. In addition, the increasing elderly population also poses challenges to the existing healthcare systems. Therefore, activity recognition is expected to address these challenges, for instance by automatically giving recommendations encouraging more active lifestyle, or helping elderly people to live more safely and independently.

Lifestyle monitoring and recommendation systems utilize the user activity information provided by the recognition engine to promote a more active and healthy lifestyle, or to actively support elderly people performing their daily activities. [3] uses ActiReg (PreMed AS, Oslo, Norway), a multi-sensor wearable device, to monitor the energy expenditure of children through the level of walking and running that they perform in their daily life. Based on the result of the energy consumption analysis, it is possible to recommend a suitable nutrition plan to avoid disease like obesity. [74] supports mentally disabled people in public transportation (tell the user where to get off and which bus to take) by utilizing user's location information. [2] encourages a more active lifestyle by combining the user activity recognized by a multi-sensor engine with the user location and the transportation information. The system can suggest the user to take a walk to the next bus stop instead of waiting at the current stop if it detects that the time is enough for the user to do that.

Another type of healthcare related system aims to detect potentially dangerous situations in a person's life to response urgently and automatically (for example calling to family members or doctors). [43] utilizes three-axis accelerometers to measure subject kinematics and detect the occurrence of falls. With a similar goal, [68] analyzes the characteristics of postural transition such as the time of sit-to-stand, the time of stand-to-sit transitions and their duration through the use of on-body gyroscope sensors to detect falling. [54] uses physiological sensors to detect arrhythmia, the system is connected with a call center and location service center to support the patient if his vital body signs indicate imminent health threats.

Industrial Applications: In the industrial area, activity recognition systems can support workers in performing their task efficiently. [56] uses a wearable device consisting of accelerometers and audio sensors to recognize the worker's activities in assembling tasks. The system is able to follow the progress thus can provide some necessary relevant information for the worker during his working section. The authors of [51] also investigate the use of on-body sensors to support workers in aircraft maintenance. In these scenarios, activity recognition systems are used to conveniently provide hands-free access to necessary information like manual, guideline or training documentations. Another typical work in this area published in [96], the system records activity data using wearable and environmental sensors for recognizing the worker activities in car manufactory. Based on the recognized activities, the system provides the workers upcoming assembly steps or warns the workers of the improper operations.

Entertainment and Games: [7] describes a wearable sensor based activity recognition system for performing interactive dancing. [116] uses portable, wireless motion-sensing clamp that can be attached to everyday objects to turn them into game controllers for playing video games. Recently, the application of activity recognition in entertainment and gaming is not only reported by research work but also widely available as commercial products such as Nintedo's Wii (Nintendo, 2006), Kinect (Microsoft, 2010).

The above paragraphs address the application of activity recognition systems in some typical domains. There are also many other potential application such as smart hospital [27], [95], advertising [84], military [66].

2.2 Sensors Used in Activity Recognition

There is a wide range of available sensors for activity recognition. As pointed out in section 1.1, those sensors can be categorized into video sensor, object-attached sensor, and wearable sensor.

Even in the category of wearable sensor based system, there exist different solutions utilizing a number of different sensor types.

There are a number of sensors providing environmental attributes such as temperature, humidity, audio level, etc.. Various activity recognition systems utilize audio, light, or humidity sensors to extract activity information such as [58], and [73]. However, it should be noted that those environmental sensors may not be comprehensive enough for activity recognition. Therefore, they are often compensated by other sensors such as accelerometer, gyroscope, etc..

Accelerometers are among the most widely used sensors for recognizing ambulatory activities (for instance walking, running, lying, etc.) because of the low cost, low power consumption, easy deployment. Using accelerometers, some papers report quite high recognition accuracies: 85% [8], 90% [37], [81]. There are also a number of works analyzing the effect of different accelerometer's configurations in the classification accuracy. For example, [58] studies the relationship between the recognition accuracy and the accelerometer's sampling rate. Interestingly, the work points out that increasing the sampling rate over 20 Hz does not give much significant increase in the accuracy. In addition, the acceleration amplitude of $\pm 2g$ is sufficient for ambulatory activity recognition. [38] is another work investigating in the effect of the accelerometer position on the accuracy. In that work, trousers pocket is pointed out to be one of the best positions.

GPS (Global Positioning System) is another commonly used sensor. Current mobile phones are equipped with GPS receiver making this sensor convenient for recognizing transportation activities. The GPS is perhaps not enough for recognizing some activities but it may provide helpful information to support the recognition process [82]. For example if the user location is a park, the possible activity can be walking but hardly can be teeth brushing. [2] combines GPS with other wearable sensors to recognize if the user is standing at a bus stop and then recommends the user

to take a walk if there is enough time for him to arrive at the next stop.

Physiological sensors which provide vital signs data such as heart rate, skin resistance, etc. have also been utilized in several works. Some typical works are [54] that uses physiological sensors to detect arrhythmia, and [87] showing that vital signs can be exploited to improve the recognition accuracy when it is combined with accelerometers.

2.3 Activity Recognition Methodology

In any activity recognition system, **feature extraction**, **feature selection**, and **classification algorithm** are the most important components. Therefore, in the following sections, a summary of the related work to each component is presented.

2.3.1 Feature Extraction

Over the past decades, there have been a great number of researchers investigating in the problem of feature extraction for activity recognition using wearable accelerometer and gyroscope [94]. A range of different approaches has been proposed to deriving some features from a frequency analysis [8], [73], [76], [78], and others on the time domain [1], [24], [25], [59]. In addition, wavelet-based methods have been also used to compute the so-called time-frequency features [88], [89], [111].

Regarding the frequency domain approach, in [8], one of the most-cited work in the area of wearable sennsor based activity recognition, the authors use fast Fourier transform (FFT) to compute the mean, energy, entropy and correlation of the acceleration. With that feature set, the system's accuracy is around 85% when classifying 10 activities performed by 20 subjects. Using exactly the same feature set, the author of [81] is able to classify 8 activities of 2 subjects with an accuracy higher than 90%. Other reports from [94] also indicate that extracting features on the

frequency domain is a good approach.

Arguing for time domain features, the authors of [59] are based on the low computational complexity and the fact that the frequency features may not well represent non-periodical signals. [94] shows that simple features extracted directly from time varying signal such as mean, standard deviation, cross-correlation, are strong enough to classify activities with a reasonable average accuracy (higher than 85%). Moreover, recently some more complicated features, for example autoregressive coefficients, are utilized and the authors report very promising results [38], [45].

Another approach is to utilize both time and frequency features by using wavelet transform methods. However, while the authors of [37] present very high accuracies (higher than 90% when classifying 3 activities: walking up-stair, wlaking down-stair, and normal walking on a flat plain), [94] points out that time and frequency features outperform wavelet features with the classification of 8 activities (walking, up-stair, down-stair, jogging, running, hopping with the left leg, hopping with the right leg, jumping).

It can be seen that there is actually not any single feature set for all the activity recognition systems. That is the reason why a robust feature selection method can be a good solution to select the best features from different categories.

2.3.2 Feature Selection

Feature selection is a technique for selecting a subset of relevant features, which contain information to help classifying one class from the others, from a large number of extracted features.

In pattern recognition [10], [102], the identification of the most discriminative features is an important step [19], since it is common to have a large number of features, including relevant as

well as irrelevant features, at the beginning of the pattern recognition process [28], [36]. Feeding a large set of features into a recognition model not only increases the computation burden but also causes the problem, commonly known as the curse of dimensionality [39]. Therefore, removing irrelevant features speeds up the learning process and alleviates the effect of *the curse of dimensionality*.

As pointed out in section 2.3.1, a range of feature extraction techniques are available in activity recognition, and which technique provides the best quality features is still a matter of controversy. Therefore, a mechanism for selecting good features from a combination set of features extracted by well-known techniques is a smart approach to take the advantages of different feature extraction methods. In [73], the authors selected features based on the visualization of the activity data. However, visually selection only works with features having high classification power. Nevertheless, there often some features, which have no classification power individually but are really helpful in a combination. The visualization method is not able to figure out this case, thus results in a bad selection.

So far, there is a great number of methods in the automatic feature selection research area. Those methods can be categorized into three main directions namely *wrapper*, *embedded* and *filter*. *Wrapper* [92], [113] approaches make use of the classification accuracy to evaluate the usefulness of features at each step. This approach can be used for real-time activity recognition systems dealing with short-length input frames. For example, the authors of [63], [71], [99], [117] already successfully applied the wrapper feature selection in a real-time accelerometer-based activity recognition system. However, in long-term activity recognition, the computational complexity is often directly proportional to the length of the data sequence; therefore repeatedly training/evaluating the system is impractical.

Embedded methods [16], [104], [115] also utilize a particular classifier to evaluate the strength of features. However, the selection process is *embedded* in the training phase of the classifier (often by using L1 regularizer); thus it overcomes the speed limitation of the *wrapper* methods. [17], [57], [104] are typical works applying the embedded feature selection method for activity recognition. Nevertheless, because the selection happens in the training phase, this approach can only be used if the feature vectors are directly input to the classifiers. Furthermore, embedding feature selection into the classifier training may increase the complicatedness of the classification algorithm.

Filter algorithms [9], [23], [75] utilize simple measurements such as correlation, mutual information to estimate the goodness of features. As a result, *filter* methods are classifier-independent and effective regarding computational cost. That is the reason why *filter* approach is often applied in many classification systems. In the area of activity recognition, several *filter* methods including RELIEF-F [46], CFS [36], Information Gain [79], MIFS [9], and mRMR [75], are evaluated in [5], [6], [15], [17], [47], [55], and [117]. From the conclusions of these work, mRMR is pointed out to be the most accurate feature selection method in these activity recognition systems. Since the filter approach has low computational cost and is classifier-independent, it is really suitable for the long-term activity recognition. Therefore, we would like to derive our feature selection method from this approach.

For the *filter* based methods, the two most popular criteria used to evaluate the goodness of features are correlation [36] and mutual information [75]. In [36], a typical work and one of the most cited work in correlation-based feature selection, a subset of features (S) is selected so that the below potential measurement is maximized

$$P_S = \frac{k\overline{r_{cf}}}{\sqrt{k + k(k-1)\overline{r_{ff}}}}, \tag{2.1}$$

where S is a subset of k features, $\overline{r_{cf}}$is the mean feature-class correlation ($f \in S$), and $\overline{r_{ff}}$ is the

average feature-feature inter-correlation. The correlation in equation (2.1) is computed by

$$\overline{r_{xy}} = \frac{E\left[(x-\mu_x)(y-\mu_y)\right]}{\sigma_x \sigma_y}, \tag{2.2}$$

where μ_x, μ_y, σ_x, and σ_y are the mean and standard deviation values of x and y, respectively. However, it is well known that the correlation is not able to describe non-linear relationships among variables as depicted in Figure 2.1. Furthermore, the computation of equation (2.2) requires that all the features must be numerical variables, it is another weakness of the correlation-based feature selection method.

The information-based method utilizes a simple measurement, hence it also has the advantage of low computation cost. In addition, the mutual information is capable of capturing the non-linear relationship (as illustrated in Figure 2.1), and is suitable for both numerical and categorical data. In the recent work [23], [75] mutual information criteria is preferred to the correlation one.

In mutual information based feature selection methods, mutual information is used to quantitatively analyze the relationship between any two features or between a feature and a class variable. The following definition of the mutual information has been used as the basis of recent existing work [9], [23], [75]

$$I(X;Y) = \int_{\Omega_Y} \int_{\Omega_X} p(x,y) log_2 \left(\frac{p(x,y)}{p(x)p(y)} \right) dxdy, \tag{2.3}$$

where Ω_X and Ω_Y are the sample spaces of X and Y, $p(x)$, $p(y)$, and $p(x,y)$ are the probability density functions of X, Y, and (X,Y), respectively. In the case of discrete variables, the integration notation is replaced by the summation notation as

$$I(X;Y) = \sum_{y \in \Omega_Y} \sum_{x \in \Omega_X} p(x,y) log_2 \left(\frac{p(x,y)}{p(x)p(y)} \right). \tag{2.4}$$

Equation (2.4) computes the mutual information based on probability distributions of discrete variables, hence we can apply that to both numerical as well as categorical data. An example of

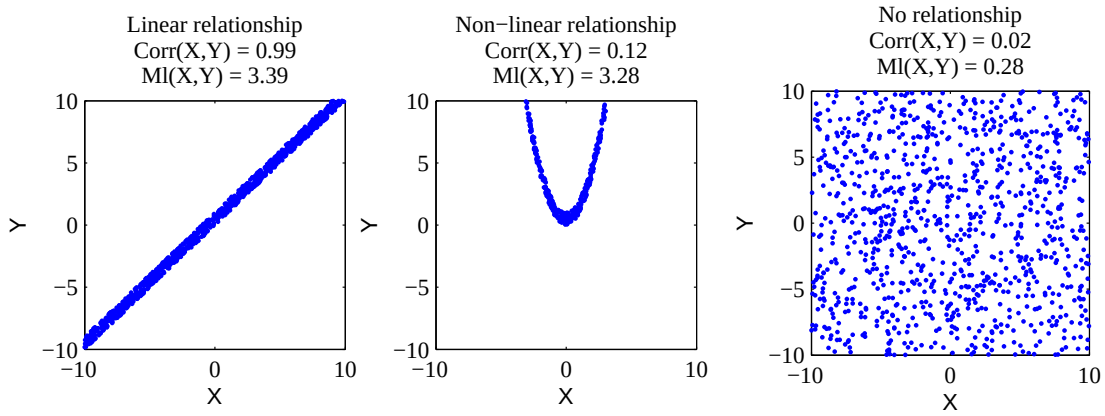


Figure 2.1: Different types of the relationship between two random variables (X and Y). The left figure shows a linear relationship captured by high values of both the correlation (Corr) and the mutual information (MI). The middle figure shows a non-linear relationship which is still well described by the high MI value, but Corr fails to reflect this relationship. The right figure shows two unrelated variables, hence both Corr and MI produce very low values

Object	A_1	A_2
1	M	A
2	M	B
3	F	B
4	F	A
5	M	C
6	F	C
7	M	C
8	F	C
9	F	A
10	M	B

	A	B	C	$P(A_1)$
M	0.1	0.2	0.2	0.5
F	0.2	0.1	0.2	0.5
$P(A_2)$	0.3	0.3	0.4	

$$I(A_1, A_2) = P(M,A)\log_2\left(\frac{P(M,A)}{P(M)P(A)}\right) + P(M,B)\log_2\left(\frac{P(M,B)}{P(M)P(B)}\right) +$$
$$P(M,C)\log_2\left(\frac{P(M,C)}{P(M)P(C)}\right) + P(F,A)\log_2\left(\frac{P(F,A)}{P(F)P(A)}\right) +$$
$$P(F,B)\log_2\left(\frac{P(F,B)}{P(F)P(B)}\right) + P(F,C)\log_2\left(\frac{P(F,C)}{P(F)P(C)}\right) = 0.05$$

Figure 2.2: Mutual information of categorical variables. The left table contains ten objects which have two categorical attributes A_1 and A_2. The right table shows the joint and marginal probabilities

computing the mutual information of categorical data is given in Figure 2.2. Additionally, Figure 2.1 demonstrates that non-linear relationships can be well described by the mutual information.

In [9], Battiti proposed to use bivariate mutual information functions including feature-feature mutual information $I(X_i; X_j)$ and class-feature mutual information $I(C; X_i)$ to estimate the feature's goodness. The selection criterion aimed at maximizing the class-feature mutual information (CFMI) and minimizing the feature-feature mutual information (FFMI). Since the CFMI represents the discrimination ability of a feature (relevance), while the FFMI contains information about the redundancy or the similarity among features, the Battiti's method serves as a starting point for the later max-relevance and min-redundancy approaches [75].

Battiti's feature selection algorithm (MIFS) selects a feature (X_i) at each step so that the following feature potential measurement is maximized

$$f(X_i) = I(C; X_i) - \beta \sum_{X_s \in S_{i-1}} I(X_s; X_i), \tag{2.5}$$

where function f measures the goodness of a feature, S_{i-1} is the set of selected features in the previous $i-1$ steps, X_i is any non-selected feature, and β is a manually tuned parameter used to make the left and the right terms in the subtraction comparable.

In [49], the author analyzed the disadvantages of Battiti's criterion and then proposed an improved one, the MIFS-U, represented by

$$f(X_i) = I(C; X_i) - \beta \sum_{X_s \in S_{i-1}} \frac{I(C; X_s)}{H(X_s)} I(X_s; X_i). \tag{2.6}$$

Despite the improvement made by the later work, both of the above methods require a parameter (β) to be estimated manually. If β is too large, the right term dominates, so both algorithms tend to select features based on minimum redundancy. In contrast, if β is too small, the algorithms favor maximum-relevance features. Unfortunately, there is no way to optimize the value of β.

The authors of [75] presented a parameter-free feature selection algorithm, called max-relevance and min-redundancy (mRMR), maximizing the below function

$$f(X_i) = I(C; X_i) - \frac{1}{|S_{i-1}|} \sum_{X_s \in S_{i-1}} I(X_s; X_i). \tag{2.7}$$

The basic idea of the above-mentioned work is similar to the one introduced by Battiti. However, Peng and his colleges provided a solid theoretical background of the method and eliminated the manually tuned parameter by averaging the feature-feature mutual information in the right term of the subtraction in (2.5). Although, mRMR does not always produce better results than do MIFS and MIFS-U [23], it eliminates the difficulty of parameter selection while producing results comparable to those of MIFS and MIFS-U.

Recently, the authors of [23] pointed out the drawback of mRMR, which was still the unbalance between the two terms of the subtraction. It is pointed out in [23] that

$$I(C;X_i) = H(C) - H(C|X_i) \leq H(C) = -\sum_{c\in\Omega_C} p(c)log_2(p(c)), \tag{2.8}$$

where Ω_C is the sample space of the class variable C. Based on Jensen's inequality, it is clear that

$$I(C;X_i) \leq log_2\left(\sum_{c\in\Omega_C} p(c)\frac{1}{p(c)}\right) = log_2(|\Omega_C|). \tag{2.9}$$

Therefore, in a two-class recognition problem ($|\Omega_C| = 2$), $I(C;X_i)$ is bounded in the range [0,1]. Similar proof leads to the following inequality

$$\frac{1}{|S_{i-1}|}\sum_{X_s\in S_{i-1}} I(X_s;X_i) \leq log_2(|\Omega_X|), \tag{2.10}$$

where Ω_X is the sample space of the features. Since $|\Omega_X|$ can have any arbitrary large value, the right term of the subtraction in (2.7) greatly varies and can dominate the left term (bounded in [0,1]). In such a case, the algorithm is biased toward the less redundant features.

Based on the above observation, Pablo *et al* introduced so-called normalized mutual information [23]. The authors showed that the mutual information between two random variables should be divided by the minimum value of the entropies in order to produce a normalized value in the range [0,1]. Then they presented a selection strategy (NMIFS) using the following feature quality estimation

$$f(X_i) = I(C;X_i) - \frac{1}{|S_{i-1}|}\sum_{X_s\in S_{i-1}} \frac{I(X_s;X_i)}{min(H(X_s),H(X_i))}. \tag{2.11}$$

It can be seen from (2.11) that NMIFS achieves a good balance between relevance and redundancy in two-class recognition systems, wherein both terms of the subtraction are within the range $[0,1]$. Problems may occur when the number of classes increases [108]. In that case, the left-side mutual information breaks the upper bound and may dominate the right term. Hence, NMIFS may

suffer from the same limitation as that in MIFS and MIFS-U when β is too small. Furthermore, because NMIFS assigns different normalizing weights to the features, it may select unexpected features. For example, if X_i and X_j are two features with the same relevance; X_i, however, is less random than X_j or $\frac{1}{H(X_i)} > \frac{1}{H(X_j)}$. In such a case, X_j may have a smaller weight in the right term of (2.11), making the overall potential $f(X_j)$ bigger than that of X_i; as a result, the method biases toward the noisier feature.

To summarize the common problem of the existing works [9], [23], [49], [75] in mutual information based feature selection, we reformulate the problem as the following: given a dataset with N features $X_1, X_2, ..., X_N$, and a set of $i-1$ selected indexes ($S_{i-1} = \{s_1, s_2, ..., s_{i-1}\}$), the next feature ($X_{s_i}$)is selected so that the redundancy $\left(RD(X_{s_i}) = \sum_{s \in S_{i-1}} I(X_s; X_{s_i}) \right)$ is minimized and the relevance $(RL(X_{s_i}) = I(C; X_{s_i}))$ is maximized. However, because the two problems may not have a common solution, we would like to find a scale factor (β) so that a feature X_{s_i} maximizing $RL(X_{s_i}) - \beta \times RD(X_{s_i})$ is a feasible solution for the minimization as well as the maximization. The existing solutions are summarized below

- MIFS and MIFS-U: β is manually selected by experiments,
- mRMR: $\beta = \frac{1}{|S_{i-1}|}$,
- NMIFS: $\beta(X_s; X_{s_i}) = \frac{1}{|S_{i-1}|} \times \frac{1}{min(H(X_s), H(X_{s_i}))}$.

Although a significant improvement has been made [23], there are still some limitations of the existing works pointed out in the above analysis. Hence, one of main focuses of this work is to propose a novel feature selection method to overcome those limitations.

2.3.3 Classification Method

So far, many algorithms have been proposed for activity recognition. K-Nearest neighbors (KNN), artificial neural network (ANN), decision tree (DT), support vector machine (SVM) and some

other kinds of classification methods were evaluated in [81]. These classification algorithms can deal with vector data only (each training instance must be represented in the form of a vector or a point in a multi-dimensional space). Therefore, to detect activities from a long stream of data, sliding windows are used. However, because each window is processed independently from the others, there may be some cases in which incorrect results appear between two correct results (for example, the result sequence can be: eating, running, eating). This fragmentation can be overcome if the relationship and duration of activities are taken into account.

To make use of the sequential structure of activities, Hidden Markov Model (HMM) was used in [98]. Recently, Conditional Random Fields model (CRF) was introduced as a much better approach compared to HMM in sequential data modeling [50], [90], [91], [110]. Thus, some researchers have successfully applied CRF to activity recognition [52], [104], [107]. The authors of [107] also pointed out that offline classification (the whole input sequence is processed at a time instead of processing each individual frames), produces higher accuracy than that of the online recognition.

However, a limitation of both the conventional HMM and the first-order CRF is the Markovian property, which assumes that the current state depends only on the previous state. Because of this assumption, the labels of two adjacent states must be supposed to occur successively in the observation sequence. Unfortunately, the presumption is not always satisfied in reality. For example, in the activity recognition problem, two expected activities (activities that we want to recognize) are often separated by irrelevant activities (activities that we do not intend to detect). This problem is called the long-range relationship modeling. Furthermore, constant self-transition probabilities cause the distribution of state's duration to be geometric [80] which is inappropriate to the real activity duration model. This is the so-called duration modeling problem

In [85], Sarawagi and Cohen have shown that Semi-Markov conditional random fields model (Semi-CRF) is capable using an explicit duration model. It is, however, still not able to solve the long-range relationship problem. In the following paragraphs, a brief introduction of the Semi-CRF is presented to make its limitations clear.

Since the conventional CRF is limited to the Markovian assumption that the label y_t at time t depends only on the previous label y_{t-1}, it is not able to capture the duration distribution as well as the interdependency of segments (a segment is a sequence of consecutive states with the same label) [80]. Therefore, Semi-Markov model is proposed to handle these issues. In [85], Sarawagi and Cohen describe a method for learning and inferring with Semi-CRF. The authors include in each state a label, a beginning time and an ending time. Thus, a new state is defined as

$$s_i = (y, b, e)\ i = 1, 2, ..., P, \tag{2.12}$$

where P is the length of the sequence $S = s_1...s_P$, which is constructed from input labels $Y = (y_1, y_2, ..., y_T)$. y, b, and e are label, beginning time, and ending time of the state s_i, respectively. For example, if we have a sequence of activities Y=(eating, eating, cleaning, cleaning, cleaning, unknown, sleeping, sleeping) then S=$\{$(eating,1,2), (cleaning,3,5), (unknown,6,6), (sleeping,7,8)$\}$. The beginning and ending time must satisfy the following constraints.

$$s_i.b \leq s_i.e\ i = 1, 2, ..., P, \tag{2.13}$$

$$s_i.e + 1 = s_{i+1}.b\ i = 1, 2, ..., P - 1, \tag{2.14}$$

$$s_1.b = 1, \tag{2.15}$$

$$s_P.e = T. \tag{2.16}$$

Now, instead of computing the likelihood of Y given X, the likelihood of S given X is estimated by

$$P(S|X) = \frac{\prod_{i=1}^{P} \Psi(s_{i-1}, s_i, X)}{Z_X}, \tag{2.17}$$

$$Z_X = \sum_{S'} \prod_{i=1}^{P'} \Psi(s'_{i-1}, s'_i, X), \tag{2.18}$$

where $\Psi(s_{i-1}, s_i, X)$ encodes the potential of the transition from s_{i-1} to s_i. In the following equations, $\Psi(s_{i-1}, s_i, X)$ can be rewritten in an alternative form $\Psi(s_{i-1}.y, s_i.y, X, s_i.b, s_i.e)$. For example, with the sequence S above, $\Psi(s_1, s_2, X)$ may also be written as $\Psi(1, 2, X, 3, 5)$. The Ψ function is defined in the below form

$$\Psi(s_{i-1}, s_i, X) = e^{W^T F(s_{i-1}, s_i, X)}, \tag{2.19}$$

where

$$W = [w_1, w_2, ..., w_N]^T, \tag{2.20}$$

is a column vector of model parameters,

$$F(s_{i-1}, s_i, X) = \begin{bmatrix} f^1(s_{i-1}, s_i, X) \\ f^2(s_{i-1}, s_i, X) \\ ... \\ f^N(s_{i-1}, s_i, X) \end{bmatrix}, \tag{2.21}$$

is a column vector of feature functions. In equations (2.17) and (2.18), the product of potential functions Ψ over all transitions of a sequence can be considered as the potential of the sequence. Thus, (2.17) is equal to

$$P(S|X) = \frac{Pol(S)}{\sum_{S'} Pol(S')}, \tag{2.22}$$

where

$$Pol(S) = \prod_{i=1}^{P} \Psi(s_{i-1}, s_i, X) \tag{2.23}$$

is the potential of the sequence $S = s_1, s_2, ..., s_P$. The forward algorithm and parameter estimation are implemented based on the following equations [85]

$$\alpha(t, y) = \sum_{d=1}^{D} \sum_{y'} \alpha(t - d, y') \Psi(y', y, X, t - d, t) \; t = 1, 2, ..., T, \tag{2.24}$$

$$Z_X = \sum_y \alpha(T, y), \tag{2.25}$$

$$\frac{dZ_X}{dw_k} = \sum_y \eta^k(T, y), \tag{2.26}$$

$$\eta^k(t, y) = \sum_{d=1}^{D} \sum_{y'} \left(\begin{pmatrix} \eta^k(t-d, y') + \\ \alpha(t-d, y') f^k(y', y, X, t-d, t) \end{pmatrix} \times \Psi(y', y, X, t-d, t) \right), \tag{2.27}$$

for $t = 1, 2, ..., T$ and $k = 1, 2, ..., N$. Where N is the number of model's parameters, D is the maximum duration of a label.

Based on equations from (2.17) to (2.27), the derivative of log likelihood of S given X is calculated as

$$\frac{d}{dw_k} log(P(S|X)) = \sum_{i=1}^{P} f^k(s_{i-1}, s_i, X) - \frac{\sum_y \eta^k(T, y)}{\sum_y \alpha(T, y)}. \tag{2.28}$$

From (2.24) and (2.27), it is clear that the Semi-CRF model increases the computation complexity of forward and backward algorithms by D times from $O(TM^2)$ to $O(TM^2D)$, where T, M, D are the length of the input sequence, the number of possible label values, and the maximum duration length, respectively. If we have N parameters to be optimized, the computation load is $O(NTM^2D)$. Truyen *et al.* [103] introduced a more complicated model, called Hierarchical Semi-Markov Conditional Random Fields (HSCRF) and demonstrated that HSCRF could be converted to Semi-CRF as a special case. Nevertheless, their conversion did not show any improvement in the complexity required for the optimization of the model's parameters. In [70], the authors proposed a method to decrease the computational cost of Semi-CRF. They, however, utilized a Bayes filter to eliminate some sequences from the computation. The approach, therefore, did not keep the originality of the problem. Furthermore, because of the constraints (2.13-2.16) the model proposed by Sarawagi and Cohen is still not able to model the long-range dependency.

Therefore, another focus of this work is to overcome the above limitations of the existing work by introducing our novel Semi-CRF to model both the duration and the long-range relationship of activities. Additionally, a fast training algorithm is developed making the proposed model suitable for large scale activity recognition applications.

Chapter 3

Signal Processing and Feature Selection

This chapter provides the step-by-step detail of the signal processing and feature selection modules in the proposed activity recognition system illustrated in Figure 3.1. In the first section, techniques for processing the input data and extracting features are presented. In the second section, the proposed feature selection method is described. The next section validates the proposed feature selection algorithm using different datasets and classifiers to prove that the proposed algorithm selects better feature set than do the existing one.

3.1 Signal Processing

The first block in the system is used for processing the input acceleration signals. Because the input signals are long sequences of temporal data, the system first divides these long sequences into shorter frames by using sliding windows as illustrated in Figure 3.2. Then from each frame data (one example is presented in Figure 3.2), the features are extracted.

