

EXPLAINABLE PATTERN MODELLING AND SUMMARIZATION IN SENSOR
EQUIPPED SMART HOMES OF ELDERLY

A Dissertation

presented to

the Faculty of the Graduate School

at the University of Missouri-Columbia

In Partial Fulfillment

of the Requirements for the Degree

Doctor of Philosophy

by

AKSHAY JAIN

Dr. James Keller, Dissertation Supervisor

July 2020

The undersigned, appointed by the dean of the Graduate School, have examined the
dissertation entitled

EXPLAINABLE PATTERN MODELLING AND SUMMARIZATION IN SENSOR
EQUIPPED SMART HOMES OF ELDERLY

presented by Akshay Jain,
a candidate for the degree of Doctor of Philosophy,
and hereby certify that, in their opinion, it is worthy of acceptance.

Dr. James Keller

Dr. Mihail Popescu

Dr. Marjorie Skubic

Dr. Richelle Koopman

In memory of my friend, brother, and cousin
Ruchit Jain

Acknowledgements

I would like to thank my mentor Dr. James Keller for his invaluable guidance throughout my graduate studies. He continues to be a source of inspiration and I could not have asked for a better adviser. I would like to thank Dr. Mihail Popescu for providing me the opportunity to work on his grants, which along with providing me funding, helped me gain important work experience. His insights throughout my graduate studies have been of immense help. I am grateful to Dr. Marjorie Skubic for her guidance and the opportunity to work with a great interdisciplinary group at the Center for Eldercare and Rehabilitation Technology. I am also thankful to Dr. Richelle Koopman for giving me an opportunity to collaborate with an excellent team for her grant at the Medical School.

I am grateful to my parents for providing me means to come to the University of Missouri to pursue graduate studies. I am lucky to have met Laila – the smartest person I know. Special thanks to her for copyediting this dissertation. I am thankful to Tushar for continuing to be an invaluable support. I would like to thank all my friends who made my time at Mizzou delightful.

Lastly – I am fortunate to have found the music of Bob Dylan.

Acknowledgements.....	ii
Abstract.....	v
1. Introduction	1
1.1. Problem Statement	1
1.2. Overview	3
2. Background	10
2.1. In-Home Sensor Monitoring in Eldercare	10
2.1.1. TigerPlace	12
2.2. Routine Modelling	14
2.2.1. Routine Modelling in Eldercare.....	14
2.2.2. Routine Modelling in Other Domains	18
2.3. Explainable AI	20
2.4. Natural Language Generation in Decision Making	28
2.5. Linguistic Protoform Summaries	31
2.5.1. LPS generation.....	34
2.5.2. Comparison of Truth-value Methods.....	43
3. Explainable AI for Dataset Comparison	49
3.1. Dissimilarity between LPS sets.....	50
3.1.1. LPS Sets Dissimilarity Method of (Wilbik, et al. 2014).....	52
3.1.2. Proposed Method	54
3.2. Method Validation	61
3.2.1. Gradual Change in Data.....	62
3.2.2. Correlation between distance measures	65
4. Routine Modelling	69
4.1. Overview	69
4.2. Behavioral Routine.....	73
4.2.1. Using LPSs to Represent Behavioral Patterns	74
4.2.2. Dissimilarity between Daily Behavioral Patterns	84
4.2.3. Prototype Behavioral Routine.....	87
4.2.4. Natural Language Explanation of Routine	92
4.2.5. Changes in Behavioral Routine	98
4.2.6. Retrospective Case Studies	105
4.3. Physiological Routine	134
4.3.1. Using LPSs to Represent Physiological Measurements	134
4.3.2. Prototype Pulse Rate Routine	137
4.3.3. Natural Language Explanation of Routine	139
4.3.1. Changes in Pulse Rate Routine.....	140
4.3.2. Retrospective Case Studies	141
4.4. Conclusions.....	150
5. Linguistic Summarization of In-Home Sensor Data.....	151
5.1. Overview	151
5.2. Existing Infrastructure.....	152
5.2.1. Alerts Based on Significant Changes from Baseline	158
5.2.2. Alerts Based on Hard Limits	158

5.2.3. Gait Alerts	159
5.3. Summarization Process	159
5.3.1. Tuple Summary	160
5.3.2. Trend Summary	164
5.3.3. Simple LPS Summary	171
5.4. Results	175
5.4.1. Validation of the Tuple Summary	175
5.4.2. Validation of the Trend Summary	178
5.4.3. Validation of the Simple LPS Summary	182
5.5. Discussion	185
5.5.1. Case I	188
5.5.2. Case II	189
5.5.3. Case III	190
5.5.4. Case IV	191
5.5.5. Case V	192
5.6. Conclusions & Future Work	193
List of Publications	195
Bibliography	196
VITA	206

Abstract

In the next several decades, the proportion of the elderly population is expected to increase significantly. This has led to various efforts to help live them independently for longer periods of time. Smart homes equipped with sensors provide a potential solution by capturing various behavioral and physiological patterns of the residents. In this work, we develop techniques to model and detect changes in these patterns. The focus is on methods that are explainable in nature and allow for generating natural language descriptions. We propose a comprehensive change description framework that can detect unusual changes in the sensor parameters and describe the data leading to those changes in natural language. An approach that models and detects variations in physiological and behavioral routines of the elderly forms one part of the change description framework. The second part comes from a natural language generation system in which we identify important health-relevant features from the sensor parameters. Throughout this dissertation, we validate the developed techniques using both synthetic and real data obtained from the homes of the elderly living in sensor-equipped facilities. Using multiple real data retrospective case studies, we show that our methods are able to detect variations in the sensor data that are correlated with important health events in the elderly as recorded in their Electronic Health Records.

1. Introduction

1.1. Problem Statement

Smart homes equipped with sensors provide a great opportunity to help older adults live independently. Since sensors are capable of recording the activities of the residents round the clock, they are ideal to monitor their day-to-day routines. A routine is a collection of activities performed in a repetitive fashion, which enables us to conduct our tasks in a structured manner. From physical day-to-day activities, to biological circadian rhythms in the human body, to social events like going to work every day, routines are present everywhere. Routines are a manifestation of us attempting to perform our daily activities in an efficient manner. Therefore, a deviation from the routine is our natural response towards some change in the environment that resulted in the formation of the routine in the first place.

Although there has been a significant amount of work in which sensor-based routines of the elderly have been correlated with changes in their health, most of it has been done with a focus on only the researchers being able to make the correlations. In their current form, the techniques are not designed with the caretakers in mind and require expertise to really make use of them. In order to enable these techniques to have a significant impact on the lives of the elderly, they need to be designed such that they are more suitable to be used by people who take care of the residents on a day-to-day basis. An Explainable AI approach along with Natural Language Generation (NLG) can offer a potential solution by

presenting the information about routine and other sensor parameters in intuitive natural language.

The central theme of this research involves using in-home sensor data to extract information about routines and other important health relevant sensor parameters and expressing them in natural language. To this end, this research attempts to address the following threefold goal:

- Model and explain routines of elderly using in-home sensor data
- Build natural language explanation of changes in routines
- Design an NLG system to supplement description of changes in routine with linguistic summaries of other important health relevant sensor parameters

We propose methods to discover patterns in the data generated from sensors installed in apartments of the elderly living in smart homes. The data is multimodal in nature and encapsulates information about activity patterns, sleeping patterns and patterns of physiological measurements (such as patterns in heart rate and respiration rate) of the elderly. We introduce a routine modelling approach that captures the regular behavioral and physiological patterns in the in-home sensor data. We emphasize that the models produced by our approach are explainable and allow for natural language descriptions. We propose a technique to detect deviations in the modelled routines and express them in natural language. We also develop a method to supplement the explanations of deviation

in routine with summaries of other important health relevant sensor parameters. To this end, we introduce techniques to extract important health relevant features from the sensor parameters. While we validate various parts of our system separately, with the help of retrospective case studies, we envision that our routine modelling and deviation detection method would generate an alarm on detecting significant changes in the routines of elderly residents. The notifications of the alarms would include textual explanations of changes in routine along with natural language descriptions of other important health relevant parameters. We believe that the linguistic nature of the alarms would help their recipients to act on them in a more effective manner.

1.2. Overview

The block diagram in Figure 1.1 shows the connection between different building blocks that are developed in this work. The routine modeling and deviation detection system along

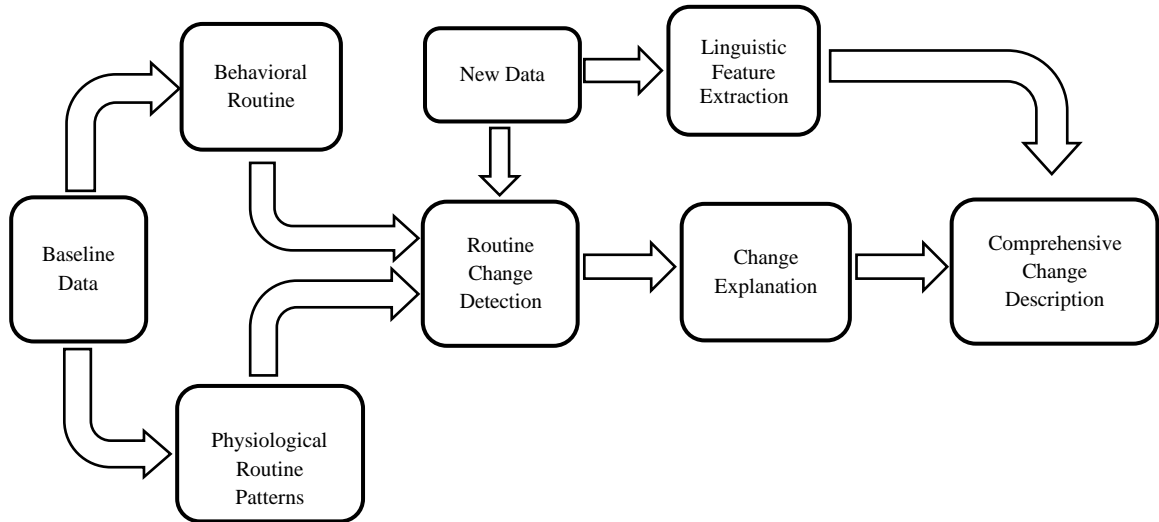


Figure 1.1: Block Diagram of the envisioned Comprehensive Change Description Framework

with the linguistic summarization of various sensor parameters can form a comprehensive textual change description of the in-home sensor data. Although the full potential of our comprehensive change description system would be realized when all the building blocks work in concert as depicted in Figure 1.1, the validation of the individual components is the necessary first step. Accordingly, we separately validate the different components of the comprehensive change description framework with the help of multiple retrospective case studies taken from the sensor-equipped apartments of the elderly.

We model and provide natural language explanations of the routines of the elderly over a baseline time period that may be identified as normal by the nursing staff at the facility where they are residing, or with the help of the Electronic Health Records (EHR). The behavioral routines capture the usual activity patterns of the residents while the physiological rhythm, which we interchangeably call their physiological routine, models their pulse rate and respiration rate patterns. We also develop an explainable change detection method that compares the behavioral and physiological routine models with the data outside the baseline period and quantifies the dissimilarity in numerical as well as linguistic terms. We employ this method to describe the changes in routines in natural language. This forms one part of the comprehensive change description of in-home sensor data mentioned in Figure 1.1. Although identifying changes in routine can provide important insights towards changes in the health of the elderly, the in-home sensor data is believed to contain much richer information that can be of assistance in monitoring the residents' wellbeing. To this end, we design a separate Natural Language Generation

(NLG) system, where we identify important health relevant features present in the sensor data and convert them into intuitive, concise and informative natural language statements.

In Chapter 2, we review the literature related to various aspects of our routine modelling and natural language summarization system. In the first section, we provide the motivation for using in-home sensor technology to help the elderly live independently for longer periods of time. We then provide a detailed review of existing routine modelling techniques concerning in-home sensor data. We also visit some methods that model routines over data obtained from non-smart home settings. One of the most important and distinctive features of our routine modelling approach is that it allows natural language explanations. In the next section, we provide a detailed review of Explainable AI (XAI) and Natural Language Generation (NLG) systems. The central elements of our explainable routine modelling and sensor data summarization system is a fuzzy-logic based technique called Linguistic Protoform Summaries (LPSs). We trace the history of LPSs to their current form. We also list various methods to produce LPSs and use synthetic data to compare the most popular methods relevant to this work.

In Chapter 3, we describe an LPS based explainable dissimilarity measure, which is part of a paper that was presented at the IEEE International Conference on Fuzzy Systems (FUZZ-IEEE), 2019. There, we propose a distance/similarity measure to compare two sets of objects. We first describe the two object sets using LPSs and then use a Fuzzy Inference System (FIS) to compare the LPS sets. Along with the numerical distance, the proposed method also quantifies the differences in the two datasets in linguistic terms, which can be

used to explain why the two sets of objects are dissimilar. We study the properties of our method with the help of synthetic data examples and show that it produces high correlation with the well-known Euclidean distance measure. This explainable distance measure forms the heart of our routine modeling technique, which we describe in the next chapter.

Chapter 4 introduces our LPS based routine modelling approach, which forms the basis of finding behavioral and physiological routines in in-home sensor data. We design various synthetic data experiments to study the various parameter settings of our algorithm. We adapt our explainable distance measure to detect deviations in both behavioral and physiological routines. For behavioral routines, we make use of synthetic data to demonstrate that our algorithm is able to differentiate between different behavioral patterns, while for physiological data, we illustrate our method using real pulse rate data obtained from a sensor equipped apartment of an elderly resident. For both behavioral and physiological data, we follow this with multiple retrospective case studies where we use our method to model routines of the elderly and develop techniques to express it in natural language. We also show that our method is able to detect and provide natural language explanations of changes in the routines that are correlated with the residents' Electronic Health Records.

The first step in determining the model of the routine of a person is to define what constitutes a routine. For the case of elderly, knowing their living pattern is important. By living patterns, we mean how and where they spend their time during the course of a day. To this end, we represent the behavioral routine of the residents by modelling their location

patterns at different times of days. For example, we would like to quantify how much of the morning is spent sleeping or being inside the apartment. Hence, in order to capture the activity patterns, we represent the residents' whereabouts during morning, daytime and nighttime in four activities, namely:

- Inside the bathroom: Time spent in the bathroom
- On the bed: Time spent on bed
- Out of the apartment (OOA): Time spent outside the apartment
- Inside the apartment (INA): Time spent in the living room, kitchen, bedroom

Physiological parameters such as the resting heart rate are also important aspects of a resident's routine. To this end, we incorporate the daily pulse rate and respiration rate measurements of a resident, measured by a hydraulic bed sensor, as part of their physiological rhythm. We model these parameters such that we can assess variations in their measurements. For example, we would like to determine if there are more measurements with a higher pulse rate than usual. Hence, we define the physiological routine of a resident with the following two parameters:

- Instances of pulse rate measurements taken during a day
- Instances of respiration rate measurements taken during a day

In Chapter 5, we present a Natural Language Generation system to supplement changes in routine with linguistic summaries of other important health relevant features from the in-home sensor data generated in the apartments of the elderly. The system described here is part of a manuscript that was published at the *Journal of Biomedical Informatics* (JBI). In

this work, we start by identifying important attributes in the sensor data that are relevant to the health of the elderly. We then develop algorithms to extract these important health related features from the sensor parameters and summarize them in natural language. Such a system will help the clinicians to focus more on making higher-level health related inferences by saving their time and effort in interpreting the numerical data. For example, in our long collaboration with clinicians dealing with the sensor data, we have found that they often seek trends in the sensor streams to determine changes in the health of the elderly. A system that can reliably detect trends and present them in natural language will be beneficial as the time saved in looking for trends in the data would be put to better use in making higher-level inferences. Moreover, since clinical staff turnover in nursing home is high, a linguistic format of the data will help new personnel in interpreting the sensor data. The benefit of such a linguistic summarization system would further increase as in-home technology becomes more pervasive.

To the best of our knowledge, the only system that expressed data generated by in-home sensors in natural language was presented in (Jain and Keller 2015). There, we summarized the state of a resident as the sensor measurements obtained currently, as compared to measurements over the last two weeks. We also described if there was any increasing or decreasing trends leading to the current day. The linguistic alert summarization system described here is an extension of that previous work. While the clinicians in our team appreciated the previous system in that it described the sensor data in natural language, their main critique was that the summaries were too verbose, and they would like them to

be shorter and more informative. Mindful of this, we build the summarization system described in this work from the ground up, with three main objectives. First, we make the summaries more **informative** by including sensor parameters that cover multiple aspects of the health of the elderly. While the earlier work only summarized 3 sensor parameters, here, we process data from 17 sensor parameters. We also extract multiple data features from the sensor parameters to mine different types of information content. Second, we test the **accuracy** of each data feature by conducting multiple surveys with human subjects. Third, we make the summaries **concise** by controlling the amount of information included in the summaries, without removing useful content. To make the summaries easy to read, we carefully format the textual content. This summarization methodology is deployed as part a National Library of Medicine (NLM) funded project (Popescu, NIH-NLM #R01LM012221).

2. Background

2.1. In-Home Sensor Monitoring in Eldercare

In the next several decades, the number of people with age 65 and older is expected to rise significantly in the United States (Ortman, et al.). As the proportion of the elderly population grows, there is an increasing need to find ways to help them age independently (Gross 2007). In the recent past, in-home sensor technology has been shown to have a lot of potential to help the elderly live independently for longer lengths of time by monitoring their health in an effective manner. In (Rantz, et al. 2005), it was postulated that the technology can help older adults live a healthier life for longer durations of time, as illustrated in Figure 2.1. The figure depicts the usual decline of the functional ability of

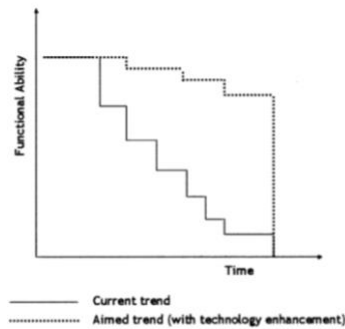


Figure 2.1: Functional ability of older adults VS Time with and without technology enhancement

people as they age. It also shows the trend in decline in the presence of the sensor technology. In the traditional aging model, the functional ability of people drops uniformly as they age, with the flat plateaus followed by major health events which renders them less able than before. The use of technology, as suggested by the aging model in the figure, can

help to reduce the rate of decline in their functional ability, thus, helping them remain healthier for longer durations.

In the in-home sensor technology for aging model, homes of the elderly residents are equipped with sensors to conduct longitudinal monitoring of different health related aspects of their lifestyle. A significant number of projects have employed this model by installing/developing various types of non-intrusive sensors and have successfully made correlations between change in the sensor data and the health of the elderly residents. For instance, increase in bed restlessness has been shown to be linked with pain (Rantz, et al. 2008), change in daily living patterns may suggest changes in mental health (Galambos, et al. 2013), or decrease in the gait speed may be correlated to increase of fall risk in near future (Phillips, et al. 2017). In a Randomized Control Trial (Rantz, et al. 2017), residents with sensor monitoring system were found to have better gait right and other health related scores as compared to the participants with traditional care models. While most of the projects follow a similar general framework of monitoring sensor data longitudinally to look for changes, they differ in terms of the types of sensors being used as well as the information extracted from the obtained data (QuietCare , Abowd, et al. 2002, Intille, et al. 2003, Alwan, et al. 2006, Helal and Chen 2009, Kaye, et al. 2011, Cook, et al. 2013, Popescu 2015).

2.1.1. TigerPlace

TigerPlace is an independent living environment for the elderly, situated in Columbia, Missouri, USA. It was developed as a partnership between the Sinclair School of Nursing at the University of Missouri and Americare (Rantz, et al. 2005). It follows the aging in place model where clinicians are present on site to assist the elderly in living independently for as long as possible. Researchers at the School of Nursing and Electrical & Computer engineering have had a long history of collaboration in using technology to augment the TigerPlace model in supporting the elderly. To this end, apartments at TigerPlace are equipped with sensors to monitor different health related aspects of the elderly. In the following, we provide a brief overview of the different types of sensors installed at TigerPlace.