In this work, some commonly used feature extraction techniques from the existing work are utilized including:

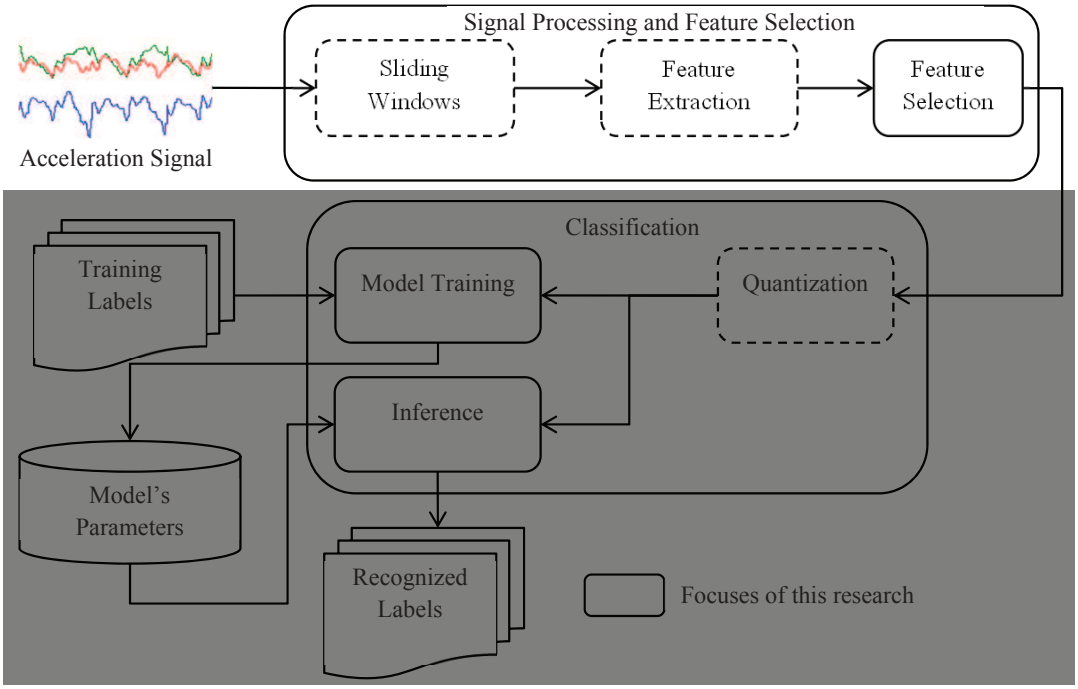


Figure 3.1: Block diagram of the signal processing and feature selection modules (not shaded area)

- Time domain features [11], [41], [59], [94].
- Frequency domain features [8], [42], [77], [81], [94].
- LPC (Linear Predictive Coding) features [38], [65].

Time domain features: assume that for an accelerometer sensor, a frame data consists of three signals corresponding to the three acceleration axes $X = \{x_1, x_2, ..., x_L\}$, $Y = \{y_1, y_2, ..., y_L\}$, and $Z = \{z_1, z_2, ..., z_L\}$, where L is the length of the window. The widely used time features are calculated as below.

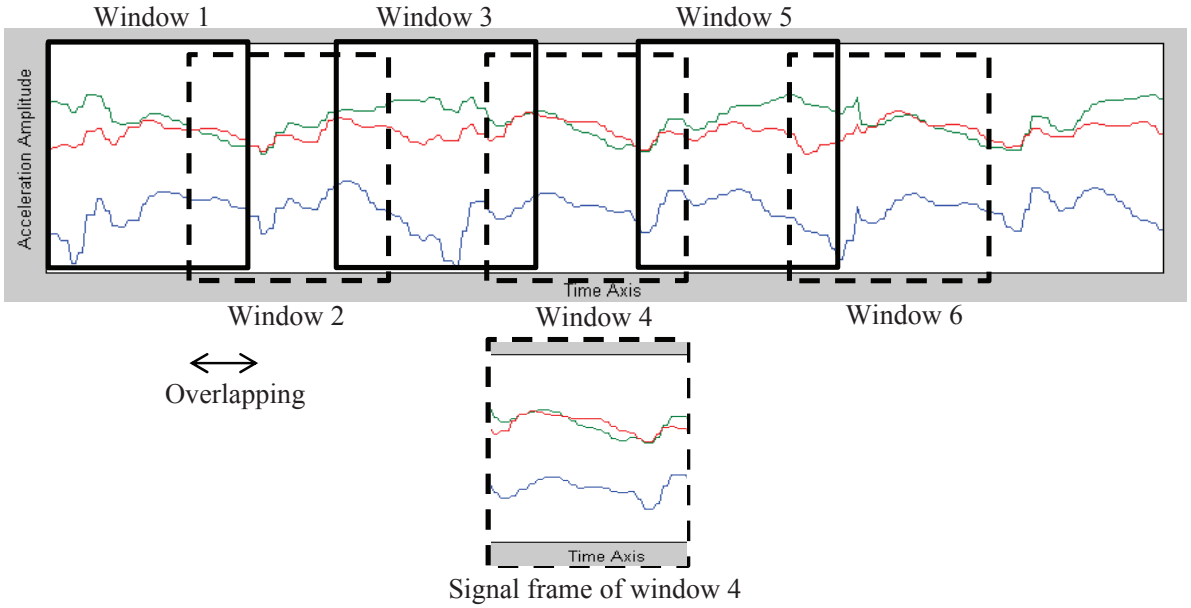


Figure 3.2: Using sliding windows to make data frames

Mean values:

$$\mu_X = \frac{1}{L}\sum_{i=1}^{L} x_i, \tag{3.1}$$

$$\mu_Y = \frac{1}{L}\sum_{i=1}^{L} y_i, \tag{3.2}$$

$$\mu_Z = \frac{1}{L}\sum_{i=1}^{L} z_i. \tag{3.3}$$

Standard deviation values:

$$\sigma_X = \sqrt{\frac{1}{L}\sum_{i=1}^{L}(x_i - \mu_x)^2}, \tag{3.4}$$

$$\sigma_Y = \sqrt{\frac{1}{L}\sum_{i=1}^{L}(y_i - \mu_y)^2}, \tag{3.5}$$

$$\sigma_Z = \sqrt{\frac{1}{L}\sum_{i=1}^{L}(z_i - \mu_z)^2}. \tag{3.6}$$

Correlation values:

$$\rho_{XY} = \frac{\frac{1}{L}\sum_{i=1}^{L}(x_i - \mu_X)(y_i - \mu_Y)}{\sigma_X \sigma_Y}, \tag{3.7}$$

$$\rho_{YZ} = \frac{\frac{1}{L}\sum_{i=1}^{L}(y_i - \mu_Y)(z_i - \mu_Z)}{\sigma_Y \sigma_Z}, \tag{3.8}$$

$$\rho_{ZX} = \frac{\frac{1}{L}\sum_{i=1}^{L}(z_i - \mu_Z)(x_i - \mu_X)}{\sigma_Z \sigma_X}. \tag{3.9}$$

The time domain feature set is constructed by combining the above values with the day time of the frame (dt); thus the final set is $\{\mu_X, \mu_Y, \mu_Z, \sigma_X, \sigma_Y, \sigma_Z, \rho_{XY}, \rho_{YZ}, \rho_{ZX}, dt\}$.

Frequency domain features: to compute the frequency domain features, a frame signal (for example $X = \{x_1, x_2, ..., x_L\}$) is transformed into the frequency domain by using the discrete Fourier transform (DFT) as below

$$X_k = \left|\sum_{n=1}^{L} x_n e^{-i2\pi k \frac{n}{L}}\right|, k = 1, 2, ..., L, \tag{3.10}$$

after that some commonly used features are calculated in the following equations:

Spectral energy:

$$E_X = \frac{1}{L}\sum_{k=1}^{L} X_k. \tag{3.11}$$

Spectral Coefficients:

$$C_k^X = \sum_{n=\frac{(k-1)L+1}{M}}^{\frac{kL}{M}} X_k, k = 1, 2, ..., M, \tag{3.12}$$

where $M = 6$ is the number of frequency bands [42].

Similarly, we can compute the frequency features for other axes (Y and Z). Finally the frequency domain feature set is $\{E_X, E_Y, E_Z, C_1^X ... C_M^X, C_1^Y ... C_M^Y, C_1^Z ... C_M^Z\}$.

LPC (Linear Predictive Coding) features: The LPC features of a given signal $X = \{x_1, x_2, ..., x_L\}$ is computed based on the autoregressive model (AR) of the signal as presented below

$$\tilde{x}_n = \sum_{p=1}^{P} a_p^X x_{n-p}, n = 1, 2, ..., L, \tag{3.13}$$

where $\tilde{x}_n$ is the linear prediction of sample value x_n based on the previous samples, P is the order of the model, and $a_p^X, p = 1, 2, ..., P$ are the prediction coefficients. The prediction error is calculated by

$$e_X = \sqrt{\frac{1}{L}\sum_{n=1}^{L}(x_n - \tilde{x}_n)^2}, \tag{3.14}$$

the prediction coefficients and error are calculated for all acceleration axes. Therefore, the final LPC features include $\{a_1^X, a_2^X, ..., a_P^X, e_X, a_1^Y, a_2^Y, ..., a_P^Y, e_Y, a_1^Z, a_2^Z, ..., a_P^Z, e_Z\}$.

In the training phase of the system, all the training sequences are divided into shorter frames by sliding windows, and then the above features are extracted. For one frame, those features are combined to form a vector denoted by $\{f_1, f_2, ..., f_N\}$, where N is the total number of features. After that a data matrix is constructed by adding the feature vectors, prefixed by the corresponding frame's activity label, row-by-row. Figure 3.3 illustrates an example of a data matrix with M frames and N features.

3.2 Feature Selection

As pointed out in the discussion in chapter 2, even though Pablo *et al* proposed NMIFS to overcome the limitations of the previous methods including MIFS, MIFS-U and mRMR, there are still

situations in which NMIFS may cause unexpected feature selections. Therefore, in this section, the focus is to resolve the limitations of NMIFS addressed in section 2.3.2.

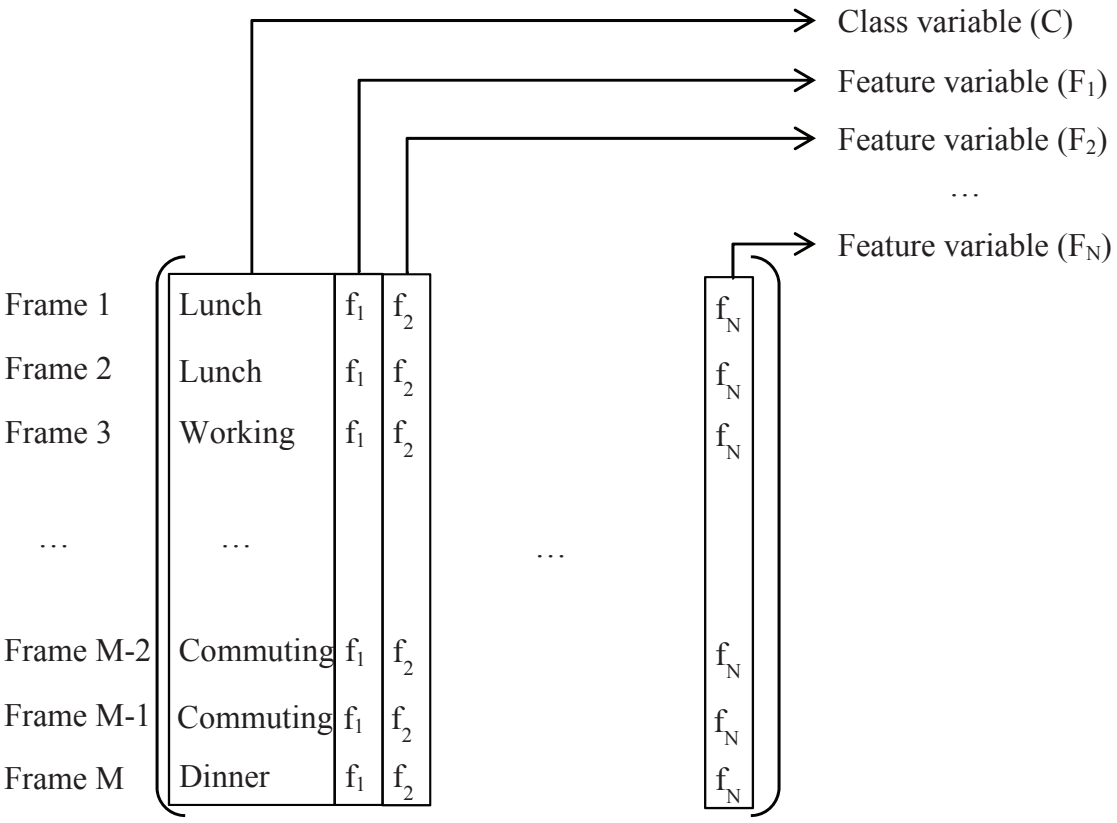


Figure 3.3: Example of a data matrix

To avoid the imbalance between the feature's relevancy and redundancy, it is necessary to normalize them to a same value range using their upper bounds. Therefore, we first consider the upper bound of the mutual information of any two variables in the following paragraphs.

Assume that a data matrix $F(M - by - N)$ is the output of the feature extraction phase as illustrated in Figure 3.3. Consider any two feature variables F_i and F_j, the joint mutual information

of F_i and F_j is computed using equation (2.4). It is already proven that [23]

$$I(F_i; F_j) \leq min\left(H(F_i), H(F_j)\right), \tag{3.15}$$

where $H(F_i)$, and $H(F_j)$ are the entropy of variables F_i and F_j, respectively; the entropy is computed by

$$H(F_i) = -\sum_{f \in \Omega_F} p(f) log_2(p(f)), \tag{3.16}$$

where Ω_F is the state space of feature variables.

We further develop the above upper bound by applying Jensen's inequality [44] to the definition of the entropy as below

$$H(F_i) \leq log_2\left(\sum_{f \in \Omega_F} p(f)\frac{1}{p(f)}\right), \tag{3.17}$$

$$H(F_i) \leq log_2\left(|\Omega_F|\right). \tag{3.18}$$

From (3.15) and (3.18), the proposed upper bound of the mutual information is

$$I(F_i; F_j) \leq log_2(|\Omega_F|). \tag{3.19}$$

In the proposed method, all the features are quantized using the same number of levels (Q), which is decided so that the expected quantization error is achieved. The quantization algorithm is depicted in Algorithm 1. As can be seen, the number of quantization levels is gradually increased until the quantization error is smaller than a predefined small constant ξ, the expected quantization error. In the experiments, $\xi = 0.01$ is selected because smaller values did not make any improvement regarding the accuracy but created extra computation burden. Obviously, $log_2(|\Omega_F| = Q)$ is an upper bound of the mutual information $I(F_i, F_j)$ and does not depend on F_i or F_j (hence, $log_2(|\Omega_F|)$ is a feature-independent upper bound).

Algorithm 1: Feature Quantization algorithm

Input : N - Total number of features

$F(M - by - N)$ - Data matrix

ξ - The quantization error

Output: Q - Number of quantization levels

$\overline{F}(M - by - N)$ - Quantized data

Quantization

$Q = 2$

$Done = False$

while $Done = False$ **do**

$MaxError = -INFINITE$

for $n = 1$ *to* N **do**

$Upper = max(F_n)$

$Lower = min(F_n)$

$Step = (Upper - Lower)/Q$

$Partition = [Lower : Step : Upper]$

$CodeBook = [Lower - Step, Lower : Step : Upper]$

$[\overline{F}_n, QError] = Quantiz(F_n, Partition, CodeBook)$

if $Qerror > MaxError$ **then**

$MaxError = QError$

if $MaxError < \xi$ **then**

$Done = True$

Break

$Q = Q + 1$

Algorithm 2: Mutual Information-based Feature Selection Using Greedy Forward Searching

Input : M - Total number of data samples

N - Total number of features

K - Number of features to be selected

$F(M - by - N)$ - Data matrix

$C(M - by - 1)$ - Class labels

Output: S_k - The selected feature index, where $k = 1, 2, ..., K$

Forward

$S = \varnothing$

//Normalize and quantize features

$\overline{F} = Normalize(F)$

$\overline{F} = Quantiz(\overline{F})$

//Start selecting features

for $k = 1$ *to* K **do**

$BestScore = -INFINTE$

for $i = 1$ *to* N *and* $i \notin S$ **do**

$g = 0$

$count = 0$

for $s \in S$ **do**

$count = count + 1$

$g = g + NI(\overline{F}_s; \overline{F}_i)$

$g = NI(C; \overline{F}_i) - g/count$

if $g > BestScore$ **then**

$BestScore = g$

$BestIndex = i$

$S = S \bigcup BestIndex$

To eliminate the problem of unequal normalizing weights, the proposed algorithm makes use of the feature-independent upper bound in (3.19) to normalize the mutual information instead of using (3.15) as in [23]. Therefore, the proposed normalized feature-feature mutual information is calculated by

$$NI(F_i; F_j) = \frac{I(F_i; F_j)}{log_2(|\Omega_F|)}. \tag{3.20}$$

Clearly, the proposed feature-feature mutual information is always within the range [0,1]. Therefore, to achieve a balance between the relevance and the redundancy, we propose to divide the class-feature mutual information by $log_2|\Omega_C|$ to make it also in the same value range. Hence, the proposed class-feature mutual information is now defined as

$$NI(C; F_i) = \frac{I(C; F_i)}{log_2\left(|\Omega_C|\right)}. \tag{3.21}$$

Using the normalized mutual information functions defined in (3.20) and (3.21), our feature selection method measures the goodness of a feature (F_i) at selection step j as

$$g(F_i) = NI(C; F_i) - \frac{1}{|S_{j-1}|} \sum_{s \in S_{j-1}} NI(F_s; F_i), \tag{3.22}$$

where S_{j-1} is the set of selected feature indices after step $j-1$. The feature searching process is done by using the greedy forward algorithm presented in Algorithm 2. Suppose that S_K is the final set of K selected features, the output of the feature selection is a new data matrix of the size $M - by - K$: $\overline{F} = F(1:M, S_K)$.

3.3 Validation of the Proposed Feature Selection Method

Because the proposed feature selection is dataset-independent and classifier-independent, it can be validated with different kind of datasets and classifiers. Therefore, in this section we borrow some commonly used classifiers: k-Nearest-Neighbors (kNN), Support Vector Machine (SVM), and Linear Discriminant Analysis (LDA) to conduct classification experiments with several datasets from UCI repository [4]. The experimental results show that the proposed selection algorithm is

able to overcome the limitations of the existing works addressed in Chapter 2; thus it produces higher accuracies.

Table 3.1 provides brief information about these datasets. To ensure objective and accurate comparison results and to avoid data-specific statements, datasets of different class number, sample number, and feature type are selected as depicted in Table 3.2.

Regarding classification methods, three common methods including k-Nearest-Neighbor (kNN, k=3), Support Vector Machine (SVM) and Linear Discriminant Analysis (LDA) are used; these methods are implemented in the MatlabArsenal toolbox [114] with WEKA [29], [35] integrated. The accuracy is measured using the ten-fold cross validation rule.

Tables 3.3-3.8 summarize the classification rates of the three classifiers with the 12 datasets. Each sub-table contains the number of features in the first column and the recognition accuracies in columns 2 to 5, which correspond to the four feature potential measurements: the proposed method (f^1), mRMR (f^2), NMIFS (f^3), and f^4 (NMIFS with normalized relevancy). Besides the average accuracy, the significance of the difference between the proposed method and the others is also measured by using paired t-tests [31]. Those t-values are put on the right side of each accuracy.

Although the tables are convenient for highlighting differences in detail, they are limited in representing the overall trend. Therefore, more general view of the results is displayed in Figures 3.6, 3.7, and 3.8. In the following paragraphs, the results are analyzed to show that the proposed method achieves better result compared to the others.

Table 3.1: Descriptions of the UCI datasets used in the feature selection experiments

	Dataset	Description
1	Arrhythmia [32]	This dataset contains ECG recordings of people with and without the presence of cardiac arrhythmia.
2	Hill Valley [4]	Each record represents 100 points on a two-dimensional graph. When plotted in order as the Y co-ordinate, the points create either a Hill or a Valley.
3	Image Segmentation [4]	The instances were drawn randomly from a database of 7 outdoor images. The images were manually segmented to create a classification for every pixel.
4	Ionosphere [93]	This dataset is used for the classification of radar returns from the ionosphere.
5	Isolet [26], [21], [22]	This dataset contains data of spoken words (name of letters).
6	Libras Movement [20]	This dataset contains 15 classes of 24 instances each. Each class references to a hand movement in Brazilian signal language.
7	Madelon [33], [34]	An artificial dataset, which was part of the NIPS 2003 feature selection challenge.
8	Multiple Features [106], [105]	This dataset consists of features of handwritten numerals (0-9).
9	Landsat Satellite [4]	Multi-spectral values of pixels in 3x3 neighbourhoods in a satellite image, and the classification associated with the central pixel in each neighbourhood.
10	(Connectionist Bench) Sonar [30]	This targets at discriminating between sonar signals bounced off a metal cylinder and those bounced off a roughly cylindrical rock.
11	Spambase [4]	This dataset contains the feature vectors of spam and non-spam emails.
12	Breast Cancer (Diagnostic) [64]	This dataset stores features computed from digitized images and is used for breast cancer diagnosis.

Table 3.2: Properties of the UCI datasets used in the feature selection experiments

	Dataset	# Class	# Samples	# Features	Type of features
1	Arrhythmia [32]	16	452	279	Continuous, Discrete
2	Hill Valley [4]	2	1212	100	Continuous
3	Image Segmentation [4]	7	2310	18	Continuous
4	Ionosphere [93]	2	351	33	Continuous, Discrete
5	Isolet [26], [21], [22]	26	7797	617	Continuous
6	Libras Movement [20]	15	360	90	Continuous
7	Madelon [33], [34]	2	2600	500	Continuous
8	Multiple Features [106], [105]	10	2000	649	Continuous, Discrete
9	Landsat Satellite [4]	6	6435	36	Discrete
10	(Connectionist Bench) Sonar [30]	2	208	60	Continuous
11	Spambase [4]	2	4601	57	Continuous, Discrete
12	Breast Cancer (Diagnostic) [64]	2	569	31	Continuous

Arrhythmia dataset: It can be seen that f^1 produces the highest accuracy, which is averagely about 26% higher than that of f^2, and the difference increases as the number of features increases. f^4 shows a slightly better result than f^3 (about $3-5\%$ higher, especially when combined with an LDA classifier); it, however, is still worse than f^1, which has the highest results in 15 out of 18 tests with the Arrhythmia dataset.

Hill Valley dataset: This dataset sees almost the same accuracies in all four selection methods. With kNN and LDA classifiers, f^2 and f^1, respectively, produce higher results than do the other methods although the disparity is often not greater than 2%. As can be seen, t-values are rarely higher than 2.26 (or $p-value < 0.05$). It means that the accuracy differences are not statistically significant.

Image Segmentation dataset: Although the four results approach to each other as the feature number goes up, f^2 and f^3 are often the lowest accurate methods with high t-values (high significant differences). On average, f^4 is slightly better than f^1 (about 1.3% higher accuracies).

Ionosphere dataset: While the four feature selection methods do not create any significant differences when combined with kNN and SVM classifiers (almost all t-values are much smaller than 2.26). f^3 and f^4 have about 3% lower average accuracies than those yielded by f^1 and f^2 in case of using LDA recognition model.

Isolet dataset: With this dataset, f^3 often produces the worst results, this significant weakness is also supported by the very high t-values. f^1 is a little better than f^4 when the number of features is greater than 5. f^1 and f^2 are almost similar with only about 0.6% average distance in the accuracy

Libras Movement dataset:It is obvious that f^3's accuracies are often significantly lower than those of the others (lower accuracies, high t-values). The differences among f^1, f^2, and f^4 are insignificant since almost all the t-values are much smaller than 2.26.

Madelon dataset: No significant disparity is presented with LDA recognition model; however, when combined with kNN and SVM, f^1 proves to be the superior measurement, with about 5% higher accuracies than those of the others.

Multiple features dataset: f^1 and f^2 are a slightly better than the other two methods if the number of features is less than 10. However, with 10 to 15 features, f^3 and f^4 are better than f^1 and f^2. The four methods approach to similar results when the number of features keeps increasing.

Landsat Satellite and Breast Cancer datasets: Similar results are observed in these datasets regardless of the classifier or the feature selection method. There is no dominant measurement among the four, and the difference of classification rates (between any two selection criterions) is approximately $1 - 2\%$

Sonar dataset: There is no superior among the four methods when the LDA classifier is used. However, while f^1, f^3, and f^4 maintain similar accuracies with kNN and SVM, f^2 loses its competitiveness and obviously becomes the weakest method (about 10% lower recognition rates in almost all the kNN tests).

Spambase dataset: when kNN and LDA classifiers are used, the average accuracies are similar; however, f^3 and f^4 are significantly better than f^1 because they have small standard deviations leading to high t-values as can be seen in Table 3.3, 3.4, 3.7 and 3.8. Although, f^2's accuracies are clearly lower than those of f^1 when they are used with SVM, the differences are not statistically significant because of the low t-values.

Overall, it can bee see that f^2 is often the worst criterion; f^1, in contrast, is often one of the two best measurements. Even if it does not have the highest result (for example with the Sonar dataset), the difference between f^1 and the best method is not significant. It is also worth noting that f^4 often produces better results than does f^3. Furthermore, Tables 3.3-3.8 summarize the number of times that each feature selection method produces the highest results and show the statistics in Figure 3.5. It is obvious that f^1 proves to be the most outstanding method. Among the other three selection criterions, f^4, in general, is a little better than f^2 and f^3. Hence, the statistics provide another reason to conclude that f^1 is the most superior method with f^4 occupying the second position, f^2 and f^3 competing for the lowest rank. Since f^4 differs from f^3 only in the class-feature normalization, it is clear that the normalization of class-feature mutual information

has a positive effect on the quality of the selected feature set. The superiority of f^1 illustrates the efficiency of the constant normalizing weights in our methods because f^1 and f^4 are different only in these weights. In addition, Figure 3.4 shows an analysis of the redundancy (RD) and relevancy

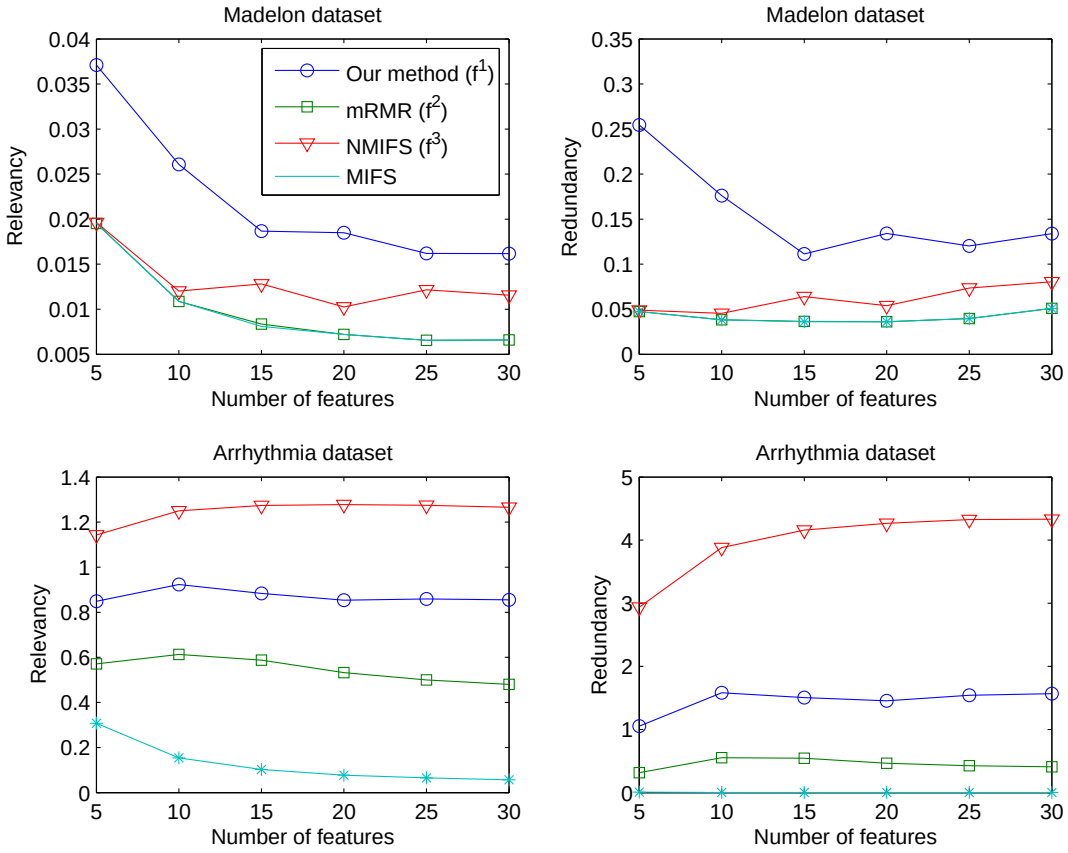


Figure 3.4: Redundancy and relevancy of the selected features

(RL) of the selected features. Those two quantities are computed as below (derived from the method in [118]).

$$RD(X_1, X_2, ..., X_N) = \frac{1}{N(N-1)} \sum_{i \neq j} I(X_i; X_j), \tag{3.23}$$

$$RL(X_1, X_2, ..., X_N) = \frac{1}{N} \sum_{i} I(C; X_i). \tag{3.24}$$

As can be seen, the proposed method gives higher priority to selecting the relevant features in case of low-redundancy dataset (Madelon). Whereas, it pays more attention to selecting the less redundant features if the dataset has high redundancy (Arrhythmia). In other words, the proposed method is less prone to any specific kind of features than the others.

So far, it has been proven that the proposed method not only inherits the advantages of the parameter-free methods like mRMR and NMIFS but also overcomes their limitations. By using the normalized class-feature mutual information, the imbalance between the relevance and the redundancy, which can be seen in mRMR and NMIFS, is lessen. To resolve the problem of unequal normalizing weights, a feature-independent upper bound of the mutual information is proposed, which then acts as the normalizing factor.

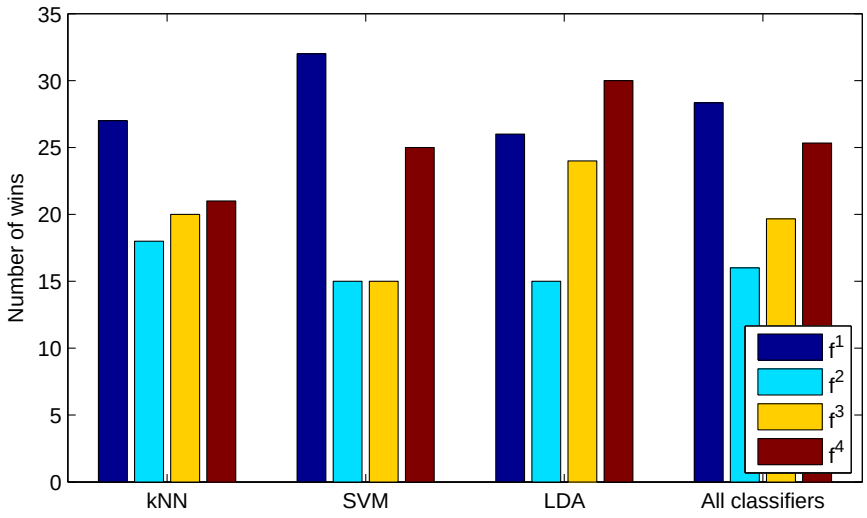


Figure 3.5: Number of times each method achieves the highest accuracy. The rightmost group shows the average number of all the three classifiers. There are $12 \times 6 = 72$ tests for each classifier and a total of $72 \times 3 = 216$ tests for all three classifiers. The results of those tests are presented in Tables 3.3-3.8

Table 3.3: kNN classification accuracies with different feature selection methods. Bold items highlight significant differences in comparison with f^1 (t-value > 2.26 or p-value < 0.05)

k-Nearest-Neighbor (kNN, k=3)									
Arrhythmia					Hill Valley				
# Fea	f^1	f^2/t	f^3/t	f^4/t	# Fea	f^1	f^2/t	f^3/t	f^4/t
5	51.53	47.99/0.57	55.17/1.66	54.22/1.24	5	53.47	53.40/0.04	52.31/0.54	52.31/0.54
10	**60.00**	58.21/0.85	**55.52/2.35**	57.98/1.41	10	**55.20**	**52.48/2.69**	54.05/0.62	54.05/0.62
15	**60.63**	51.53/1.49	**54.60/2.91**	60.60/0.02	15	53.63	54.29/0.62	54.13/0.33	54.13/0.33
20	**63.99**	**50.29/3.60**	**56.17/3.90**	60.38/2.24	20	53.72	51.82/1.58	54.46/0.39	54.46/0.39
25	**64.19**	**54.24/5.30**	**56.84/4.15**	63.30/0.52	25	53.88	53.39/0.54	53.96/0.05	53.96/0.05
30	**65.24**	**30.58/4.79**	**57.06/4.67**	61.11/1.74	30	53.14	53.80/1.06	53.39/0.16	53.39/0.16
Image Segmentation					Ionosphere				
# Fea	f^1	f^2/t	f^3/t	f^4/t	# Fea	f^1	f^2/t	f^3/t	f^4/t
3	**92.81**	**83.77/5.38**	**88.35/2.70**	**96.71/2.97**	5	89.18	90.30/0.93	88.32/0.99	88.32/0.99
6	**96.19**	**95.11/2.26**	**93.68/2.41**	96.02/0.58	10	87.13	88.27/1.06	87.18/0.03	87.18/0.03
9	**95.84**	95.41/0.87	95.89/0.40	95.76/1.50	15	87.98	88.03/0.04	85.69/1.81	85.69/1.81
12	**95.84**	**94.55/3.30**	95.58/1.20	95.58/1.20	20	85.99	85.47/0.39	84.84/1.32	84.84/1.32
15	**94.98**	**94.33/2.57**	**96.10/2.65**	**96.10/2.65**	25	83.40	85.16/1.17	84.27/1.17	84.27/1.17
18	95.50	95.50/0.00	95.50/0.00	95.50/0.00	30	84.32	84.89/0.69	84.32/0.00	84.32/0.00
Isolet					Libras Movement				
# Fea	f^1	f^2/t	f^3/t	f^4/t	# Fea	f^1	f^2/t	f^3/t	f^4/t
5	**48.67**	49.02/1.80	**41.12/9.58**	**51.84/5.96**	5	**63.43**	63.91/0.24	**47.17/7.16**	61.66/0.72
10	**64.22**	**61.40/7.14**	**51.71/17.07**	**61.86/4.39**	10	**69.77**	72.13/1.14	**56.58/4.97**	71.00/0.53
15	**70.55**	70.75/0.54	**59.31/20.52**	**64.92/6.46**	15	69.84	72.41/2.07	64.28/1.95	70.40/0.28
20	**72.98**	72.18/1.42	**64.41/17.74**	71.73/1.77	20	71.85	72.96/0.64	66.52/1.55	72.72/0.47
25	**75.05**	**73.44/4.20**	**65.35/14.76**	**73.22/2.79**	25	73.72	73.54/0.11	**66.51/2.82**	73.25/0.20
30	**76.07**	**74.89/5.21**	**65.77/19.22**	**73.99/4.09**	30	74.10	74.89/0.58	72.16/1.14	75.36/0.77

Table 3.4: kNN classification accuracies with different feature selection methods. Bold items highlight significant differences in comparison with f^1 (t-value > 2.26 or p-value < 0.05)

k-Nearest-Neighbor (kNN, k=3)									
Madelon					Multiple Features				
# Fea	f^1	f^2/t	f^3/t	f^4/t	# Fea	f^1	f^2/t	f^3/t	f^4/t
5	**59.92**	**54.96/3.65**	**56.08/2.70**	**56.19/2.56**	5	**89.35**	90.65/1.72	**86.25/3.18**	**87.45/2.71**
10	**58.77**	**54.08/5.27**	**52.38/6.29**	**52.38/6.29**	10	**93.60**	94.60/1.96	94.35/1.39	**96.60/6.80**
15	**58.00**	**52.50/3.34**	**51.19/4.86**	**51.19/4.86**	15	**96.95**	**95.35/2.42**	**98.00/5.55**	97.45/1.50
20	**56.00**	**52.38/5.34**	**53.42/2.93**	**53.42/2.93**	20	**97.95**	**97.05/3.67**	98.05/0.30	98.20/1.34
25	**55.81**	**51.77/3.51**	54.00/2.07	54.00/2.07	25	97.85	97.35/1.79	98.35/2.12	98.20/1.91
30	**55.73**	**51.08/5.94**	**52.42/3.99**	**52.42/3.99**	30	**98.25**	**97.35/2.86**	97.90/1.48	97.95/1.11
Landsat Satellite					Sonar				
# Fea	f^1	f^2/t	f^3/t	f^4/t	# Fea	f^1	f^2/t	f^3/t	f^4/t
5	85.76	85.97/0.54	86.03/0.49	85.95/0.37	5	74.04	70.65/1.42	75.51/0.78	75.51/0.78
10	88.70	89.40/1.52	88.28/2.18	88.83/0.35	10	**79.84**	**70.65/2.99**	85.45/1.81	85.45/1.81
15	**89.77**	**90.85/3.77**	89.70/0.29	89.62/1.07	15	**83.18**	**72.68/2.61**	85.13/0.75	85.13/0.75
20	90.19	90.94/2.10	89.93/1.30	90.50/1.57	20	**85.07**	**70.73/6.89**	86.13/0.46	86.13/0.46
25	90.74	90.67/0.25	90.27/2.13	90.86/1.10	25	**87.49**	**76.01/5.75**	87.47/0.02	87.47/0.02
30	90.89	90.85/0.27	90.78/1.17	91.03/1.49	30	**86.09**	**75.10/3.29**	87.04/0.51	87.04/0.51
Spambase					Breast Cancer				
# Fea	f^1	f^2/t	f^3/t	f^4/t	# Fea	f^1	f^2/t	f^3/t	f^4/t
5	86.63	80.92/1.66	79.18/1.85	79.18/1.85	5	**94.02**	**91.22/2.58**	95.07/1.96	95.07/1.96
10	**89.91**	**88.29/2.93**	89.13/0.69	89.09/0.74	10	**93.84**	**90.68/2.43**	93.84/0.00	93.84/0.00
15	90.46	89.57/1.94	90.59/0.34	90.57/0.28	15	93.50	94.36/1.15	93.14/0.80	93.14/0.80
20	**89.59**	90.24/1.34	**91.65/3.54**	**91.63/3.57**	20	96.32	95.25/0.85	96.32/0.00	96.32/0.00
25	**90.13**	90.46/0.64	91.04/1.01	**91.57/2.36**	25	97.36	96.48/1.17	97.19/1.00	97.19/1.00
30	**90.26**	90.59/0.80	**91.05/3.63**	**91.05/3.63**	30	96.83	97.01/0.27	96.83/0.00	96.83/0.00

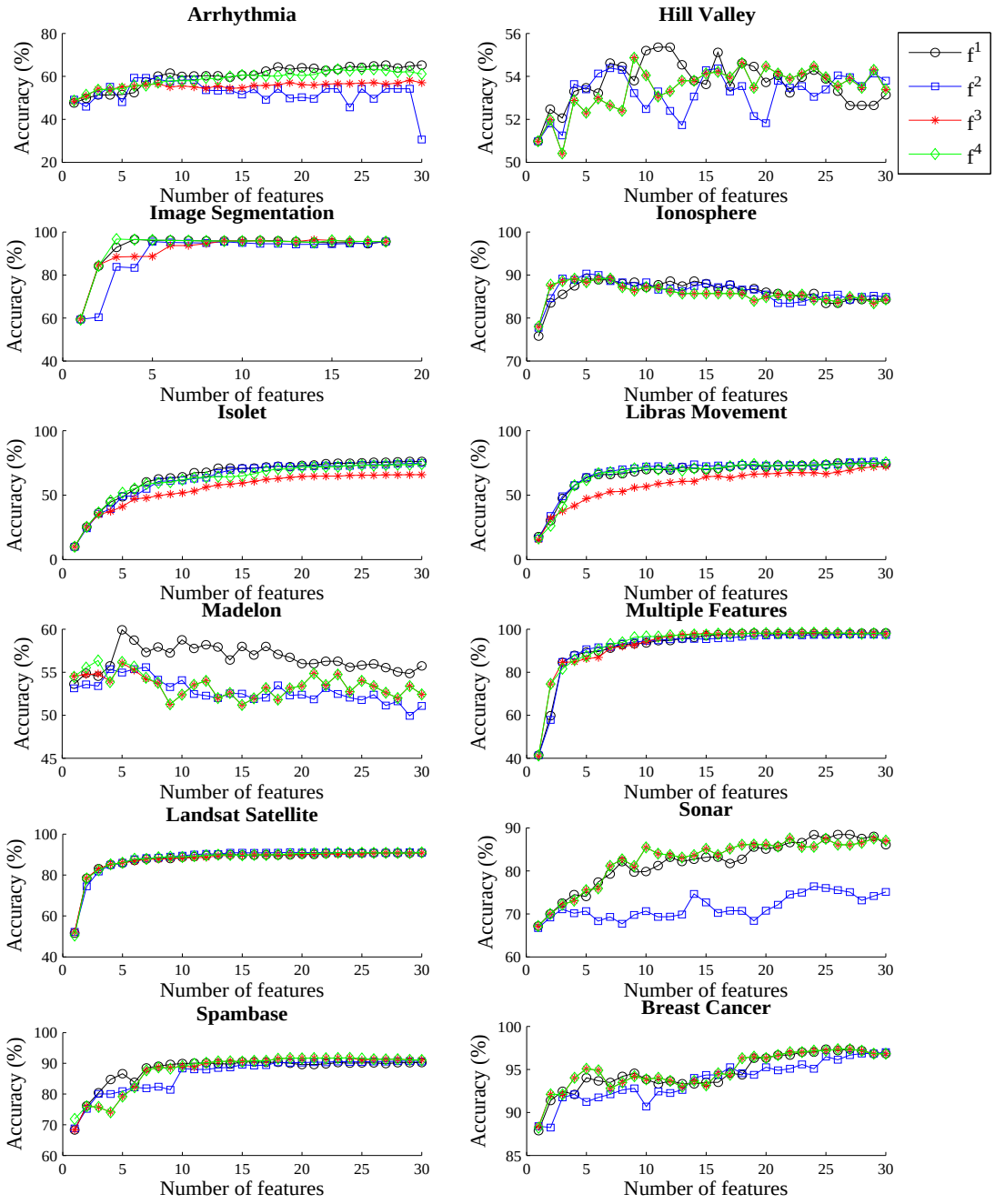


Figure 3.6: kNN classification accuracies of the 12 datasets with different feature selection methods

Table 3.5: SVM classification accuracies with different feature selection methods. Bold items highlight significant differences in comparison with f^1 (t-value > 2.26 or p-value < 0.05)

Support Vector Machine (SVM)									
Arrhythmia					Hill Valley				
# Fea	f^1	f^2/t	f^3/t	f^4/t	# Fea	f^1	f^2/t	f^3/t	f^4/t
5	**60.52**	46.94/1.57	**56.73/2.72**	57.87/1.35	5	50.49	51.07/0.96	51.24/1.79	50.58/0.19
10	**65.55**	**42.38/2.40**	**58.26/7.35**	**59.17/4.36**	10	51.15	51.32/0.25	51.15/0.00	50.74/1.10
15	**64.65**	**25.93/4.29**	**60.23/2.45**	63.37/0.61	15	51.15	50.49/1.50	51.32/0.52	50.66/1.20
20	**67.27**	**9.78/37.30**	**60.67/5.74**	64.08/1.43	20	**50.74**	50.74/0.00	**51.81/2.90**	50.99/0.61
25	**67.28**	**10.63/45.53**	**60.47/4.45**	64.73/1.07	25	51.15	50.99/0.51	51.15/0.01	51.40/0.58
30	**68.18**	**10.41/44.36**	**60.07/3.88**	66.11/0.71	30	51.32	51.48/0.42	50.82/1.40	50.91/0.96
Image Segmentation					Ionosphere				
# Fea	f^1	f^2/t	f^3/t	f^4/t	# Fea	f^1	f^2/t	f^3/t	f^4/t
3	**77.97**	75.19/1.54	75.19/1.60	**88.35/5.97**	5	88.27	89.69/1.12	90.02/1.55	90.02/1.55
6	**90.00**	**87.32/9.34**	**84.42/4.41**	**91.77/3.46**	10	**92.28**	92.02/0.39	**90.84/3.00**	**91.14/2.45**
9	**92.64**	**87.58/12.41**	92.38/1.51	92.77/1.41	15	93.16	92.86/0.45	94.60/2.21	94.60/2.21
12	**93.29**	**91.47/4.92**	93.29/0.00	93.33/1.00	20	94.89	94.00/1.04	93.77/1.82	93.77/1.82
15	**93.38**	**92.42/4.14**	**94.24/2.33**	**94.24/2.37**	25	94.88	94.85/0.04	95.19/0.47	95.19/0.47
18	93.90	93.90/0.00	93.85/0.55	93.90/0.00	30	95.16	95.14/0.04	95.73/1.50	95.73/1.50
Isolet					Libras Movement				
# Fea	f^1	f^2/t	f^3/t	f^4/t	# Fea	f^1	f^2/t	f^3/t	f^4/t
5	**53.24**	52.69/0.68	**46.70/6.39**	**56.28/3.99**	5	**47.54**	51.96/1.63	**40.77/2.34**	**44.77/2.37**
10	**68.90**	**65.56/6.72**	**57.12/17.58**	**66.14/6.48**	10	**61.68**	64.05/0.72	**51.63/4.78**	64.86/1.44
15	**73.66**	**74.70/2.65**	**64.35/20.25**	**70.99/3.69**	15	**72.19**	71.70/0.30	**57.46/5.36**	71.02/0.67
20	**76.40**	76.14/0.57	**70.19/18.47**	77.02/1.19	20	**75.65**	77.04/0.92	**63.49/4.33**	75.39/0.16
25	**78.81**	78.50/0.74	**71.04/17.15**	78.62/0.40	25	**77.30**	80.05/1.80	**72.29/2.38**	77.34/0.05
30	**80.11**	80.12/0.03	**71.99/19.72**	79.83/0.58	30	77.57	81.93/2.14	74.84/1.67	79.33/1.47

Table 3.6: SVM classification accuracies with different feature selection methods. Bold items highlight significant differences in comparison with f^1 (t-value > 2.26 or p-value < 0.05)

Support Vector Machine (SVM)									
Madelon					Multiple Features				
# Fea	f^1	f^2/t	f^3/t	f^4/t	# Fea	f^1	f^2/t	f^3/t	f^4/t
5	**61.65**	**53.42/8.52**	**53.73/6.80**	**54.12/6.87**	5	**90.10**	91.65/1.85	**86.85/3.33**	**86.60/3.68**
10	**64.00**	**53.81/9.37**	**56.62/6.27**	**56.69/6.90**	10	**93.65**	94.45/1.65	**94.95/5.46**	**96.80/7.47**
15	**62.35**	**53.85/5.69**	**55.31/5.50**	**55.12/5.77**	15	**97.85**	**95.85/4.67**	97.85/0.00	97.65/0.65
20	**61.92**	**53.50/5.74**	**57.73/5.59**	**57.46/6.14**	20	98.20	97.95/0.86	98.45/0.96	98.45/0.83
25	**61.31**	**54.12/8.29**	**57.23/6.87**	**57.31/5.68**	25	**98.65**	**98.10/2.40**	98.55/0.69	98.25/1.56
30	**61.54**	**53.35/8.27**	**55.73/5.55**	**55.96/5.65**	30	**98.75**	98.35/1.50	**98.35/2.45**	98.45/1.41
Landsat Satellite					Sonar				
# Fea	f^1	f^2/t	f^3/t	f^4/t	# Fea	f^1	f^2/t	f^3/t	f^4/t
5	85.50	84.82/2.02	85.56/0.17	85.27/1.24	5	73.67	75.94/0.98	72.67/0.47	73.17/0.28
10	87.19	87.72/1.53	87.27/0.36	87.13/0.27	10	76.06	75.94/0.05	76.46/0.14	77.37/0.42
15	**88.94**	88.94/0.01	**88.38/2.93**	88.75/1.10	15	78.44	75.53/1.32	80.30/1.05	80.77/1.22
20	**89.17**	**89.90/3.84**	89.32/0.80	89.36/1.65	20	81.77	75.53/2.20	81.82/0.04	82.75/0.82
25	**89.79**	89.95/1.93	89.98/0.87	**90.01/2.95**	25	**83.18**	**75.51/2.74**	83.75/0.35	84.25/0.51
30	90.35	90.23/0.69	90.33/0.44	90.35/0.02	30	**82.25**	**74.10/3.59**	**85.63/3.30**	**85.63/3.30**
Spambase					Breast Cancer				
# Fea	f^1	f^2/t	f^3/t	f^4/t	# Fea	f^1	f^2/t	f^3/t	f^4/t
5	88.22	87.31/2.11	87.87/0.75	87.78/0.93	5	**95.42**	**92.60/2.75**	94.54/1.62	94.54/1.62
10	**90.85**	85.57/1.79	**89.83/2.80**	**89.87/2.64**	10	**95.78**	**92.42/3.36**	95.60/1.00	95.78/0.00
15	90.94	85.78/1.67	90.72/0.69	90.74/0.63	15	95.25	94.36/0.96	94.72/1.96	94.89/1.01
20	91.46	86.81/1.49	91.59/0.63	91.57/0.46	20	97.00	96.29/0.85	97.18/0.42	97.18/0.54
25	92.09	86.22/1.96	91.55/1.65	91.52/1.71	25	97.53	97.36/0.36	97.53/0.00	97.36/1.00
30	92.02	86.33/1.95	91.72/1.14	91.72/1.08	30	97.53	97.88/0.79	97.53/0.00	97.53/0.00

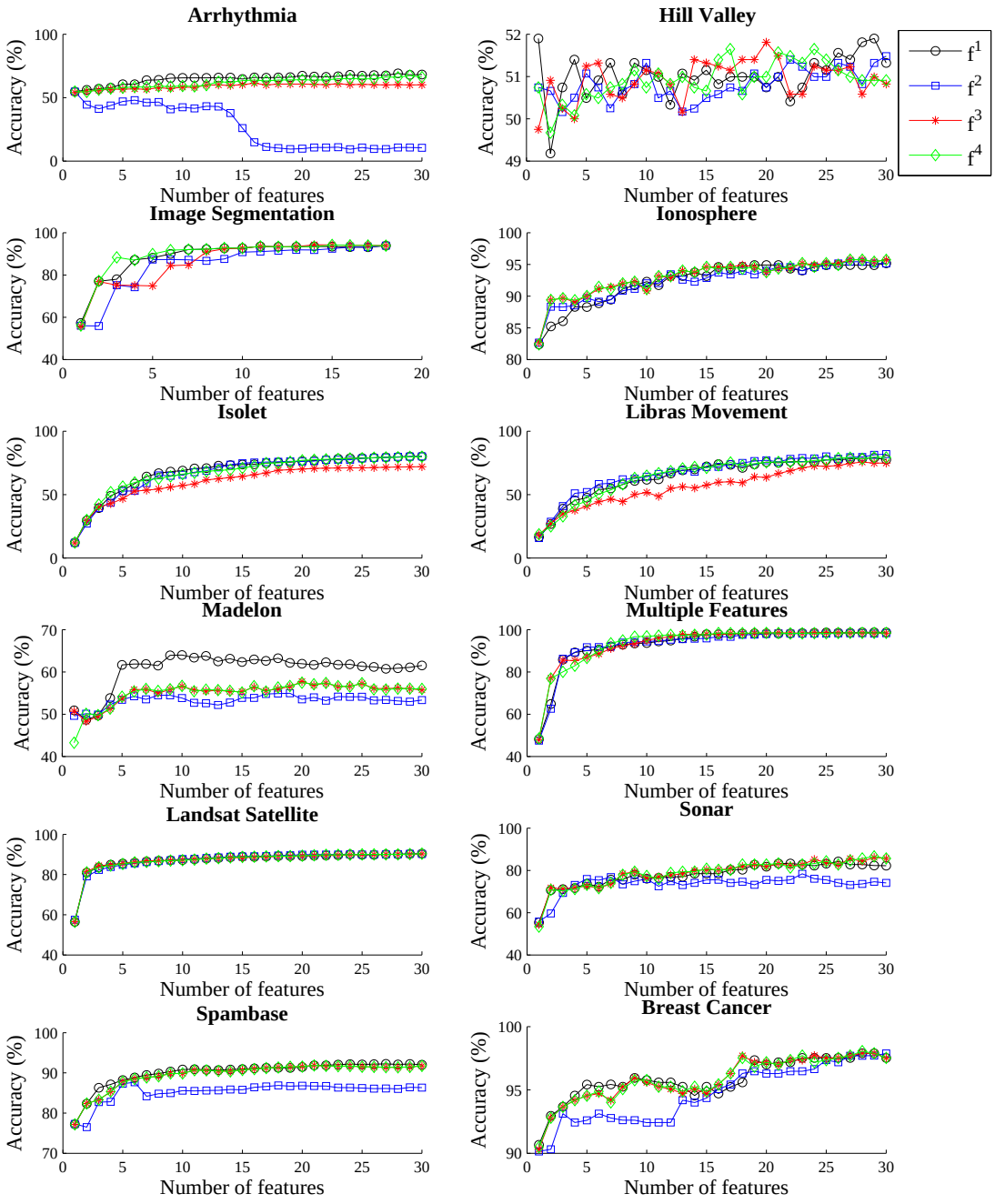


Figure 3.7: SVM classification accuracies of the 12 datasets with different feature selection methods

Table 3.7: LDA classification accuracies with different feature selection methods. Bold items highlight significant differences in comparison with f^1 (t-value > 2.26 or p-value < 0.05)

Linear Discriminant Analysis (LDA)									
Arrhythmia					Hill Valley				
# Fea	f^1	f^2/t	f^3/t	f^4/t	# Fea	f^1	f^2/t	f^3/t	f^4/t
5	**24.17**	21.92/0.44	**14.41/2.80**	22.75/0.40	5	51.32	50.91/1.62	51.07/1.00	51.07/1.00
10	**33.96**	25.04/1.54	**27.28/3.06**	35.83/0.78	10	51.40	51.15/1.15	51.57/0.61	51.57/0.61
15	**44.74**	**27.04/3.14**	**34.63/4.44**	38.90/1.74	15	51.57	51.07/1.33	51.40/0.40	51.40/0.40
20	**50.89**	**7.00/11.68**	**39.99/4.07**	**41.34/2.53**	20	51.65	51.15/1.97	51.48/0.56	51.48/0.56
25	**55.22**	**10.27/11.47**	**39.41/5.14**	**45.32/2.64**	25	51.57	51.24/1.31	51.65/0.36	51.65/0.36
30	**55.30**	**15.54/8.90**	**41.40/5.13**	49.10/2.16	30	**51.49**	**50.66/2.74**	51.90/1.47	51.90/1.47
Image Segmentation					Ionosphere				
# Fea	f^1	f^2/t	f^3/t	f^4/t	# Fea	f^1	f^2/t	f^3/t	f^4/t
3	**78.14**	**68.14/5.01**	**73.03/3.13**	**81.65/2.31**	5	**84.29**	84.29/0.01	**80.56/2.51**	**80.56/2.51**
6	**84.59**	**73.16/14.71**	**78.18/4.76**	**87.32/5.81**	10	**84.83**	85.72/0.74	**81.44/2.55**	**81.44/2.55**
9	**87.92**	**81.77/8.70**	87.62/1.00	87.97/0.22	15	**83.67**	85.11/1.89	**81.11/3.23**	**81.11/3.23**
12	**89.87**	**88.23/2.98**	89.78/1.00	89.87/0.00	20	85.68	86.55/1.01	81.70/2.03	81.70/2.03
15	**89.09**	**88.66/2.38**	88.57/0.87	88.57/0.87	25	86.85	87.71/1.96	84.82/1.17	84.82/1.17
18	88.79	88.79/0.00	88.79/0.00	88.79/0.00	30	**85.99**	**87.70/2.71**	84.23/0.91	84.23/0.91
Isolet					Libras Movement				
# Fea	f^1	f^2/t	f^3/t	f^4/t	# Fea	f^1	f^2/t	f^3/t	f^4/t
5	**48.31**	47.86/1.00	**41.07/8.55**	49.99/2.10	5	**46.21**	46.81/0.28	**39.18/2.62**	44.62/0.82
10	**61.25**	**57.37/7.10**	**48.04/15.71**	**58.05/5.19**	10	**54.37**	54.94/0.32	**46.06/2.86**	52.62/1.45
15	**63.68**	**65.55/3.76**	**54.14/14.40**	**61.70/2.64**	15	**59.30**	60.98/0.88	**47.77/6.65**	58.72/0.70
20	**65.27**	**66.90/3.70**	**60.83/10.63**	**66.60/2.63**	20	**60.43**	**65.09/2.37**	**52.01/4.85**	62.77/1.74
25	**67.31**	66.27/1.72	**60.81/9.52**	67.82/0.68	25	**64.54**	67.58/1.80	**56.84/4.76**	63.61/0.64
30	**68.21**	68.05/0.27	**62.18/13.03**	69.31/1.84	30	**65.01**	67.55/2.13	**58.33/4.58**	64.81/0.23

Table 3.8: LDA classification accuracies with different feature selection methods. Bold items highlight significant differences in comparison with f^1 (t-value > 2.26 or p-value < 0.05)

Linear Discriminant Analysis (LDA)									
Madelon					Multiple Features				
# Fea	f^1	f^2/t	f^3/t	f^4/t	# Fea	f^1	f^2/t	f^3/t	f^4/t
5	60.62	60.04/0.56	60.19/0.42	60.19/0.42	5	**88.35**	87.50/1.08	**81.75/8.45**	**81.95/6.59**
10	60.42	59.62/0.65	59.88/0.43	59.88/0.43	10	90.30	90.90/1.05	90.20/0.26	**91.70/2.35**
15	60.69	59.81/0.72	59.58/1.34	59.58/1.34	15	**94.55**	**92.40/2.90**	**95.90/2.61**	94.60/0.18
20	60.54	60.08/0.53	60.08/0.61	60.08/0.61	20	95.30	94.95/0.70	95.80/1.63	95.00/0.97
25	60.92	59.38/2.08	59.54/1.96	59.54/1.96	25	95.75	95.85/0.19	96.10/1.41	95.95/0.45
30	60.15	59.08/1.05	59.46/1.19	59.46/1.19	30	96.15	95.75/0.95	96.35/0.69	96.75/1.86
Landsat Satellite					Sonar				
# Fea	f^1	f^2/t	f^3/t	f^4/t	# Fea	f^1	f^2/t	f^3/t	f^4/t
5	81.03	80.22/2.11	81.38/1.36	80.98/0.19	5	69.72	68.27/0.54	71.22/0.59	71.22/0.59
10	82.33	81.94/1.42	82.14/1.09	81.99/1.13	10	70.65	67.77/0.93	72.58/1.29	72.58/1.29
15	82.64	82.39/0.94	82.50/0.59	82.41/1.60	15	75.05	69.32/1.59	75.94/0.51	75.94/0.51
20	82.44	82.38/0.28	82.60/0.83	82.60/1.63	20	77.39	72.54/2.18	78.01/0.28	78.01/0.28
25	82.13	82.24/0.63	82.33/1.17	82.27/0.93	25	76.96	73.47/1.13	78.39/0.82	78.39/0.82
30	82.38	82.38/0.01	82.33/1.00	82.55/1.94	30	78.05	73.51/1.39	77.96/0.05	77.96/0.05
Spambase					Breast Cancer				
# Fea	f^1	f^2/t	f^3/t	f^4/t	# Fea	f^1	f^2/t	f^3/t	f^4/t
5	83.48	83.63/0.30	84.57/2.24	84.57/2.24	5	**94.03**	**91.04/3.79**	94.20/0.54	94.20/0.54
10	**86.57**	85.89/1.05	**87.92/4.35**	**87.92/4.35**	10	**94.56**	**91.56/3.43**	94.56/0.00	94.56/0.00
15	87.00	87.48/0.69	87.42/1.19	87.42/1.19	15	94.38	94.20/0.42	94.02/1.50	94.02/1.50
20	**87.42**	87.57/0.28	**88.37/3.00**	**88.37/3.00**	20	95.96	95.25/1.21	95.96/0.00	95.96/0.00
25	**87.94**	88.07/0.25	**89.13/4.50**	**89.13/4.50**	25	**96.49**	**95.60/3.00**	96.67/1.00	96.67/1.00
30	**88.70**	88.59/0.27	**89.81/4.02**	**89.81/4.02**	30	96.67	96.14/1.41	96.67/0.00	96.67/0.00