Although many types of sensor technologies have been employed at TigerPlace over the years, currently, most of the apartments are equipped with three sensor types: motion sensors, bed sensors and depth camera sensors.

- The motion sensors record the activity of the residents around the apartment. They are strategically placed at various locations in order to cover the maximum area. The raw data generated by the motion sensors is processed to log the time stamps and the location of movements occurring in their field of view. The motion data is recorded at a resolution of 7 seconds. That is, for each sensor, the time between consecutive logged time stamps is at least 7 seconds. In most of the apartments, the motion sensors are

placed in the bathroom, bedroom, living room, den, front door etc. Along with the general activity patterns of the residents, data obtained from the motion sensors has been used to recognize activities like the times the resident goes outside the apartment or makes a visit to the bathroom (Wang 2011).

- The bed sensor monitors the quality of sleep of the residents by measuring restlessness and other physiological parameters like the pulse rate and respiration rate, non-intrusively. This is done with the help of a proprietary hydraulic bed sensor designed by researchers at the Center for Eldercare and Rehabilitation Technology (Heise, et al. 2011). The hydraulic sensors are placed underneath the mattress and produces a Ballistic Cardiogram (BCG) of a person lying on the mattress. The BCG signal is a measure of the flow of the blood inside a person's body. In addition to being able to detect the time spent in the bed, the quality of the BCG signal can be used to find the measure of restlessness of a person lying on the bed. Moreover, the BCG signal is processed with the help of various signal-processing algorithms to produce continuous measurements of important physiological parameters, such as the pulse rate and respiration rate (Rosales, et al. 2017).
- The depth cameras at the apartments at TigerPlace assess different parameters associated with the gait of the residents. The depth sensors only record the silhouette of the resident and hence do not present issues concerning privacy of the residents. In most of the apartments, the depth sensors are placed in the living room, and only record videos when there is an activity inside its field of view. The recorded videos are

processed to measure gait parameters such as walking speed, stride time and stride length (Stone and Skubic 2013). The data from the depth camera is also used to generate alarms on detecting a fall in its field of view (Stone and Skubic 2014).

In the last few years, researchers at the University of Missouri have implemented the in-home sensor model of TigerPlace to various sites around Columbia, Missouri. The data used in this research is obtained from these sites. We use the data generated by the three types of sensors mentioned above. We identify important changes in high-level parameters such as the pulse rate, respiration rate, time out of apartment, time in bathroom etc. and, also employ them to make inferences about the behavioral and physiological patterns/routines of the residents.

2.2. Routine Modelling

2.2.1. Routine Modelling in Eldercare

A widely used approach to assess changes in the health of the elderly residents is by using the in-home sensor data to model and detect deviations in their day-to-day routines. Maintaining a stable daily routine has been positively correlated with the well-being of older adults. In a study in (Zisberg, et al. 2010), the sleep quality of older adults was found to improve with performing activities in a routinely manner. In another study, general daily routines were found to help in maintaining well-being in elderly women (Ludwig 1997). The results of these studies have led to the development of a significant number of techniques that use in-home sensors to understand behavioral patterns of the elderly, which

can be considered as their routines. In the majority of the routine modelling techniques, sensor data originating from various sources are fused to obtain a baseline representation of their living patterns. The change in the routine over a time period is then compared to gain insight into changes in the health of the elderly (Cardinaux, et al. 2008, Virone, et al. 2008, Shin, et al. 2011, Wang, et al. 2012, Yin, et al. 2015, Dawadi, et al. 2016, Yefimova 2016, van Kasteren, et al. 2017, White 2018). Next, we briefly describe some of these techniques.

Visualization is a common strategy to study routines of the elderly living in homes equipped with sensors. Yin et al (Yin, et al. 2015), represented routines of residents by their transition patterns inside the apartment and visualized the transition matrices with acyclic graphs to compare routines over different periods of time. In (Yefimova 2016), the author went through the data generated by the bed sensors, motion sensors and the depth sensors, manually, to determine the activities performed by them. The living patterns for a period of time were then presented in patch plots, which are temporal representations of the residents' activity patterns. They found different patterns on weekends as compared to weekdays. In (van Kasteren, et al. 2017), motion sensors and power sensors were used to model the routine of a person. The motion sensors were used to track the residents around the apartments while the power sensors were used to monitor the TV watching activity or use of electric appliances such as microwave or kettle. The activity patterns obtained by the sensors were visualized using what the authors call a radar plot, which is basically the distribution of the activities performed throughout a day, aggregated over multiple days.

The radar plots of a resident over two weeks showed that the plots for the week when the resident was ill depicted less movement inside the apartment as compared to a normal week for the resident. In another case study, the radar plots over two months showed the resident's gradual return to normal routine after the death of their spouse.

In (Virone, et al. 2008), the authors use motion sensor data to assess changes in the circadian rhythms of the elderly residents. They compute the circadian activity patterns using the number of hits that a motion sensor recorded along with the location the hits were produced. Using a visual presentation of their method to represent the behavioral activity, they show that the residents have differing patterns on weekdays and weekends. They also present a mechanism to compute deviation from the circadian activity patterns. A number of case studies were presented to validate the technique.

In (Shin, et al. 2011), the behavioral patterns of the residents were represented by three features: activity level, mobility level and non-response interval. The three features were used to train a Support Vector Machine (SVM) for a time period over an expert labeled data consisting of normal and abnormal days. The behavioral patterns represented by the three features were then classified using the trained SVM model. In order to validate their method, they collected data from apartments of 9 elderly residents over a period of 694 days. Patterns were labeled as normal or abnormal by experts, which formed the training/testing set for the SVM algorithm. Overall, the algorithm produced the sensitivity and specificity of 74.2% and 85.8%, respectively.

A different method was presented in (Wang, et al. 2012) where the motion activity of the residents along with the time away from apartment was represented pictorially, in what is called a motion density map. The authors also introduced a dissimilarity method based on the textural features of the motion density maps to quantitatively compare activities between two or more time periods. Using a case study, it was shown that the method could produce a relatively higher dissimilarity between the baseline and a month before surgery, as compared to a month after the surgery. Another case study was used to show a trend in the dissimilarity from the baseline as the lifestyle of the resident changed over-time. In another work, the motion density maps successfully differentiated patterns of activity of five residents suffering from depression from the time when they didn't have depression (Galambos, et al. 2013).

Recently, in (Ibrahim, et al. 2019), activities of the elderly residents as recorded by the motion and the bed sensors were converted into sequences of varying lengths. These sequences of sensor firings collected over a period of time were used to represent the normal activity pattern of the residents. The authors also defined a method to compute distance between sequences, which was then used to cluster the collection of sequences into multiple groups. The model of sequences built over the normal days was used to flag any abnormal days in the future using streaming clustering.

Dawadi et al. represented the routine of residents in terms of activities of daily living, like: sleep, bed to toilet, cook, eat, personal hygiene, leaving home and so on (Dawadi, et al. 2016). These previously determined activity labels were used to compute the probability

that an activity was performed at a given time of day. The probabilities across multiple days were combined to form what the authors call an aggregated activity curve, which is the representation of the routine of the residents observed over several days. They also define a dissimilarity measure between two activity curves to determine how the current set of activity patterns compare to the routine of the resident. They applied their method to data obtained from 18 residents living in smart homes and computed correlations between two standard clinical scores (TUG and RBANS) and the changes in their routines. A positive correlation was found between the change in routine as measured by their algorithm and the TUG score. However, no significant correlation was obtained with the RBANS score. With the help of two case studies, it was shown that the algorithm detected a change in routine when a resident's health deteriorated over a 24 month period, while no change in routine was detected for a resident whose health did not change significantly over time.

2.2.2. Routine Modelling in Other Domains

The importance of monitoring life routines has been studied in many disciplines other than eldercare. In the seminal work of (Eagle and Pentland 2006), called the Reality Mining project, students and faculty at the MIT Media Lab were given cell phones to record their location patterns and social interactions. A technique called Eigenbehaviors was introduced in (Eagle and Pentland 2009), where this data was used to extract routines of the participants. There, the location activity for each day was represented as vectors and the principal components of the activity over several days was used to represent routines.

The authors were able to approximate the behavior of residents with 90% accuracy using the primary six Eigen vectors. It was postulated that the technique could also be used as a tool to generate synthetic data.

An approach using topic models to represent daily routines was introduced in (Huynh, et al. 2008). There, routines of people were described in terms of fine and course level activities. Activities such as *sitting at desk*, *going to toilet*, *having dinner*, were termed as fine level activities, while activities comprised of multiple fine level activities such as *lunch routine*, *commuting*, were characterized as course activities. The fine level activities formed the words for the topic models and were labeled by a separate activity recognition algorithm. Thirty-minute moving windows over the fine activity labels were considered as the documents. In order to validate the technique, accelerometers were used to label fine level activities performed by a participant over six days. Using these activity labels, they recognized four topics in the data, representing the activities: *dinner*, *commuting*, *lunch* and *office work*. They obtained a mean precision and recall of 86.1% and 67.2% on comparing the results of the algorithm with the manually labeled activities. The topic modelling algorithm provided the probabilities of a topic (high level) given the fine level activities happening inside a 30-minute window. The learned model was applied to successive 30-minute windows over a day to assess the routine of a person. This approach was later used by White (White 2018) to find routines of elderly residents.

Begole et al. (Begole, et al. 2003) describe modelling and visualization techniques to study work rhythms of employees in offices. (Jiang, et al. 2012) use an open source data where

32,366 residents in the Chicago metropolitan region answered questions about their locations on a weekday and a weekend. They used this data to study moving patterns of the residents around the city. They transformed the data into vectors by dividing each day into five-minute intervals and selecting one out of 9 locations for each interval (*work*, *home*, *shop* etc.). They used Principal Component Analysis (PCA) to find the eigen vectors as representative of the daily location activity of the residents. They also performed K-means clustering on the data to group the activity patterns of the participants into a number of clusters. The data was found to contain eight types of routines, representing eight different ways in which the residents in the region spend their weekdays and weekends.

The numerous routine modelling techniques presented in this section are a testament to the importance of studying routines in helping detect changes in health of the elderly. All of the techniques attempt to present routines obtained from sensor data in one visual form or another, while some of them also describe ways to quantify deviations in routine. However, the deviation is always quantified numerically where the explanation of the changes in data that led to the deviation in routine are not readily available. We attempt to address this problem by proposing a routine modelling technique that can model and detect the changes in the routines as well as provide explanations in natural language.

2.3. Explainable AI

As the Artificial Intelligent (AI) systems are becoming exceedingly smart, they are being deployed in our environment at an equally fast pace. If we look around, they are present

everywhere, crunching data generated by the sensors enabling the driving assistance systems to the numerous health tracking devices. While their assistance has been mostly appreciated, they have also drawn some criticisms, especially when these AI enabled systems are used for critical applications such as driverless cars or clinical decision making. This has led to an increased interest in developing systems that are not just black boxes, but methods that can provide an insight into their “thought process”. Explainable AI (XAI) is an emerging area of interest where the focus is on developing models that are interpretable and capable of explaining themselves, which in turn makes them understandable and trustworthy to their users.

Black Box AI models which are solely driven by data without any regard to the implications for decision making can pose serious ethical problems due to the inherent biases that may be present in the data (Mittelstadt, et al. 2016). For instance, in a study conducted to evaluate various machine learning algorithms for applications in health care, it was found that even though a black box model could produce high objective accuracy, the results might not be desirable on ethical grounds (Caruana, et al. 2015). For the task of predicting risk of death in patients of pneumonia based on their previous health data, neural networks out-performed a rule-based algorithm. However, apart from providing the probability of death, the rule-based method also provided correlations present in the data that led to the final outcome. One such ‘rule’ made a connection that the patients who had a history of asthma were less likely to die. On further probing, the reason for this correlation was found to be that the asthma patients had already received good care that decreased

their probability of death. Therefore, even though models like the neural networks may provide results that are more accurate objectively, their black box nature prevents a deep probing into their inner working. Another application that has drawn a lot of focus recently is the deployment of AI for autonomous driving. Unless these algorithms allow a window inside their workings, it would become really difficult to pin-point the exact reason why an autonomous driving vehicle performed a certain action. The current opaqueness of the self-driving systems has made the regulators hesitant to allow them on public streets (Doshi-Velez, et al. 2017).

The increased calls to develop Explainable AI systems is a testament to their importance in modern AI driven lifestyles. One such step is the European Union's General Data Protection Regulation (GDPR) which requires AI systems to be able to explain themselves (Goodman and Flaxman 2016). The XAI initiative of DARPA, which calls researchers to produce more transparent models that are understood and trusted by human users (Gunning 2017), is another evidence of the importance of explainable AI systems.

Although the meteoric rise in the popularity of explainable AI systems is only recent, researchers have been interested in developing XAI techniques for quite some time. These systems have not always been described as explainable but are frequently termed as interpretable, understandable or transparent models. In-fact, in order to review explainable AI literature, researchers often query for papers having all or any of these terms (Doran, et al. 2017, Alonso, et al. 2018). Biran et al. (Biran and Cotton 2017) trace the history of machine learning techniques designed to have explanation capabilities to as early as 1980s.

According to them, the earliest systems to provide explanations of their workings were rule-based expert systems, which later evolved into Bayesian networks. (Lacave and Díez 2002) provides a comprehensive survey of explanation techniques for Bayesian networks. They classify the explanation methods based on their content, communication and the adaptation. The content of the explanation controls what aspects of the network are to be explained, while the communication is the method by which the explanations are conveyed, for example visually or in natural language. The adaptation property drives the level of detail to be included in the explanation of the network based on the level of the user's knowledge. While these properties are only specified for Bayesian networks, they can form a good guiding principal for explanations of AI systems in general.

A machine learning method that has been quite popular traditionally is Support Vector Machines (SVMs). Most commonly, SVMs are used for the task of classifying data into separate classes. Although the objects are assigned to a class based on their distances to a decision boundary, the interpretation of the model requires technical expertise; hence, the results are not readily explainable. An approach to make SVM results more understandable is to use them along with rule extraction techniques (Martens, et al. 2008). In this approach, the data is first relabeled based on the results produced by SVM. This 'cleaned' data is then used to extract rules, which are then said to mimic the hard to decipher SVM model. Such methods where an explainable model is used along with a black box model are termed as model-agnostic interpretable techniques, as the explainable model is not dependent on the real model which is tasked to perform the actual decision making (Ribeiro, et al. 2016).

It has been argued that there is a tradeoff between the interpretability of a model and its capability in handling complex datasets. Therefore, in order to maintain the ability of AI models to perform highly complex tasks, the model agnostic interpretability approach offers a potential solution. However, the explanations produced by model agnostic approaches can be locally inconsistent as these methods are limited by recording the response of the black box model for only a small set of input.

The impressive predictive capabilities of the deep learning models have brought a lot of focus on techniques to make these models interpretable/explainable. The non-linear structure of these models that makes them highly accurate and flexible, also makes it difficult to understand their inner workings for a given task. To interpret the deep learning models, researchers have approached the explainability by producing visual or textual explanations. To name a few, for the task of classifying names of birds from images, (Hendricks, et al. 2016) produced explanations such as *“This is western grebe because this bird has a long white neck, pointy yellow beak and a red eye.”* In another work, the textual explanations like *“This is a healthy meal because it contains a variety of vegetables”* were accompanied by the highlighting the elements of the image that led to this explanations (Huk Park, et al. 2018). A recent survey of the explainability in deep learning models is presented in (Gilpin, et al. 2018).

Fuzzy logic-based models are a class of algorithms that have a high degree of interpretability deep-rooted in their design. Due to the ability of fuzzy sets to represent linguistic variables in mathematical form, they provide opportunities to develop models

that are more natural for human understanding. In the bibliometric analysis of machine learning related literature with a focus on works related to explainable AI, papers related to fuzzy sets were found to be most prolific (Alonso, et al. 2018). In their analysis, the authors found 3000 papers (since the year 2000) containing terms related to explainable AI. Out of these 3000, about 28% were found to also contain terms related to fuzzy sets and systems, which attests the focus of the fuzzy set community towards development of interpretable models.

The most cited work in this regard was found to be a paper that described designing fuzzy inference systems (FIS) from data (Guillaume 2001) for classification and/or data exploration tasks. FIS are fuzzy rule-based systems where the rules can either be designed by experts or learned from data. Fuzzy rules provide a tool to design highly interpretable systems since the rules can be described in intuitive natural language statements. Another popular work which employs fuzzy rules for the task of classification of high dimensional data was presented in (Jin 2000). There, the main focus was on pruning the number of fuzzy rules, since a high dimensional data can result in a very large number of fuzzy rules that makes it very difficult to interpret the learned model.

A fuzzy logic-based system to detect falls from videos was presented in (Anderson, et al. 2009). The activities performed in the video were first described using fuzzy set based linguistic variables. These variables were then used with a fuzzy rule base to detect falls in the video. The authors argued that a system driven by fuzzy logic was better than traditional fall detection activity models in terms of interpretability. The use of linguistic variables

and fuzzy rules also made the system more flexible in that the rules could be easily modified based on the systems performance for a given dataset. Fuzzy clustering algorithms (Bezdek, et al. 1999) are another class of popular fuzzy set based models that provide higher interpretability as compared to their crisp counterpart. With the membership degree of a data point denoting its belongingness in a certain cluster, they provide a better insight into the data.

Linguistic Protoform Summaries (LPSs), have been recently used to produce pattern recognition systems which are capable of producing results which are explainable in textual format (Wilbik and Keller 2013, Wilbik, et al. 2014). LPSs are template-based sentences that employ fuzzy logic to produce natural language summaries of data. In (Wilbik, et al. 2014), LPSs were used to represent baseline patterns of sensor data, which were then used to find anomalies in data at some later point in time. The representation of data using LPSs readily provided linguistic explanations behind the anomalies, which makes this system inherently explainable.

The large array of approaches to design systems that allow a peek inside their workings has made it difficult to have a widely agreed global definition of explainable AI. Doran et. Al (Doran, et al. 2017) performed an analysis of the machine learning literature to define different types of AI systems with respect to their explaining capabilities. They classify the works in the AI related literature into three types: opaque, interpretable and comprehensible systems. An opaque system is one where the model does not allow the user to understand how the inputs are processed to produce a certain output. For example, an opaque

classification system would not provide any information other than the class a certain input belongs to. On the other hand, interpretable models are transparent in the sense that a user could mathematically map the inputs to outputs. These types of models require the users to possess an expertise to understand the underlying mathematical model of the AI system. Comprehensible systems provide a visual or a textual explanation of why a certain input was mapped to an output. However, in a comprehensible system, the actual AI model need not be transparent. Deep neural networks fall under the category of opaque systems while SVM, data clustering, decision trees, Bayesian networks etc. are common examples of interpretable models. Models that explain the results of deep learning models visually or in textual form are comprehensible systems. Methods based on fuzzy sets can potentially have both the interpretability and the comprehensibility property. To illustrate the classification of the XAI systems into the three aforementioned categories, the authors of (Doran, et al. 2017) describe the communication between a doctor and a patient as a comprehensible system where the doctor provide the patient with a high-level explanation of why their symptoms led to a certain diagnosis. On the other hand, the communication among doctors is analogous to an interpretable system where the details of the diagnosis are discussed on a technical level.

Due to the critical nature of clinical applications, it is imperative that the AI models designed to make decisions based on health care data needs to be explainable in nature (Caruana, et al. 2015, Holzinger, et al. 2017, Ahmad, et al. 2018). Using the classification of AI systems of Doran et al. (Doran, et al. 2017), the models need to be interpretable as

well as comprehensible. The high cost of inaccurate diagnosis in the clinical domain, makes it essential for the models to be transparent in nature. Moreover, the models need to be interpretable in order to be trustworthy by the health care personnel using them (Ahmad, et al. 2018). On the other hand, the comprehensibility property of the models is required for them to be used to the fullest by both clinicians and the patients.