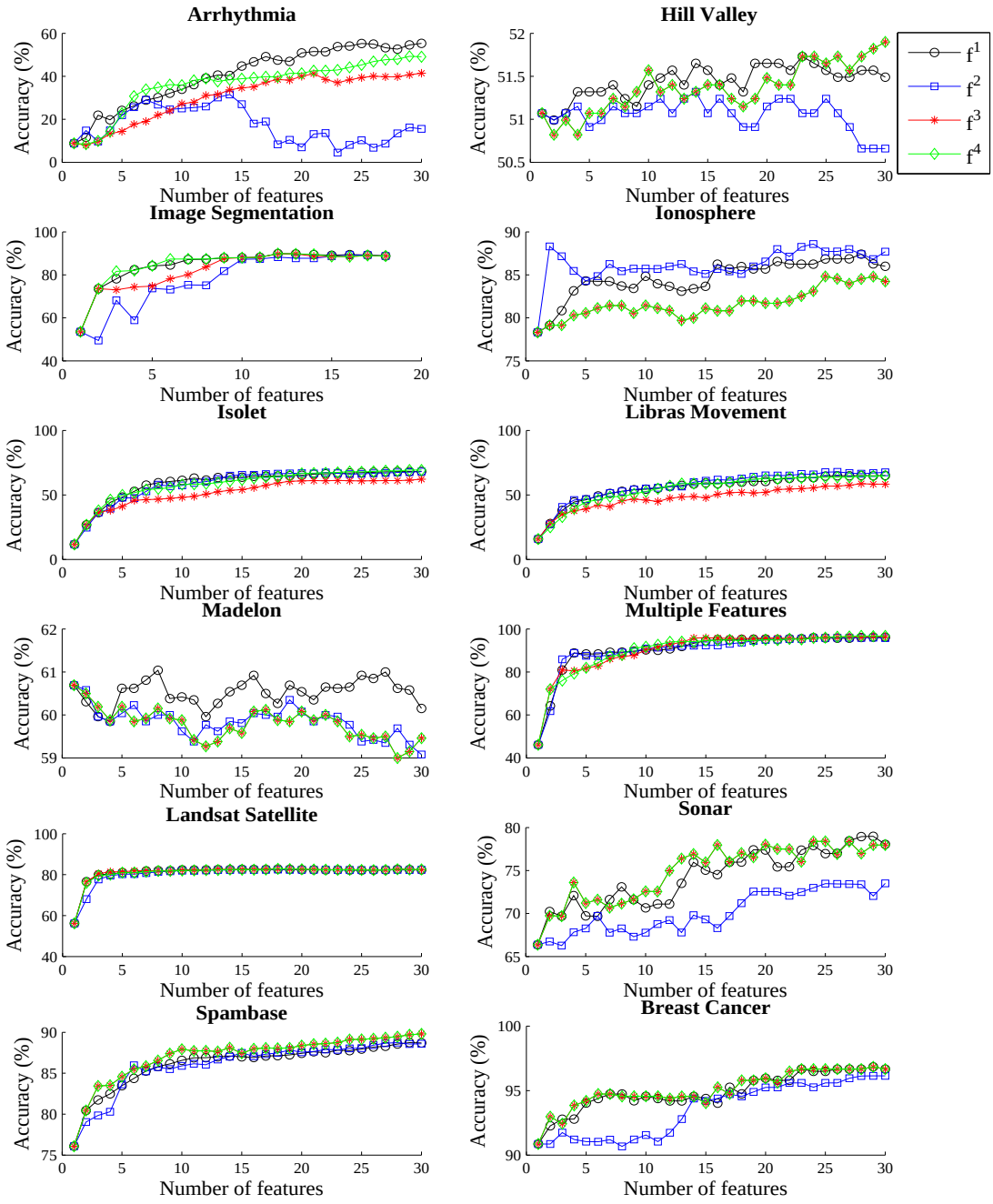


Figure 3.8: LDA classification accuracies of the 12 datasets with different feature selection methods

Chapter 4

Classification

This chapter provides the details of the classification module in the proposed activity recognition system illustrated in Figure 4.1. The first section describes how the quantization module converts continuous (scalar) feature vectors into discrete values. The sequences of discrete values are input to the classification algorithm called Semi-Markov Conditional Random Fields model (Semi-CRF), which is presented in the next section. The two last sections show the experiments to validate the operation of the Semi-CRF model individually and of the whole proposed system.

4.1 Quantization

Given the data matrix from the feature selection step $\overline{F}(M - by - K)$, this quantization step aims at constructing a code-book of V vectors by Linde Buzo Gray (LBG) clustering algorithm [53] presented below.

LBG clustering algorithm

Step 1: Initialization

$$V = 1, C_1 = \frac{1}{M} \sum_{m=1}^{M} \overline{F}_m,$$

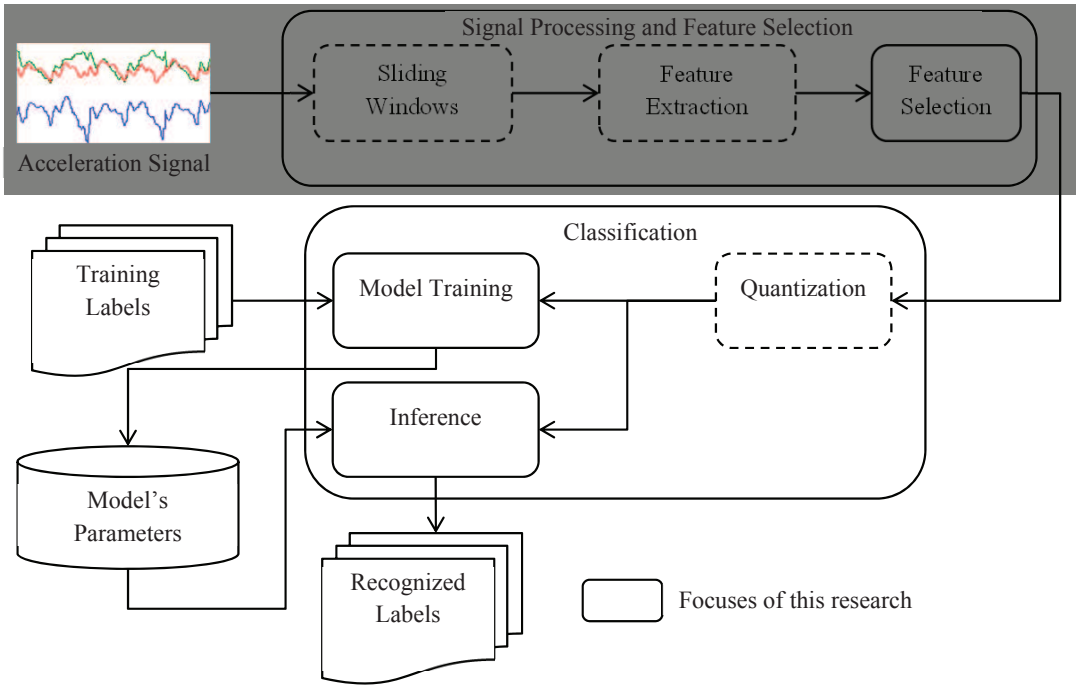


Figure 4.1: Block diagram of the classification module (not shaded area)

Step 2: Splitting

For $i = 1, 2, ..., V$

$$C_{V+i} = (1 - \epsilon)C_i,$$

$$C_i = (1 + \epsilon)C_i,$$

$$V = 2V,$$

Step 3: Updating

For $i = 1, 2, ..., V$, find J_i, a cluster centered at C_i .

For $i = 1, 2, ..., V$

$$C_i = \frac{\sum_{m=1, \overline{F}_m \in J_i}^{M} \overline{F}_m}{|J_i|},$$

Step 4: Repeating step 2 and step 3 until the desired number of code-book vectors is obtained.

Once the code-book vectors $C_1, C_2, ..., C_V$ are determined, all the feature vectors are matched to the index of the nearest code-book vector. Therefore, after the quantization step, the input acceleration is converted to a sequence of discrete (integer) values denoted by $X = \{x_1, x_2, ..., x_T\}$, where T is the total number of frames. This process is depicted in Figure 4.2. Hereafter, X represents a sequence of discrete values, and Y represents a sequence of the corresponding labels.

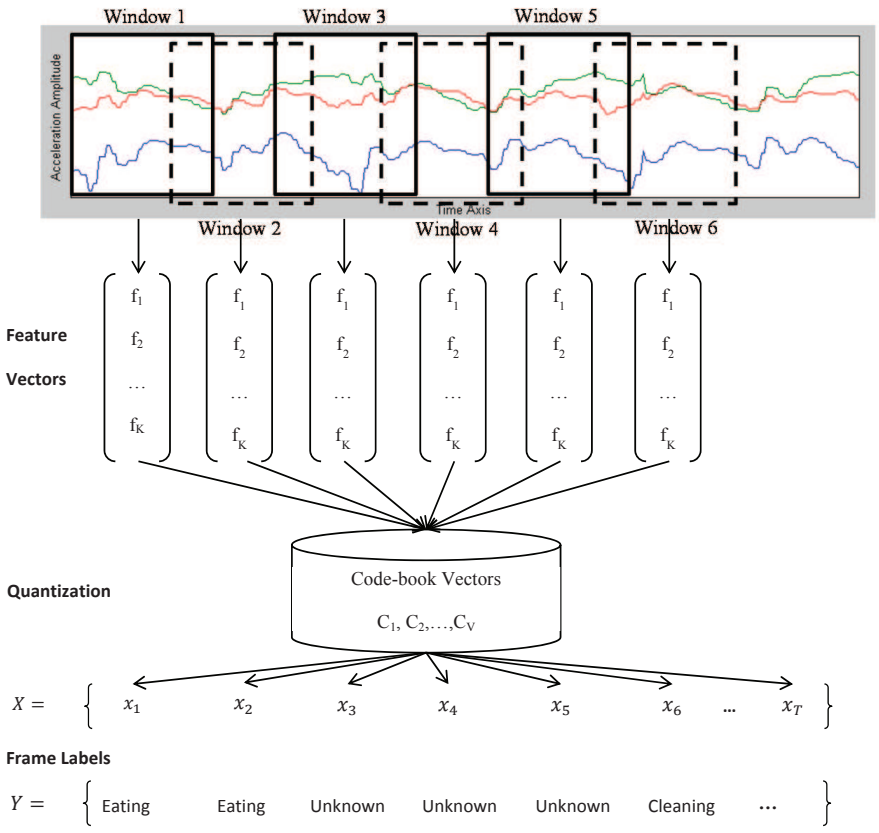


Figure 4.2: Making a sequence of discrete values from a continuous temporal input

4.2 Semi-Markov Conditional Random Fields (Semi-CRF)

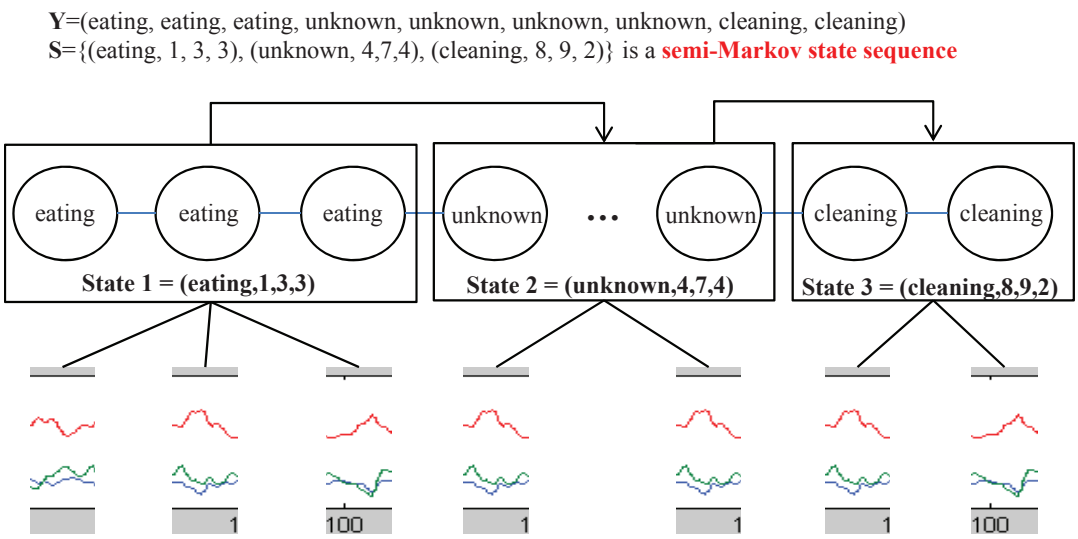


Figure 4.3: A Semi-Markov sequence (S) constructed by [85]

As mentioned in section 2.3.3, in a Semi-CRF model, the state sequence S=(y_1, b_1, e_1, d_1), (y_2, b_2, e_2, d_2),...,(y_P, b_P, e_P, d_P) (y, b, e and d are the label, beginning time, ending time and duration respectively, P is the length of S) is constructed by grouping all the same consecutive labels from the original label sequence (Y) as illustrated in Figure 4.3. Because these Semi-Markov states are subject to constraints (2.13-2.16) preventing the algorithm from learning the long-range relationship of activity labels. For example, consider the sequence in Figure 4.3, because the unknown activity happens between the two target activities the model is not able to learn the relationship between eating and cleaning.

It is possible to tackle the problem if the Semi-Markov sequence is constructed without the unknown activities, for instance S=(eating, 1, 3, 3), (cleaning, 8, 9, 2) (shown in Figure 4.4).

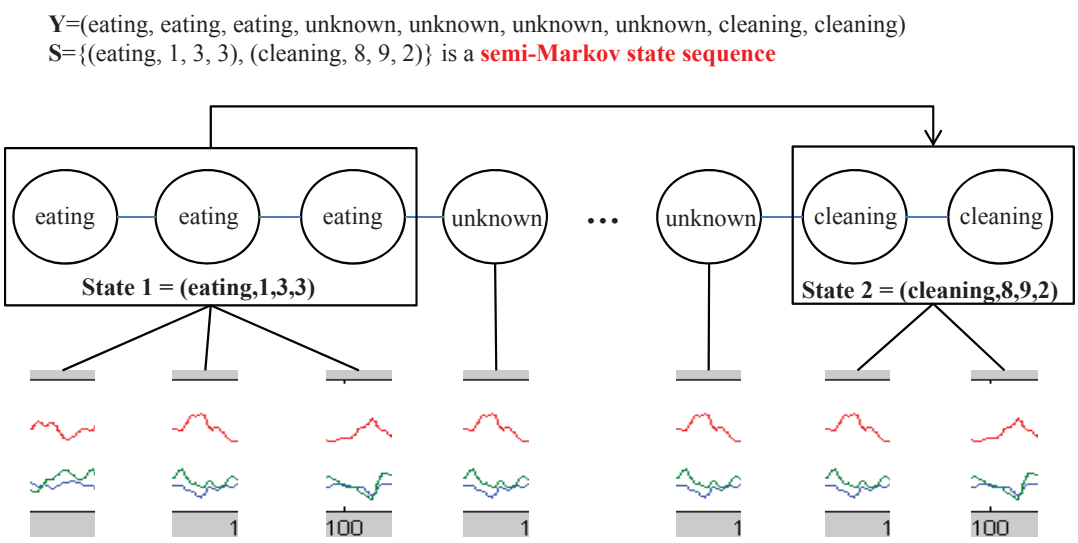


Figure 4.4: A Semi-Markov sequence (S) constructed by our proposed method

Therefore, inequalities should be used instead of the equalities in (2.13-2.16). This leads to our below constraints

$$0 < s_i.b \leq s_i.e < s_{i+1}.b \leq s_{i+1}.e \leq T \; i = 1, 2, ..., P-1. \tag{4.1}$$

Now, based on the Semi-Markov sequence S, the log-likelihood of a training sequence is computed by [85]

$$P(S|X) = \frac{\prod_{i=1}^{P} \Psi(s_{i-1}, s_i, X)}{Z_X}, \tag{4.2}$$

$$Z_X = \sum_{S'} \prod_{i=1}^{P'} \Psi(s'_{i-1}, s'_i, X). \tag{4.3}$$

Because the proposed model is designed to learn three characteristics:

- the relationship between any two consecutive activities (transition),

- the duration distribution of activities (duration),
- the relationship between sensory data and activity labels (observation),

we define Ψ functions in the below form

$$\Psi(s_{i-1}, s_i, X) = \begin{pmatrix} e^{Q^{Tr}(s_{i-1},s_i,X)\times} \\ e^{Q^{D}(s_{i-1},s_i,X)\times} \\ e^{Q^{O}(s_{i-1},s_i,X)} \end{pmatrix}. \tag{4.4}$$

The weighted transition potential function is given by

$$Q^{Tr}(s_{i-1}, s_i, X) = \sum_{y',y} w^{Tr}(y', y)\delta(s_{i-1}.y = y', s_i.y = y), \tag{4.5}$$

where $w^{Tr}(y', y)$ is the weight of transition from label y' to label y and δ is given by

$$\delta(A) = \begin{cases} 1 \text{ if A is true} \\ 0 \text{ if A is false} \end{cases}. \tag{4.6}$$

The weighted duration potential function of an target activity is calculated as

$$\begin{aligned} Q^{D}(s_{i-1}, s_i, X) &= \sum_{y,d} G^{D}(y, d)\delta(s_i.y = y, d = s_i.e - s_i.b + 1) \\ &= \sum_{y,d} w^{D}(y)\frac{(d - m_y)^2}{-2\sigma_y^2}\delta(s_i.y = y, d = s_i.e - s_i.b + 1), \end{aligned} \tag{4.7}$$

where $w^{D}(y)$ is the duration weight of label y. m_y and σ_y are the empirical average and standard deviation of label y's duration, respectively, which can be easily extracted from training data. Figure 4.5 depicts the shape of the corresponding potential function $e^{G^{D}(y,d)}$ with three different values of y. Clearly, the most likely duration (at the center of the bell) has the highest potential value. Since unknown activities can have an arbitrary length, it is not practical to model the duration of such activities because the modeling may increase the possible maximum duration length, D, which may result in high complexity.

Note that in (4.7), it is assumed that the duration of an activity has a Gaussian-like distribution. Although the assumption is not always true, it is reasonable to assume that since most activities often last around a constant amount of time.

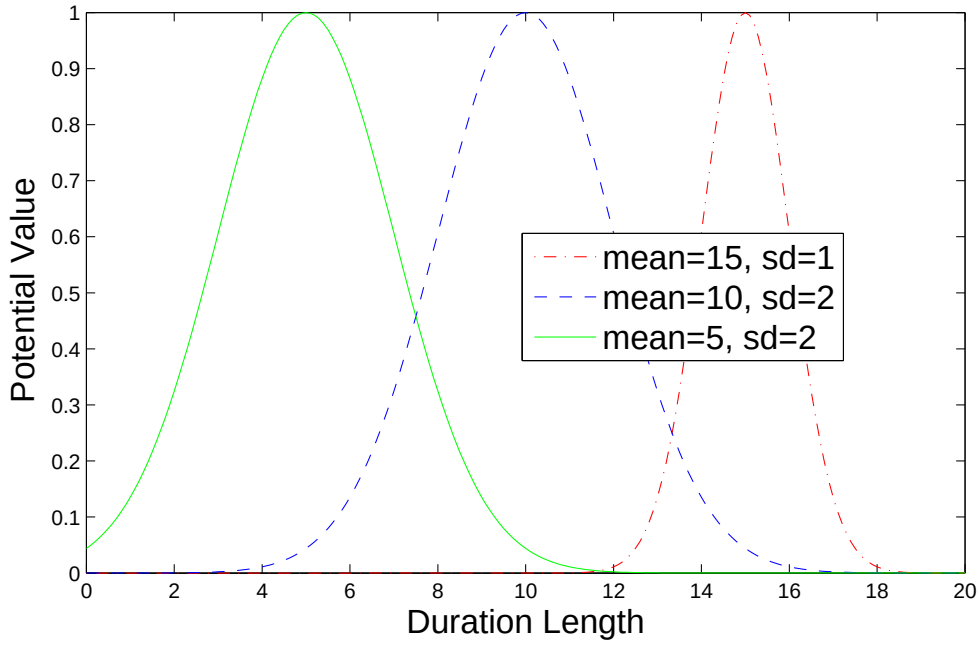


Figure 4.5: Duration potential with different values of mean and standard deviation. If a detected segment has a length of, for example 6, then it most likely belongs to the same class with the label, whose duration potential is presented in the green

The weighted observation potential function is defined as

$$Q^O(s_{i-1}, s_i, X) = \sum_{y,t_1,t_2} \begin{pmatrix} G_y(y,t_1,t_2)\times \\ \delta(s_i.y = y, s_i.b = t_1, s_i.e = t_2)+ \\ G_u(u,t_1,t_2)\times \\ \delta(s_{i-1}.e + 1 = t_1, s_i.b - 1 = t_2) \end{pmatrix}, \tag{4.8}$$

where

$$G_y(y, t_1, t_2) = \sum_{t=t_1}^{t_2} \sum_{o} w^O(y, o)\delta(x_t = o), \tag{4.9}$$

$$G_u(u, t_1, t_2) = \sum_{t=t_1}^{t_2} \sum_{o} w^O(u, o)\delta(x_t = o), \tag{4.10}$$

where $w^O(y, o)$ and $w^O(u, o)$ in that order are the weights of the observation given that input symbol o is observed in state with label y (target activity) and u (unknown activity). Because the observation term often goes together with the duration term, therefore in the presentation of the following equations, a combined potential function of the below form is used

$$G(y, t_1, t_2) = G_y(y, t_1, t_2) + G^D(y, t_2 - t_1 + 1). \tag{4.11}$$

It can be seen that, the proposed approach is similar to that in [85], but it allows discontinuity in the time of state by using $s_{i+1}.b > s_i.e$ instead of $s_{i+1}.b = s_i.e + 1$. The inequality enables our model to skip unknown activities and directly model the transition between two target activities.

The training phase requires finding the maximum point of the log-likelihood function. Since there is no analytic solution, a well-known "climbing-hill" method (L-BFGS) [69] is used. In order to find the local maximum point using L-BGFS algorithm, all the gradients have to be computed repeatedly at every step. Therefore an efficient gradient computing algorithm may significantly decrease the whole training time. In the following sections, all the detail about the proposed computation algorithms is presented including: forward, backward, and individual gradient computing algorithms. Furthermore, the Viterbi algorithm [80], [109], which is is adapted to the proposed model in the inference phase, is presented section 4.2.5

4.2.1 Forward Algorithm

We proposed the following forward algorithm to compute the normalization factor Z_X efficiently by using the dynamic programming method. Firstly, α, which is denoted as a forward variable, is

computed by

$$\alpha(y,t) = \sum_{S^t \in \Gamma_t^y} Pol(S^t) = \sum_{S^t \in \Gamma_t^y} \prod_{i=1}^{q} \Psi(s_{i-1}, s_i, X), \tag{4.12}$$

where $\Gamma_t^y = \{S = s_1, s_2, ..., s_q\}$ is a set of all Semi-Markov sequences, which have an original label sequence $(y_1, y_2, ..., y_t)$ with the last expected label is y. Thus, every $S^t = s_1, s_2, ..., s_q \in \Gamma_t^y$ satisfies $s_q.e \leq t$ and $s_q.y = y$.

$$\gamma(y,t) = \sum_{S^t \in \Lambda_t^y} Pol(S^t) = \sum_{S^t \in \Lambda_t^y} \prod_{i=1}^{q} \Psi(s_{i-1}, s_i, X), \tag{4.13}$$

where $\Lambda_t^y = \{S = s_1, s_2, ..., s_q\}$, is a set of all Semi-Markov sequences, which have an original label sequence $(y_1, y_2, ..., y_{t-1}, y_t = y)$. Therefore, every $S^t = s_1, s_2, ..., s_q \in \Lambda_t^y$ satisfies $s_q.e = t$ and $s_q.y = y$. Let ϕ^t represent a special sequence of length t, which contains only "*u*" labels. Equation (4.3) is equal to

$$\begin{aligned} Z_X &= \sum_{S^T} Pol(S^T) \\ &= \sum_{y} \sum_{S^T \in \Gamma_T^y} Pol(S^T) + Pol(\phi^T) \\ &= \sum_{y} \alpha(y,T) + e^{G_u(u,1,T)}. \end{aligned} \tag{4.14}$$

To compute $\alpha(y,t)$ efficiently, it is noted that

$$\begin{aligned} \alpha(y,t) &= \sum_{S^t \in \Gamma_t^y} Pol(S^t) \\ &= \sum_{S^{t-1} \in \Gamma_{t-1}^y} Pol(S^{t-1} \oplus u) + \sum_{S^t \in \Lambda_t^y} Pol(S^t), \end{aligned} \tag{4.15}$$

where $S^{t-1} \oplus u$ denotes the concatenation of a label "*u*" to the end of the original sequence of S^{t-1}. It is easy to see that

$$Pol(S^{t-1} \oplus u) = Pol(S^{t-1}) e^{G_u(u,t,t)}. \tag{4.16}$$

From (4.2.1) and (4.16) it is clear that

$$\alpha(y,t) = \alpha(y,t-1)e^{G_u(u,t,t)} + \gamma(y,t). \tag{4.17}$$

Deriving $\gamma(y,t)$ from (4.13) leads to

$$\gamma(y,t) = \sum_{S^t \in \Lambda_t^y} Pol(S^t)$$

$$= \sum_{d=1}^{D} \sum_{S^{t-d}} Pol(S^{t-d} \oplus (y, t-d+1, t)), \tag{4.18}$$

where $S^{t-d} \oplus (y, t-d+1, t)$ represents the appending of d labels y to the original sequence of S^{t-d}. In case S^{t-d} contains at least one state, it can be assumed that (y^*, b, e) is the last state of S^{t-d}, then the Pol function is calculated by

$$Pol(S^{t-d} \oplus (y, t-d+1, t)) = Pol(S^{t-d}) e^{w^{Tr}(y^*,y)+G(y,t-d+1,t)}. \tag{4.19}$$

In the other case, $S^{t-d} = \emptyset$ or its original sequence comprises of only "*u*". It is clear that

$$Pol(S^{t-d} \oplus (y, t-d+1, t)) = e^{G_u(u,1,t-d)} e^{G(y,t-d+1,t)}. \tag{4.20}$$

From (4.18), (4.19), (4.20) we conclude that

$$\gamma(y,t) = \sum_{d=1}^{D} \begin{pmatrix} \sum\limits_{y'} \alpha(y', t-d) e^{w^{Tr}(y',y)+G(y,t-d+1,t)} \\ + e^{G_u(u,1,t-d)+G(y,t-d+1,t)} \end{pmatrix}. \tag{4.21}$$

Obviously in (4.21) only $\alpha(y', t-d)$ and $w^{Tr}(y', y)$ depend on y', therefore by pre-caching

$$\lambda(y,t) = \sum_{y'} \alpha(y', t) e^{w^{Tr}(y',y)}, \tag{4.22}$$

γ can be efficiently computed by

$$\gamma(y,t) = \sum_{d=1}^{D} \begin{pmatrix} \lambda(y, t-d) e^{G(y,t-d+1,t)} \\ + e^{G_u(u,1,t-d)+G(y,t-d+1,t)} \end{pmatrix}. \tag{4.23}$$

Based on (4.14), (4.17), (4.22), and (4.23) the forward algorithm can be implemented as in below pseudo-code.

Algorithm 3: Forward algorithm for calculating Z_X

Forward

- **for** $t = 1$ *To* T **do**
  - **for** $y = 1$ *To* $StateNum$ **do**
    - $\alpha[y][t] = 0$
    - $\gamma[y][t] = 0$
    - $\lambda[y][t] = 0$
    - **for** $d = 1$ *To* D **do**
      - **if** $t - d + 1 > 0$ **then**
        - $\gamma[y][t]$ += $\lambda[y][t-d]e^{G(y,t-d+1,t)}$
        - $\gamma[y][t]$ += $e^{G_u(u,1,t-d)+G(y,t-d+1,t)}$
      - **else**
        - $Break$
    - **if** $t > 1$ **then**
      - $\alpha[y][t] = \alpha[y][t-1]e^{G_u(u,t,t)} + \gamma[y][t]$
    - **else**
      - $\alpha[y][t] = \gamma[y][t]$
    - **for** $y^{'} = 1$ *To* $StateNum$ **do**
      - $\lambda[y][t] = \lambda[y][t] + \alpha[y^{'}][t]e^{w^{Tr}(y^{'},y)}$
- $Z_X = e^{G_u(u,1,T)}$
- **for** $y = 1$ *To* $StateNum$ **do**
  - $Z_X = Z_X + \alpha(y, T)$

4.2.2 Backward Algorithm

Similar to the forward algorithm, we propose the following backward algorithm

$$\beta(y,t) = \sum_{S^{T-t+1} \in \Omega_t^y} Pol(S^{T-t+1})$$

$$= \sum_{S^{T-t+1} \in \Omega_t^y} \prod_{i=1}^{q} \Psi(s_{i-1}, s_i, X), \tag{4.24}$$

where $\Omega_t^y = \{S = s_1, s_2, ..., s_q\}$ is a set of all Semi-Markov sequences, which have an original label sequence $(y_t, y_{t+1}, ..., y_T)$ with the first expected label is y.

$$\eta(y,t) = \sum_{S^{T-t+1} \in \Upsilon_t^y} Pol(S^{T-t+1}), \tag{4.25}$$

where $\Upsilon_t^y = \{S = s_1, s_2, ..., s_q\}$ is a set of all Semi-Markov sequences, which have an original label sequence $(y_t = y, y_{t+1}, ..., y_T)$. Following similar steps in forward algorithm results in

$$\beta(y,t) = \beta(y,t+1)e^{G_u(u,t,t)} + \eta(y,t), \tag{4.26}$$

$$\eta(y,t) = \sum_{d=1}^{D} \begin{pmatrix} \sum\limits_{y'} \beta(y', t+d) e^{w^{T_r}(y,y') + G(y,t,t+d-1)} \\ + e^{G(y,t,t+d-1) + G_u(u,t+d,T)} \end{pmatrix}$$

$$= \sum_{d=1}^{D} \begin{pmatrix} \zeta(y, t+d) e^{G(y,t,t+d-1)} \\ + e^{G(y,t,t+d-1) + G_u(u,t+d,T)} \end{pmatrix}, \tag{4.27}$$

where

$$\zeta(y,t) = \sum_{y'} \beta(y', t) e^{w^{T_r}(y,y')}. \tag{4.28}$$

The following pseudo-code illustrates how the backward algorithm can be implemented.

Algorithm 4: Backward algorithm for calculating Z_X

Backward

- **for** $t = T$ *Down To* 1 **do**
  - **for** $y = 1$ *To* $StateNum$ **do**
    - $\beta[y][t] = 0$
    - $\eta[y][t] = 0$
    - $\zeta[y][t] = 0$
    - **for** $d = 1$ *To* D **do**
      - **if** $t + d - 1 \leq T$ **then**
        - $\eta[y][t]$ += $\zeta[y][t + d]e^{G(y,t,t+d-1)}$
        - $\eta[y][t]$ += $e^{G(y,t,t+d-1)+G_u(u,t+d,T)}$
      - **else**
        - $Break$
    - **if** $t < T$ **then**
      - $\beta[y][t] = \beta[y][t + 1]e^{G_u(u,t,t)} + \eta[y][t]$
    - **else**
      - $\beta[y][t] = \eta[y][t]$
    - **for** $y^{'} = 1$ *To* $StateNum$ **do**
      - $\zeta[y][t] = \zeta[y][t] + \beta[y^{'}][t]e^{w^{Tr}(y,y^{'})}$
- $Z_X = e^{G_u(u,1,T)}$
- **for** $y = 1$ *To* $StateNum$ **do**
  - $Z_X = Z_X + \beta(y, 1)$

4.2.3 Gradient Estimation

The gradients of our proposed model's target function are computed as below.

$$L(S|X) = \sum_{i=1}^{P} \begin{pmatrix} Q^{Tr}(s_{i-1}, s_i, X)+ \\ Q^{D}(s_{i-1}, s_i, X)+ \\ Q^{O}(s_{i-1}, s_i, X) \end{pmatrix} - log(Z_X). \tag{4.29}$$

Hence

$$\frac{dL}{dw^*} = \sum_{i=1}^{P} \frac{dQ^*(s_{i-1}, s_i, X)}{dw^*} - \frac{1}{Z_X}\frac{dZ_X}{dw^*}. \tag{4.30}$$

Herein Q^* and w^* are used to refer to any kind of the potential function and weight (* can be D, Tr, or O for duration, transition, or observation respectively). Computing the first term of the right side in (4.30) is trivial, Z_X is calculated by using forward or backward variables. Therefore, here the main focus of this section is to evaluate $\frac{dZ_X}{dw^*}$ for different kind of weights. From (4.3) and (4.4) it can be seen that

$$\frac{dZ_X}{dw^*} = \sum_{S^T} \left(\left(\sum_{i=1}^{P} \frac{dQ^*(s_{i-1}, s_i, X)}{dw^*} \right) \prod_{i=1}^{P} \Psi(s_{i-1}, s_i, X) \right). \tag{4.31}$$

Gradient of the transition weight

Since

$$\frac{dQ^{Tr}(s_{i-1}, s_i, X)}{dw^{Tr}(y', y)} = \delta(s_{i-1}.y = y', s_i.y = y), \tag{4.32}$$

it brings about that

$$\frac{dZ_X}{dw^{Tr}(y', y)} = \sum_{t=1}^{T} \sum_{S^{Tr} \in \Lambda_t^{y'} \oplus \Omega_{t+1}^{y}} \prod_{i=1}^{P} \psi(s_{i-1}, s_i, X), \tag{4.33}$$

where each $S^T = s_1, s_2, ..., s_P \in \Lambda_t^{y'} \oplus \Omega_{t+1}^{y}$ can be defined as the concatenation of two sub sequences $S_{prev}^{t} \in \Lambda_t^{y'}$ and $S_{post}^{T-t} \in \Omega_{t+1}^{y}$. Therefore

$$\frac{dZ_X}{dw^{Tr}(y', y)} = \sum_{t=1}^{T} \gamma(y', t)\beta(y, t+1)e^{w^{Tr}(y', y)}. \tag{4.34}$$

Gradient of the duration weight

From the definition of the duration potential function, it is obvious that

$$\frac{dQ^D(s_i, s_{i-1}, X)}{dw^D(y)} = \sum_{y,d} \delta(s_i.y = y, s_i.e - s_i.b + 1 = d) \frac{(d - m_y)^2}{-2\sigma_y^2}. \tag{4.35}$$

As a result

$$\frac{dZ_X}{dw^D(y)} = \sum_{d=1}^{D} \sum_{t=1}^{T} \frac{(d - m_y)^2}{-2\sigma_y^2} \sum_{S^T \in \chi_y^{d,t}} \prod_{i=1}^{P} \psi(s_{i-1}, s_i, X), \tag{4.36}$$

where $\chi_y^{d,t} = \{S = s_1, ..., s_P\}$ is a set of all Semi-Markov sequences whose original sequences contain d continuous labels y from time t. To utilize the advantages of the caching technique, some intermediate terms are defined below

$$\theta(y, t, d) = \sum_{S^T \in \chi_y^{d,t}} \prod_{i=1}^{P} \psi(s_{i-1}, s_i, X). \tag{4.37}$$

Obviously, each $S^T \in \chi_y^{d,t}$ can be represented as a concatenation

$$S^T = S_{prev}^{t-1} \oplus (y, t, t + d - 1) \oplus S_{post}^{T-t-d+1}, \tag{4.38}$$

where

$$S_{prev}^{t-1} \in \bigcup_{y'} \Gamma_{t-1}^{y'} \bigcup \{\phi^{t-1}\}, \tag{4.39}$$

and

$$S_{post}^{T-t-d+1} \in \bigcup_{y^*} \Omega_{t+d}^{y^*} \bigcup \{\phi^{T-t-d+1}\}. \tag{4.40}$$

Equations (4.37), (4.38), (4.39), and (4.40) imply that

$$\theta(y, t, d) = \begin{pmatrix} \lambda(y, t-1)\zeta(y, t+d)e^{G(y,t,t+d-1)} \\ +\zeta(y, t+d)e^{G_u(u,1,t-1)+G(y,t,t+d-1)} \\ +\lambda(y, t-1)e^{G(y,t,t+d-1)+G_u(u,t+d,T)} \\ +e^{G_u(u,1,t-1)+G(y,t,t+d-1)+G_u(u,t+d,T)} \end{pmatrix}. \tag{4.41}$$

Using $\theta(y,t,d)$ the gradient is calculated as

$$\frac{dZ_X}{dw^D(y)} = \sum_{d=1}^{D}\sum_{t=1}^{T}\frac{(d-m_y)^2}{-2\sigma_y^2}\theta(y,t,d). \tag{4.42}$$

Gradient of the observation weight

To estimate the gradient of the observation weight, two different cases should be considered. Firstly, the observation weights of target labels is computed. Based in equations (4.8), (4.9), and (4.10), the observation gradient is computed by

$$\frac{dQ^O(s_{i-1},s_i,X)}{dw^O(y,o)} = \sum_{k=s_i.b}^{s_i.e}\delta(s_i.y=y,x_k=o). \tag{4.43}$$

Combining (4.43) and the definition of θ in (4.41) leads to

$$\frac{dZ_X}{dw^O(y,o)} = \sum_{\substack{k,t,d \\ k\in[t,t+d-1]}} \theta(y,t,d)\delta(x_k=o). \tag{4.44}$$

Similarly, observation gradient with respect to the weight of unknown label is computed by

$$\frac{dZ_X}{dw^O(u,o)} = \sum_{t=1}^{T} \nu(t)\delta(x_t=o), \tag{4.45}$$

where

$$\nu(t) = \sum_{S^T}\prod_{i=1}^{P}\psi(s_{i-1},s_i,x), \tag{4.46}$$

sequence $S^T = s_1, s_2, ..., s_P$ does not contain any expected activities at time t. So, S^T can be decomposed as

$$S^T = S_{prev}^{t-1} \oplus \boldsymbol{u} \oplus S_{post}^{T-t}, \tag{4.47}$$

where

$$S_{prev}^{t-1} \in \bigcup_{y'}\Gamma_{t-1}^{y'}\bigcup\{\phi^{t-1}\}, \tag{4.48}$$

and

$$S_{post}^{T-t} \in \bigcup_{y} \Omega_{t+1}^{y} \bigcup \{\phi^{T-t}\}. \tag{4.49}$$

Finally, ν equals to

$$\nu(t) = \begin{pmatrix} \sum_{y'} \sum_{y} \alpha(y', t-1)\beta(y, t+1) e^{w^{Tr}(y',y)+G_u(u,t,t)} \\ +\alpha(y', t-1) e^{G_u(u,t,T)} \\ +\beta(y, t+1) e^{G_u(u,1,t)} \\ +e^{G_u(u,1,T)} \end{pmatrix}. \tag{4.50}$$

As mentioned above, L-BFG is used to find the local maximum point of the target function with L1 regularizer ($-\frac{1}{2}W^T W$) to avoid overfitting. Hence the actual target function is

$$L(S|X) = \sum_{i=1}^{P} \begin{pmatrix} Q^{Tr}(s_{i-1}, s_i, X)+ \\ Q^{D}(s_{i-1}, s_i, X)+ \\ Q^{O}(s_{i-1}, s_i, X) \end{pmatrix} - log(Z_X) - \epsilon \frac{1}{2} W^T W, \tag{4.51}$$

where ϵ is a smoothing constant, which is manually estimated.

4.2.4 Concavity of the Target Function

In the previous sections, the methods for calculating the gradients of the log-likelihood target function have been presented. These gradients are used to find the maximum point of the target function using L-BFG algorithm. A common issue of the gradient-based optimization methods is the local optimum point. However, in this section the target function is proven to be a concave function, which is guaranteed to have a unique global maximum. Therefore, the proposed system does not suffer from the local optimum problem with the gradient-based optimization method.

From equations (4.2-4.10), we can rewrite the likelihood function as below

$$P(S|X) = \frac{Q(S, X)}{Z_X}, \tag{4.52}$$

$$Z_X = \sum_{S'} Q(S', X), \tag{4.53}$$

$$Q(S, X) = e^{f_S(W)}, \tag{4.54}$$

$$f_S(W) = \sum_{i=1}^{P} \left\{ \begin{array}{c} \sum_{t=s_i.b}^{s_i.e} w^O(s_i.y, X_t) + w^D(s_i.y) \times \frac{(s_i.e - s_i.b + 1 - m_y)^2}{-2\sigma_y^2} + \\ w^{Tr}(s_i.y, s_{i+1}.y) + \sum_{t=s_i.e+1}^{s_{i+1}.b-1} w^O(u, X_t) \end{array} \right\}, \tag{4.55}$$

$$\frac{d^2Q}{dW^2} = \frac{d^2 f_S}{dW^2} e^{f_S(W)} + \left(\frac{df}{dW}\right)^2 e^{f_S(W)}. \tag{4.56}$$

Because $f_S(W)$ is a linear function, $\frac{d^2 f_S}{dW^2} = 0$, therefore we have

$$\frac{d^2Q}{dW^2} = \left(\frac{df}{dW}\right)^2 e^{f_S(W)} \geq 0, \tag{4.57}$$

hence Q is a convex function. Since Z_X is a sum of convex functions, its log function is also a convex function or

$$\frac{d^2 log(Z_X)}{dW^2} \geq 0. \tag{4.58}$$

From equation

$$L(S|X) = log(P(S|X)) - \epsilon \frac{1}{2} W^T W = log(Q(S, X)) - log(Z_X) - \epsilon \frac{1}{2} W^T W, \tag{4.59}$$

it can be seen that

$$\frac{d^2 L}{dW^2} = \frac{d^2 f_S}{dW^2} - \frac{d^2 log(Z_X)}{dW^2} - \epsilon = -\frac{d^2 log(Z_X)}{dW^2} - \epsilon. \tag{4.60}$$

From equations (4.58) and (4.60), it is obvious that $\frac{d^2 L}{dW^2} \leq 0$. In another word, $L(S|X)$ is a concave function.

4.2.5 Inference Using Viterbi Algorithm

Inference in the proposed Semi-CRF is done by adapting Viterbi algorithm with a complexity of $O(TM^2D)$. The goal of the inference phase is to find the best matched sequence Y given an input X so that $P(Y|X)$ is maximized. Let denote

$$V(y, t) = \max \log \left(P(S^t = s_1, s_2, ..., s_q | x_1, x_2, ..., x_t) \right), \tag{4.61}$$

$S^t = s_1, s_2, ..., s_q$ has the last expected label is y, or equivalently $s_q.e \leq t$ and $s_q.y = y$.

$$V(y,t) = \max_{d} \begin{cases} A = V(y,t-1) + G_u(u,t,t) \\ B = G_u(u,1,t-d) + G(y,t-d+1,t) \\ V(y',t-d) + w^{Tr}(y',y) + G(y,t-d+1,t) \end{cases} . \tag{4.62}$$

For backtracking, $\Delta^{State}(y,t)$ and $\Delta^{Duration}(y,t)$ are used to store the previous trace of $V(y,t)$ as following

$$\Delta^{State}(y,t) = \begin{cases} y \text{ if } V(y,t)\text{=A} \\ u \text{ if } V(y,t)\text{=B} \\ y' \text{ otherwise} \end{cases} , \tag{4.63}$$

$$\Delta^{Duration}(y,t) = \begin{cases} 0 \text{ if } V(y,t)\text{=A} \\ d \text{ if } V(y,t)\text{=B} \\ d \text{ otherwise} \end{cases} . \tag{4.64}$$

Using $V(y,t)$, $\Delta^{State}(y,t)$ and $\Delta^{Duration}(y,t)$ it is easy to track the optimized path by following the below steps.

4.3 Validation of The Proposed Algorithms

In this section, to verify the performance of the proposed algorithms, a well-known dataset of long-term activities, which has been published at http://www.mis.infor matik.tu-darmstadt.de/data, is used. The dataset contains 7 days of continuous data from one user (except the sleeping time), measured by two fixed-position triaxial accelerometers (ADXL330), one on the wrist and the other in the right pocket. The sensor's sampling frequency was set to 100Hz. The sensor device has on-board memory for storing data temporarily.

To annotate the activity data, the subject was periodically notified by an application on his mobile phone, which presented a set of multiple-choice questions about his current activities.

Algorithm 5: Viterbi algorithm for tracking the best matched sequence

Step 1. Initialization

$y^* = \text{argmax } V(y, T)$

$t = T$

$y = y^*$

$i = 1$

Step 2. Backtracking

while $y \neq$ u *and* $t > 0$ **do**

  while $\Delta^{Duration}(y, t) = 0$ **do**

    $t = t - 1$

  $s_i.y = y$

  $s_i.e = t$

  $s_i.b = t - \Delta^{Duration}(y, t) + 1$

  $i = i + 1$

  $t = t - \Delta^{Duration}(y, t)$

  $y = \Delta^{State}(y, t)$

$P = i - 1$

Step 3. Finalization

$y_1, y_2, ..., y_T = OriginalSequence(s_P, s_{P-1}, ..., s_1)$

Table 4.1: Low level activities which are annotated [41]

Activity	Average duration	Occurrences	Total
sitting / desk activities	49.41 min	54	3016.0 min
unlabeled	1.35 min	239	931.3 min
having dinner (eating)	17.62 min	6	125.3 min
walking freely	2.86 min	38	124.2 min
driving car	10.37 min	10	120.3 min
having lunch (eating)	10.95 min	7	75.1 min
discussing at white board	12.80 min	5	62.7 min
attending a presentation	48.9 min	1	48.9 min
driving a bike	11.82 min	4	46.3 min
walking while carrying something	1.43 min	10	23.1 min
walking	2.71 min	7	23.0 min
picking up mensa food	3.30 min	7	22.6 min
sitting / having a coffee	5.56 min	4	21.8 min
queuing in a line	2.89 min	7	19.8 min
using the toilet	1.95 min	2	16.7 min
washing dishes	3.37 min	3	12.8 min
standing / having a coffee	6.7 min	1	6.7 min
preparing food	4.6 min	1	4.6 min
washing hands	0.32 min	3	2.2 min
running	1.0 min	1	1.0 min
wiping the whiteboard	0.8 min	1	0.8 min

Table 4.2: The occurrence of low level activities in activity routines of the dataset [41]

	commuting	office work	lunch	dinner
sitting / desk activities		x	x	x
unlabeled	x	x	x	x
eating			x	x
walking freely	x	x	x	x
driving car	x			
discussing at white board		x		
attending a presentation		x		
driving a bike	x			
walking carrying something		x	x	
walking		x		
picking up mensa food			x	
sitting / having a coffee			x	
queuing in a line			x	
using the toilet		x		
washing dishes				x
standing / having a coffee		x	x	
preparing food				x
washing hands				x
running	x			
wiping the whiteboard		x		

Table 4.3: Daily routines which are annotated

Routine	Occurrences
dinner	7
commuting	14
lunch	7
office work	14
unknown(null)	> 50

Besides, the subject also entered labels, starting times, ending times of activities into a diary. In total, 34 labeled activities were monitored, of which a subset of 24 activities occurred during the activity routines. Table 4.1 lists all the annotated activities, which were grouped [41] into 5 daily routines as seen in Tables 4.2 and 4.3.

Individual Validation of The Proposed Semi-CRF Model

To validate the Semi-CRF algorithm individually, the feature selection module is not used, we compare our achievement with that of [11] and [41] using the same feature set. Those features are time domain features and are clearly described in section 3.1. Figure 4.6 and 4.7 demonstrate the class distributions on different feature planes.

Because there are few parameters which are decided manually: the length of a sliding window, the overlapping between two consecutive sliding windows, and the number of code-book vectors. Therefore, experiments are conducted to analyze the effect of different parameter's values on the achieved results. To quantitatively evaluate and compare results, we compute individual precision and recall rates from the recognized routines using equations (4.65) and (4.66).

Precision:

$$PRE_r = \frac{TP_r}{TP_r + FP_r}, \tag{4.65}$$

Recall:

$$REC_r = \frac{TP_r}{TP_r + FN_r}, \tag{4.66}$$

where r is a routine (can be lunch, commuting, office, or dinner), TP_r (true positive) is the number of r frames which are correctly recognized as r, FP_r (false positive) is the number frames which are misrecognized (or unexpectedly recognized) as r, and FN_r is the number of r frames which are not recognized as r. Based on the precision and recall rates of individual routines, the average precision, recall and f-score values are calculated using equations (4.67), (4.68), and (4.69).

$$PRE_{avg} = \frac{PRE_{lunch} + PRE_{commuting} + PRE_{office} + PRE_{dinner}}{4}, \tag{4.67}$$

$$REC_{avg} = \frac{REC_{lunch} + REC_{commuting} + REC_{office} + REC_{dinner}}{4}, \tag{4.68}$$

$$F - SCORE_{avg} = 2\frac{PRE_{avg}REC_{avg}}{PRE_{avg} + REC_{avg}}. \tag{4.69}$$

Table 4.4: Average precision, recall and f-score with different parameters' values

	Overlapping (%)			Frame Length			Codebook Size		
	25	50	75	256	512	1024	32	64	128
Precision	82.07	88.47	82.32	82.36	88.47	75.61	80.96	88.47	78.14
Recall	83.46	86.68	83.93	82.65	86.68	75.14	82.65	86.68	77.65
F-Score	82.26	87.57	83.11	82.50	87.57	75.37	81.80	87.57	77.89

Figure 4.8a, 4.8b, and 4.8c show the results obtained with different values of these parameters. Table 4.4 contains the average precision, recall and f-score. Based on the experiment results, with an assumption that the performance of the system with other parameter values can be linearly interpolated, we select the parameter set producing the highest f-score: the overlap portion is 50%, the window's length is 512 samples, and the number of codebook vectors is 64.

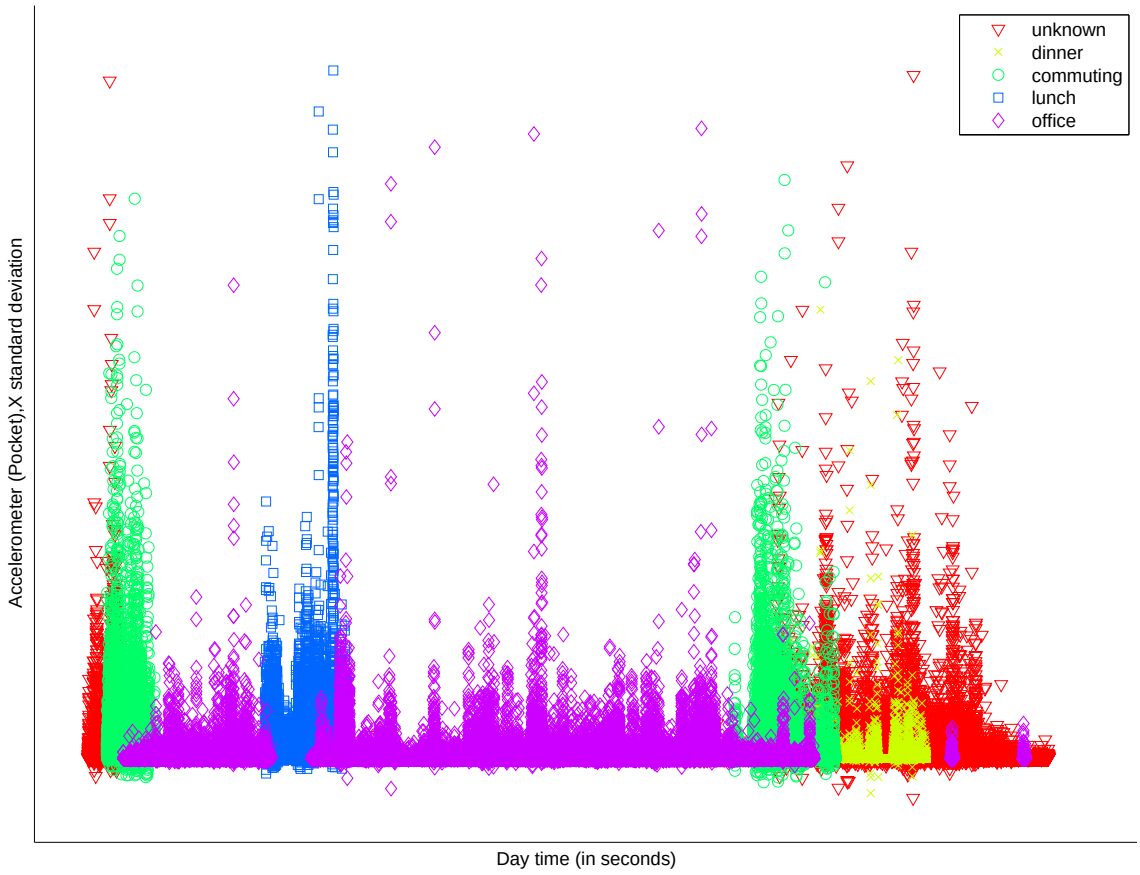


Figure 4.6: Distribution of all the classes

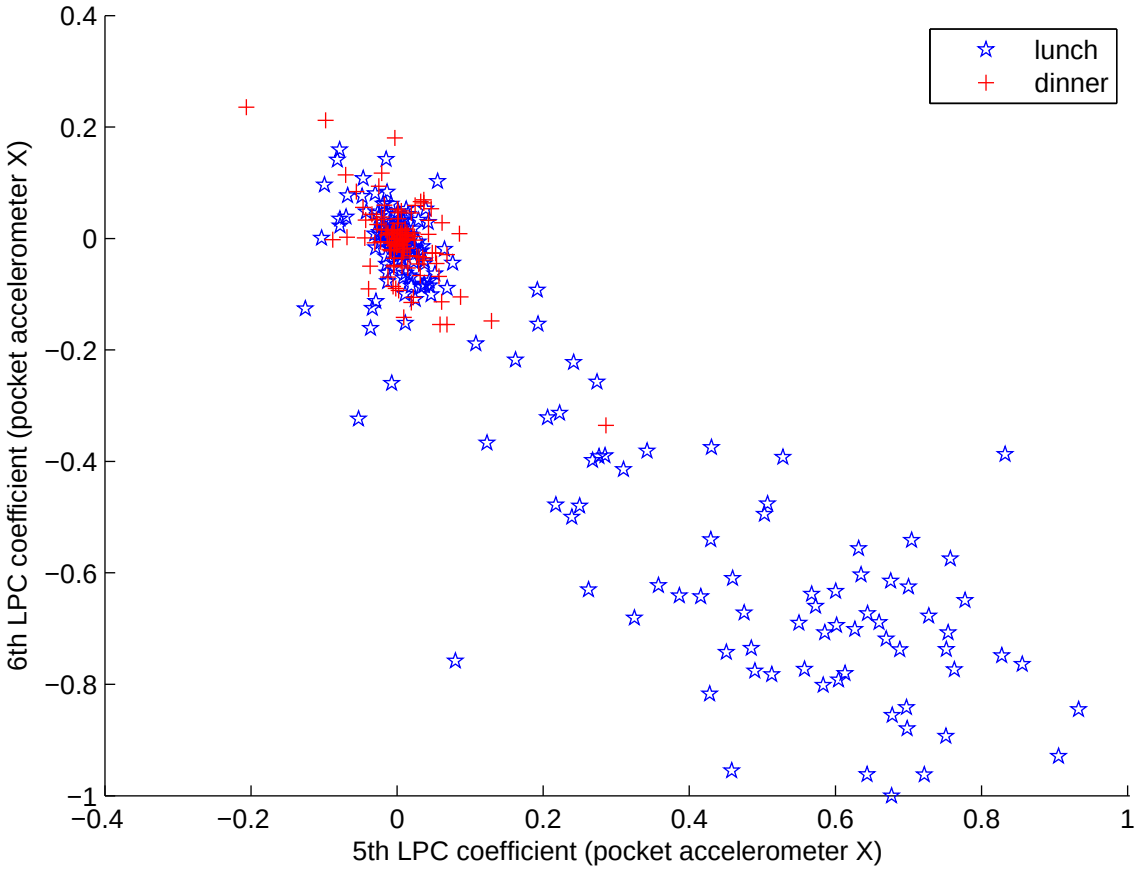
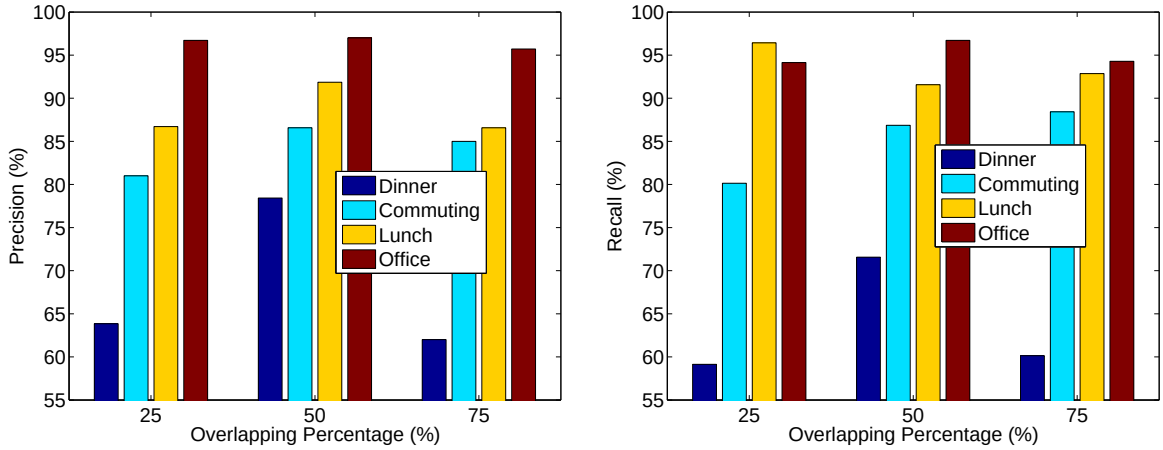
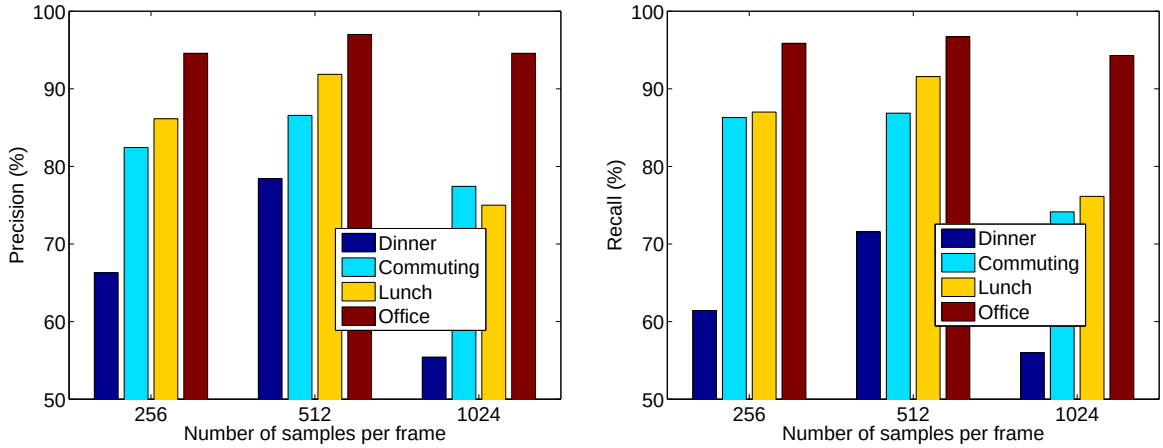