The explainable models could either use visual or textual representations to express the reasoning behind their results. Visualization techniques are able to explain important details of the data, but they require some amount of expertise and a learning curve to be fully useful. Moreover, due to the short amount of time available to the clinical personnel to review a patient's records, visualization techniques are not ideal in the healthcare domain. Natural language representations of these models could provide an intuitive and efficient way to understand the results of the comprehensible models. Although the idea of using Natural Language Generation (NLG) as a communication agent for intelligent systems is still quite new, it is believed to be a key element of the explainable AI framework (Biran and McKeown 2015, Biran and Cotton 2017).

2.4. Natural Language Generation in Decision Making

With the advent of big data, the field of Natural Language Generation (NLG) has seen a significant amount of work in the last two decades. NLG techniques are designed to extract important information from different forms of data and express that in natural language. Traditionally the de-facto method to understand data has been to employ visualization

techniques that present a coherent picture of important content in the data. Visualization makes it easier to make sense of the data and draw important observations from it. However, this becomes less feasible as the data becomes more complex and ‘big’. Visualization comes with its own set of shortcomings. First, graph literacy is not uniform across different groups of people. When presented with simple graphs, the ability to extract relevant information from the graphs varies among people based on levels of education, familiarity with the underlying data in the graphs (Friel, et al. 2001) or even culture (Galesic and Garcia-Retamero 2011). Second, most of the visualization techniques involve selection of parameters like axis scale, size of the graphs, color etc. which makes it difficult to objectively interpret graphs even for experts in their respective domains. Moreover, natural language has been shown to be helpful in interpreting numerical data and improve decision making in the presence of uncertain information (Gkatzia, et al. 2016).

In the recent past, NLG technology has proven its utility in numerous applications ranging from weather reporting to health care data summarization (Gatt and Krahmer 2018). One of the earliest applications where NLG was postulated to have significant impact was in health care (Cawsey, et al. 1997). Since then, a lot of work has been done to help health care processes by summarizing their data in natural language. In (Gatt, et al. 2009), nursing discharge summaries were automatically generated from sensor data in Neonatal Intensive Care Units (NICUs). The summaries produced by the NLG system were found to be comparable to the ones produced by nurses. Hospital records of patients were converted into reports in natural language in (Scott, et al. 2013). The system was designed so that the

summaries could be customized in terms of the content based on the audience. The system was evaluated for accuracy and efficiency by presenting health care personnel with the auto generated summaries and summaries produced by humans. While the auto-generated summaries were found to be on par with the human written summaries, the doctors found them more efficient in terms of the time it took to go through them. (Di Eugenio, et al. 2014) designed an NLG system to combine physician discharge summaries and nursing documentation. The discrepancies between the discharge summaries and information logged by the nurses was the motivation for this work. (Goldstein and Shahar 2016) designed a system called CliniText, which summarized multivariate numeric and linguistic data generated in a cardiac ICU during a patient's stay. Although the over-all quality of the clinician written summaries were superior to the auto generated summarizations, the CliniText system was found to be better than the summarizations produced by clinicians in terms of the completeness of the content and extraction of information.

The continuous streams of data generated by the sensors monitoring different types of processes has attracted NLG techniques towards these applications. In (Yu, et al. 2007), large amounts of data generated by sensors monitoring the functioning of gas turbines was textually described to aid engineers in maintaining different part of the turbine system. There, corpus of text written by experts to describe the time series were used to design various features of the NLG system. The time series were summarized in a hierarchical fashion by focusing on the larger picture and then drilling down into finer patterns of the temporal data. This NLG architecture was also used in (Sripada, et al. 2003) to produce

weather forecasts. In (Banaee, et al. 2013), physiological data comprising of respiration and heart rate measurements were obtained from wearable sensors. Features such as trends, along with basic statistical parameters, were used to summarize the information content of the data, which were then expressed in natural language. In (Alexander, et al. 2014), Linguistic Protoform Summaries (LPSs) were shown to be successful in correlating in-home sensor data with changes in health. The output of this system expressed the changes in sensor patterns in linguistic terms. However, the language of the output was not very intuitive. To the best of our knowledge, the only system that expressed data generated by in-home sensors in natural language was presented in (Jain and Keller 2015). There, we used LPSs to summarize the state of a resident as the sensor measurements obtained currently, as compared to measurements over the last two weeks. We also described if there was any increasing or decreasing trends leading to the current day.

2.5. Linguistic Protoform Summaries

Linguistic Protoform Summaries (LPSs) are template-based natural language statements that summarize data in linguistic terms. They are capable of quantifying one or more properties of data. For instance, LPSs can be used to generate statements like: ***Most** of the students in class are **tall***, summarizing the heights of collection of students, or weather during a month: ***Few** of the **cold** days in this month were **windy***. The use of Fuzzy Sets to represent terms like most, few, tall, cold etc. gives LPSs the ability to produce ‘human’ like natural language sentences.

In the fuzzy set literature, the term linguistic summary is a general term that encompasses many different techniques to produce textual output. An in-depth review about the different types and applications of linguistic summaries can be found in (Boran, et al. 2016). Linguistic Protoform Summaries are fuzzy quantified sentences and are just one of the types of linguistic summaries that employ fuzzy sets. LPSs were introduced by Yager (Yager 1982) to summarize numerical data in easy to understand natural language like sentences. In comparison to the traditional data summarization operators like mean, median, and standard deviation, LPSs allow for a more informative, but still concise, representation of data. The first use of the term protoform (short for prototypical form) to describe these fuzzy quantified sentences was done by Zadeh (Zadeh 2002) in 2002. The protoform is an abstracted form of a quantified sentence. For example, the protoform of the proposition “Most students are tall” is “Q y’s are A” where Q is a fuzzy quantifier and A is the property of objects, y, being summarized.

Since the proposal of Yager in 1982, Linguistic Protoform Summaries have seen a lot of work in terms of sentence templates (protoforms) as well as in various application domains. Among the different applications, linguistic summarization of time series has especially seen a lot of attention. For instance, (Kacprzyk, et al. 2010) segment the time-series comprising of mutual fund data and relate different features in the segments by the use of fuzzy quantifiers. Features such as trends, variability and duration are extracted from the time-series segments to produce summaries like: *Most of slowly increasing trends are of a medium length*. In another work, Sanchez et al (Sanchez-Valdes, et al. 2016) make use of

fuzzy quantifiers to produce a hierarchical system describing daily/weekly patterns of activities inferred from data recorded by smart phones. The smart phone sensor derived short terms activity patterns like *walking, running, idling* are aggregated to denote a day as *sedentary, active* etc. These daily patterns are in turn accumulated to describe the weekly activity pattern of a user. The LPS based hierarchical scheme is used to describe a week as “*In summary, this week the user had a moderate lifestyle, because all the days the lifestyle was moderate.*” In (Ramos-Soto, et al. 2015), LPSs were used along with a template based NLG system to produce linguistic descriptions of short term weather forecasts. Wilbik et al. (Wilbik and Dijkman 2016) employ LPSs to summarize process data, which is comprised of sequences of activities performed in a process. To keep the description provided by the LPSs short, they use a set of rules to determine LPSs that convey similar information. In (Wilbik, et al. 2011), LPSs were used to summarize activities of elderly residents using the data obtained from their apartments that were equipped with sensors. There, the amount of activity performed over a time period was summarized by LPSs of the form: *On most of the nights the resident had a medium level of motion.* With the help of a case study, they pointed out the existence or absence of summaries when the person was having health issues.

Recognizing the potential of LPSs to summarize numerical data, authors in (Wilbik and Keller 2012) introduced a distance measure over the space of LPSs which enabled their use as features of data. This allowed the authors to compare two object sets by first summarizing them using LPSs and then quantifying the difference between the two sets

numerically. This approach was used in (Wilbik and Keller 2013) to compare nightly sleeping and movement patterns in sensor data obtained from homes of the elderly. The technique was extended in (Wilbik, et al. 2014) to cluster LPSs generated from sensor data over a certain time period. A similar approach was used to compare patterns of activity of two elderly residents suffering from depression (Jain, et al. 2016). There, we focused on the out-of-apartment activity of the residents during a month when they were suffering from depression, a month before, and a month after taking depression medication. The use of LPS allowed us to linguistically compare the patterns of out of apartment activity during the three time periods. We were also able to show that, quantitatively, the LPS representation of the out of apartment activity prior to the onset of depression were closer to the month after the medication use than the month when the residents were suffering from depression. In (Jain, et al. 2017), we used LPSs as input to fuzzy rules to produce a decision support system for in-home blood pressure data. We showed that the DSS based on LPSs was able to match judgements of three physicians with high accuracy.

2.5.1. LPS generation

The central task in using Linguistic Protoform Summaries to summarize numerical data is to measure their validity based on the underlying information. This assessment of LPSs that takes into account the fuzzy set representation of terms forming the LPSs is itself a fuzzy value indicating the validity of a given LPS corresponding to the dataset being summarized. Since the seminal work of Yager (Yager 1982), a substantial number of LPS evaluation techniques have been proposed in the literature. Although there are many

metrics measuring different aspects of the LPS (Wilbik 2010), the most widely studied is called the degree of truth or the truth-value metric, which gives the validity of the LPS in terms of how true it is with respect to the corresponding data. Sanchez et al. (Delgado, et al. 2014) provides a comprehensive review of different approaches to compute the truth-value in relation to the format of the LPSs and the design of fuzzy sets representing linguistic terms. In the following, we describe a few of these techniques concerning this work.

At the very basic level, LPSs or fuzzy quantification is nothing but quantifying one or more properties of data in terms of the fuzzy sets representing the quantifier and the said properties. With this definition in mind, the two most widely used LPS protoforms are called: Simple and Extended protoforms (Wilbik 2010) or type I and type II protoforms (Delgado, et al. 2014). An example of a type I LPS is *Few days were cold*, which, in general, can be represented by the protoform: *Q y's are P* where *Q* and *P* denote the quantifiers and the summarizers, respectively. A type II LPS can be illustrated by: *Most of the windy days were cold* where the *windy days* in the above example are used to qualify the type I LPS expression. Type II LPS are represented by protoforms of the form: *Q R y's are P* where the terms *Q* and *P* represent the quantifiers and summarizers as in type I LPS, while the term *R* is aptly called the qualifier. Another component that plays a key role in the computation of the truth-values of LPSs is the definition of the fuzzy set representing the quantifiers. As noted in (Delgado, et al. 2014), the quantifiers can be defined in two ways according to the semantic meaning of the linguistic term being

represented: *coherent* and *non-coherent* quantifiers. Coherent quantifiers are defined by monotonically non-decreasing membership functions and are used to represent terms like: *almost all, at least a few, at least some* etc. The non-coherent quantifiers represent terms like: *a few, about 25%* and so on and can be modelled by monotonically non-increasing or non-monotonic membership functions. We are mostly interested in linguistic terms modeled by non-coherent quantifiers, as they are more applicable in producing statements that are natural in communication.

The quantifiers can also be classified as relative or absolute in terms of the represented quantity. Quantifiers like *few, some, many, about 25%* are deemed as relative quantifiers, as they are quantifying a proportion of the given objects. Absolute quantifiers, as their name suggests, are used to represent the absolute quantity of objects being quantified, for example, *about 10, exists, all* etc. In the following, we describe the various truth-value computation techniques available in the literature with a focus on methods for LPSs comprising of non-coherent, relative quantifiers.

The truth-value computation technique employed in the seminal paper of Yager is arguably the most widely used in the LPS literature for both type I and type II LPSs. This technique is based on works of Zadeh; hence, it is often called the Zadeh's method of truth-value computation. Although it performs as expected for coherent quantifiers, it has been shown to produce non-intuitive truth-values for non-coherent quantifiers like *few*. In (Wilbik, et al. 2015), the authors showed that even when all the objects being summarized have a small

belongingness to a certain property, Zadeh's method produced a high truth-value for an LPS comprising of a non-coherent quantifier (*a few*).

Many truth-value techniques presented in the literature are only applicable to LPS composed of coherent quantifiers. Yager's ordered weighted average has been used to compute truth-value for coherent quantifiers (Delgado, et al. 2014). The truth-value is basically the weighted average of the memberships of the objects in the property being quantified. The weights are obtained using the membership function of the quantifier in the LPS. The OWA based method is equivalent to the truth-value computation method based on Choquet Integral (Delgado, et al. 2014), where the membership function of the quantifier forms the fuzzy measure which is used to aggregate the memberships of the objects in the given property.

Similar to the use of Choquet integral, Sugeno integral has also been used to compute truth-value of the LPSs. Initially, the method involving Sugeno integral was only defined for coherent quantifiers (Kacprzyk, et al. 2006). However, it was later extended for non-coherent quantifiers (Wilbik, et al. 2015). We present this method in detail in the following. Note that although most of the truth-value computation methods described next are applicable to both absolute and relative quantifier, we only focus on relative quantifiers as they are the most relevant to our work.

2.5.1.1. Truth-Value computation using Sugeno Integral

Consider the type I LPS shown in Equation 2.1, where Q and P are the quantifiers and the summarizers, respectively.

$$Q \text{ } y's \text{ are } P \quad (2.1)$$

The Sugeno integral formulation of the truth-value computation of this LPS is presented in Equation 2.2.

$$T(Q \text{ } y's \text{ are } P) = \max_{\alpha \in unique(P(y))} (\alpha \wedge Q(P_\alpha)) \quad (2.2)$$

Here, \wedge is the min operator, $P(y)$ is the membership function of the summarizer P , $P_\alpha = \frac{|\{y_i \in Y | P(y_i) \geq \alpha\}|}{N}$ is the proportion of objects whose membership $P(y)$ is greater than or equal to α , $unique(P(y))$ is the set of α cuts of $P(y)$, $||$ denotes the cardinality of a set. $Q(x)$ is a normal, convex and monotonically non-decreasing membership function of the quantifier Q . This method was adapted in (Wilbik, et al. 2015), to compute truth-values of LPS comprising of quantifiers having non-coherent (non-monotonic) membership functions. There, the non-coherent quantifiers, $Q(x)$ are split into two monotonically non-decreasing functions, $Q_1(x)$ and $\overline{Q_2}(x)$ (which is the inverse of $Q_2(x)$) and the truth-value is computed as shown in Equation 2.3. Since both Q_1 and $\overline{Q_2}$ are both monotonically non-decreasing quantifiers, the truth-values of the corresponding LPS can be computed using Equation 2.2.

$$T(Q \text{ } y's \text{ are } P) = T(Q_1 \text{ } y's \text{ are } P) - (1 - T(\overline{Q_2} \text{ } y's \text{ are } P)) \quad (2.3)$$

Although, this method provided a way to compute truth-values of LPS comprising of non-coherent quantifiers, it still produced non-intuitive truth-values in some cases. To be specific, the truth-values computed by this method were not semantically ordered with the quantifier definitions. This was addressed in (Jain and Keller 2015) by replacing the difference operator with a *min* operator in Equation 2.3, as shown in Equation 2.4. There, we proved that the truth-values computed by this method follows the semantic order of the quantifiers, which makes it more intuitive to use LPSs as a linguistic representation of numeric data.

$$T(Q \text{ } y's \text{ are } P) = T(Q_1 \text{ } y's \text{ are } P) \wedge (1 - T(\overline{Q_2} \text{ } y's \text{ are } P)) \quad (2.4)$$

In that work, we also adapted the Sugeno integral method for type II summaries of the form $Q \text{ } R \text{ } y's \text{ are } P$, as shown in Equation 2.5.

$$T(Q \text{ } R \text{ } y's \text{ are } P) = \max_{\beta \in \text{unique}(R(y_i))} \beta \wedge \left(\max_{\alpha \in \text{unique}(P^{R_\beta}(y_i))} (\alpha \wedge Q(P_\alpha^{R_\beta})) \right) \quad (2.5)$$

where, $R_\beta = \{y_i \in Y | R(y_i) \geq \beta\}$

$$P^{R_\beta} = \{P(y_i) | y_i \in R_\beta\}$$

$$P_\alpha^{R_\beta} = \frac{|\{y_i \in R_\beta | P(y_i) \geq \alpha\}|}{|R_\beta|}, \text{ for } |R_\beta| > 0$$

For the case of type II summaries, we take beta cuts of the qualifier data. For each of the objects falling in the current beta cut, we follow the procedure for the simple protoforms.

The intuition behind this is that when computing truth-values of summaries like *Some of the big balls are heavy*, we should only focus on the balls that are *big* under certain condition, which is the beta cut in Equation 2.5.

2.5.1.2. Truth-Value computation using Fuzzy Cardinality

A family of truth-value computation techniques that are applicable to non-coherent quantifiers was presented in (Delgado, et al. 2000). For type I LPS of the form *Q y's are P*, these methods obtain the truth-value by computing the degree of compatibility between the cardinality of the fuzzy set, *P* and the quantifier, *Q*, as depicted in Equation 2.6.

$$\bigoplus_{i \in \{0, \dots, n\}} (Q(i) \otimes C(P, i)) \quad (2.6)$$

Here, $Q(i)$ and $C(P, i)$ are the values of the quantifier and the cardinality of the summarizer at some point i . \oplus and \otimes are the t-conorm (for ex. max operator) and the t-norm (for ex. min operator), respectively,

Similarly, for type II sentences, of the form *Q R y's are P*, the degree of compatibility is given by Equation 2.7. In this case, the truth-value is evaluated by computing the compatibility between relative cardinality of the fuzzy set *P* with respect to the qualifier, *R*, and the quantifier, *Q*.

$$\bigoplus_{c \in \{0, \dots, n\}} (Q(c) \otimes C(P/R, c)) \quad (2.7)$$

Using this model of evaluating the degree of compatibility, several truth-value computation equations were derived, which differ by the methods to obtain the cardinality of the fuzzy sets and the choice of t-norm and t-conorm.

The cardinality of a crisp set $A = \{1,4,7\}$ is simply the number of elements in the set. However, the computation of cardinality is not so straightforward for the case of fuzzy sets where each element of the set belongs to it with a certain degree. For example: consider the fuzzy set $A = \{0.8/x_1, 0.66/x_2, 0.58/x_3, 0.54/x_4, 0.43/x_5, 0.4/x_6\}$. The cardinality of a fuzzy set can be a scalar quantity, or it can be a fuzzy set itself (Zadeh 1983). For example, the power of the fuzzy set A , calculated by $\sum A(x_i) = 3.41$, is an example of a scalar method where the cardinality is a single crisp number. Delgado et al. (Delgado, et al. 2000) argue that although using a scalar cardinality of a fuzzy set for the task of truth-value computation is simpler, useful information is lost in the process. To this end, they list a number of fuzzy cardinality methods, which are then used to obtain various truth-value techniques for type I and II LPS. Recognizing their ability to be compatible with non-coherent quantifiers, we provide a detailed description of these methods in the following.

- GD method for type I LPS

$$T_{GD}(Q \text{ y's are } P) = \sum_{i=0}^N ED(P, i) \times Q\left(\frac{i}{N}\right) \quad (2.8)$$

Here, ED is the fuzzy cardinality of the fuzzy set P and is given by:

$$ED(P, k) = b_k - b_{k+1} \quad (2.9)$$

with $b_0 = 1$ and $b_{n+1} = 0$

where, b_k is the k^{th} largest value of membership of the fuzzy set P .

- ZS method for type I LPS

$$T_{ZS}(Q \text{ y's are } P) = \max_{k \in \{0, \dots, n\}} \left(\min(Z(P, k), Q\left(\frac{k}{n}\right)) \right) \quad (2.10)$$

Here, Z is the Zadeh fuzzy cardinality (Zadeh 1983), which is defined as in Equation 2.11.

$$Z(A, k) = \begin{cases} 0 & \text{if does not exist } \alpha \mid |A_\alpha| = k \\ \sup\{\alpha \mid |A_\alpha| = k\} & \text{otherwise} \end{cases} \quad (2.11)$$

The *GD* and *ZD* methods have been extended to compute the truth-value of type II LPSs.

The only difference between the truth-value computation of the type I and type II LPSs is process of computing the cardinality. For the type II sentences, the cardinality of the fuzzy set P is computed with respect to the fuzzy set R .

- GD method for type II LPS

$$T_{GD}(Q \text{ R y's are } P) = \sum_{c \in CR(P/R)} ER(P/R, c) \times Q(c) \quad (2.12)$$

The relative cardinality, $ER(P/R)$ is computed by:

$$ER(P/R, c) = \sum_{\forall \alpha_i \mid CR(P/R)=c} \alpha_i - \alpha_{i+1} \quad \forall c \in CR(P/R) \quad (2.13)$$

$$CR(P/R) = \left\{ \frac{|(P \cap R)_\alpha|}{|R_\alpha|} \mid \alpha \in M(P/R) \right\} \quad (2.14)$$

Where $M(P) = \{\alpha \in (0,1): \exists x_i \in X \text{ such that } P(x_i) = \alpha\}$ that is, $M(P)$ is the set representatives of alpha cuts of the set P with $1 = \alpha_1 < \alpha_2 < \dots < \alpha_n < \alpha_{n+1}$. $M(P/R) = M(P \cap R) \cup M(R)$ is the set of representative alpha cuts of the set obtained by the union of the set $P \cap R$ and set R .

- ZS method for type II LPS

$$T_{ZS}(Q \text{ R } y's \text{ are } P) = \max_{c \in CR(P/R)} (\min(ES(P/R, c), Q(c))) \quad (2.15)$$

$$ES(P/R, c) = \max_{\forall \alpha_i | CR(P/R)=c} \alpha_i \quad \forall c \in CR(P/R) \quad (2.16)$$

That is, the relative cardinality of set P with respect to R , at each c is the highest α value which resulted in c .

Both the GD and ZS methods for type II LPS require the set R to be a normal fuzzy set ($\max(R(y_i)) = 1$). Therefore, before we begin the process, the method requires for the set R to be normalized, and the use of the same scaling factor to scale the fuzzy set $P \cap R$.

2.5.2. Comparison of Truth-value Methods

The existence of various truth-value computation techniques brings forth the question of how these methods compare with each other. The authors in (Delgado, et al. 2014) study the evaluation techniques for type I and type II sentences based on a set of properties like time complexity and their ability to produce fuzzy assessment, among others. Although the behavior of the computation techniques with respect to the theoretical properties help in having a better understanding, they do not provide a practical comparison. To this end, we

take a different approach towards studying these truth-value computation techniques by observing their response with respect to data. We design a set of synthetic data experiments to see whether these methods produce intuitive results.

2.5.2.1. Type I LPSs

Consider the five quantifiers shown in Figure 2.2 and a bag containing 100 balls where for each ball, we have its degree in bigness (between 0 and 1). We summarize the size of the balls in the bag by computing truth-value of LPS of the form: Q balls are big with Q being the quantifiers of Figure 2.2. In other words, the bigness of balls in the bag are summarized by an LPS set with one LPS for each of the five quantifiers. We start the experiment with all the balls having a small membership in the property bigness. That is, all the 100 balls are relatively small. Now we replace one small ball at a time with a big ball until all the balls in the bag have a high membership in bigness. At each change of the ball, we compute the truth of the LPS set summarizing the contents of the current bag.

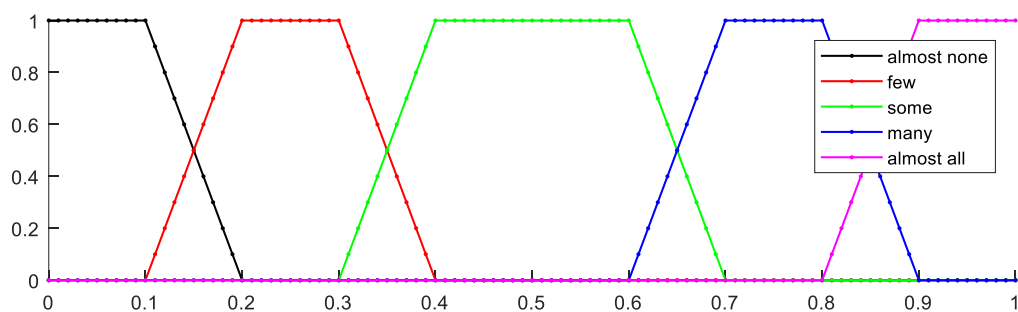


Figure 2.2: Membership functions of Quantifiers

Figure 2.3 compares three methods based on the variation of truth-values of all the five LPS in the set, quantifying the number of big balls in the bag at each stage.

For the Sugeno and Delgado GD method, we see that at the beginning when the bag contains all relatively small balls, the truth-value of the LPS with the quantifier *almost none* is the highest, and as the smaller balls are replaced with big ones, the truth-value with *almost none* drops and *few* increases. At the middle of the experiment, when around half of the balls are big, the truth-value with the quantifier *some* is highest which is then followed by truth-value with *many* and which then finally gives way to the truth-value of

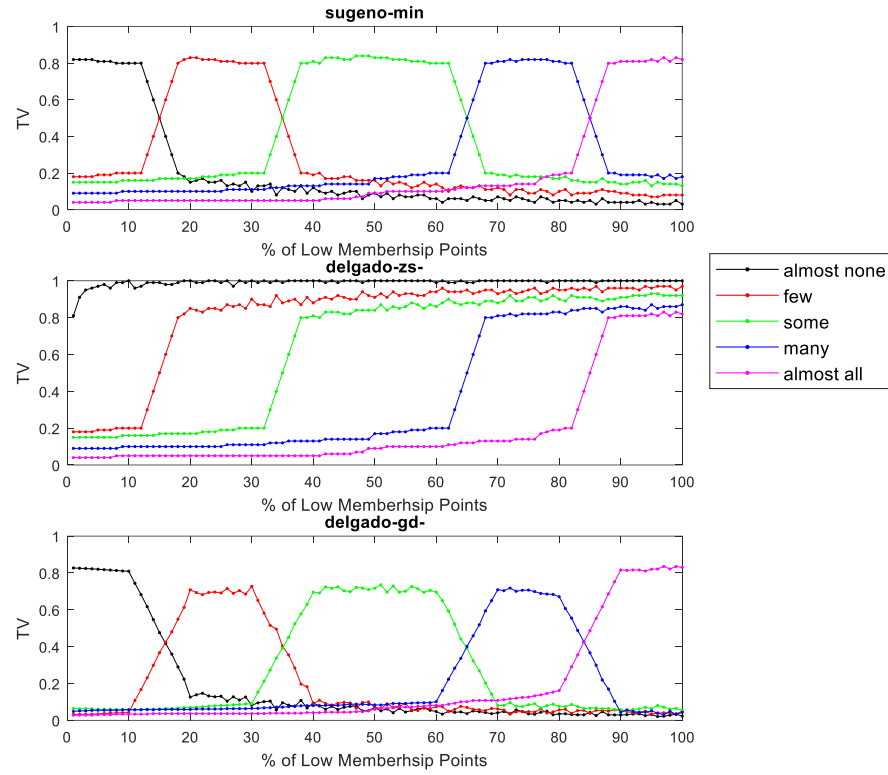


Figure 2.3: Variation in truth-value of type I LPSs of the form: Q balls are big as the size of balls in the bag changes.

almost all being highest. The ZS method does not produce this intuitive variation in truth-values in response to the contents of the bag. Hence, with the results of this experiment,

we conclude that both the Sugeno and Delgado-GD truth-value computation methods produce equally acceptable values throughout the course of the experiment described above. However, as mentioned before, the Sugeno integral based method has an advantage in that it produces semantically ordered truth-values, which is not true for the Delgado GD method. Hence, in the rest of this work, we employ the Sugeno integral based method to compute the truth-values of type I LPSs.

2.5.2.2. *Type II LPSs*

Similar to the type I LPSs, we design a synthetic data experiment to test the truth-value computation methods for type II LPSs. Again, consider a bag of 100 balls and suppose we want to summarize the size of the balls in this bag relative to their color by sentences of the form: *Q of the red balls are big*. Here, the features red and big are fuzzy and form the qualifier (R) and the summarizer (P) of the type II summaries of the form: *Q R y's are P*. At the beginning of the experiment, we start with a bag with 100 ‘small’ and ‘red’ balls. Now similar to the pervious experiment, we replace the small red balls with big red balls, one by one until all the balls in the bag are big and red. That is, we keep the redness property of the balls fixed and just change the size of the balls. At each step, we compute the truth-value of the LPSs of the form *Q of the red balls are big* using the quantifiers of Figure 2.2. The variation of truth-values comparing the Sugeno integral method and the two Delgado et al. methods, GD and ZS are shown in Figure 2.4. We expect a gradual variation in truth-value as we change the size of balls in the bag, starting from the truth-values depicting that the bag contains no red balls that are really big, while at the other side of the

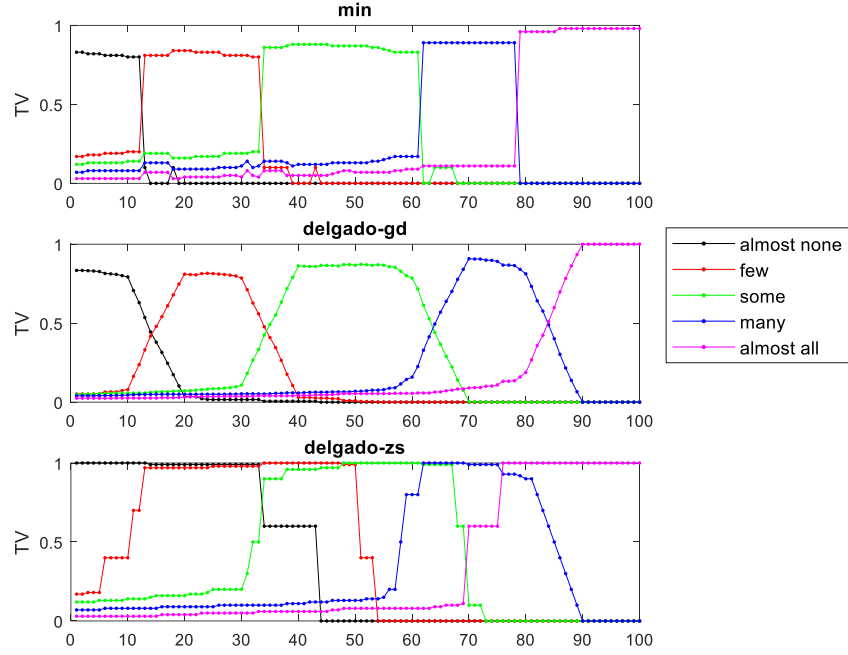


Figure 2.4: Variation in truth-values of the form: Q of the red balls are *big* as the size of the balls in the bag changes, with all the balls having high degree in the redness property.

experiment, the truth-values should suggest that almost all the red balls in the bag are big. We see that with all the three methods, the truth-values convey the contents of the bags correctly at the start and end of the experiment. Also, with all the methods, the LPS with the highest truth-value switches sequentially from the one with quantifier *almost none* to the LPS with the quantifier *almost all*. However, with the Sugeno and Delgado-ZS method, the variation is not gradual. In the Sugeno integral based method, the truth-value drops abruptly from a higher value to a very lower value. For example, when about 10 percent of the red balls in the bag are *big*, the truth-value of the LPS with quantifier *almost none* is quite high at one point and suddenly drops to a very low value with just change of one small ball to a big ball, which is very non-intuitive. Also, with the Delgado-ZS method, the

truth-value variation looks very non-intuitive relative to the other two methods. Contrary to these two methods, the Delgado-GD method produces truth-values that vary gradually as the size of the balls in the bag changes. We can see that, as with the type I LPS, the truth-values follow the pattern of the quantifier membership functions, which is what we expect. Therefore, due to the intuitive and gradual variation in the truth-values computed by the Delgado-GD method, for our purposes, we deem it best among the available truth-value computation methods for type II LPSs. Hence, in the rest of this work, we use it for computation of truth-values of type II LPSs.

3. Explainable AI for Dataset Comparison

One of the central elements of any pattern recognition task is the distance/similarity measure. Numerous distance measure techniques have been proposed in the literature. Most of the time, the choice of a distance measure is made based on the application concerned. Perhaps the most well-known, and well-used, of these distances is the Euclidean distance. A survey of various distance measures available is presented in (Cha 2007). Specifically, they use synthetic data to study techniques that find distance between probability density functions (pdfs). In these methods, histograms of sets of objects are converted to pdfs, which are then compared to get a measure of dissimilarity between the object sets. Histograms are one of the most common ways to visualize and analyze the distribution of sets of objects. Almost all of the available methods to quantify differences between histograms provide a number that gives an estimate of how similar or dissimilar two given histograms are. However, in some applications a method that provides the comparison between sets of objects in linguistic terms can be more beneficial.

We make use of Linguistic Protoform Summaries in tandem with Fuzzy Rules to design a system that can compare datasets numerically, as well as explain the difference in linguistic terms. We validate our method with the help of synthetic data and show that it produces high correlation with the well-known Euclidean distance measure.

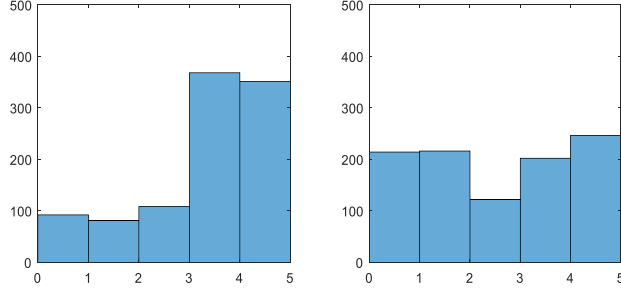


Figure 3.1 Distribution of size of balls in two bags

3.1. Dissimilarity between LPS sets

Consider two bags of balls, with each containing 1000 balls of varying sizes. The distribution of the size of balls in the two bags is shown in Figure 3.1. To summarize the balls in the two bags using Linguistic Protoform Summaries, we define a Protoform of the form *Q of the balls in the bag are P*, where *Q* and *P* are the quantifier and the summarizer, respectively. Suppose we want to partition the size of the balls with the linguistic terms: *small* and *big*, and quantify the number of balls by linguistic variables: *almost none*, *few*, *some*, *many* and *almost all*. Figure 3.2 and Figure 3.3 show the fuzzy set definitions of the aforementioned linguistic terms. Using these definitions, we compute the truth-value of all the LPSs comprising of all combinations of the quantifiers and summarizers. That is, for 5 quantifiers and 2 summarizers, we get a total of 10 LPSs. The truth-value of the summaries representing both the bags are shown in Table 3.1. If we compare the truth-values for the two bags, we can infer that bag 1 has a higher number of bigger balls compared to bag 2, which has a higher number of the smaller balls. Figure 3.1 also conveys this information.

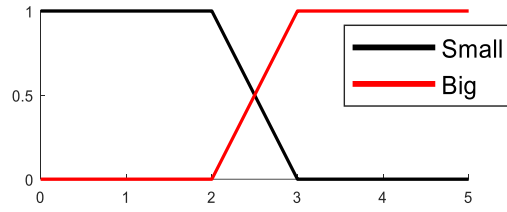


Figure 3.2: Membership functions of the summarizers representing size of balls

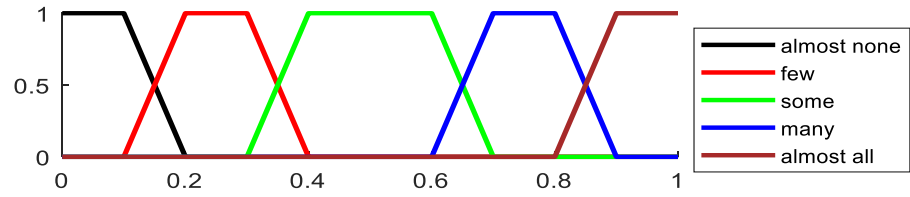


Figure 3.3 Membership functions of the Quantifiers

Table 3.1: Truth-values of the LPSs representing the two bags of balls

Quantifiers	Summarizers	Bag 1 Truth-value	Bag 2 Truth-value	Difference (Bag 2-Bag1)
almost none	small	0.2	0	-0.2
few	small	0.8	0	-0.8
some	small	0	1	1
many	small	0	0	0
almost all	small	0	0	0
almost none	big	0	0	0
few	big	0	0	0
some	big	0	1	1
many	big	0.8	0	-0.8
almost all	big	0.2	0	-0.2

3.1.1. LPS Sets Dissimilarity Method of (Wilbik, et al. 2014)

The LPSs are a concise representation of data. Hence, a comparison between them is analogous to comparison between the underlying data. To this end, in (Wilbik and Keller 2012), a method was presented to quantify the dissimilarity between LPSs. The authors also proved that the distance measure is a metric. This method was later used in (Wilbik, et al. 2014) to compute dissimilarity between two datasets composed of sensor data. Next, we provide a brief description of that method.

Consider two LPSs: Q_1 y's are $P_1 @ T_1$ and Q_2 y's are $P_2 @ T_2$, where Q_1 and Q_2 are the quantifiers, P_1 and P_2 are the summarizers, and T_1 and T_2 are the truth-values of the two LPSs. Equation 3.1 gives the similarity between the two LPSs.

$$\begin{aligned} &sim((Q_1 \text{ y's are } P_1 @ T_1), (Q_2 \text{ y's are } P_2 @ T_2)) \\ &= \min(sim(Q_1, Q_2), sim(P_1, P_2), sim(T_1, T_2)) \end{aligned} \quad (3.1)$$

That is, the overall similarity between the two LPSs is given by their least similar elements.

The similarity between the two quantifiers, Q_1 and Q_2 is given by Equation 3.2.

$$sim(Q_1, Q_2) = \frac{\int(\mu_{Q_1} \cap \mu_{Q_2})}{\int(\mu_{Q_1} \cup \mu_{Q_2})} \quad (3.2)$$

Here, μ_{Q_1} and μ_{Q_2} are the membership functions of the two quantifiers. Basically, the similarity is the ratio of the overlapping area between the two membership functions and the sum of the area covered by them together. Similarly, the similarity between the two

summarizers P_1 and P_2 , having membership function μ_{P_1} and μ_{P_2} , is given by the ratio of the overlapping area and the sum of the two areas.

$$sim(P_1, P_2) = \frac{\int(\mu_{P_1} \cap \mu_{P_2})}{\int(\mu_{P_1} \cup \mu_{P_2})} \quad (3.3)$$

Note that, this equation is only valid when the two summarizers are describing the same attribute of the objects being summarized. That is, the membership functions for both P_1 and P_2 are defined on a common domain. Finally, the similarity between the truth-values, T_1 and T_2 is given by the complement of the difference between the two truth-values (Equation 3.4).

$$sim(T_1, T_2) = 1 - abs(T_1 - T_2) \quad (3.4)$$

Consider two LPS sets, $S_1 = \{s_{11}, s_{12}, \dots, s_{1n}\}$ and $S_2 = \{s_{21}, s_{22}, \dots, s_{2m}\}$ containing n and m number of summaries, respectively. Then the similarity between the two sets is given by Equation (3.5).

$$sim(S_1, S_2) = \frac{1}{n + m} \left(\sum_{i=1}^n \max_{j=1,2,\dots,m} sim(s_{1i}, s_{2j}) + \sum_{j=1}^m \max_{i=1,2,\dots,n} sim(s_{1i}, s_{2j}) \right) \quad (3.5)$$

Here, for each summary in S_1 we find the most similar summary in set S_2 , and vice versa. The final similarity value is the average of all the individual similarities. Note that the two LPS sets, S_1 and S_2 , do not have same number of summaries. This is because, in the method of (Wilbik, et al. 2014), after generating the truth-values of all the summaries (as shown in

Table 3.1), only the summaries with truth-values higher than a threshold are kept. For example, in Table 3.1, if the truth-value threshold is 0.5, then only summaries 1 and 8 are kept for bag 1 and summaries 3 and 7 are kept for bag 2. Note, that the distance/dissimilarity between the two LPS sets, is the inverse of their similarity. We call this dissimilarity computation method as d_{LPS} in the rest of this Chapter.

$$d_{LPS} = dissim(S_1, S_2) = 1 - sim((Q_1 \text{ y's are } P_1), (Q_2 \text{ y's are } P_2)) \quad (3.6)$$

This method was used in (Wilbik and Keller 2013) to compare nightly sleeping and moving patterns of elderly residents who have sensors installed inside their apartments. The data coming from the sensors was summarized by summaries of the form: *Many* of the 15-minute slots had *high bed restlessness*, or *Few* of the 15-minute slots had *low bedroom motion*. These types of summaries were generated for each night during a time-period. The distance between the summaries was then used to estimate how similar or dissimilar each of the nights are from each other.