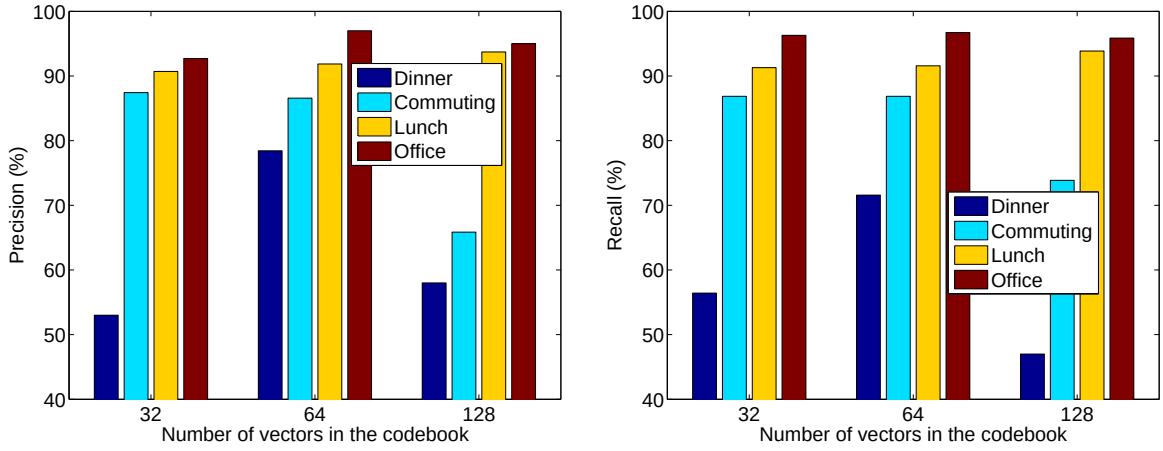
Figure 4.7: Distribution of lunch and dinner classes



a. Achieved precision and recall with different values of the overlapping percentage



b. Achieved precision and recall with different values of the frame's length



c. Achieved precision and recall with different values of the codebook's size

Figure 4.8: Achieved precision and recall with different parameter's values

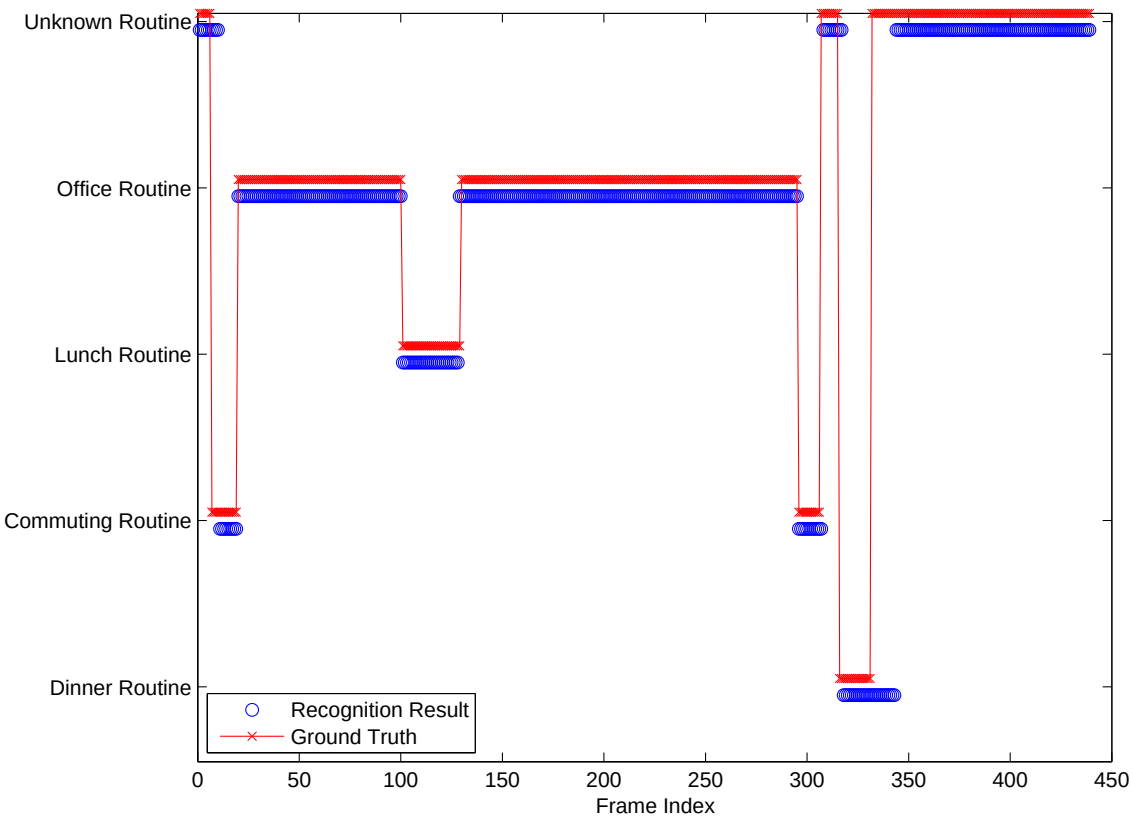


Figure 4.9: A single day recognized routines

Leave-one-out cross validation rule is utilized to measure the results of recognition as can be seen in Table 4.5. Table 4.6 shows the corresponding confusion matrix. Figure 4.9 demonstrates a typical example of recognized routines together with the ground truth.

The experimental results show a considerable improvement of the proposed method compared to other methods in [11] and [41], except dinner routine which has lower precision that of HMM [41] and lower recall than that of Boosting [11]. Since the same feature set is used in all the experiments, the improvement can be explained as the result of taking the relationship together with the duration of activities into account.

Table 4.5: Recognition results of the existing systems and the proposed Semi-CRF model without the feature selection

	Baseline (HMM)		Huynh et al.		Ulf Blanke et al.		Our method	
Routines	Pre(%)	Rec(%)	Pre(%)	Rec(%)	Pre(%)	Rec(%)	Pre(%)	Rec(%)
Dinner	88.60	27.30	56.90	40.20	85.27	90.48	78.43	71.57
Commuting	72.60	31.50	83.50	71.10	81.77	82.36	86.57	86.86
Lunch	84.40	80.70	73.80	70.20	84.56	90.04	91.86	91.57
Office	89.20	91.10	93.40	81.80	98.12	93.63	97.00	96.71

Table 4.6: The classification confusion matrix (%) of the proposed Semi-CRF model

	Lunch	Commuting	Dinner	Office	Unknown
Lunch	91.57	0	0	0	8.43
Commuting	0	86.86	0	2.20	10.94
Dinner	0	4.65	71.57	0.21	23.57
Office	3.29	0	0.35	96.71	0.65
Unknown	5.82	8.82	19.33	0.58	65.45

Table 4.7: Sequence of activities, which occur in the dinner routine. As can be seen, a dinner routine often contains some related activities such as having dinner, sitting, and washing dishes. However, in day 2 the dinner routine is interrupted by walking and carrying something

Day	Sequence of activities	Precision (%)	Recall (%)
1	having dinner, sitting	100	100
2	having dinner, walking, sitting, walking, sitting, carrying something, walking, washing dishes	0	0
3	having dinner, sitting, washing dishes	60	100
4	having dinner	100	76
5	having dinner, washing dishes	95	95
6	having dinner, sitting	100	45
7	having dinner, sitting, washing dishes	94	85

Table 4.8: Transition weights after training

	Dinner	Commuting	Lunch	Office
Dinner	-9.67	-10.99	-9.34	-8.99
Commuting	-2.50	-8.38	-6.80	-4.20
Lunch	-7.08	-9.39	-8.23	-6.12
Office	-9.60	-9.38	-7.51	-8.38

Table 4.9: Features selected by different methods

Feature	FINMIFS	mRMR [75]	NMIFS [23]
Time (in seconds)	x	x	x
Correlation coeffcients. Acc. 1's X-Z axes			x
Spectral energy. Acc. 1's X axis	x	x	x
Spectral energy. Acc. 1's Y axis	x	x	x
Band 4 energy. Acc. 1's Y axis	x		
Band 2 energy. Acc. 1's Z axis	x	x	x
Band 4 energy. Acc. 1's Z axis			x
Linear predictive coefficients 6. Acc. 1's X axis	x	x	
Spectral energy. Acc. 1's Z axis	x	x	x
Linear predictive coefficients 2. Acc. 1's Y axis			x
Linear predictive coefficients 5. Acc. 1's Y axis		x	
Linear predictive coefficients 6. Acc. 1's Y axis	x	x	x
Linear predictive coefficients 7. Acc. 1's Y axis	x	x	
Linear prediction error. Acc. 1's Z axis	x	x	x
Correlation coefficients. Acc. 2's X-Y axes			x
Correlation coefficients. Acc. 2's Y-Z axes			x
Correlation coefficients. Acc. 2's X-Z axes	x	x	x
Band 2 energy. Acc. 2's X axis	x	x	
Spectral energy. Acc. 2's Y axis			x
Spectral energy. Acc. 2's Z axis	x	x	x
Band 2 energy. Acc. 2's Z axis		x	
Linear predictive coefficients 3. Acc. 2's X axis	x	x	
Linear predictive coefficients 5. Acc. 2's X axis	x	x	
Linear predictive coefficients 6. Acc. 2's X axis	x	x	x
Linear predictive coefficients 4. Acc. 2's Y axis	x		x
Linear predictive coefficients 6. Acc. 2's Y axis	x	x	x
Linear predictive coefficients 3. Acc. 2's Z axis			x
Linear prediction error. Acc. 2's Z axis	x	x	x

Nevertheless, a limitation of the proposed method is that the precision and recall are affected by the fragmentation of the routine. As seen in Table 4.7, the worst result, decreasing the overall achievement, is achieved with dinner routine of day 2, which is interrupted by other activities such as walking, and carrying something. Meanwhile, the proposed system still obtains good results (on average higher than 90.00%) with other days, which are not fragmented. This poor result is mainly caused by the assumption of the duration distribution as shown in Figure 4.5 and the use of activity transitions described in equation (4.5). In a highly fragmented data, the activity duration can be strangely long; thus it does not fall into the desired area of the duration distribution making the duration potential significantly low for the truth labels. Besides the transitions among fragmented activities are not modeled. These limitations prevent the model from correctly recognizing the fragmented activity routines.

Table 4.8 shows an example of transition matrix after training. It can be seen that the potential of most likely transitions such as ”office work - lunch”, ”commuting - dinner” or ”lunch - office” is always the highest to encourage the prediction of these events.

Validation of The Proposed Integrated System

Table 4.10: Recognition results of the proposed system without and with the feature selection module

	Without FS		Our method		mRMR [75]		NMIFS [23]	
Routines	Pre(%)	Rec(%)	Pre(%)	Rec(%)	Pre(%)	Rec(%)	Pre(%)	Rec(%)
Dinner	78.43	71.57	81.20	74.33	84.73	78.72	88.55	82.27
Commuting	86.57	86.86	88.95	87.57	85.14	86.63	82.58	85.47
Lunch	91.86	91.57	94.80	92.20	92.12	94.87	91.13	96.58
Office	97.00	96.71	96.85	96.57	96.23	93.22	94.53	90.57
Average	88.47	86.68	90.45	87.67	89.56	88.36	89.20	88.73

So far, the proposed algorithms including the feature selection algorithm and the Semi-CRF model are already validated individually. Therefore, in this section the whole integrated system depicted in Figure 1.3 is validated using the same dataset as the above section. However, instead of using the predefined time domain features, a number of different features are extracted as described in section 3.1, then the feature selection module is executed to select the same number of features as in the previous experiments. Three different feature selection methods are evaluated including our feature-independent and normalized mutual information based method (FINMIFS), maximum relevancy and minimum redundancy method (mRMR) [75], and normalized mutual information based method (NMIFS) [23]. Table 4.9 shows features that are selected by different methods. From results depicted in Table 4.10, it obviously can be seen that the use of feature selection module increases the system's precision and recall rates. Although in this experiment, different feature selection methods produce quite similar results, we still prefer our selection algorithm because of its stable performance for different datasets as pointed out in chapter 3.

4.4 Computational Complexity Analysis

In addition to the accuracy, the proposed Semi-CRF model also has a practical training time. With 72 hours of training data, the algorithm takes about 2 hours to complete parameter estimation in a system with Intel dual core 1.83GHz processor and 512 MB RAM. More detail about the computational complexity is presented below.

In [85], the estimation of equation (2.28) requires that $\alpha(t, y)$ and $\eta^k(t, y)$ are pre-calculated for all possible values of t and y. Each of them needs a complexity of $O(MD)$ as can be seen in (2.24) and (2.27). Therefore, the complexity per gradient computation is proportional to $O(TM^2D)$. Hence, the estimation of gradients for all N model's parameters takes $O(NTM^2D)$.

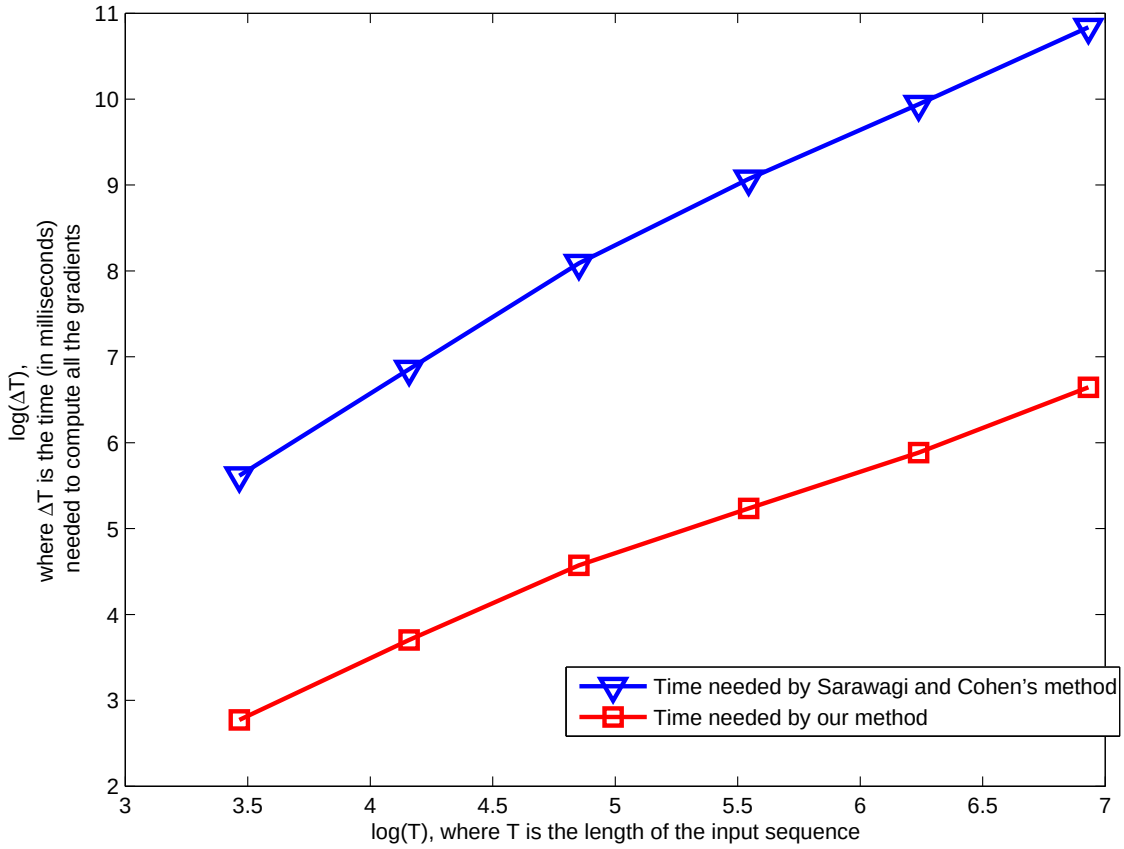


Figure 4.10: Average time needed for computing all the gradients. Herein, the number of labels (M) is 4, the maximum duration (D) is 16, the codebook's size (V) is 128, the length of the input sequence (T) changes from 32 to 1024

In this work, gradients are computed by using equations (4.34), (4.42), (4.44), and (4.45). It is obvious that if α, γ, λ, β, θ, ζ, and ν are cached, then estimating the above equations requires a maximum complexity of $O(TD)$. Hence, for optimizing N parameters, the proposed Semi-CRF model requires only $O(NTD)$ to completely calculate all gradients. Nevertheless, the extra time of estimating the cached variables should be taken into account. As shown in the pseudo-code for the forward and backward algorithm, α, γ, λ, β, η, and ζ can be computed with $O(2TM(M+D))$.

Meanwhile, equations (4.41) and (4.50) show that θ and ν take $O(TMD)$ and $O(TM^2)$, respectively. Totally caching these variables requires a complexity of $O(3\,TM(M+D))$.

It can be seen that the improvement completely comes from the caching mechanism. In the gradient equations (4.34), (4.42), (4.44), and (4.45), different partitioning methods, which take into account the characteristics of the gradients, are utilized. For example, in (4.34) the set of all length-T label sequences (S^T) is partitioned into subsets ($\bigcup_t \Lambda_t^{y'} \oplus \Omega_{t+1}^{y}$) based on the occurrence time of a transition from y' to y. Meanwhile in [85] a fixed partitioning method, where S^T was always divided into subsets based on the length of the last segment, was used regardless of different characteristics of the gradients. This is the reason why the proposed calculation method achieves much more efficient caching perfomance.

Herein, a numerical example is given to compare $O(NTM^2D)$, which is the estimated complexity in [85], to $O(3TM(M+D))+O(NTD)$, the proposed algorithm's complexity. Suppose that there are $N = 1000$ gradients of an input sequence, which has the length $T = 1000$, the maximum duration $D = 100$, the number of labels $M = 8$, then the former complexity is about 64×10^9, the latter one is about 10^9. In addition, Figure 4.10 illustrates another comparison of the two complexities with N=532, M=4, D=16 and T changes from 32 to 1024. The amount of time which was required by the Sarawagi and Cohen's method in [85] is marked in blue triangles, time consumed by the proposed algorithm is marked in red squares. Obviously, the proposed method achieves a remarkable improvement in the computational complexity.

The above analysis points out that the proposed Semi-CRF model not only improve the recognition result but also significantly reduce the calculation time in the training phase. That decrease is really meaningful especially for the long-term activity recognition system, in which the computational cost is directly proportional to the length of the input sequence.

Chapter 5

Conclusion and Future Work

Human activity recognition (AR) research area has a big potential of being applied in different applications such as health care services, Human-Computer interaction, industrial manufacture. Therefore many researchers have been investigating their time and effort in proposing practical solutions for activity recognition. So far the two of the most challenges in the area of activity recognition research area are how to obtain good features from the input sensory data and how to construct a good classification model for the features.

In this dissertation, the first challenge is addressed by proposing a mutual information based selection method to select good features extracted from several well-known existing feature extraction methods. The advantage of this approach is the ability to combine the strengths of different extraction techniques. However, the selection process can be biased due to the imbalance between the feature's classification power and the feature's redundancy. To avoid that biased selection, we normalize both terms using our proposed feature-independent upper bound of the mutual information function.

Regarding the classification model, since the existing works process the input frames independently they are not able to utilize the characteristics of long data sequences such as the relationship between two consecutive frames and the frame duration. In this work, we overcome that limitation by proposing a novel semi Markov conditional random fields to model the transition as well as the duration. Although taking the sequential data characteristics into account helps increasing the recognition accuracy, it however gives an extra computation burden to the system. Therefore we utilize a smart caching mechanism based on our own factorization of the model's target function. The proposed classification method has been proved to be efficient regarding both accuracy and computational complexity aspects.

With the above proposed solutions, this dissertation technically contributes novel algorithms for solving the two important problems in activity recognition:

- Feature selection algorithm: the proposed method follows filtering approach to take the advantage of low computational cost. Therefore it is suitable to be used in large scale application. More importantly, the proposed method overcome the limitations of the previously proposed algorithms. Hence it produces much better recognition accuracy in comparison with the existing one.

- Sequential modeling algorithm: the semi-Markov conditional random fields model proposed in this dissertation makes use the long-term characteristics of Human activities to eliminate the fragmentation in the classification result. All the necessary algorithms for training and inferring with the model are presented in chapter 4 including:

  - Forward algorithm
  - Backward algorithm
  - Gradient computation algorithms
  - Viterbi algorithm

- Furthermore, a novel smart-caching gradient computation method is introduced. Both theoretical and experimental analyses prove that the proposed method remarkably reduces the computational complexity in the training phase of the classification model.

Besides the achievement, we also pointed out the limitations of our solutions, which require further research effort to be solved completely. Firstly, the feature selection algorithm works based on the assumption that each input frame is long enough to be labeled as one activity instance. Therefore, the frame length of around 3 seconds is reasonable. For a system which uses low-sampling-rate sensors such as accelerometer or gyroscope such window length is practical. However with high-sampling rate sensors like microphone (at least 8000 sample-per-second), the window length is often less than 200 milliseconds. In such a case, the feature selection may not be applicable. Secondly, in the proposed semi-Markov conditional random fields model there is an assumption that the input sequence contains only discrete symbols. That is the reason why the proposed system needs a quantization module to discretize the sensory data. However, it is obvious that the quantization step clears some detail information of the input and may result in bad accuracy. Thirdly, although the proposed Gaussian-like duration distribution is more reasonable than the geometric one used in the original conditional random fields model, it may not always true in realistic applications. Hence, hardly utilization of the Gaussian-like distribution may result in a bad accuracy if one activity is frequently corrupted by other activities as pointed out in section 4.3.

The above analysis motivates our future work to solve those problems, namely:

- Feature selection for sequential data: a possible solution for selecting good features directly from the sequential data is to embed the selection process into the training phase of the classification model. Nevertheless, this approach requires the feature vectors to be directly handled by the classification model; thus it should be resolved together with the problem of modeling continuous input feature vectors.

- Direct modeling of continuous input sequences for the semi-Markov conditional random fields. Gaussian mixture model is a well-known model for handling such kind of problem. However integrating a mixture of Gaussian density functions into the classification model breaks the current factorization mechanism and results in a high computational cost.

- Utilize Gaussian mixture duration distributions for the semi-Markov conditional random fields. Nevertheless, using the Gaussian mixture imposes difficulties in computing the target function at a practical computational cost. As can be seen a good factorization of the target function (to improve the calculation speed by using caching mechanism) can potentially solve the last two problems. Therefore, it is one of our most focused future work related to this dissertation.

In conclusion, in this dissertation we have proposed our solutions for the long-term activity recognition problem. We address and overcome the limitations of the existing work to improve the performance of the proposed system regarding the recognition accuracy as well as computational cost. However, there are still some limitations which require much more effort to solve. Those challenges together with the bright future of activity recognition motivate us to continue our work in this research area.

Publications

Journal Publications

1. **La The Vinh**, Le Xuan Hung, Ngo Quoc Hung, Hyoung Il Kim, Manhyung Han, Young-Koo Lee and Sungyoung Lee, Semi Markov Conditional Random Fields for Accelerometer Based Activity Recognition, Applied Intelligence, ISSN: 0924-669X, 2010.

2. Jehad Sarkar, **La The Vinh**, Young-Koo Lee and Sungyoung Lee, "GPARS: a General-Purpose Activity Recognition System", Applied Intelligence, Springer(SCI), ISSN: 0924-669X, 2010.

3. **La The Vinh**, Sungyoung Lee, Young-Tack Park and Brian J. d'Auriol, A Novel Feature Selection Method Based on Normalized Mutual Information, Applied Intelligence, ISSN: 0924-669X, 2011.

4. Asad Masood Khattak, Phan Tran Ho Truc, Le Xuan Hung, **La The Vinh**, Viet-hung Dang, Donghai Guan, Zeeshan Pervez, Manhyung Han, Sungyoung Lee, Young-koo Lee, Towards Smart Homes Using Low Level Sensory Data, Journal of Sensors, ISSN: 1424-8220, 2011.

Conference Publications

1. Le Xuan Hung, Sungyoung Lee, Phan Tran Ho Truc, **La The Vinh**, Asad Khattak, Manhyung Han, Viet-Hung Dang, Mohammad M. Hassan, Miso (Hyung-Il) Kim, Kyo-Ho Koo, Young-Koo Lee and Eui-Nam Huh, Secured WSN-integrated Cloud Computing for u-Life Care, 7th Annual IEEE CCNC 2010 Conference http://www.ieee-ccnc.org/2010/, Las Vegas, USA, January 9-12, 2010.

2. Asad Masood Khattak, **La The Vinh**, Viet-Hung Dang, Phan Tran Ho Truc, Le Xuan Hung, D. Guan, Zeeshan Pervez, Manhyung Han, Sungyoung Lee and Young-Koo Lee, Context-aware Human Activity Recognition and Decision Making, 12th International Conference on e-Health Networking, Application Services (IEEE HealthCom 2010), Lyon, France, July 1-3, 2010.

3. **La The Vinh**, Nguyen Duc Thang, and Young-Koo Lee, An Improved Maximum Relevance and Minimum Redundancy Feature Selection Algorithm Based on Normalized Mutual Information, International Workshop on Computing Technologies and Business Strategies for u-Healthcare(CTBuH 2010), Seoul, Korea, July 19-23, 2010.

4. **La The Vinh**, Sungyoung Lee and Young-Koo Lee, A Fast Implementation of the Semi-Markov Conditional Random Fields, International Conference on Signal Processing, Image Processing and Pattern Recognition (SIP 2011), Jeju Island, Korea, Dec 08, 2011.

5. **La The Vinh**, Sungyoung Lee and Young-Koo Lee, Emotional Speech Classification using Hidden Conditional Random Fields, the 2nd International Symposium on Information and Communication Technology (SoICT) , Ha Noi, Vietnam, Oct 15, 2011.

International Patents

Sungyoung Lee, Young-Koo Lee, **La The Vinh**, Le Xuan Hung, Ngo Quoc Hung, Hyoung Il Kim and Manhyung Han, Method of Recognizing Activity on Basis of Semi-Markov Conditional Random Field Model, Patent Application No. 12/886,800, September 21, 2010.