3.1.2. Proposed Method

The block diagram in Figure 3.4 presents an overview of our proposed method. The process starts by computing truth-values of the summaries comprising of all the quantifiers for a given summarizer. In the method depicted in Figure 3.4, only summaries with a common summarizer are compared with each other (For example: All the summaries with summarizer as *big* in Table 3.1). In case of multiple summarizers, we can obtain an overall

comparison by aggregating distances for each of the summarizers. We discuss this in detail later on in this Chapter.

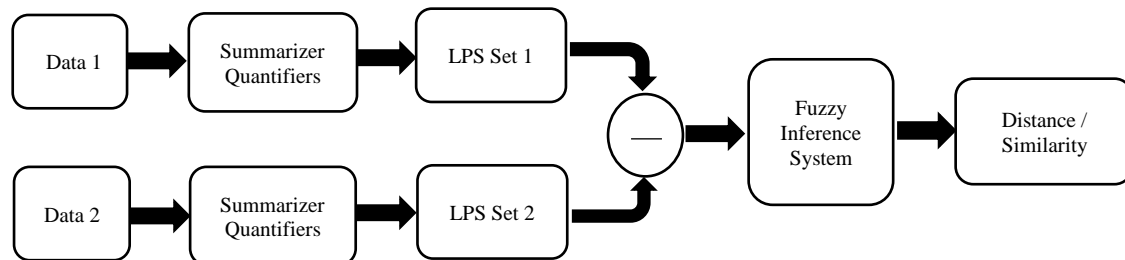


Figure 3.4: Overview of the system comparing two sets of objects using Linguistic Protoform Summaries and Fuzzy Inference System

Consider a summarizer P and n quantifiers Q_1, Q_2, \dots, Q_n . We start by computing truth-values of summaries: Q_i y's are P , for $i = 1$ to n for the two datasets which we wish to compare. Then for each quantifier, we compute the difference between the truth-values of the corresponding LPSs obtained for dataset 1 and dataset 2.

The difference in the truth-values form the input to a Mamdani type Fuzzy Inference System (FIS) (Keller, et al. 2016). Fuzzy Inference Systems are very apt at representing human intuition in a computational system. Since with the use of LPSs, we already have the two datasets represented in intuitive linguistic terms, FIS is a suitable choice for dealing with these linguistic representations.

The fuzzy rule base is the central element of an FIS. Table 3.2 shows a fuzzy rule base for the quantifiers defined in Figure 3.2, *almost none, few, some, many, almost all*. The rules encode the relationships between the differences in the truth-values corresponding to

Table 3.2: Fuzzy rule base with five antecedents and one consequent. The terms dec, sim and inc represents decrease, similar and increase, respectively

rule #	Antecedents					Consequent
	almost none	few	some	many	almost all	change
1	dec	inc				inc
2	dec		inc			inc
3	dec			inc		significant inc
4	dec				inc	significant inc
5	inc	dec				dec
6	inc		dec			dec
7	inc			dec		significant dec
8	inc				dec	significant dec
9		dec	inc			inc
10		dec		inc		inc
11		dec			inc	significant inc
12		inc	dec			dec
13		inc		dec		dec
14		inc			dec	significant dec
15			dec	inc		inc
16			dec		inc	significant inc
17			inc	dec		dec
18			inc		dec	significant dec
19				dec	inc	inc
20				inc	dec	dec
21	sim	sim	sim	sim	sim	sim

the five quantifiers. For example, the first rule reads as: *If the difference between the truth-value of LPS corresponding to **almost none** decreases, and the difference between the truth-value of LPS corresponding to **few** increases, then, there is an **increase** in the object property represented by the summarizer.* For instance, if the LPS is used to describe the size of the balls in a bag by a summarizer *big*, with the Protoform: ***Q** balls are **big***, then

the consequent of the fuzzy rules denotes the change in the number of *big* balls in two bags. The fuzzy sets used to represent antecedents and the consequent of the fuzzy rule base are

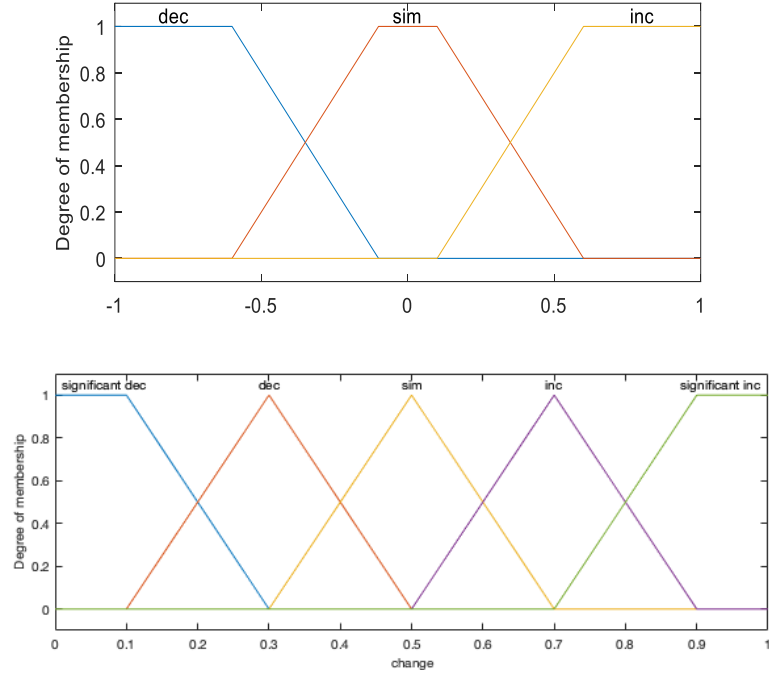


Figure 3.5a & b: Membership functions of the antecedents (3.5a) and consequents (3.5b) used in the Fuzzy Inference System

shown in Figure 3.5. Here, the definition of *decrease*, *similar* and *increase* are common to all the five inputs (difference in truth-value). The antecedents are defined over $[-1, 1]$, which is the range of the difference between the truth-values.

When the difference in the truth-value are divided into 3 linguistic labels (*increase*, *decrease* and *similar*), then for five quantifiers, in order to cover the complete input space, we need to define a total of $3^5 = 243$ rules. However, since the truth-values corresponding to the quantifiers are not independent of each other, we only need a small set of rules to cover most of the input space. The rules of Table 3.2 are defined based on the reasoning

that for every significant change in the underlying data, the truth-values of at least two LPSs change significantly. Moreover, the change in the truth-value is in the opposite direction (if one increases, then the other decreases).

Naturally, this behavior is dependent on the way we define the fuzzy sets used to represent the quantifiers. Mindful of this, we define a fuzzy rule for every pair of quantifiers. This leads to $\frac{5}{2}C + 1 = 21$ fuzzy rules. If the truth-values corresponding to all the quantifiers do not change by much (i.e. remain similar), then the fuzzy rules suggest no change in the data (rule # 21). Note that although it's very unlikely, but if a situation arises when none of the fuzzy rules fire with a strength greater than zero, we deem the two datasets as similar.

The distance based on the fuzzy rule firings is obtained as the defuzzified value of the FIS. We employ centroid defuzzification to combine the firing strengths of all the fuzzy rules. For the consequent membership functions shown in Figure 3.5b, the smallest defuzzified value is obtained when a rule with the consequent *decrease* fires at a maximum strength of

Table 3.3: Interval of defuzzified values for each of the five consequents of Figure 3.5b

Defuzzified value	Change
[0,0.15)	Significant Decrease
[0.15,0.4)	Decrease
[0.4,0.6)	Similar
[0.6,0.8)	Increase
[0.8,1.0]	Significant Increase

one, while the highest defuzzified value is obtained when a rule with the consequent *increase* fires at a strength of one. Based on this, the defuzzification value ranges from 0.1 to 0.9. We linearly scale the defuzzified value so that it spans from zero to one. To obtain the direction and strength of dissimilarity in linguistic terms, we take the linguistic label that has the highest membership for the scaled defuzzified output. Table 3.3 presents the mapping of the dissimilarity and the linguistic labels assigned to each value.

Accordingly, if the scaled defuzzified value is between zero and 0.15, then it is assigned the label *significant decrease*, if the value is between 0.15 and 0.4. then it is assigned the label *decrease* and so on. Based on this, the change is least when the defuzzified value is 0.5, while a defuzzified value less than 0.5 suggests a decrease and a value greater than 0.5 implies an increase. As per the normal convention, we would like a distance value of zero to represent no change, and a distance value of one to represent maximum change. To this end, we perform the computation given in Equation 3.7 to convert the defuzzified value to a distance measure.

$$d_{FIS} = distance = \frac{defuzzified\ value - 0.5}{0.5} \quad (3.7)$$

We carry on this process separately for each summarizer that describes the dataset. For example, in the example in Table 3.1, the dissimilarity computation process is carried out separately for the LPSs describing the datasets in terms of the summarizer *big* and the summarizer *small*. The distance computed with respect to each of the summarizer are then

aggregated to obtain an overall distance between the two datasets. We can use any of the well-known aggregation operators like *mean*, *median*, *maximum*, *minimum* and so on.

We illustrate our dissimilarity computation method with the help of the two bags of balls example presented before in Section 3.1. There, Figure 3.1 showed the distribution of sizes of the two bags and Table 3.1 displayed the truth-values of the LPSs and the difference between them. The difference vector is input to the FIS to compute the distance. Figure 3.6a and b shows the output of the FIS for the two summarizers: *Small* and *Big*, respectively. Here, the shaded portion of the fuzzy sets of the FIS consequents show the fuzzy set obtained as a result of firing of fuzzy rules. Since the two summarizers, *small* and *big*, are complements of each other, the FIS output for the two are also complements. However, this may not be the case for a different set of summarizers.

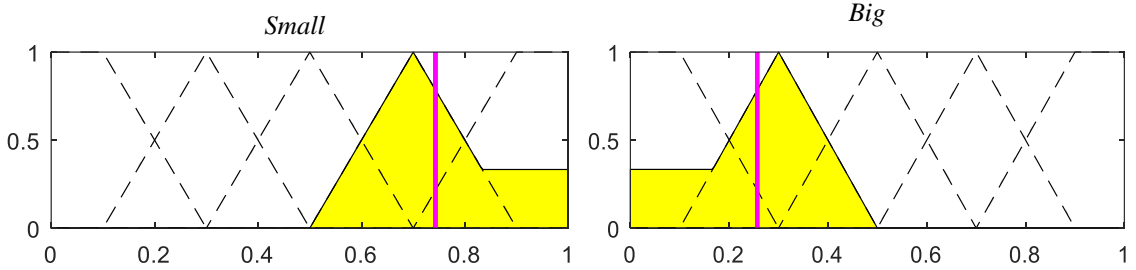


Figure 3.6a & b: Fuzzy output of the FIS for the two summarizers: Small and Big. The vertical line shows the defuzzified value of the fuzzy sets.

We observe that for the summarizer *Small*, the truth-value corresponding to the quantifier *few* decreases and that for *some* increases. This leads to the firing of rule # 9 listed in Table 3.2. The truth-value corresponding to the quantifier *almost none* also decreases a little which leads to firing of rule # 2, but with less strength. Similarly, for the summarizer *Big*, there is increase in the truth-value corresponding to the quantifier *some*, a decrease with

that for *many* and a small decrease with that of *almost all*. This leads to the firing of rule #17 and #18. The scaled defuzzified output for the two summarizers, *Big* and *Small*, are 0.19 and 0.81, respectively. This leads to the individual distance between both of them to be 0.62. Using the mean operator to aggregate the two distances the overall dissimilarity between the two bags of balls is 0.62. The explanation for the dissimilarity is given by the consequent that has the highest membership for each defuzzified value, that is, *decrease* for the summarizer *big* and *increase* for *small*. Based on this, the following statement describes the comparison between the contents of the two bags: *The dissimilarity between the two bags is 0.62. There is an increase in the number of small balls in bag 2 while a decrease in the number of big balls.*

3.2. Method Validation

In this section, we present results of various experiments that we conducted to study and validate our distance/similarity computation algorithm. In the two experiments, we study the functioning of our algorithm with the help of synthetic data, where we compare the LPS based methods with two histogram distance measures, the Euclidean and the Canberra metric (Cha 2007). We briefly describe the two techniques in the following. Suppose X is a set of n elements, then its histogram $H(X)$ represents the frequency of each value falling inside a user-defined bin. A pdf of the corresponding histogram $H(X)$ is produced by dividing each level by the total size of X , $P(X) = H(X)/n$. If X and Y are two sets of objects with their histograms as $H(X)$ and $H(Y)$ and pdfs as P and Q , respectively, each

having d number of bins, then the Euclidean and the Canberra distance between the two pdfs is given by Equation 3.8 and 3.9, respectively.

$$Euclidean, d_{Euc} = \sqrt{\sum_{i=1}^d |P_i - Q_i|^2} \quad (3.8)$$

$$Canberra, d_{can} = \sum_{i=1}^d \frac{|P_i - Q_i|}{P_i + Q_i} \quad (3.9)$$

3.2.1. Gradual Change in Data

In this experiment, we show the working of our method by computing distance between two sets of objects, which gradually grow further apart from each other. To this end, we start with two bags of balls with both of them having 1000 *small* balls. We gradually change the distribution of the size of balls in the second bag by replacing 5 *small* balls with balls of *bigger* size. That is, at first step, we replace five *small* balls in bag 2 with *big* balls, then in step 2 we replace 10 *small* balls in bag 2 with *big* balls and so on. This would leave us with all *big* balls in bag 2 at the end of the experiment. Note that, the small balls are drawn from a Gaussian distribution with the Mean and the Standard deviation of one, with the distribution restricted between 0 and 2.5. Similarly, the big balls are drawn from a Gaussian distribution with a Mean value of 4 and a Standard Deviation of 1 and is restricted between 2.5 and 5.

At each step, we compute how dissimilar bag 1 is from bag 2. Figure 3.7 shows the distance computed by the proposed method (d_{FIS}), the method of (Wilbik, et al. 2014) (d_{LPS}) and the method based on Euclidean distance (d_{Euc}). For the LPS based methods, we use the summarizers and the quantifiers of Figure 3.2 and Figure 3.3, and for the Euclidean distance based method, we divide the histogram into 5 bins. We see that d_{Euc} increases gradually throughout the experiment. d_{LPS} and d_{FIS} also increase gradually until a certain point, after which they saturate. In d_{FIS} the distance value saturates at the point when the majority of the balls in bag 2 are *big* (around 500), while with d_{LPS} the saturation occurs much earlier. Also, in d_{LPS} , the change in distance is less gradual than the proposed method, as suggested by large leaps in Figure 3.7.

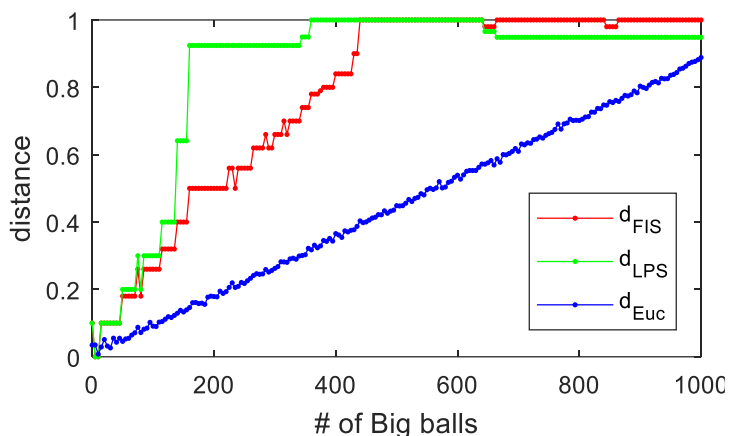


Figure 3.7: Distance plot for the three distance measures used in Experiment 3.2.1

The main advantage of our method is that it can specify the dissimilarity between the two datasets in linguistic terms. This, in turn, can be used to explain the differences between datasets. Based on the consequents of Figure 3.5b, the change is described as an *increase*, a *decrease*, a *significant decrease* or a *significant increase* in the property represented by the summarizer. In the above example, the FIS suggests a similar number of small balls in bag 2 up to a certain point, which then changes to a *decrease* in the number of small balls and then finally saturates at a *significant decrease* in the number of small balls. This variation is shown in Figure 3.8 for both small and big balls. We see that the two bags are judged as similar until there are less than 80 *big* balls in bag 2. After this, the FIS suggests that bag 2 has a little higher number of big balls, and a little fewer small balls, as compared to bag 1 (shown by *inc*) until there are about 250 big balls out of 1000. Then, the dissimilarity saturates and suggests that there are a lot of big balls (shown by *significant inc*) and fewer small balls in bag 2 (shown by *significant dec*). We note that there is a small spike in the change in the number of *small* and *big* balls just before it switches from *inc* to *significant inc*, and *dec* to *significant dec*, respectively. Figure 3.9 shows the histograms of the distribution of bag 2 which led to the spike and just before (spike -1) and after the spike

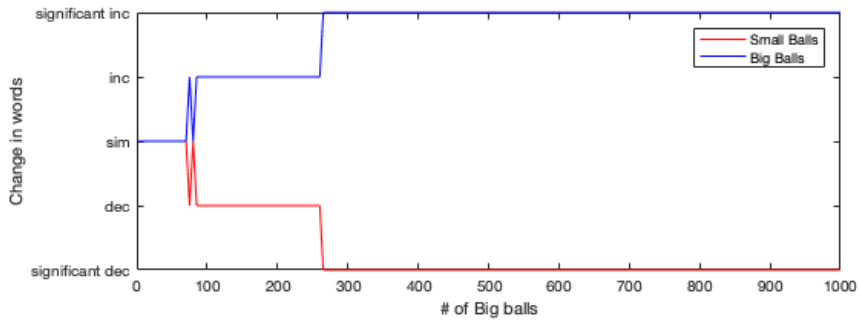


Figure 3.8: Representation of linguistic terms showing the change detected by the FIS

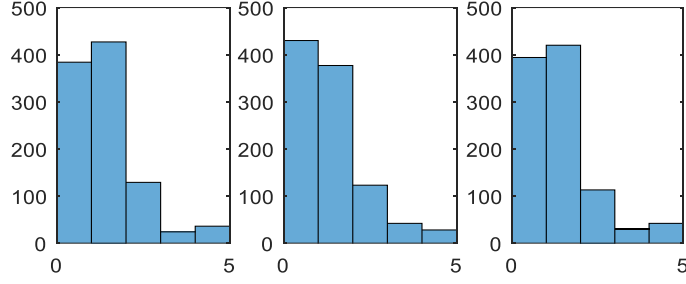


Figure 3.9a, b & c: Histogram of the distribution of bag 2 that resulted in spike in Figure 3.8, distribution before the spike and after the spike

(spike +1). The three histograms suggest a similar distribution among the corresponding bags. Also, the Euclidean distance of these three distributions to the distribution of bag 1 is 0.07 (spike-1), 0.09 (spike), 0.07 (spike+1). That is, for both d_{FIS} and d_{Euc} , the bag with the distribution of Figure 3.9b, is different than the bags with distribution of Figure 3.9a and 3.9c.

3.2.2. Correlation between distance measures

In this experiment, we study the behavior of the proposed method in terms of how it correlates with d_{LPS} and d_{Euc} . We use the synthetic data setting consisting of two bags of balls, as used before. We start by filling two bags with randomly sized balls and compute distance between the two bags. This process is carried out for N number of times, and the dissimilarity measured by the different methods are plotted against each other. The random assignment of the balls in the two bags is done as follows: For each bag, we randomly select the number of small balls by drawing an integer between 1 and 1000 from a uniform distribution. Then, the number of big balls are equal to 1000 minus the number of small balls. Note that the *small* and *big* balls are drawn from the Gaussian distributions used in

the last experiment. Figure 3.10 shows the correlation plots between the proposed method, d_{LPS} and d_{Euc} . For reference, we also show correlation between the Euclidean measure (d_{Euc}) and the Canberra measure (d_{Can}). For each correlation, we also compute the correlation coefficient using Equation 3.10 (Cha 2007).

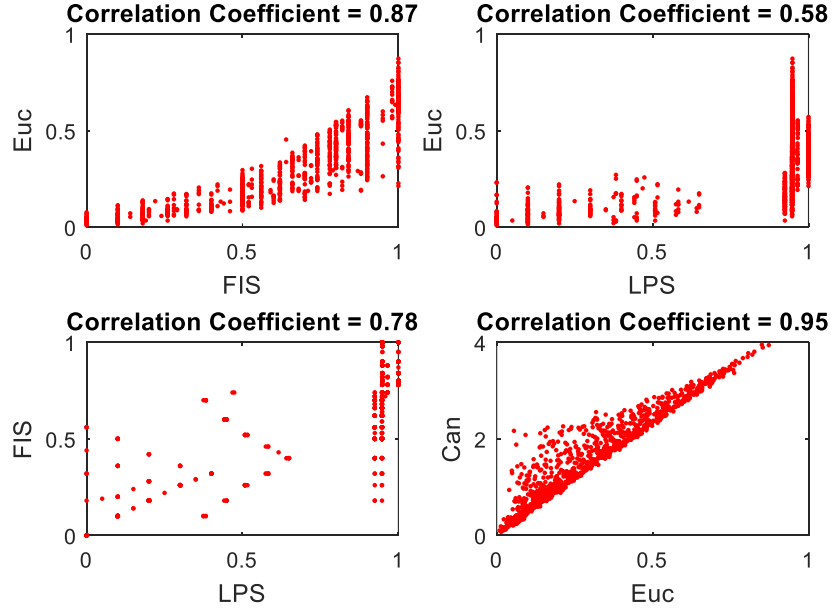


Figure 3.10: Correlation plots comparing distance values produced by d_{Euc} , d_{LPS} , d_{FIS} , d_{Can}

$$Corr(d_1, d_2) = \frac{\sum_{i=1}^N (d_1(p_i, q_i) - \bar{d}_1)(d_2(p_i, q_i) - \bar{d}_2)}{\sqrt{\sum_{i=1}^N (d_1(p_i, q_i) - \bar{d}_1)^2 \sum_{i=1}^N (d_2(p_i, q_i) - \bar{d}_2)^2}} \quad (3.10)$$

where $\bar{d}_1 = (\sum_{i=1}^N d_1(p_i, q_i)) / N$

Here, d_1 and d_2 represent the two distance measures, and p_i and q_i are the datasets between which we are computing the dissimilarity. A higher correlation coefficient suggests a high degree of correlation.

We see that d_{Euc} and the proposed method exhibit a fairly high correlation coefficient of 0.87, although not as high as the correlation between the Euclidean and the Canberra metric (0.95). The correlation plot suggests that, in general, d_{FIS} produces smaller dissimilarities as compared to d_{Euc} . The correlation coefficient between d_{LPS} and d_{Euc} is much lower as compared to the proposed method and the correlation plot between them shows that d_{LPS} saturates around 0.7, after which all of the distance values are around 1.

In order to have a better understanding of the two LPS based methods, we conduct the same experiment as above, but with a different set of summarizers. Instead of just two summarizers (*small* and *big*), we use five summarizers: *about 0*, *about 1*, *about 2*, *about 3*, *about 4* and *about 5*, as shown in Figure 3.11. This is akin to increasing the number of bins in a histogram. With these summarizers, we can express the size of the balls on a finer level, as compared to just two summarizers. The correlation plots, along with the correlation coefficients for the new set of summarizers are shown in Figure 3.12. We observe a better correlation to d_{Euc} for both the LPS based distance measures (0.98 and

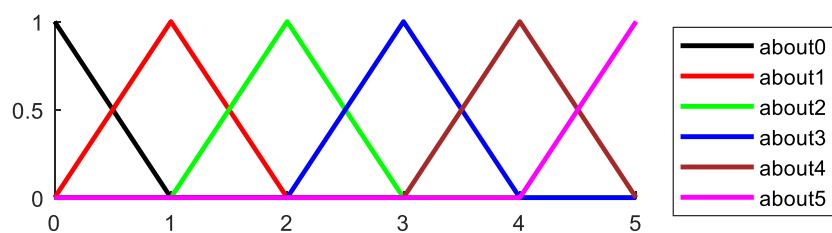


Figure 3.11: Membership functions of summarizers describing size of balls in a bag

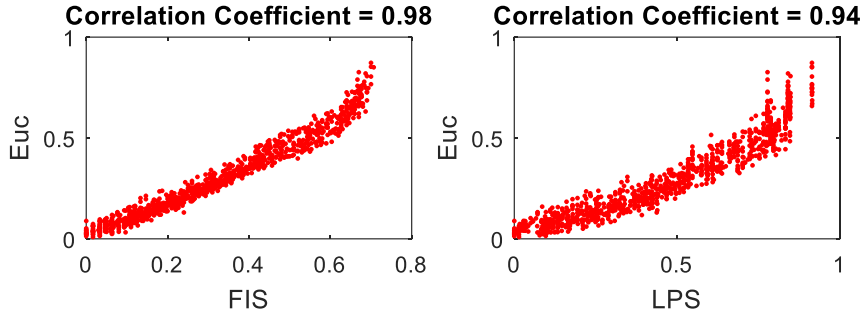


Figure 3.12: Correlation plots comparing distance values produced by d_{EUC} , d_{LPS} , d_{FIS} when d_{LPS} & d_{FIS} are used with Summarizers of Figure 3.11

0.94, for d_{FIS} & d_{LPS} , respectively). Moreover, the correlation with the proposed method and the Euclidean distance measure (0.98) surpasses the correlation between the Canberra and Euclidean distance measure (0.95). This suggests that, with an increased number of summarizers used to describe the property being compared, the proposed method correlates very highly with a standard measure like the Euclidean distance.

4. Routine Modelling

4.1. Overview

We outline our routine modelling and deviation detection approach in Figure 4.1. We start with the assumption that a time period that can be considered as normal in terms of an

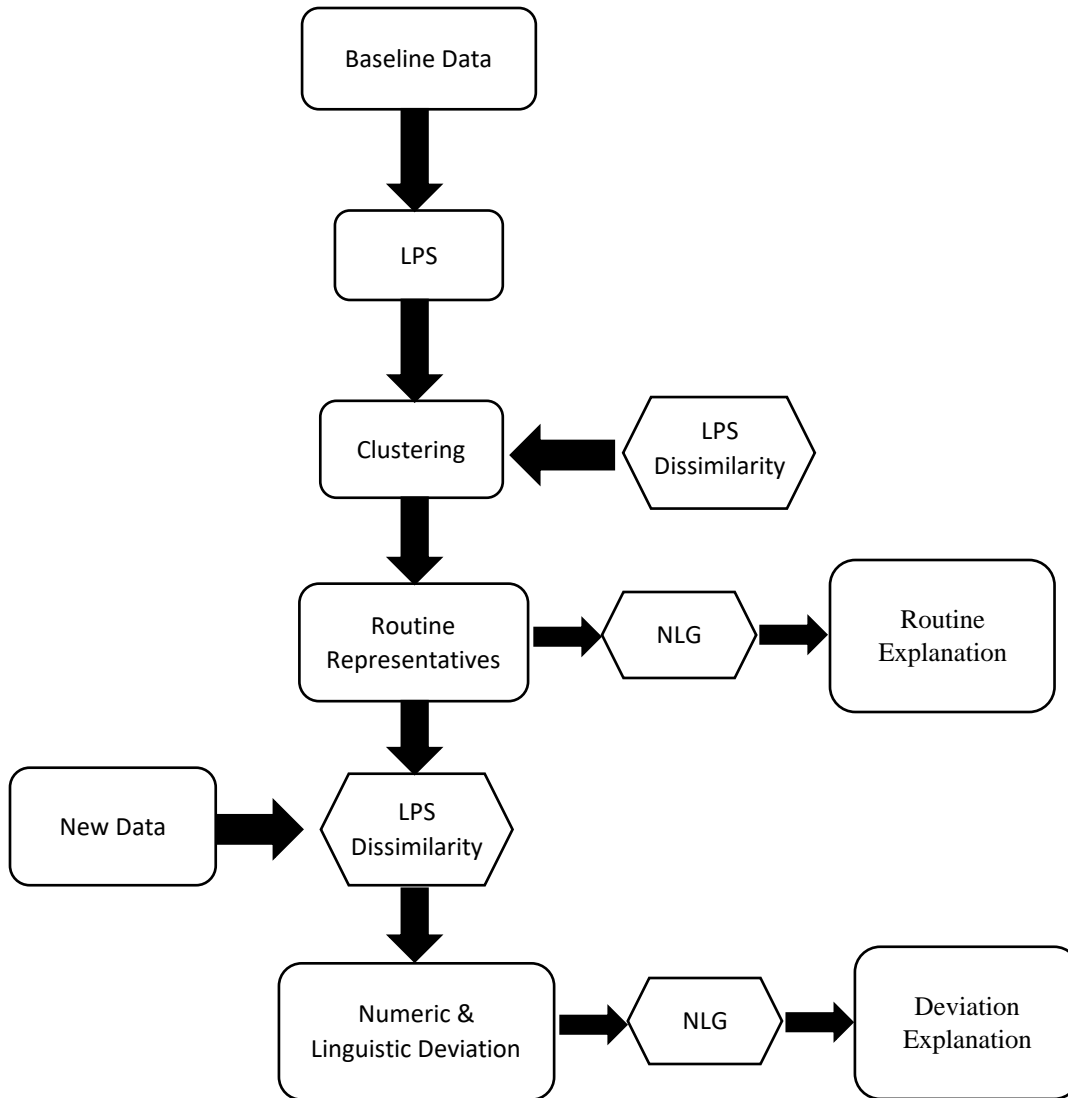


Figure 4.1: Block diagram showing computation of routine and deviation from it. Linguistic Protoform Summaries are first used to represent routine based on baseline time period, which is then compared to new data. The NLG block is responsible for describing results in natural language.

elderly resident's health has been identified as baseline by either their caretakers or using their health records. We model the daily routine of the resident over this baseline time period and use this model to find changes in their routine with time.

The first step in our routine modelling and deviation detection approach is to express the daily behavioral and physiological data in linguistic terms using the Linguistic Protoform Summaries (LPSs). This involves selection of the protoforms that are most appropriate for the underlying data. The type of LPSs drives the choice of the truth-value computation method. The representation of data into LPSs also requires the design of fuzzy sets modeling the quantifiers, summarizers and the qualifiers. The choice of the LPS protoforms, fuzzy definitions of the linguistic terms and the truth-value computation method come together to represent the daily physiological and behavioral patterns of the elderly residents.

Given the daily LPS representation of behavioral and physiological patterns of activities, we need an aggregation mechanism that combines this information over the baseline time period and produces an overall representation of the routine. A straightforward and simplistic way to achieve this is to aggregate all the daily representations into one global representative. Since one of our aims is to be able to explain the discovered routine in natural language, this strategy might not produce very intuitive results if a person has multiple routines during a time period, for example weekend and weekday routines. A different technique that helps in dealing with this existence of multiple patterns is to identify the presence of more than one routine, and if so, then model each individual routine

separately. These individual routines together form the overall routine of a resident. To this end, we employ clustering to find groups of similar days in both the behavioral and the physiological data.

Clustering is a very well-known and widely used technique to discover patterns in data in an unsupervised manner. The clustering techniques present in the literature are broadly classified into two categories: algorithms for object data and relational data, where the objects are the entities over which clustering is performed (Bezdek, et al. 1999). The clustering methods for the object data requires the objects to be defined by a set of feature vectors and the relation between them are assessed by some distance measure over the feature vectors. The clustering is obtained by using the distance measure between the objects to analytically optimize some objective function. On the other hand, the relational clustering techniques use the pairwise relations (dissimilarities) between the objects to perform clustering, without depending on the analytical form of the method to compute the relationship. Since the fuzzy-rule based method directly produces a dissimilarity between the LPS sets representing the data, only the relational clustering methods are appropriate to our routine modelling approach. To this end, we employ the dissimilarity technique presented in Chapter 3 to compute pairwise dissimilarities between daily behavioral and physiological activities, which is in-turn is used to find groups of similar days using a relational clustering method. We then use the medoids of the obtained clusters to represent the routine.

Once we have a model of the general routine of a resident, our next task is to compare it with the data outside of the baseline time period. This entails the representation of this new data using LPSs and then use the method to compare LPS sets to determine deviation between the baseline model and new data in numeric as well as linguistic terms.

The use of LPSs to both represent the routine and then to linguistically quantify the deviation between the baseline data and the new data allows us to produce natural language explanations of the baseline routine as well as changes in the data. We make use of template based NLG techniques to generate natural language descriptions of the behavioral and physiological patterns in the data and produce explanations of the changes in the routines, as the new data arrives.

In the following we present our approach to model the behavioral and physiological patterns. In Section 4.2, we describe our process to represent the behavioral routine of a hypothetical elderly resident. We design several synthetic datasets to illustrate our method. We also describe our technique to generate natural language summaries of the obtained routine model. The data dissimilarity method of Chapter 3 is employed to find deviations in the routine that can be expressed in natural language. Section 4.3 lays out our approach to model the physiological routines comprising of the pulse rate and the respiration rate data. We make use of pulse rate measurements of a resident taken over a three-month period to show the method can detect changes in physiological measurements, which can be described in natural language. For both behavioral and physiological routines, we present multiple retrospective case studies to validate our method.

4.2. Behavioral Routine

As mentioned before, the behavioral routine is responsible for capturing the activity patterns of residents longitudinally, over a given time-period. We describe our behavioral routine modelling algorithm with the help of synthetic datasets with an assumption that a set of activity detection algorithms provides us with information about the whereabouts of the resident throughout the day in one of the four activities: inside apartment (INA), outside apartment (OOA), in the bed or in the bathroom. The use of synthetic data allows us to better study the various properties of our algorithm, as well as to probe it with respect to different parameter settings.

Consider a hypothetical resident, named Mr. Jones, who lives in a sensor-equipped aging in place facility. In his normal routine, Mr. Jones sleeps in his bedroom during the night and gets up around 7 am in the morning. After getting up, he goes through his usual morning routine in the bathroom. After getting ready around 9 am, he goes out to the dining place outside the apartment to get breakfast. He returns around 10 am and spends time watching TV or performing normal chores inside the apartment. Then the resident leaves the apartment again for lunch around noon. He spends about an hour and a half having lunch before returning back to the apartment around 1:30 PM. Then he usually watches some TV before taking a nap in the bedroom. He gets up around 3:30, fixes himself coffee in the kitchen, and watches some TV in the living room. He stays in the living room watching TV, reading and doing other activities until 6 in the evening when he goes out of the apartment again for dinner. He comes back around 7:30 and then watches some more

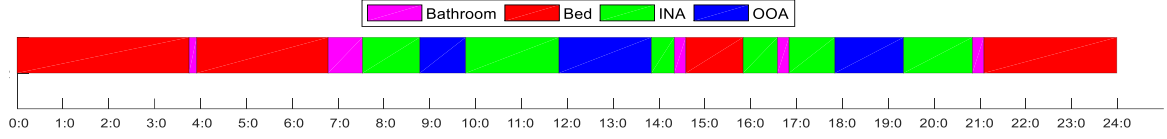


Figure 4.2: Patch plot showing location pattern of a hypothetical resident over a day. The time on x-axis varies from midnight to midnight, from left to right. The four locations are depicted by different colors.

TV before going to bed around 9 PM. Adding to this overall routine, he goes to the bathroom at around 7 AM, 11 AM, 3 PM, 5 PM, 9 PM, 4 AM, where the first visit in the morning is longer. This activity is presented in a pictorial form in Figure 4.2. We call this plot a patch plot.

4.2.1. Using LPSs to Represent Behavioral Patterns

Given the location information presented in Figure 4.2, our goal here is to assess how much time a person spends at a certain location at a given time of day. For instance, we would like to know how the person spent their morning hours. In this statement, morning is a general as well as a personal term. People tend to have their own definition of what is morning time, however, universally, morning comprises of early hours of the day. In other words, the definitions of different times of days, like morning, daytime, nighttime are not crisp, but have a fuzzy nature. To this end, we first divide a day into slots of Δt minutes

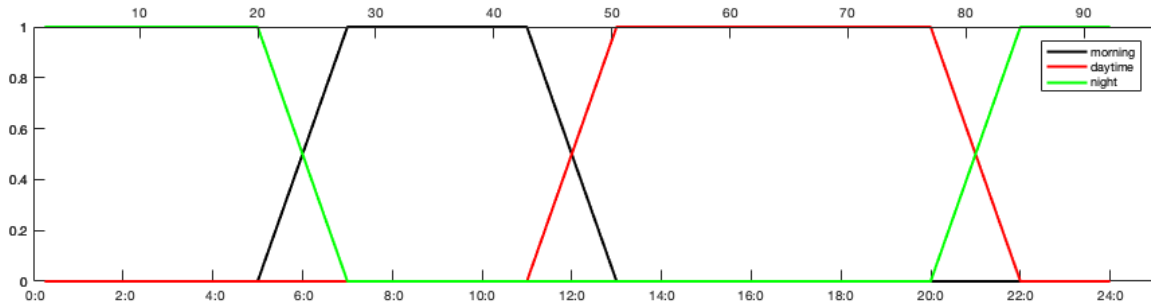


Figure 4.3: Fuzzy sets representing sub division of a day. The lower x-axis shows the time of the day in Hour:Min format. The upper x-axis has 15 min slot numbers for corresponding times

and then to each slot, assign a degree of belongingness to the different subdivisions of a day, as shown in Figure 4.3. We divide the 24 hours of a day into three sub-time divisions, namely: morning, daytime and nighttime.

For example, if $\Delta t = 15$ minutes, then in total we would have $24*60/15 = 96$ slots. Next, for each of the Δt slots, we assign a degree of membership in each of the four locations. Due to quantization, a time slot can have some portions of it in different locations. Therefore, for each slot we determine the fraction of time spent in each of the four categories. For example, consider the first 4 minutes of a slot were spent outside of the apartment, and then the person goes to bathroom for 5 minutes and then stays in the living room for rest of the 15 minutes. Then the 15 min slot will have $4/15 = 0.27$ in OOA, 0.33 in the bathroom, 0.4 in INA and 0 in bed. Given the data in Figure 4.2, we can compute a membership in all the four locations for each of the 96 slots, as shown in Figure 4.4. We see that during the nighttime, almost all of the slots have highest degree in the location bed, while during the day there are certain intervals where the degree of OOA is highest, these are the times the person went outside the apartment.

Now for each of the Δt slots, we have a degree in each of the four activities and a degree in the four sub-divisions of the day. This information enables us to use Linguistic Protoform Summaries as an aggregation mechanism to calculate the amount of time that

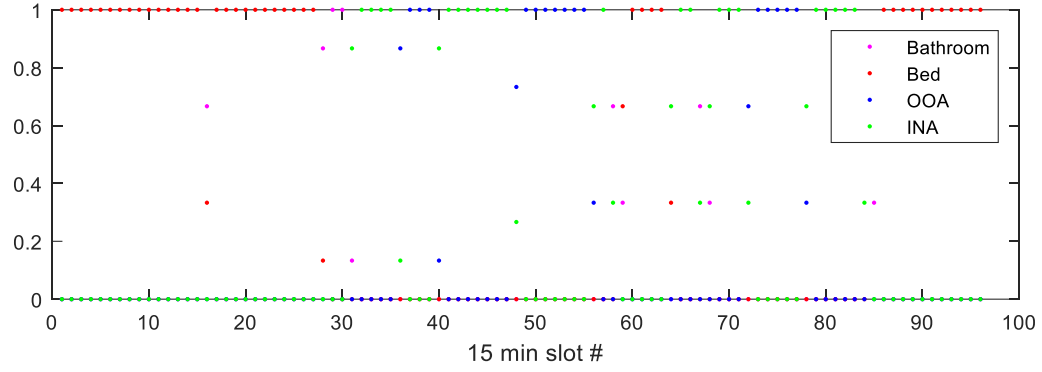


Figure 4.4: Membership of the 96 fifteen minute slots in four locations representing pattern of Figure 4.2

was spent in a certain activity, at a given time of day. To this end, we define type II LPS (described in Section 2.5) with the protoform:

Q** of the Δt slots in the **R** were spent in **P

For example,

Many** of the 15 minute slots in the **morning** were spent in **bed

Where, *Q* (*Many*) is the quantifier, *R* (*morning*) is the qualifier and *P* (*in bed*) is the summarizer.