Appendix A: Compare the proposed feature selection criteria (f^1) with those used in MIFS [9], MIFSU [49], Gain Ratio(GR) [35], and SBMLR [16]

k-Nearest-Neighbor (kNN, k=3)									
Arrhythmia					Hill Valley				
# Fea	f^1	$MIFS/t$	$MIFSU/t$	GR/t	# Fea	f^1	$MIFS/t$	$MIFSU/t$	GR/t
5	50.43	52.01/0.72	47.46/0.41	50.71/0.12	5	51.98	51.73/0.30	52.31/0.21	52.39/0.26
10	**60.98**	**51.58/5.11**	**44.73/2.54**	**55.64/2.76**	10	**52.72**	53.05/0.43	52.63/0.10	**50.33/2.35**
15	**63.36**	**47.06/4.21**	**50.06/3.37**	**53.72/4.37**	15	52.47	52.56/0.08	53.79/1.42	52.63/0.15
20	**62.32**	**41.78/3.18**	**45.21/2.50**	**54.90/5.00**	20	53.21	52.97/0.23	53.30/0.11	52.72/0.43
25	**64.05**	**30.55/3.77**	**49.28/3.26**	54.39/1.64	25	53.29	52.88/0.54	52.80/0.98	52.30/1.12
30	**65.47**	**54.28/8.49**	**41.00/3.49**	**59.12/2.51**	30	52.88	52.30/0.59	53.30/0.65	52.31/0.32
Image Segmentation					Ionosphere				
# Fea	f^1	$MIFS/t$	$MIFSU/t$	GR/t	# Fea	f^1	$MIFS/t$	$MIFSU/t$	GR/t
3	**92.64**	**62.34/21.99**	**88.61/2.74**	**88.14/2.86**	5	87.43	87.44/0.01	87.15/0.22	84.26/1.78
6	**96.10**	**84.76/13.01**	94.33/1.60	**88.79/11.67**	10	85.42	85.43/0.01	85.98/0.98	87.13/1.76
9	**95.97**	**92.55/3.86**	**92.55/3.86**	93.81/2.07	15	82.17	84.56/0.69	84.86/0.84	85.69/1.11
12	**95.76**	**93.51/4.62**	**93.55/3.39**	95.58/0.94	20	79.62	83.12/1.09	83.70/1.37	83.44/1.15
15	**94.33**	95.02/1.40	**93.64/2.84**	94.72/1.40	25	80.86	82.27/1.04	81.99/0.71	82.85/1.76
18	**95.06**	95.06/0.00	95.06/0.00	95.06/0.00	30	81.14	81.97/0.69	81.97/0.79	81.40/0.28
Isolet					Libras Movement				
# Fea	f^1	$MIFS/t$	$MIFSU/t$	GR/t	# Fea	f^1	$MIFS/t$	$MIFSU/t$	GR/t
5	**49.16**	**36.89/13.61**	49.15/0.03	**12.08/73.07**	5	**62.97**	58.27/2.01	**58.10/2.84**	**46.06/4.84**
10	**64.67**	**43.81/14.35**	**52.20/15.81**	**19.99/44.03**	10	**69.43**	69.73/0.12	71.75/1.20	**50.79/4.98**
15	**70.72**	**45.53/38.02**	**57.52/21.87**	**28.56/37.62**	15	**69.75**	**73.52/2.31**	73.46/1.94	**53.13/6.89**
20	**72.72**	**47.98/60.18**	**57.88/26.20**	**32.99/43.25**	20	**70.91**	73.01/1.08	74.04/2.00	**56.20/6.83**
25	**74.87**	**49.71/38.95**	**57.33/23.66**	**34.73/25.93**	25	**72.84**	74.70/1.09	76.65/2.15	**66.15/2.31**
30	**75.70**	**51.44/51.21**	**57.80/27.50**	**40.08/16.89**	30	**73.43**	76.09/1.73	**77.21/2.34**	**68.68/3.04**

k-Nearest-Neighbor (kNN, k=3)									
Arrhythmia					Hill Valley				
# Fea	f^1	$MIFS/t$	$MIFSU/t$	GR/t	# Fea	f^1	$MIFS/t$	$MIFSU/t$	GR/t
5	**61.00**	**53.35/10.69**	**72.12/4.47**	59.19/1.21	5	**89.25**	87.75/1.13	87.10/2.14	**77.85/4.15**
10	**58.54**	**52.00/4.94**	**62.19/2.78**	**73.65/9.52**	10	**93.95**	**89.65/5.02**	92.05/1.91	**89.10/5.99**
15	**58.42**	**51.15/8.28**	57.81/0.55	**80.42/12.86**	15	**97.25**	**92.40/7.83**	**94.40/5.90**	**93.40/7.61**
20	**56.69**	**51.96/3.16**	56.81/0.14	**76.12/20.03**	20	**97.85**	**94.45/5.35**	**95.20/3.57**	**93.65/8.78**
25	**56.88**	**51.50/4.38**	55.12/2.25	**71.46/13.26**	25	**98.15**	**95.25/6.50**	**95.85/4.64**	**95.30/5.57**
30	**56.46**	**51.00/4.85**	**53.42/2.79**	**67.58/7.83**	30	**98.35**	**95.65/4.10**	**95.70/3.40**	**95.60/6.40**
Landsat Satellite					Sonar				
# Fea	f^1	$MIFS/t$	$MIFSU/t$	GR/t	# Fea	f^1	$MIFS/t$	$MIFSU/t$	GR/t
5	86.31	86.11/0.48	86.22/0.22	82.45/1.70	5	**73.88**	74.37/0.23	72.97/0.41	**70.04**/3.22
10	**88.95**	89.06/0.33	**89.76/3.12**	**87.93/3.58**	10	**80.78**	**72.15/2.30**	**67.68/3.43**	74.49/1.84
15	90.33	90.12/0.57	90.40/0.14	89.39/2.06	15	**80.23**	**71.14/2.54**	79.23/0.29	76.90/0.95
20	90.38	90.47/0.24	90.74/1.06	90.19/0.71	20	**85.11**	**74.97/5.14**	**75.33/3.21**	81.18/1.69
25	90.99	90.63/0.94	90.69/1.10	90.82/0.87	25	**85.57**	**73.02/4.03**	**75.87/3.94**	85.09/0.16
30	90.77	90.88/0.31	90.97/0.65	90.72/0.40	30	**88.47**	**77.90/4.27**	**76.92/3.96**	84.57/1.44
Spambase					Breast Cancer				
# Fea	f^1	$MIFS/t$	$MIFSU/t$	GR/t	# Fea	f^1	$MIFS/t$	$MIFSU/t$	GR/t
5	**86.74**	**73.59/3.10**	83.81/0.94	86.20/0.54	5	**93.83**	**90.85/2.69**	92.09/1.32	94.38/0.45
10	**89.20**	**78.13/3.82**	88.46/0.62	88.44/0.63	10	94.38	91.36/1.60	92.61/1.41	94.56/0.58
15	**89.66**	**78.37/3.90**	88.65/1.00	89.33/0.37	15	**93.31**	92.96/0.43	**95.07/2.72**	94.02/1.50
20	**89.57**	**81.53/3.68**	89.07/0.38	88.57/0.68	20	95.77	95.08/1.49	95.07/1.06	95.42/0.48
25	**89.55**	86.15/2.20	88.68/0.68	89.18/0.29	25	**97.00**	96.48/0.66	95.61/2.05	**95.76/3.29**
30	**89.63**	87.39/1.97	89.59/0.06	90.16/1.31	30	97.18	97.36/0.31	97.18/0.00	97.01/1.00

Support Vector Machine (SVM)									
Arrhythmia					Hill Valley				
# Fea	f^1	$MIFS/t$	$MIFSU/t$	GR/t	# Fea	f^1	$MIFS/t$	$MIFSU/t$	GR/t
5	**58.20**	**39.66/2.45**	**33.19/2.92**	58.64/0.34	5	50.00	50.00/0.01	51.65/1.21	50.17/0.13
10	**63.53**	**39.22/2.95**	**8.86/44.01**	60.20/2.23	10	50.33	50.25/0.23	51.57/1.65	50.49/0.32
15	**63.73**	**39.23/3.11**	**10.18/56.01**	60.20/2.13	15	50.66	50.41/0.51	50.66/0.01	49.92/1.07
20	**65.05**	**34.11/3.92**	**10.41/40.70**	60.87/2.20	20	51.07	50.99/0.18	51.15/0.20	50.09/1.33
25	**66.58**	**14.42/8.18**	**7.78/24.51**	50.10/1.94	25	51.07	50.58/0.94	50.99/0.28	49.59/1.63
30	**67.92**	**18.15/8.36**	**8.65/36.23**	**43.59/2.70**	30	50.58	51.32/1.44	50.58/0.01	51.15/1.35
Image Segmentation					Ionosphere				
# Fea	f^1	$MIFS/t$	$MIFSU/t$	GR/t	# Fea	f^1	$MIFS/t$	$MIFSU/t$	GR/t
3	**78.79**	**53.16/11.03**	79.09/0.17	75.54/1.67	5	83.92	84.48/0.98	85.61/1.74	84.45/0.37
6	**89.70**	**75.84/14.04**	**86.93/5.58**	**74.81/22.40**	10	87.59	87.31/0.32	88.43/1.15	87.86/0.30
9	**92.60**	**86.67/6.79**	**87.53/8.92**	**83.72/6.31**	15	89.29	88.72/0.56	89.01/0.42	88.14/0.95
12	**93.16**	93.29/0.44	**92.21/4.30**	93.29/0.57	20	**90.71**	89.30/2.22	**88.45/2.44**	89.86/1.96
15	**93.38**	93.98/2.09	93.16/1.10	**93.64/2.70**	25	90.72	89.88/1.96	89.87/1.96	90.13/1.03
18	**94.11**	94.20/1.01	94.11/0.02	**94.29/2.45**	30	89.58	90.16/1.50	90.15/0.99	90.43/1.14
Isolet					Libras Movement				
# Fea	f^1	$MIFS/t$	$MIFSU/t$	GR/t	# Fea	f^1	$MIFS/t$	$MIFSU/t$	GR/t
5	**53.26**	**42.12/9.66**	52.66/1.42	**13.15/42.09**	5	**53.31**	52.85/0.13	54.27/0.30	**33.59/4.39**
10	**69.14**	**49.56/14.06**	**58.15/16.21**	**23.74/29.65**	10	**63.64**	69.36/2.10	66.00/1.14	**40.16/9.26**
15	**74.00**	**53.29/22.51**	**64.13/21.82**	**32.86/36.84**	15	**67.99**	**75.55/3.18**	72.77/1.58	**46.58/8.31**
20	**76.77**	**57.46/24.99**	**66.17/18.52**	**37.84/44.69**	20	**71.42**	**76.45/2.77**	**76.62/2.47**	**54.23/6.21**
25	**78.85**	**59.07/27.46**	**66.97/23.07**	**38.98/44.86**	25	**79.22**	80.60/0.77	81.38/1.17	**63.90/5.79**
30	**80.15**	**61.63/40.87**	**68.05/25.39**	**43.85/20.25**	30	**80.25**	82.77/2.15	83.42/2.18	**67.03/6.04**

Support Vector Machine (SVM)									
Arrhythmia					Hill Valley				
# Fea	f^1	$MIFS/t$	$MIFSU/t$	GR/t	# Fea	f^1	$MIFS/t$	$MIFSU/t$	GR/t
5	**61.54**	**52.77/12.21**	**66.69/5.56**	**57.54/4.38**	5	**90.00**	**87.85/3.17**	**85.95/3.28**	**77.15/4.72**
10	**64.19**	**53.12/21.65**	65.19/1.21	**71.15/8.75**	10	**93.95**	92.65/1.57	94.85/1.43	**89.75/5.23**
15	**63.46**	**52.12/13.40**	62.58/1.74	**77.15/10.66**	15	**97.80**	**94.55/5.57**	**96.50/2.62**	**94.05/5.76**
20	**62.42**	**53.31/10.82**	60.85/1.96	**77.77/16.59**	20	**98.35**	**96.35/4.82**	**97.30/2.40**	**95.60/5.55**
25	**61.96**	**53.54/10.11**	**60.31/2.65**	**74.58/14.84**	25	**98.65**	**96.85/3.55**	**97.40/2.95**	**96.30/5.48**
30	**60.23**	**53.19/5.81**	**59.08/3.26**	**72.65/14.94**	30	**98.70**	**97.35/3.95**	**97.50/2.98**	**96.95/3.80**
Landsat Satellite					Sonar				
# Fea	f^1	$MIFS/t$	$MIFSU/t$	GR/t	# Fea	f^1	$MIFS/t$	$MIFSU/t$	GR/t
5	**85.05**	84.83/0.57	85.27/0.73	**83.95/2.84**	5	69.56	68.13/0.51	69.52/0.01	62.48/1.22
10	**87.21**	87.91/1.96	**87.66/2.58**	86.85/1.55	10	75.87	74.13/0.40	70.68/1.61	69.25/2.24
15	88.80	88.79/0.01	88.97/0.70	88.41/1.88	15	79.69	74.63/1.19	72.70/2.13	75.40/2.05
20	89.34	89.63/1.23	89.73/1.64	89.31/0.14	20	74.94	74.05/0.32	73.60/0.56	77.37/1.63
25	89.81	90.27/1.90	90.15/1.68	89.99/1.60	25	77.89	76.46/0.76	74.03/1.57	78.89/0.62
30	90.36	90.43/0.30	90.46/0.38	90.36/0.01	30	**79.80**	**73.06/3.08**	74.01/1.95	79.85/0.02
Spambase					Breast Cancer				
# Fea	f^1	$MIFS/t$	$MIFSU/t$	GR/t	# Fea	f^1	$MIFS/t$	$MIFSU/t$	GR/t
5	**87.98**	**74.48/6.34**	87.65/0.91	**86.37/2.32**	5	**94.56**	**92.11/2.32**	93.15/2.24	94.38/0.37
10	**90.65**	**71.75/7.02**	**88.42/4.32**	**89.52/3.69**	10	94.73	92.47/2.05	93.16/1.40	94.20/1.01
15	**91.33**	**77.09/5.31**	**86.24/2.28**	90.91/1.33	15	94.56	95.08/0.51	94.74/0.17	94.91/1.50
20	**91.70**	**78.05/5.41**	86.44/1.83	89.35/1.29	20	97.01	96.13/1.24	96.67/0.42	97.37/0.62
25	**92.44**	**84.16/3.12**	86.29/2.17	89.70/1.71	25	97.37	97.02/1.00	97.19/0.44	97.20/0.55
30	**92.59**	**85.18/2.69**	86.81/2.22	90.31/1.39	30	97.55	97.37/1.00	97.37/1.00	97.37/1.00

Linear Discriminant Analysis (LDA)									
Arrhythmia					Hill Valley				
# Fea	f^1	$MIFS/t$	$MIFSU/t$	GR/t	# Fea	f^1	$MIFS/t$	$MIFSU/t$	GR/t
5	**21.24**	21.92/0.09	**13.22/2.35**	16.16/1.36	5	50.90	50.99/0.29	51.40/1.96	51.07/0.68
10	32.34	22.58/1.27	43.27/1.74	25.41/1.68	10	50.99	50.99/0.00	51.40/1.86	51.15/0.69
15	**42.97**	**21.92/2.55**	39.92/0.50	33.16/1.92	15	51.40	51.23/0.52	51.56/0.80	51.23/0.81
20	**52.33**	**27.03/3.22**	**31.19/2.60**	**33.90/3.85**	20	51.15	50.99/0.61	51.15/0.01	51.07/0.23
25	**54.19**	43.63/1.57	**28.61/3.42**	**35.82/2.72**	25	51.40	51.15/0.63	51.23/0.80	50.74/2.07
30	**58.07**	46.68/1.90	**30.79/4.03**	**26.23/4.42**	30	51.57	51.23/0.72	51.48/0.37	52.80/1.74
Image Segmentation					Ionosphere				
# Fea	f^1	$MIFS/t$	$MIFSU/t$	GR/t	# Fea	f^1	$MIFS/t$	$MIFSU/t$	GR/t
3	**76.71**	**42.25/22.53**	**70.48/3.55**	73.68/2.00	5	84.30	83.45/0.99	84.03/0.27	82.31/1.76
6	**84.33**	**60.91/22.98**	**72.16/6.78**	**75.02/22.47**	10	84.01	84.85/1.39	84.85/1.15	83.43/0.63
9	**87.53**	**72.03/12.73**	**71.47/22.13**	**82.51/2.91**	15	84.01	84.58/0.61	85.15/1.49	83.70/0.31
12	**89.78**	89.57/0.51	**87.23/6.74**	89.65/1.00	20	84.30	84.56/0.27	84.57/0.28	83.99/0.46
15	89.26	89.18/0.21	89.57/1.65	89.48/0.99	25	83.97	85.44/1.21	84.88/0.72	83.68/0.29
18	89.05	89.05/0.00	89.05/0.00	89.05/0.00	30	83.40	85.13/1.53	84.84/1.19	83.68/0.36
Isolet					Libras Movement				
# Fea	f^1	$MIFS/t$	$MIFSU/t$	GR/t	# Fea	f^1	$MIFS/t$	$MIFSU/t$	GR/t
5	**48.15**	**35.42/9.90**	47.89/0.77	**10.57/42.96**	5	**48.17**	46.14/0.98	46.47/1.85	**37.23/3.53**
10	**61.52**	**41.12/18.83**	**48.72/29.10**	**19.20/31.87**	10	**51.57**	**58.99/2.39**	**60.36/4.65**	**45.65/2.28**
15	**64.01**	**43.91/20.84**	**53.85/22.05**	**27.57/33.54**	15	**57.68**	62.23/1.35	63.48/2.24	**48.15/3.85**
20	**65.44**	**47.24/18.25**	**56.38/14.76**	**32.26/36.08**	20	**60.83**	63.08/0.64	64.50/1.28	**49.13/3.33**
25	**67.27**	**49.19/19.32**	**57.39/10.84**	**33.22/32.61**	25	**63.83**	63.28/0.19	65.02/0.50	**50.90/4.69**
30	**68.35**	**51.62/23.51**	**58.63/14.80**	**36.32/19.85**	30	**66.31**	63.46/1.08	66.23/0.04	**53.26/4.64**

Linear Discriminant Analysis (LDA)									
Arrhythmia					Hill Valley				
# Fea	f^1	$MIFS/t$	$MIFSU/t$	GR/t	# Fea	f^1	$MIFS/t$	$MIFSU/t$	GR/t
5	60.62	61.04/2.08	60.38/0.60	60.08/0.47	5	**88.65**	**82.90/3.83**	**78.95/10.58**	**74.55/6.69**
10	60.62	60.65/0.08	59.96/1.95	61.08/1.02	10	**89.60**	88.25/1.09	88.85/0.63	**84.90/3.84**
15	60.27	60.81/1.06	60.04/0.52	61.38/1.62	15	**94.60**	**90.35/3.49**	**90.75/5.27**	**88.20/10.06**
20	60.00	60.12/0.31	59.92/0.43	60.38/0.47	20	**95.30**	**92.35/2.93**	**92.50/3.32**	**90.25/9.27**
25	60.15	60.15/0.00	59.77/1.23	60.69/0.84	25	**95.80**	**93.60/2.63**	**93.65/3.17**	**91.75/8.06**
30	60.08	59.62/0.82	59.50/1.53	60.19/0.13	30	**96.50**	**94.70/2.59**	**94.00/3.90**	**92.75/8.36**
Landsat Satellite					Sonar				
# Fea	f^1	$MIFS/t$	$MIFSU/t$	GR/t	# Fea	f^1	$MIFS/t$	$MIFSU/t$	GR/t
5	**80.84**	79.92/2.01	81.21/1.19	**78.35/4.45**	5	72.38	72.48/0.03	70.02/1.32	69.05/0.98
10	**82.36**	**81.09/3.99**	**81.32/2.77**	**80.89/3.05**	10	**74.93**	71.43/1.07	**69.10/2.46**	70.50/1.21
15	**82.77**	82.22/1.96	**82.07/2.40**	82.35/1.60	15	**78.21**	**74.38/2.46**	74.88/2.09	**71.98/2.76**
20	82.38	82.28/0.23	82.24/0.46	82.39/0.05	20	**78.24**	**70.55/6.04**	74.83/1.29	74.90/1.33
25	82.25	82.27/0.04	82.19/0.17	81.99/2.01	25	75.40	70.90/1.12	74.33/0.47	74.43/0.63
30	82.45	82.10/1.21	82.03/1.67	82.58/1.93	30	77.81	74.36/1.23	74.83/1.09	77.33/0.32
Spambase					Breast Cancer				
# Fea	f^1	$MIFS/t$	$MIFSU/t$	GR/t	# Fea	f^1	$MIFS/t$	$MIFSU/t$	GR/t
5	**83.16**	**77.07/2.84**	**85.72/5.10**	83.18/0.03	5	**93.85**	**91.21/3.50**	**92.45/2.45**	93.67/0.43
10	**86.74**	**78.25/4.92**	86.42/0.90	**85.70/2.57**	10	**94.38**	92.97/1.35	**92.80/2.38**	94.38/0.00
15	**87.18**	**80.92/3.15**	86.11/0.82	86.55/1.76	15	94.91	95.08/0.44	94.72/0.33	94.91/0.00
20	**87.03**	**82.33/2.46**	87.83/0.65	87.07/0.15	20	96.14	95.26/1.86	95.08/1.77	95.96/0.30
25	**88.18**	**84.42/3.49**	87.57/0.48	86.70/1.89	25	96.66	96.14/0.65	95.61/1.60	96.66/0.00
30	**88.79**	**85.61/3.35**	87.96/0.83	87.74/1.69	30	96.84	96.84/0.02	96.49/0.79	96.66/0.55

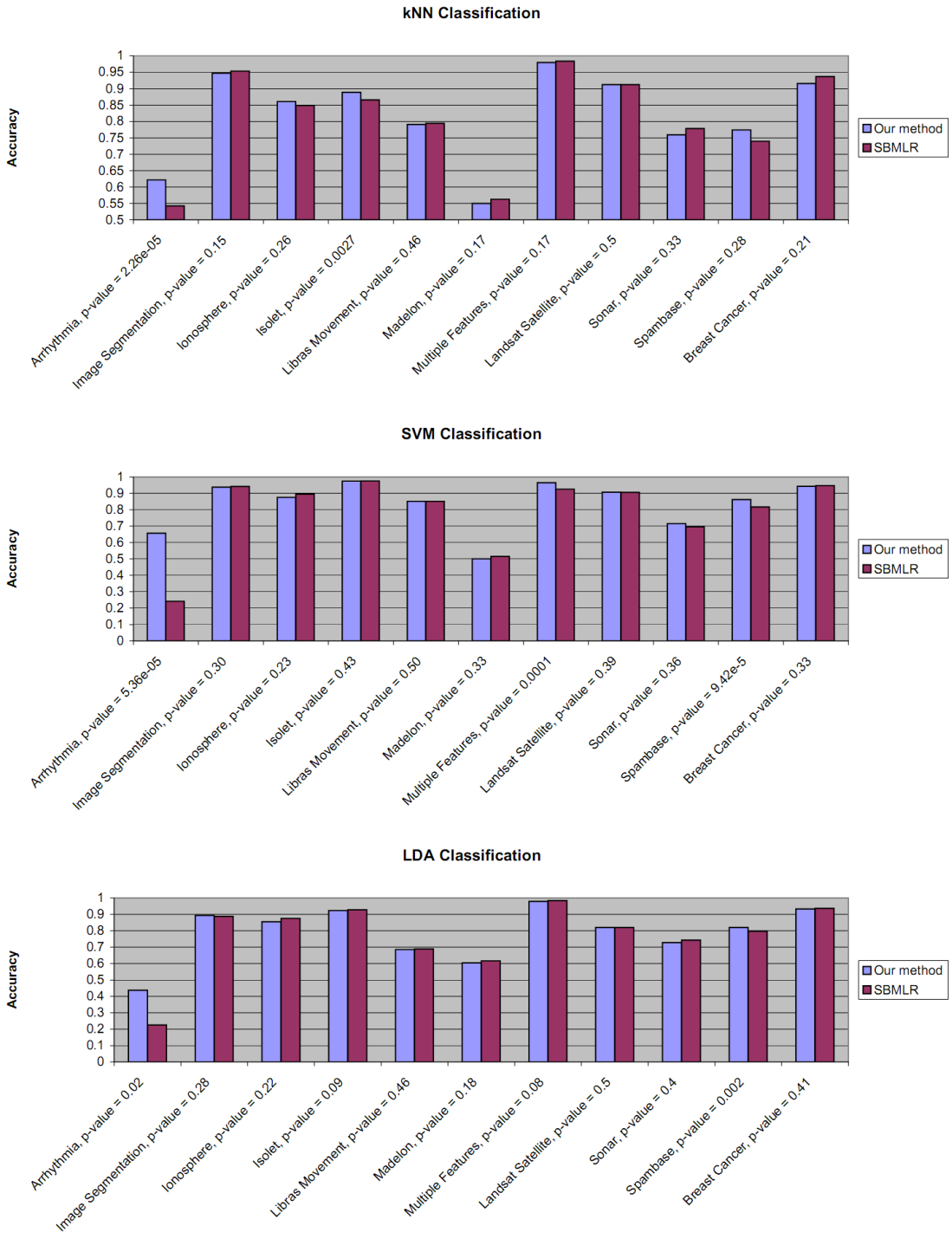


Figure A.1: Classification accuracies of the proposed feature selection method and SBMLR

Appendix B: Linear-Chain Conditional Random Fields

As pointed out in the main content of this dissertation, one of the most important contribution of this work is about the semi-Markov conditional random fields model, which is an extension of the well-known conditional random fields model. However, chapters 2 and 4 just focus on the main points in order to avoid diluting the content. Therefore, the goal of this appendix is to provide more detail of the fundamental theory behind the linear-chain conditional random fields model, which is often used to process sequential data such as speech signal, sentence of words (characters) or human activity sensory data.

A linear-chain conditional random fields model is a graphical model which is illustrated in figure B.1, wherein each white circle represents a class label such, as eating, running or walking in activity recognition, and each gray circle represents an input data, for example one can represent accelerometer input, and the other an represent gyroscope input. Although it has some similarity with the hidden Markov model, there are two obvious differences:

- State labels are observable, each state (represented by a white circle) contains a class label which is explicitly given in the training data.

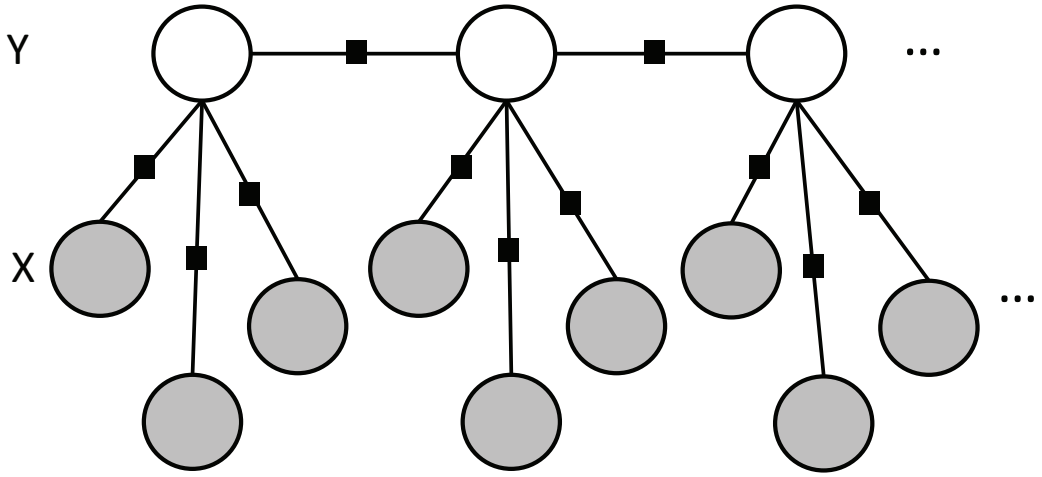


Figure B.1: Factor graph of a linear-chain conditional random fields model

- More flexible state-input dependency, as can be seen there may be more than one input sequences.

Without loss of generality, assume that a training dataset $D = \{x^{(i)}, y^{(i)}\}_{i=1}^{N}$ is given, where each $x^{(i)} = \{x_1^{(i)}, x_2^{(i)}, ..., x_T^{(i)}\}$ is a sequence of inputs, and each $y^{(i)} = \{y_1^{(i)}, y_2^{(i)}, ..., y_T^{(i)}\}$ is a sequence of class labels. The goal of the training phase is to maximize the conditional distribution (also called the likelihood of the training data)

$$p(Y|X) = \sum_{i=1}^{N} p(y^{(i)}|x^{(i)}), \tag{B.1}$$

where each individual conditional distribution is defined as

$$p(y^{(i)}|x^{(i)}) = \frac{e^{\left\{\sum_{k=1}^{K} \lambda_k f_k(y^{(i)}, x^{(i)})\right\}}}{Z^{(i)}}, \tag{B.2}$$

where

$$Z^{(i)} = \sum_{y'} e^{\left\{ \sum_{k=1}^{K} \lambda_k f_k(y', x^{(i)}) \right\}}. \tag{B.3}$$

In the above equations, K is the number of features (statistical dependencies that is learnt by the model) f_k is a feature function vector, λ_k is a weight vector. In the following paragraphs, the 3 most commonly used feature functions are considered including prior, transition and observation.

A prior feature function is defined as

$$f_s^{prior}(y^{(i)}, x^{(i)}) = \delta(y_1^{(i)} = s) \forall s, \tag{B.4}$$

then the corresponding feature function vector and weight vector are defined as

$$f_1 = \left[f_s^{prior} \right] \forall s, \tag{B.5}$$

and

$$\lambda_1 = \left[\lambda_s^{prior} \right] \forall s, \tag{B.6}$$

where s is a state value (or a class label).

Similarly, transition feature functions and weights are described below

$$f_{ss'}^{trans}(y^{(i)}, x^{(i)}) = \sum_{t=1}^{T-1} \delta(y_t^{(i)} = s, y_{t+1}^{(i)} = s') \forall s, s', \tag{B.7}$$

where T is the length of the sequences $y^{(i)}$,

$$f_2 = \left[f_{ss'}^{trans} \right] \forall s, s', \tag{B.8}$$

$$\lambda_2 = \left[\lambda_{ss'}^{trans} \right] \forall s, s'. \tag{B.9}$$

Finally, observation feature functions and weights are given as followings

$$f_{so}^{obs}(y^{(i)}, x^{(i)}) = \sum_{t=1}^{T} \delta(y_t^{(i)} = s, x_t^{(i)} = o) \forall s, o, \tag{B.10}$$

where o is a discrete value of the input sequence at time instance t.

$$f_3 = \left[f_{so}^{obs} \right] \forall s, o, \tag{B.11}$$

$$\lambda_3 = \left[\lambda_{so}^{obs} \right] \forall s, o. \tag{B.12}$$

To avoid the use of exponential functions, which can result in overflow calculation, the conditional density function in B.2 is replaced by its logarithmic value. Hence, the likelihood of training data in B.1 is also replaced by a log-likelihood function defined as

$$L(Y|X) = \sum_{i=1}^{N} log(p(y^{(i)}|x^{(i)}))$$

$$= \sum_{i=1}^{N} \left(\sum_{k=1}^{K} \lambda_k f_k(y^{(i)}, x^{(i)}) - log(Z^{(i)}) \right) \tag{B.13}$$

The maximization of $L(Y|X)$ is done by using the gradient method, therefore it is necessary to evaluate the function and its gradients. As can be seen in equation B.13, the left side is a trivial sum and can be easily computed, the right part ($Z^{(i)}$)) is however a sum of exponentially large number of values. Therefore, the dynamical programming methods called forward/backward are applied to resolve the calculation.