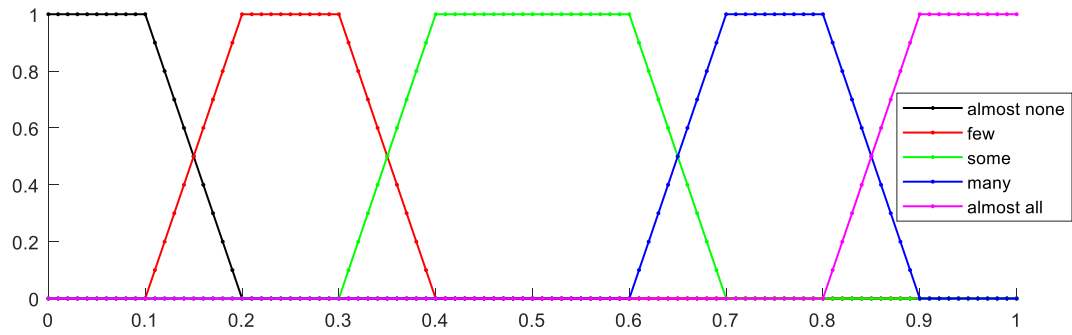


Figure 4.5: Quantifier membership functions

To generate statements of this form, we use the fuzzy definitions of the quantifiers shown in Figure 4.5.

Using the Delgado-GD truth-value computation technique for type II protoforms that was defined in Section 2.5.1, we summarize the location activity of a person for a day by generating LPSs comprising of all the combinations of quantifiers, qualifiers and summarizers. Considering the 5 quantifiers shown in Figure 4.5 (*almost none, few, some, many, almost all*), 3 qualifiers representing the time of day (*morning, daytime, night*) and 4 summarizers for the activities (*bathroom, bed, OOA and INA*), for each day we will have $5*4*3=60$ LPSs. Table 4.1 shows an example containing all the LPSs representing the activity pattern in Figure 4.2. Based on the truth-values of LPSs in the table, we can draw the following observations:

- The truth-values of LPSs pertaining to the morning time suggest that a considerable amount of the time was spent inside the apartment, with a little time in each of the other three activities.
- Most of the time during the daytime was spent inside and outside the apartment, with a little time in bed and the bathroom
- The LPSs corresponding to nighttime suggests that almost all of the night was spent in the bed.

Table 4.1: LPS representing the location activity presented in Figure 4.2. Each individual LPS is of the format, Q of the 15 in slots in R were spent in P @ (TV)

LPS #	Quantifier (Q)	Qualifier (R)	Summarizer (P)	Truth-value (TV)
1	'almost none'	'morning'	'Bathroom'	0.47
2	'few'	'morning'	'Bathroom'	0.54
3	'some'	'morning'	'Bathroom'	0.00
4	'many'	'morning'	'Bathroom'	0.00
5	'almost all'	'morning'	'Bathroom'	0.00
6	'almost none'	'morning'	'Bed'	0.41
7	'few'	'morning'	'Bed'	0.59
8	'some'	'morning'	'Bed'	0.00
9	'many'	'morning'	'Bed'	0.00
10	'almost all'	'morning'	'Bed'	0.00
11	'almost none'	'morning'	'OOA'	0.11
12	'few'	'morning'	'OOA'	0.89
13	'some'	'morning'	'OOA'	0.00
14	'many'	'morning'	'OOA'	0.00
15	'almost all'	'morning'	'OOA'	0.00
16	'almost none'	'morning'	'INA'	0.00
17	'few'	'morning'	'INA'	0.12
18	'some'	'morning'	'INA'	0.88
19	'many'	'morning'	'INA'	0.00
20	'almost all'	'morning'	'INA'	0.00
21	'almost none'	'daytime'	'Bathroom'	0.89
22	'few'	'daytime'	'Bathroom'	0.11
23	'some'	'daytime'	'Bathroom'	0.00
24	'many'	'daytime'	'Bathroom'	0.00
25	'almost all'	'daytime'	'Bathroom'	0.00
26	'almost none'	'daytime'	'Bed'	0.60
27	'few'	'daytime'	'Bed'	0.40
28	'some'	'daytime'	'Bed'	0.00
29	'many'	'daytime'	'Bed'	0.00
30	'almost all'	'daytime'	'Bed'	0.00
31	'almost none'	'daytime'	'OOA'	0.00
32	'few'	'daytime'	'OOA'	0.48
33	'some'	'daytime'	'OOA'	0.52
34	'many'	'daytime'	'OOA'	0.00
35	'almost all'	'daytime'	'OOA'	0.00
36	'almost none'	'daytime'	'INA'	0.00
37	'few'	'daytime'	'INA'	0.32
38	'some'	'daytime'	'INA'	0.68
39	'many'	'daytime'	'INA'	0.00
40	'almost all'	'daytime'	'INA'	0.00
41	'almost none'	'night'	'Bathroom'	1.00
42	'few'	'night'	'Bathroom'	0.00
43	'some'	'night'	'Bathroom'	0.00
44	'many'	'night'	'Bathroom'	0.00
45	'almost all'	'night'	'Bathroom'	0.00
46	'almost none'	'night'	'Bed'	0.00
47	'few'	'night'	'Bed'	0.00
48	'some'	'night'	'Bed'	0.00
49	'many'	'night'	'Bed'	0.00
50	'almost all'	'night'	'Bed'	1.00
51	'almost none'	'night'	'OOA'	1.00
52	'few'	'night'	'OOA'	0.00
53	'some'	'night'	'OOA'	0.00
54	'many'	'night'	'OOA'	0.00
55	'almost all'	'night'	'OOA'	0.00
56	'almost none'	'night'	'INA'	1.00
57	'few'	'night'	'INA'	0.00
58	'some'	'night'	'INA'	0.00
59	'many'	'night'	'INA'	0.00
60	'almost all'	'night'	'INA'	0.00

Next, we present a few experiments to better understand and test the LPS representation of activity patterns.

4.2.1.1. Experiment: Gradual Change in Routine

In this experiment, we design a synthetic data set that spans over multiple days. The goal here is to test and understand the variation of truth-value computation based on a synthetic dataset, where we are confident about what to expect. We take the activity information presented in Figure 4.2, and change the time of each activity by a small amount, computing the LPS representation of the activity information at each step. We shift the activities minute by minute until the activities at the current step are 60 minutes apart from the start. Figure 4.6 below depicts this variation.

Next, we compute the truth-values of for all the 60 LPSs describing the activity over the 60-day period, plotted in Figure 4.7. We can draw the following observations:

- Even though the change in the location pattern in Figure 4.7 is gradual over the course of 60 days, the truth-values have oscillations going up and down throughout the experiment, which is due to the truth-value computation method.
- Except the oscillations, the truth-value pertaining to bathroom activity remains almost constant throughout the experiment for all the three time divisions. During the morning hours, the truth-values suggests a very small fraction of amount being

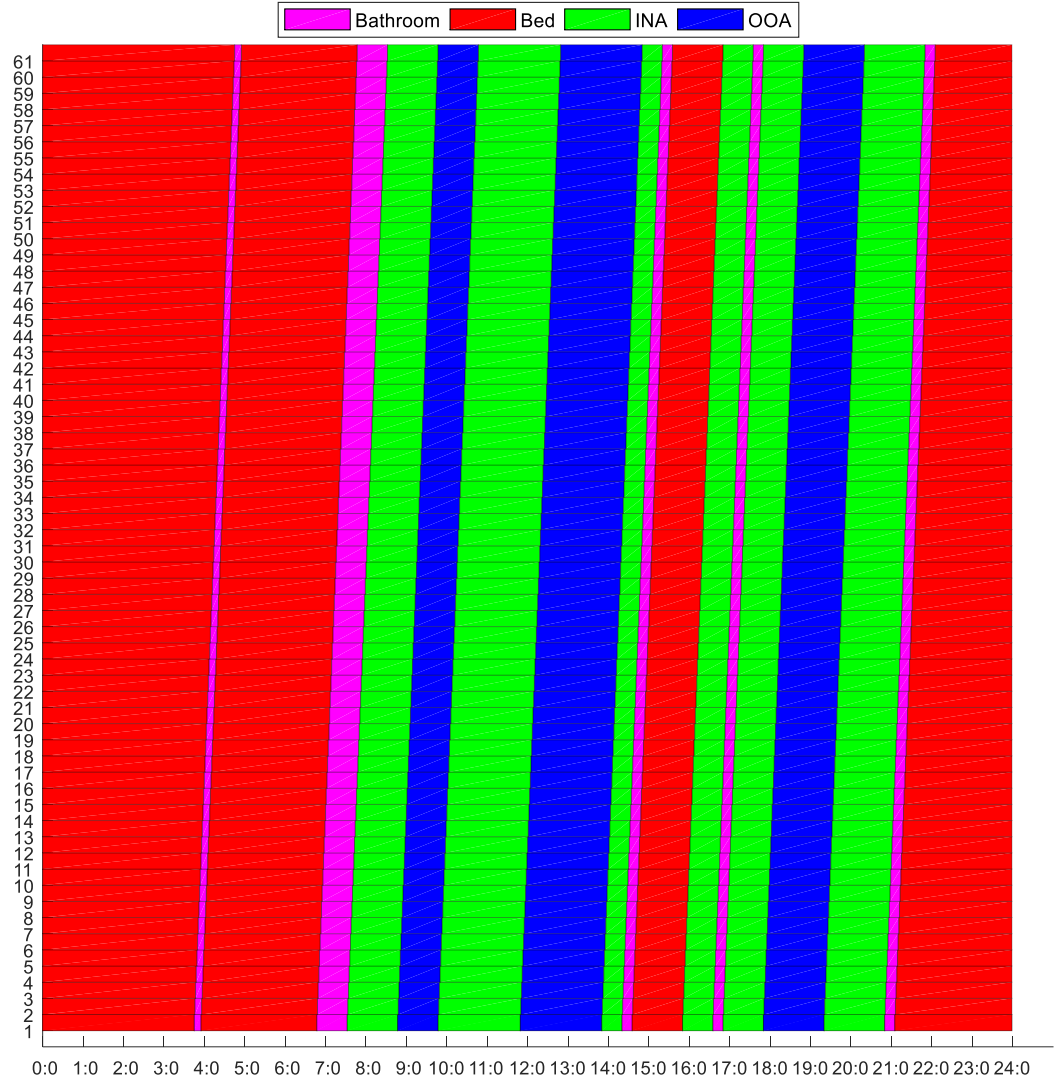


Figure 4.6: Sliding routine. Synthetic data used to test the computation of truth-values representing daily activity patterns of a resident. The activity pattern is shifted by 1 minute between each consecutive day, until the last day is 60 minute further than the first day.

spent in the bathroom, while almost no proportion of time during daytime and nighttime.

- The truth-values corresponding to the quantifier *few* and *some* related to the morning bed activity suggest an increase in the time in bed as the experiment

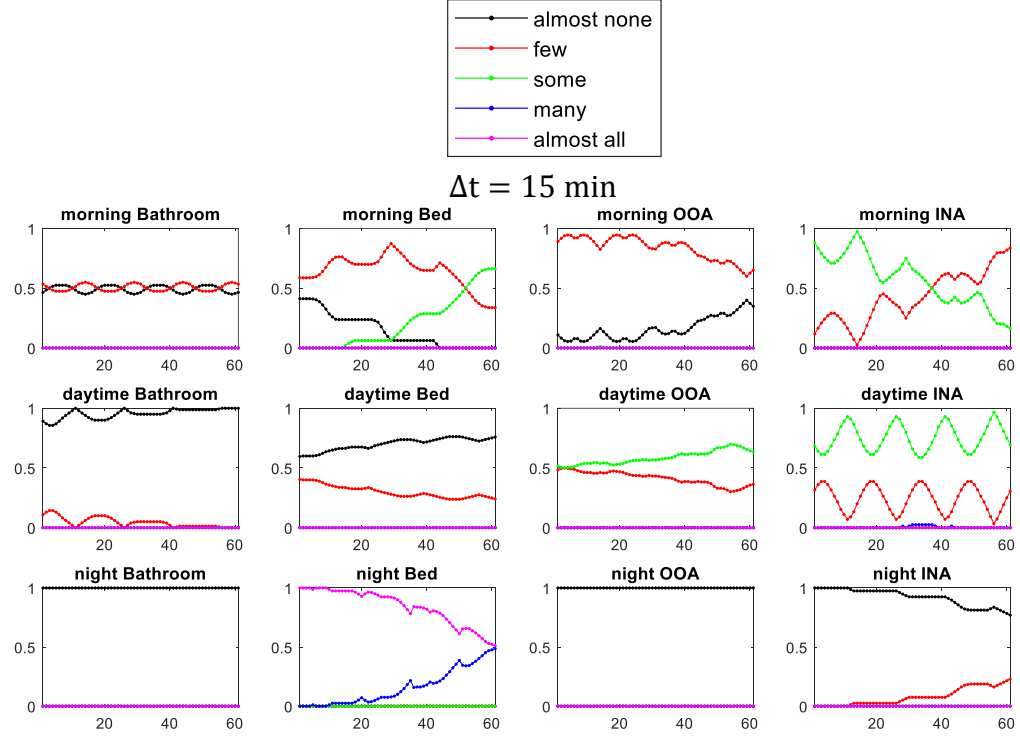


Figure 4.7: Variation of truth-values of LPSs, depicting location pattern of Figure 4.6, of the form: *Q of the Δt slots in R were spent in P*, where $\Delta t = 15 \text{ min}$, Q is the quantifier, R is the time of day (morning, daytime, nighttime) and P is one of the four locations.

proceeds. During the daytime, the proportion of time spent in bed decreases as the time of the activities shift throughout the experiment. This makes sense since the night-time bed activity moves into the morning time and the daytime bed activity moves into the night, as the experiment moves forward. This variation of course, depends on the membership functions representing different times of the day and the behavior would change if their fuzzy set definitions change.

- The OOA activity remains more or less constant during the course of the experiment, with a slight increase in the daytime.

- The truth-values representing the inside apartment activity suggest a small decrease during the morning time and increase during the nighttime, as the truth-value corresponding to the quantifier *some* decreases and *few* increases for the morning duration and that of *almost none* decreases and *few* increases, during the nighttime.

Based on these observations, we conclude that even though the oscillations are unexpected, the variations in the truth-values are intuitive according to the changes in location patterns in Figure 4.7.

4.2.1.2. Experiment: Effect of the quantization factor (Δt)

Another factor that plays an important role in the LPS representation of the activity information is the way we quantize the day, that is the value of Δt . A higher value of Δt will make the calculation of truth-values less computationally expensive since it will result in less objects to being summarized. Also, with a higher value, small changes in the data will have a lesser impact on the location memberships in the Δt slots. On the other hand, a small value of Δt would allow for the slots to represent the actual change in the data on a finer level. To this end, we compute the variation of truth-values when the value of Δt is set lower than 15 minutes. Figure 4.8 shows the variation of truth-values with $\Delta t = 5 \text{ min}$ and $\Delta t = 1 \text{ min}$, respectively.

We see that even though the overall pattern of variation in truth-values is very similar with all the three values of Δt , the change is more gradual with $\Delta t = 5 \text{ min}$ as compared to $\Delta t = 15 \text{ min}$ and even more gradual with the $\Delta t = 1 \text{ min}$. Moreover, smaller the value of

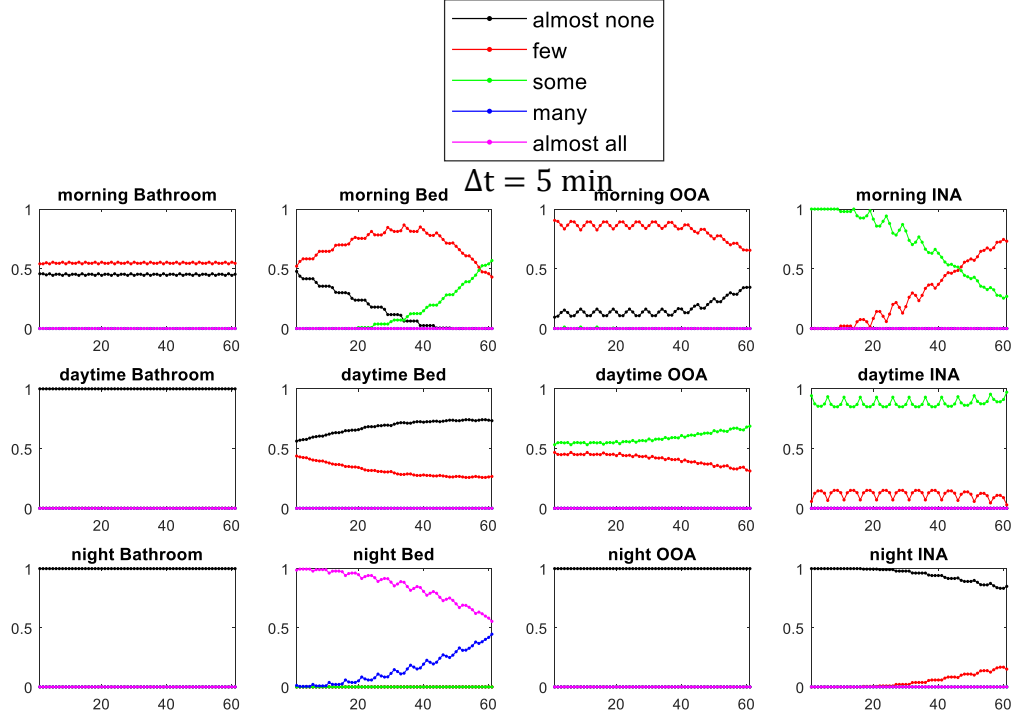


Figure 4.8: Variation of truth-values of LPSs, depicting location pattern of Figure 4.6, of the form: Q of the Δt slots in R were spent in P , where $\Delta t = 5 \text{ min}$, Q is the quantifier, R is the time of day (morning, daytime, nighttime) and P is one of the four locations.

Δt , lesser the oscillations in the truth-values. Although, the variation of truth-values looks more natural for the case of $\Delta t = 1 \text{ min}$, it is computationally expensive since it has five times more slots as compared to the value of 5 min. Hence, considering the value of $\Delta t = 5 \text{ min}$ as an acceptable trade-of between $\Delta t = 15 \text{ min}$ and $\Delta t = 1 \text{ min}$, we use it in the rest of this work.