Calculate Z using the forward algorithm

$$\alpha^{(i)}(s, 1) = e^{\left\{ \sum_{k=1}^{K} \lambda_k f_k(\{s\}, x^{(i)}) \right\}} \forall s, \tag{B.14}$$

where s can be any class label,

$$\alpha^{(i)}(s,\tau) = \sum_{y=\{y_1,y_2,...,y_\tau=s\}} e^{\left\{\sum_{k=1}^{K}\lambda_k f_k(y,x^{(i)})\right\}},$$

$$= \sum_{s'} \alpha^{(i)}(s',\tau-1) \times e^{\left\{\lambda^{trans}_{s's} + \lambda^{obs}_{s,x^{(i)}_\tau}\right\}} \forall s,\tau, \tag{B.15}$$

$$Z^{(i)} = \sum_{s} \alpha^{(i)}(s,T). \tag{B.16}$$

Calculate gradient

$$\frac{dL}{d\lambda_k} = \sum_{i=1}^{N}\left(f_k(y^{(i)},x^{(i)}) - \frac{\frac{dZ^{(i)}}{d\lambda_k}}{Z^{(i)}}\right)$$

$$= \sum_{i=1}^{N}\left(f_k(y^{(i)},x^{(i)}) - \frac{\sum_{y'} f_k(y',x^{(i)}) e^{\left\{\sum_{k=1}^{K}\lambda_k f_k(y',x^{(i)})\right\}}}{Z^{(i)}}\right). \tag{B.17}$$

The numerator of the right component in equation (B.17) is computed using a similar forward algorithm when computing $Z^{(i)}$.

Inference algorithm

Once the model is trained, the inference algorithm decides the label sequence $y(T) = \{y_1, y_2, ..., y_T\}$ of a given testing input sequence $x(T) = \{x_1, x_2, ..., x_T\}$ by solving the below maximization problem

$$y(T) = argmax_{y'(T)} p(y'(T)|x(T)),$$

Denote

$$y(\tau,s) = argmax_{y'(\tau):y'_\tau=s} p(y'(\tau)|x(\tau)), \tag{B.18}$$

then it can be easily seen that

$$y(\tau) = argmax_s p(y(\tau, s)|x(\tau)). \tag{B.19}$$

Furthermore, $y(\tau, s)$ can be computed using backtracking method as below

$$max\left(p\left(y(\tau, s)|x(\tau)\right)\right) = max_{s'} p\left(y(\tau - 1, s')|x(\tau - 1)\right) + \lambda_{s's}^{trans} + \lambda_{s,x_\tau(\tau)}^{obs} \tag{B.20}$$

References

[1] F. R. Allen, E. A. N. H. Lovell, and B. G. Celler. Classification of a known sequence of motions and postures from accelerometry data using adapted Gaussian mixture models. *Physiological Measurement*, 27:935–951, 2006.

[2] A. H. Andrew, Y. Anokwa, K. Koscher, J. Lester, and G. Borriello. Context to make you more aware. In *Proceedings of the 27th International Conference on Distributed Computing Systems Workshops*, page 49, Toronto, Ontario, Canada, June 25-29, 2007. IEEE Computer Society.

[3] D. Arvidsson, F. Slinde, and L. Hulthn. Free-living energy expenditure in children using multi-sensor activity monitors. *Clinical Nutrition*, 28(3):305–312, 2009.

[4] A. Asuncion and D. Newman. UCI machine learning repository, http://www.ics.uci.edu/∼mlearn/MLRepository.html, University of California, Irvine, School of Information and Computer Sciences, 2007.

[5] L. Atallah, B. Lo, R. King, and G.-Z. Yang. Sensor placement for activity detection using wearable accelerometers. In *Proceedings of the 2010 International Conference on Body Sensor Networks*, pages 24–29, London, United Kingdom, June 7-9, 2010. IEEE Press.

[6] L. Atallah, B. Lo, R. King, and G.-Z. Yang. Sensor positioning for activity recognition using wearable accelerometers. *IEEE Transactions on Biomedical Circuits and Systems*, 5(4):320–329, 2011.

[7] R. Aylward. *Sensemble: A Wireless Inertial Sensor System for the Interactive Dance and Collective Motion Analysis*. PhD thesis, Massachusetts Institute of Technology, 2006.

[8] L. Bao and S. S. Intille. Activity recognition from user-annotated acceleration data. In A. Ferscha and F. Mattern, editors, *Proceedings of the 2nd International Conference on Pervasive Computing*, pages 1–17, Vienna, Austria, April 21-23, 2004. Springer.

[9] R. Battiti. Using mutual information for selecting features in supervised neural net learning. *IEEE Transactions on Neural Networks*, 5(4):537–550, 1994.

[10] C. M. Bishop. *Pattern Recognition and Machine Learning*. Springer, 1st edition, 2007.

[11] U. Blanke and B. Schiele. Daily routine recognition through activity spotting. In T. Choudhury, A. Quigley, T. Strang, and K. Suginuma, editors, *Proceedings of the 4th International Symposium on Location and Context-Awareness*, pages 192–206, Tokyo, Japan, May 7-8, 2009. Springer.

[12] A. G. Bonomi and K. R. Westerterp. Advances in physical activity monitoring and lifestyle interventions in obesity: a review. *International Journal of Obesity*, 36:167–177, 2012.

[13] O. Brdiczka, J. Maisonnasse, P. Reignier, and J. L. Crowley. Detecting small group activities from multimodal observations. *Applied Intelligence*, 30:47–57, 2009.

[14] A. Brush, J. Krumm, and J. Scott. Activity recognition research: The good, the bad, and the future. In *Pervasive 2010 Workshop How to Do Good Research in Activity Recognition*, Helsinki, Finland, May 17, 2010.

[15] A. Bulling, J. A. Ward, and H. Gellersen. Multimodal recognition of reading activity in transit using body-worn sensors. *ACM Transactions on Applied Perception*, 9(1):2:1–2:21, 2012.

[16] G. C. Cawley, N. L. C. Talbot, and M. Girolami. Sparse multinomial logistic regression via Bayesian L1 regularisation. *Advances in Neural Information Processing Systems*, 19:209–216, 2007.

[17] C. Chen, B. Das, and D. J. Cook. A data mining framework for activity recognition in smart environments. In *Proceedings of the 6th International Conference on Intelligent Environments*, pages 80–83, Kuala Lumpur, Malaysia, July 19-21, 2010. IEEE Press.

[18] G. Claxton, B. Dijulio, B. Finder, J. Lundy, M. McHugh, A. Osei-Anto, H. Whitmore, J. Pickreign, and J. Gabel. Employer health benefits 2009 annual survey. Technical report, Kaiser Family Foundation and Health Research and Education Trust, 2009.

[19] M. Dash and H. Liu. Feature selection for classification. *Intelligent Data Analysis*, 1:131–156, 1997.

[20] D. B. Dias, R. C. B. Madeo, T. Rocha, H. H. Bscaro, and S. M. Peres. Hand movement recognition for Brazilian sign language: A study using distance-based neural networks. In *Proceedings of the International Joint Conference on Neural Networks*, pages 697–704, Atlanta, Georgia, United States, June 14-19, 2009. IEEE Press.

[21] T. G. Dietterich and G. Bakiri. A general method for improving multiclass inductive learning programs. In *Proceedings of the 9th National Conference on Artificial Intelligence*, page 572, Anaheim, California, United States, July 14-19, 1991. AAAI Press.

[22] T. G. Dietterich and G. Bakiri. Solving multiclass learning problems via errorcorrecting codes. *Journal of Artificial Intelligence Research*, 2:263–268, 1995.

[23] P. A. Estvez, M. Tesmer, C. A. Perez, and J. M. Zurada. Normalized mutual information feature selection. *IEEE Transactions on Neural Networks*, 20(2):189–201, 2009.

[24] J. Fahrenberg, F. Foerster, M. Smeja, and W. Muller. Assessment of posture and motion by multichannel piezoresistive accelerometer recordings. *Psychophysiology*, 34:607–612, 1996.

[25] J. Fahrenberg, W. Muller, F. Foerster, and M. Smeja. A multi-channel investigation of physical activity. *Psychophysiology*, 10:209–2172, 1996.

[26] M. A. Fanty and R. Cole. Spoken letter recognition. In R. P. Lippmann, J. E. Moody, and D. S. Touretzky, editors, *Proceedings of the Neural Information Processing Systems Conference*, pages 220–226, Denver, Colorado, United States, November 26-29, 1990. Morgan Kaufmann.

[27] J. Favela, M. Tentori, L. A. Castro, V. M. Gonzalez, E. B. Moran, and A. I. Martnez-Garca. Activity recognition for context-aware hospital applications: issues and opportunities for the deployment of pervasive networks. *Mobile Networks and Applications*, 12:155–171, 2007.

[28] I. K. Fodor. A survey of dimension reduction techniques. Technical report, Center for Applied Scientific Computing, Lawrence Livermore National Laboratory, 2002.

[29] S. R. Garner. WEKA: The Waikato environment for knowledge analysis. In *Proceedings of the New Zealand Computer Science Research Students Conference*, pages 57–64, Hamilton, New Zealand, April 18-21, 1995.

[30] R. P. Gorman and T. J. Sejnowski. Analysis of hidden units in a layered network trained to classify sonar targets. *Neural Networks*, 1:75–89, 1998.

[31] C. H. Goulden. *Methods of Statistical Analysis*. Wiley, 2nd edition, 1956.

[32] H. A. Guvenir, B. Acar, G. Demiroz, and A. Cekin. A supervised machine learning algorithm for arrhythmia analysis. In *Proceedings of the Computers in Cardiology Conference*, pages 433–436, Lund, Sweden, September 7-10, 1997. IEEE Press.

[33] I. Guyon, S. Gunn, M. Nikravesh, and L. Zadeh. *Feature Extraction, Foundations and Applications*. Springer, 2006.

[34] I. Guyon, J. Li, T. Mader, P. A. Pletscher, G. Schneider, and M. Uhr. Competitive baseline methods set new standards for the NIPS 2003 feature selection benchmark. *Pattern Recognition Letters*, 28:1438–1444, 2007.

[35] M. Hall, E. Frank, G. Holmes, B. Pfahringer, P. Reutemann, and I. H. Witten. The WEKA data mining software: An update. *SIGKDD Explorations*, 11(1):10–18, 2009.

[36] M. A. Hall. *Correlation-based Feature Selection for Machine Learning*. PhD thesis, The University of Waikato, 1999.

[37] Y. Hanai, J. Nishimura, and T. Kuroda. Haar-like filtering for human activity recognition using 3D accelerometer. In *IEEE 13th Digital Signal Processing Workshop and 5th IEEE Signal Processing Education Workshop*, pages 675–678, Marco Island, Florida, United States, Jan 4-7, 2009. IEEE Press.

[38] Z.-Y. He and L.-W. Jin. Activity recognition from acceleration data using AR model representation and SVM. In *Proceedings of the 7th International Conference on Machine Learning and Cybernetics*, pages 2245–2250, Kunming, China, July 12-15, 2008. IEEE Press.

[39] G. Hughes. On the mean accuracy of statistical pattern recognizers. *IEEE Transactions on Information Theory*, 14:55–63, 1968.

[40] T. Huynh, U. Blanke, and B. Schiele. Scalable recognition of daily activities with wearable sensors. In J. Hightower and B. Schiele, editors, *Proceedings of the 3rd International Symposium on Location and Context Awareness*, pages 50–67, Oberpfaffenhofen, Germany, September 20-21, 2007. Springer.

[41] T. Huynh, M. Fritz, and B. Schiele. Discovery of activity patterns using topic models. In *Proceedings of the 10th International Conference on Ubiquitous Computing*, pages 10–19, Seoul, South Korea, September 21-24, 2008. ACM Press.

[42] T. Huynh and B. Schiele. Analyzing features for activity recognition. In *Proceedings of the 2005 Joint Conference on Smart Objects and Ambient Intelligence: Innovative Context-Aware Services: Usages and Technologies*, pages 159–163, Grenoble, France, October 12-14, 2005. ACM.

[43] R. Jafari, W. Li, R. Bajcsy, S. Glaser, and S. Sastry. Physical activity monitoring for assisted living at home. *Electrical and Electronic Engineering*, 13:213–219, 2007.

[44] J. L. W. V. Jensen. Sur les fonctions convexes et les ingalits entre les valeurs moyennes. *Acta Mathematica*, 30(1):175–193, 1906.

[45] A. M. Khan, Y.-K. Lee, S. Lee, and T.-S. Kim. A triaxial accelerometer-based physical activity recognition via augmented features and a hierarchical recognizer. *IEEE Transactions on Information Technology in Biomedicine*, 14(5):1166–1172, 2010.

[46] K. Kira and L. A. Rendell. A practical approach to feature selection. In D. H. Sleeman and P. Edwards, editors, *Proceedings of the Ninth International Workshop on Machine Learning*, pages 249–256, Aberdeen, Scotland, July 1-3, 1992. Morgan Kaufmann.

[47] S. Knoop, S. Vacek, R. Dillmann, S. Brannstrom, and H. Christensen. Extraction, evaluation, selection and classification of motion features for human activity recognition. Technical report, Universitat Karlsruhe, 2005.

[48] M. Kurz, G. Hoelzl, A. Ferscha, A. Calatroni, D. Roggen, G. Troester, H. Sagha, R. Chavarriaga, J. Milln, D. Bannach, K. Kunze, and P. Lukowicz. Opportunity framework and data processing ecosystem for opportunistic activity and context recognition. *The International Journal of Sensors, Wireless Communications and Control*, 1(2):102–125, 2011.

[49] N. Kwak and C.-H. Choi. Input feature selection for classification problems. *IEEE Transactions on Neural Networks*, 13(1):143–159, 2002.

[50] J. Lafferty, A. McCallum, and F. Pereira. Conditional random fields: Probabilistic models for segmenting and labeling sequence data. In C. E. Brodley and A. P. Danyluk, editors, *Proceedings of the*

18th International Conference on Machine Learning, pages 282–289, Williamstown, United States, June 28-July 1, 2001. Morgan Kaufmann.

[51] M. Lampe, M. Strassner, and E. Fleisch. A ubiquitous computing environment for aircraft maintenance. In H. Haddad, A. Omicini, R. L. Wainwright, and L. M. Liebrock, editors, *Proceedings of ACM Symposium on Applied Computing*, pages 1586–1592, Nicosia, Cyprus, March 14-17, 2004. ACM Press.

[52] L. Liao, D. Fox, and H. A. Kautz. Extracting places and activities from GPS traces using hierarchical conditional random fields. *International Journal of Robotics Research*, 26:119–134, 2007.

[53] Y. Linde, A. Buzo, and R. M. Gray. An algorithm for vector quantizer design. *IEEE Transactions on Communications*, 28:84–94, 1980.

[54] K. J. Liszka, M. A. Mackin, M. J. Lichter, D. W. York, D. Pillai, and D. S. Rosenbaum. Keeping a beat on the heart. *IEEE Pervasive Computing*, 3(4):42–49, 2004.

[55] M. Losch. Feature set selection and optimal classifier for human activity recognition. In *Proceedings of the 16th IEEE International Symposium on Robot and Human interactive Communication*, pages 1022–1027, Jeju, South Korea, August 26-29, 2007. IEEE Press.

[56] P. Lukowicz, T. E. Starner, G. Troster, and J. A. Ward. Activity recognition of assembly tasks using body-worn microphones and accelerometers. *IEEE Transaction on Pattern Analysis and Machine Intelligence*, 28:1553–1567, 2006.

[57] M. Mahdaviani. *Semi-supervised and Active Training of Conditional Random Fields for Activity Recognition*. PhD thesis, The University of British Columbia, 2005.

[58] U. Maurer. Activity recognition and monitoring using multiple sensors on different body positions. In *Proceedings of the International Workshop on Wearable and Implantable Body Sensor Networks*, pages 114–116, Munchen, Germany, April 3-5, 2006. IEEE Press.

[59] U. Maurer, A. Rowe, A. Smailagic, and D. Siewiorek. Location and activity recognition using eWatch: A wearable sensor platform. *Ambient Intelligence in Everyday Life, Lecture Notes in Computer Science*, 3864:86–102, 2006.

[60] Centers for Medicare & Medicaid Services. National health care expenditures data. Technical report, Office of the Actuary, National Health Statistics Group, 2010.

[61] S&P Indices. US healthcare costs annual growth rates increase in october 2011 according to the S&P healthcare economic indices, 2011.

[62] World Health Organization. Global strategy on diet, physical activity and health. Technical report, 2004.

[63] C. McCall, K. K. Reddy, and M. Shah. Macro-class selection for hierarchical k-NN classification of inertial sensor data. In C. Benavente-Peces, F. H. Ali, and J. Filipe, editors, *Proceedings of the 2nd International Conference on Pervasive Embedded Computing and Communication Systems*, pages 106–114, Rome, Italy, February 24-26, 2012. SciTePress.

[64] R. S. Michalski, I. Mozetic, J. Hong, and N. Lavrac. The multi-purpose incremental learning system AQ15 and its testing application to three medical domains. In T. Kehler and S. J. Rosenschein, editors, *Proceedings of the 5th National Conference on Artificial Intelligence*, pages 1041–1045, Philadelphia, United States, August 11-15, 1986. Morgan Kaufmann.

[65] C.-H. Min and A. H. Tewfik. Automatic characterization and detection of behavioral patterns using linear predictive coding of accelerometer sensor data. In *Proceedings of the IEEE Annual International Conference on Engineering in Medicine and Biology Society*, pages 220–223, Buenos Aires, Argentina, August 31-September 4, 2010. IEEE Press.

[66] D. Minnen, T. L. Westeyn, D. Ashbrook, P. Presti, and T. Starner. Recognizing soldier activities in the field. In S. Leonhardt, T. Falck, and P. Mhnen, editors, *Proceedings of the 4th International Workshop on Wearable and Implantable Body Sensor Networks*, pages 236–241, Aachen University, Germany, March 26-28, 2007. Springer.

[67] L. Nachman, R. Shah, J. Huang, J. Shahabdeen, C. Wan, and G. Raffa. Intel context awareness project: Activity recognition, 2010.

[68] B. Najafi, K. Aminian, F. Loew, Y. Blanc, and P. A. Robert. Measurement of stand-sit and sit-stand transitions using a miniature gyroscope and its application in fall risk evaluation in the elderly. *IEEE Transactions on Biomedical Engineering*, 49:843–851, 2002.

[69] J. Nocedal. Updating quasi-Newton matrices with limited storage. *Mathematics of Computation*, 35:773–782, 1980.

[70] D. Okanohara, Y. Miyao, Y. Tsuruoka, and J. Tsujii. Improving the scalability of semi-Markov conditional random fields for named entity recognition. In N. Calzolari, C. Cardie, and P. Isabelle, editors, *Proceedings of the 21st International Conference on Computational Linguistics and the 44th annual meeting of the Association for Computational Linguistics*, pages 465–472, Sydney, Australia, July 17-21, 2006. The Association for Computer Linguistics.

[71] L. Palmerini. Feature selection for accelerometer-based posture analysis in parkinson's disease. *IEEE Transactions on Information Technology in Biomedicine*, 15(3):481–490, 2011.

[72] P. Palmes, H. K. Pung, T. Gu, W. Xue, and S. Chen. Object relevance weight pattern mining for activity recognition and segmentation. *Pervasive and Mobile Computing*, 6(1):43–57, 2010.

[73] J. Parkka, M. Ermes, P. Korpipaa, and J. M. et al. Activity classification using realistic data from wearable sensors. *IEEE Transaction on Information Technology in Biomedicine*, 10:119–128, 2006.

[74] D. J. Patterson, L. Liao, K. Gajos, M. Collier, N. Livic, K. Olson, S. Wang, D. Fox, and H. A. Kautz. Opportunity knocks: A system to provide cognitive assistance with transportation services. In N. Davies, E. D. Mynatt, and I. Siio, editors, *Proceedings of the 6th international conference on Ubiquitous computing (UBICOMP)*, pages 433–450, Nottingham, England, September 7-10, 2004. Springer.

[75] H. Peng, F. Long, and C. Ding. Feature selection based on mutual information criteria of max-dependency, max-relevance, and min-redundancy. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 27(8):1226–1238, 2005.

[76] S. J. Preece, J. Y. Goulermas, L. P. J. Kenney, and D. Howard. A comparison of different feature generation methods in activity classification. In J. Bussmann, H. Horemans, and H. Hurkmans, editors, *Proceedings of the International Conference on Ambulatory Monitoring of Physical Activity and Movement*, Rotterdam, The Netherlands, May 21-24, 2008. Dept. of Rehabilitation Medicine Erasmus MC.

[77] S. J. Preece, J. Y. Goulermas, L. P. J. Kenney, D. Howard, K. Meijer, and R. Crompton. Activity identification using body-mounted sensors: a review of classification techniques. *Physiological Measurement*, 30:R1–R33, 2009.

[78] T. Prill and J. Fahrenberg. Simultaneous assessment of posture and limb movements with calibrated multiple accelerometry. *Physiological Measurement*, 27:47–53, 2006.

[79] J. R. Quinlan. Induction of decision trees. *Machine Learning*, 1(1):81–106, 1986.

[80] L. R. Rabiner. A tutorial on hidden Markov models and selected applications in speech recognition. *Proceedings of the IEEE*, 77(2):257–286, 1989.

[81] N. Ravi, N. Dandekar, P. Mysore, and M. L. Littman. Activity recognition from accelerometer data. In M. M. Veloso and S. Kambhampati, editors, *Proceedings of the 20th National Conference on Artificial Intelligence*, pages 1541–1546, Pennsylvania, United States, July 9-13, 2005. AAAI Press.

[82] D. Riboni and C. Bettini. Cosar: hybrid reasoning for context-aware activity recognition. *Personal and Ubiquitous Computing*, 15(3):271–289, 2011.

[83] N. Robertson and I. Reid. A general method for human activity recognition in video. *Journal of Computer Vision and Image Understanding*, 104:232–248, 2006.

[84] M. C. Sala, K. Partridge, L. Jacobson, and J. Begole. An exploration into activity-informed physical advertising using pest. *Lecture Notes in Computer Science*, 4480:73–90, 2007.

[85] S. Sarawagi and W. Cohen. Semi-Markov conditional random fields for information extraction. In L. K. Saul, Y. Weiss, and L. Bottou, editors, *Proceedings of the 2004 Conference on Neural Information Processing Systems*, pages 1185–1192, Cambridge, Massachusetts, United States, December 13-18, 2005. MIT Press.

[86] J. Sarkar, L. T. Vinh, Y.-K. Lee, and S. Lee. GPARS: A general purpose activity recognition system. *Applied Intelligence*, 35:242–259, 2011.

[87] scar D. Laraa, A. J. Preza, M. A. Labradora, and J. D. Posadab. Centinela: A human activity recognition system based on acceleration and vital sign data. *Pervasive and Mobile Computing (In Press, Corrected Proof)*, 2011.

[88] M. Sekine, T. Tamura, M. Akay, and T. F. et al. Discrimination of walking patterns using wavelet-based fractal analysis. *IEEE Transaction on Neural Systems and Rehabilitation Engineering*, 10:188–196, 2002.

[89] M. Sekine, T. Tamura, T. Togawa, and Y. Fukui. Classification of waist-acceleration signal in a continuous walking record. *Medical Engineering and Physics*, 22:285–291, 2000.

[90] B. Settles. Biomedical named entity recognition using conditional random fields and rich feature sets. In N. Collier, P. Ruch, and A. Nazarenko, editors, *Proceedings of the International Joint Workshop on Natural Language Processing in Biomedicine and its Applications*, pages 104–107, Geneva, Switzerland, August 28-29, 2004. Association for Computational Linguistics.

[91] F. Sha and F. Pereira. Shallow parsing with conditional random fields. In *Proceedings of the 2003 Conference of the North American Chapter of the Association for Computational Linguistics on Human Language*, pages 134–141, Edmonton, Canada, May 27-June 1, 2003. Association for Computational Linguistics.

[92] K.-Q. Shen, C.-J. Ong, and X.-P. Li. Novel multi-class feature selection methods using sensitivity analysis of posterior probabilities. In *Proceedings of the IEEE International Conference on Systems, Man and Cybernetics*, pages 1116–1121, Singapore, October 12-15, 2008. IEEE Press.

[93] V. G. Sigillito, S. P. Wing, L. V. Hutton, and K. B. Baker. Classification of radar returns from the ionosphere using neural networks. *Johns Hopkins APL Tech. Dig*, 10:262–266, 1989.

[94] P. SJ, G. JY, K. LP, and H. D. A comparison of feature extraction methods for the classification of dynamic activities from accelerometer data. *IEEE Transaction on Biomedical and Engineering*, 56(3):879–879, 2009.

[95] D. Snchez, M. Tentori, and J. Favela. Activity recognition for the smart hospital. *IEEE Intelligent Systems*, 23:50–57, 2008.

[96] T. Stiefmeier, D. Roggen, G. Ogris, P. Lukowicz, and G. Trster. Wearable activity tracking in car manufacturing. *IEEE Pervasive Computing*, 7(1):42–50, 2008.

[97] L. Sun, D. Zhang, B. Li, B. Guo, and S. Li. Activity recognition on an accelerometer embedded mobile phone with varying positions and orientations. *Lecture Notes in Computer Science*, 6406:548–562, 2010.

[98] J. Suutala, S. Pirttikangas, and J. Rning. Discriminative temporal smoothing for activity recognition from wearable sensors. In H. Ichikawa, W.-D. Cho, I. Satoh, and H. Y. Youn, editors, *Proceedings of the 4th International Symposium on Ubiquitous Computing Systems*, pages 182–195, Tokyo, Japan, November 25-28, 2007.

[99] E. M. Tapia, S. S. Intille, W. Haskell, K. Larson, J. Wright, A. King, and R. Friedman. Real-time recognition of physical activities and their intensities using wireless accelerometers and a heart rate monitor. In *Proceedings of the 11th IEEE International Symposium on Wearable Computers*, pages 37–40, Boston, United State, October 11-13, 2007. IEEE Computer Society.

[100] E. M. Tapia, S. S. Intille, and K. Larson. Activity recognition in the home using simple and ubiquitous sensors. *Lecture Notes in Computer Science*, 3001:158–175, 2004.

[101] A. Teller. A platform for wearable physiological computing. *Interacting with Computers*, 16(5):917–937, 2004.

[102] S. Theodoridis and K. Koutroumbas. *Pattern Recognition*. Elsevier, 4th edition, 2008.

[103] T. T. Truyen, D. Q. Phung, H. H. Bui, and S. Venkatesh. Hierarchical semi-Markov conditional random fields for recursive sequential data. In D. Koller, D. Schuurmans, Y. Bengio, and L. Bottou, editors, *Proceedings of the 22nd Annual Conference on Neural Information Processing Systems*, pages 1657–1664, Vancouver, Canada, December 8-13, 2009. MIT Press.

[104] D. L. Vail, M. M. Veloso, and J. D. L. Lafferty. Conditional random fields for activity recognition. In E. H. Durfee, M. Yokoo, M. N. Huhns, and O. Shehory, editors, *Proceedings of the 6th International Joint Conference on Autonomous Agents and Multi-agent Systems*, page 235, Honolulu, Hawaii, United States, May 14-18, 2007.

[105] M. van Breukelen and R. Duin. Neural network initialization by combined classifiers. In *Proceedings of the 14th International Conference on Pattern Recognition*, pages 215–218, Brisbane, Australia, August 16-20, 1998. IEEE Press.

[106] M. van Breukelen, R. Duin, D. Tax, and J. den Hartog. Handwritten digit recognition by combined classifiers. *Kybernetika*, 34(4):381–386, 1998.

[107] T. van Kasteren, A. Noulas, G. Englebienne, and B. Krose. Accurate activity recognition in a home setting. In H. Y. Youn and W.-D. Cho, editors, *Proceedings of the 10th international conference on Ubiquitous computing (UBICOMP)*, pages 1–9, Seoul, Korea, September 21-24, 2008. ACM Press.

[108] L. T. Vinh, N. D. Thang, and Y.-K. Lee. An improved maximum relevance and minimum redundancy feature selection algorithm based on normalized mutual information. In *Proceedings of the 10th IEEE/IPSJ International Symposium on Applications and the Internet*, pages 395–398, Seoul, South Korea, July 19-23, 2010. IEEE Press.

[109] A. J. Viterbi. Error bounds for convolutional codes and an asymptotically optimum decoding algorithm. *IEEE Transactions on Information Theory*, 13(2):260–269, 1967.

[110] H. M. Wallach. Conditional random fields: An introduction. Technical report, University of Pennsylvania, 2004.

[111] N. Wang, E. Ambikairajah, N. H. Lovell, and B. G. Celler. Accelerometry based classification of walking patterns using time-frequency analysis. In *Proceedings of the 29th Annual Conference of the IEEE Engineering in Medicine and Biology Society*, Lyon, France, August 22-26, 2007. IEEE Press.

[112] S. B. Wang, A. Quattoni, L. P. Morency, D. Demirdjian, and T. Darrell. Hidden conditional random fields for gesture recognition. In *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, pages 1521–1527, New York, United States, June 17-22, 2006. IEEE Press.

[113] H. Xia and B. Q. Hu. Feature selection using fuzzy support vector machines. *Fuzzy Optimization and Decision Making*, 5(2):187–192, 2006.

[114] R. Yan. MatlabArsenal toolbox for classification algorithms, Informedia School of Computer Science, Carnegie Mellon University, 2006.

[115] G.-X. Yuan, K.-W. Chang, C.-J. Hsieh, and C.-J. Lin. A comparison of optimization methods and software for large-scale L1-regularized linear classification. *Journal of Machine Learning Research*, 11:3183–234, 2010.

[116] H. Zhang and B. Hartmann. Building upon everyday play. In M. B. Rosson and D. J. Gilmore, editors, *Proceedings of the 2007 Conference on Human Factors in Computing Systems*, pages 2019–2024, San Jose, California, United State, April 28-May 3, 2007. ACM.

[117] M. Zhang and A. A. Sawchuk. A feature selection-based framework for human activity recognition using wearable multimodal sensors. In *Proceedings of the International Conference on Body Area Networks*, Beijing, China, November 7-10, 2011. IEEE Press.

[118] Z. Zhao, F. Morstatter, S. Sharma, S. Alelyani, A. Anand, and H. Liu. Advancing feature selection research - ASU feature selection repository. Technical report, School of Computing, Informatics, and Decision Systems Engineering, Arizona State University, 2010.