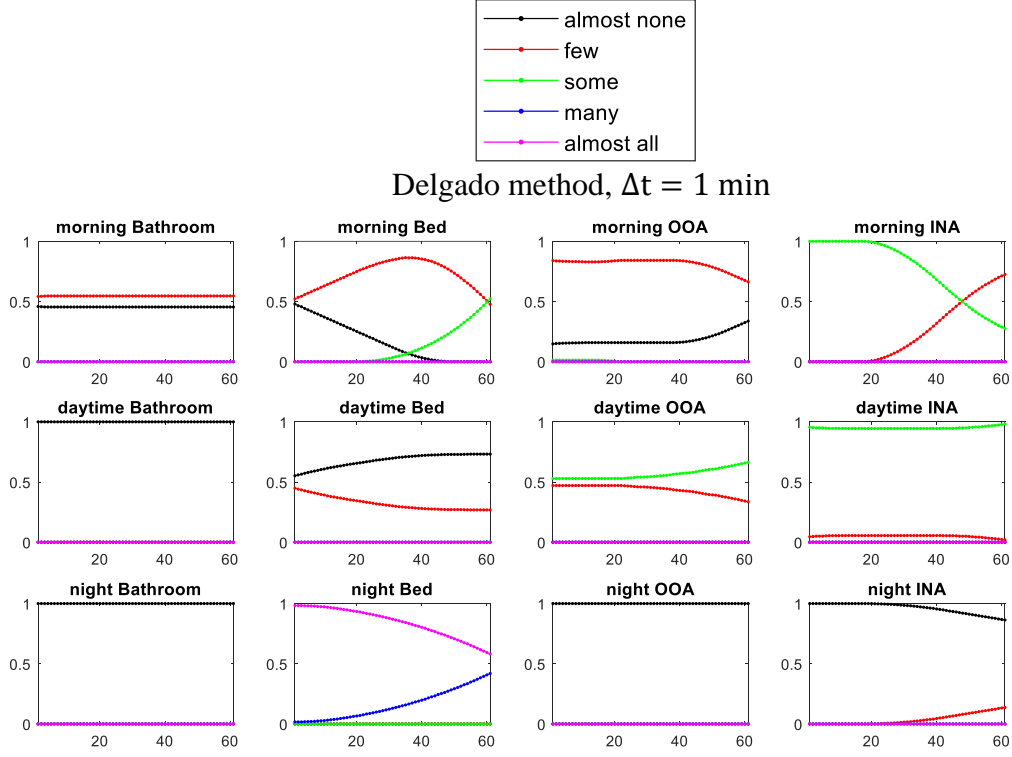


Figure 4.9: Variation of truth-values of LPSs, depicting location pattern of Figure 4.6, of the form: Q of the Δt slots in R were spent in P , where $\Delta t = 1$ min, Q is the quantifier, R is the time of day (morning, daytime, nighttime) and P is one of the four locations.

4.2.2. Dissimilarity between Daily Behavioral Patterns

The fuzzy rule based LPS dissimilarity computation method described in Chapter 3 compares two events based on some property quantified by LPSs. We utilize this method to compare two events with more than one property, for example activity during a day. If we describe activities on two days in this fashion, then we will have two LPS sets with 60 LPSs in each set. Given the quantifiers of Figure 4.5, each combination of the time of day and an activity has five LPSs associated with it. We can use the fuzzy rule based dissimilarity method to compute how dissimilar is each activity at a given time of day. This

would result in $60/5 = 12$ dissimilarity values across all activities and time of day subdivisions. That is, the dissimilarity values individually quantify the comparison between bathroom activity in morning, bathroom activity in the daytime, bathroom activity in the nighttime, bed activity in the morning, bed activity in the daytime, bed activity in the nighttime, and so on for the ONA and INA activities.

Now to compute overall dissimilarity between days, we need an aggregation mechanism to combine the 12 dissimilarities. Here we study the behavior of three such aggregation methods. Let d_{ta} be the dissimilarity between LPS set representing activity a and time of day t for day 1 & 2 (for example, $t = 1$ & $a = 1$ is the bathroom activity in the morning). Then we define three ways to aggregate the dissimilarities between all activities at different times of day by the following equations, where N_T & N_A are the number of times we divide the day in and number of activities represented by the LPS sets, respectively.

$$D_{max} = \max_{t=1:N_T} (\max_{a=1:N_A} d_{ta}) \quad (4.1)$$

$$D_{maxMean} = \frac{1}{N_T} \sum_{t=1}^{N_T} \max_{a=1:N_A} d_{ta} \quad (4.2)$$

$$D_{mean} = \frac{1}{N_T * N_A} \sum_{t=1}^{N_T} \sum_{a=1}^{N_A} d_{ta} \quad (4.3)$$

- In D_{max} we find the time of day and activity that are farthest apart from each other and deem that as the dissimilarity between the two days.

- In $D_{maxMean}$ we find the activity that is most dissimilar between the two days for each time of day. The overall dissimilarity between the two days is their average.
- In D_{mean} the overall dissimilarity is the average of the dissimilarities between each activity at each time of the day.

We compare the three methods by computing overall dissimilarity between day 1 and the rest of the days of Figure 4.6. Figure 4.10 shows the comparison. We observe that with all the three methods, barring the small bumps, the dissimilarity value increases gradually. This is intuitive since the location pattern of Figure 4.6 also varies gradually. Also, as expected, we see the dissimilarity with the *max* operator to be highest among the three, while that with *mean* is the lowest. The dissimilarity with *maxMean* is somewhere in the middle of the two, which seems to be a good trade-off between the maximum and the mean operator.

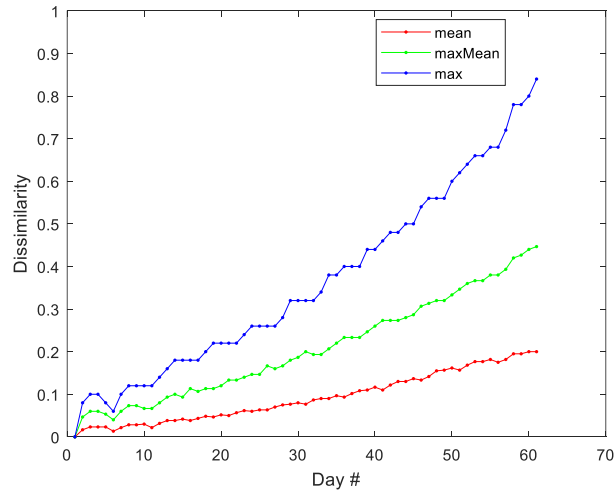


Figure 4.10: Dissimilarity between location pattern of day 1 of Figure 4.6 and the rest of the days with the three aggregation methods

4.2.3. Prototype Behavioral Routine

Using the method to compare LPS representation of daily activity patterns, we delve further into the problem of routine discovery and design another synthetic dataset. The thought behind this dataset is to make it closer to a realistic routine than the previous experiment, and hence, to facilitate a better understanding of the behavioral routine modelling problem.

To this end, we use Mr. Jones’s day activity described in Figure 4.2 in Section 4.2.1 as a starting point and design 30 days of similar activity around it. To get a month’s worth of day-to-day activity, we add small amounts of ‘noise’ (time shifts and variation in duration of each activity) to the activities of Figure 4.2. Moreover, to obtain a slightly different routine on the weekends, we assume that the person spends some part of the afternoon outside attending church or meeting family. Figure 4.11 shows a month of location information of Mr. Jones. We see that on weekends, the person gets up around the same time as weekdays but does not return to the apartment until around 2 PM after leaving at around nine in the morning. To model the behavioral routine of Mr. Jones from the data presented in Figure, we first represent each individual day’s activity in terms of LPSs. This results in 30 sets of summaries, with 60 LPSs in each set. Given the 30 sets of 60 LPSs each, we use clustering to produce a monthly representation of the activity of the resident.

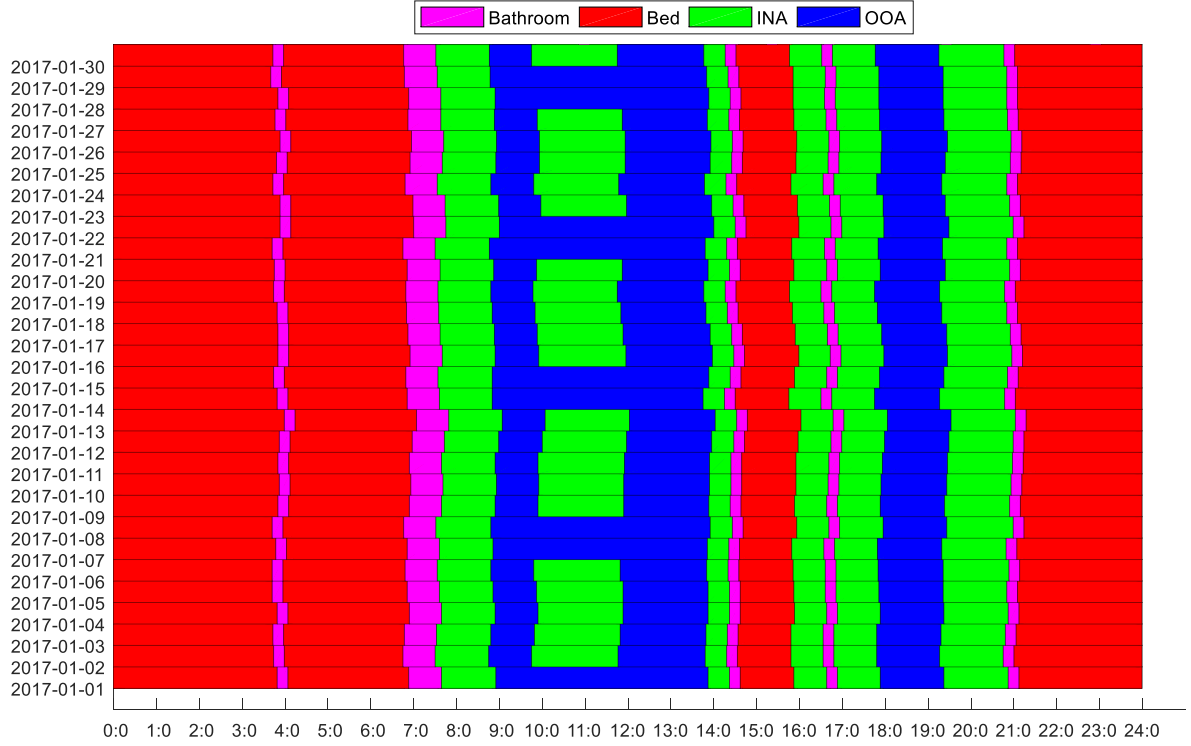


Figure 4.11: Thirty days of location pattern of a Mr. Jones, obtained by adding ‘noise’ to the pattern of Figure 4.2. Note the slightly different morning and daytime pattern on the weekdays and the weekends

Using the maxMean aggregated dissimilarity computation method to compare LPS sets, we perform relational clustering to find groups of days with similar location patterns. As a first step, in order to better understand the grouping nature of the LPS sets, we make use of iVAT (Havens and Bezdek 2012). iVAT is a popular method to study the presence of clusters in datasets. The iVAT image in Figure 4.12 is a reordered representation of the distance between the location patterns of every pair of days presented in Figure 4.11. To obtain the clusters, we can choose from many of the relational clustering techniques available in the literature (Bezdek, et al. 1999). Due to its simplicity, we employ hierarchical single linkage clustering here. Figure 4.13 shows the dendrogram plot using

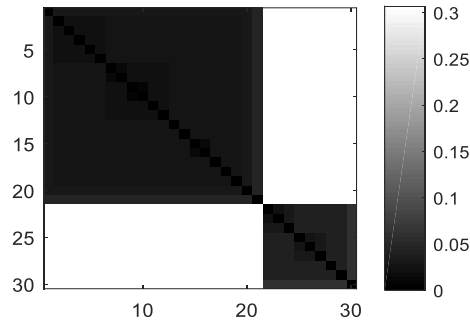


Figure 4.12: iVAT image showing grouping of days in Figure 4.11

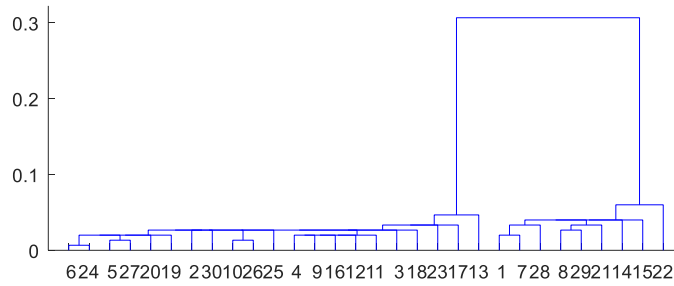


Figure 4.13: Dendrogram plot using single linkage showing dissimilarity between LPS representations of location patterns of days in Figure 4.11

single linkage hierarchical clustering. The two blocks of low dissimilarities in the iVAT image (shown by the darker colors) and the dendrogram plot strongly suggest the presence of two clusters. To this end, we cluster the LPS representation of the 30 days into two clusters using single linkage clustering.

The clustering assigns each of the 30 days of Figure 4.11 into one of the two clusters, as shown in Table 4.2 where each day belongs to either cluster number 1 or 2. It is easy to see all the weekdays fall into one cluster and weekends into the other. The existence of two clusters makes sense looking at the truth-values of all the 60 LPSs over the complete month,

Table 4.2: Table showing the cluster to which each day of Figure 4.11 belongs to.

Date	Day of week	Cluster #	Date	Day of week	Cluster #
'2017-01-01'	'Sun'	2	'2017-01-16'	'Mon'	1
'2017-01-02'	'Mon'	1	'2017-01-17'	'Tue'	1
'2017-01-03'	'Tue'	1	'2017-01-18'	'Wed'	1
'2017-01-04'	'Wed'	1	'2017-01-19'	'Thu'	1
'2017-01-05'	'Thu'	1	'2017-01-20'	'Fri'	1
'2017-01-06'	'Fri'	1	'2017-01-21'	'Sat'	2
'2017-01-07'	'Sat'	2	'2017-01-22'	'Sun'	2
'2017-01-08'	'Sun'	2	'2017-01-23'	'Mon'	1
'2017-01-09'	'Mon'	1	'2017-01-24'	'Tue'	1
'2017-01-10'	'Tue'	1	'2017-01-25'	'Wed'	1
'2017-01-11'	'Wed'	1	'2017-01-26'	'Thu'	1
'2017-01-12'	'Thu'	1	'2017-01-27'	'Fri'	1
'2017-01-13'	'Fri'	1	'2017-01-28'	'Sat'	2
'2017-01-14'	'Sat'	2	'2017-01-29'	'Sun'	2

in Figure 4.14. We see that all the LPSs have very similar truth-values during the nighttime, which is also suggested by Figure 4.11 where the night routine is very similar over the month. The variation among truth-values of LPS corresponding to OOA and INA during the morning and daytime suggest a change in these activities over these days. For example, during the weekends the truth-value associated with the quantifier *some* for the morning and daytime OOA activity is higher than on weekdays, while the truth-value associated with *few* is lower, suggesting an increase in OOA activity. Also, the truth-value corresponding to the quantifier *few* is higher for the morning and daytime INA activity, indicating a decrease in the INA activity.

To extract the routine out of the clustering results, we find the representatives of the two clusters. Two common choices are the centroid or the medoid of the clusters. The centroid

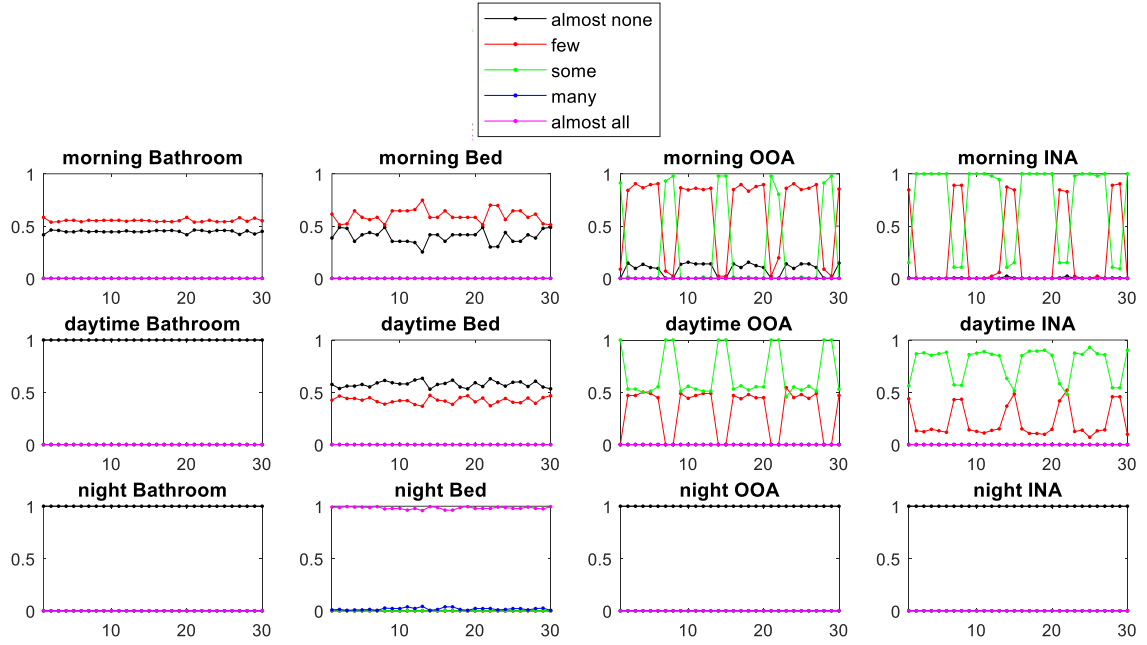


Figure 4.14 Variation of truth-values of LPSs, depicting location pattern of Figure 4.11, of the form: *Q of the Δt slots in R were spent in P*, where $\Delta t = 5 \text{ min}$, Q is the quantifier, R is the time of day (morning, daytime, nighttime) and P is one of the four locations. Notice the significantly different truth-values on weekends and weekdays.

of a cluster is obtained by computing its mean. In our case, the objects being clustered are LPSs sets, hence the average LPS set would be a set with the mean truth-value across all the sets in the cluster. Since we are interested in finding a routine that is explainable, treating the centroid as a cluster representative might not produce a very cohesive routine. That is, since the centroid is a mixture of multiple days of activity, explanation of such a representative might not be very intuitive. On the other hand, the medoid of a cluster is the element that is closest to all the other members of the cluster. If a cluster p_i has c elements, then its medoid is given by Equation 4.4, where $d(s_{ij}, s_{ik})$ is the dissimilarity between object s_{jk} and s_{ik} .

$$p_i = \arg \min_k \left(\sum_{j=1, \dots, c; j \neq k} d(s_{ij}, s_{ik}) \right) \quad (4.4)$$

For the behavioral routine modelling problem, the medoid will be the day whose activity pattern is closest to all the other days in that cluster. We call the medoid of the cluster Linguistic Medoid Prototype (LMP), as was done in (Wilbik, et al. 2014). Next, we focus on the obtained LMPs by describing them in natural language.

4.2.4. Natural Language Explanation of Routine

In this section, we describe our method to produce natural language explanations of the Linguistic Medoid Prototypes (LMPs) that represent the behavioral routine of the resident during the baseline period. We produce natural language explanations individually for each of the LMPs and together they form the linguistic explanation of the routine.

4.2.4.1. *Natural Language description of the LPS set representing daily activity*

Table 4.3a shows LPSs representing the morning activity of one of the cluster prototypes of the aforementioned synthetic routine. The activity over the whole day is represented by an LPS set, similar to Table 4.1, for morning, daytime, and nighttime. We use the LPS set of Table 4.3a to describe our natural language generation process for the morning routine and follow the same procedure for the daytime and nighttime LPS sets. Together, they form the natural language description of the activity over an entire day.

Table 4.3a & b: LPS set representing morning routine of a resident. The table on the right shows the LPS with highest truth-value for each of the four activities.

Quantifier	Daytime	Activity	Truth Value
'almost none'	'morning'	'Bathroom'	0.53
'few'	'morning'	'Bathroom'	0.48
'some'	'morning'	'Bathroom'	0.00
'many'	'morning'	'Bathroom'	0.00
'almost all'	'morning'	'Bathroom'	0.00
'almost none'	'morning'	'Bed'	0.36
'few'	'morning'	'Bed'	0.64
'some'	'morning'	'Bed'	0.00
'many'	'morning'	'Bed'	0.00
'almost all'	'morning'	'Bed'	0.00
'almost none'	'morning'	'OOA'	0.06
'few'	'morning'	'OOA'	0.94
'some'	'morning'	'OOA'	0.00
'many'	'morning'	'OOA'	0.00
'almost all'	'morning'	'OOA'	0.00
'almost none'	'morning'	'INA'	0.00
'few'	'morning'	'INA'	0.27
'some'	'morning'	'INA'	0.73
'many'	'morning'	'INA'	0.00
'almost all'	'morning'	'INA'	0.00

Quantifier	Daytime	Activity	Truth Value
'almost none'	'morning'	'Bathroom'	0.53
'few'	'morning'	'Bed'	0.64
'few'	'morning'	'OOA'	0.94
'some'	'morning'	'INA'	0.73

Figure 4.15 presents an overview of our process to convert sets of LPSs to natural language statements. The first step in our process is to select one LPS per activity. We use the LPS

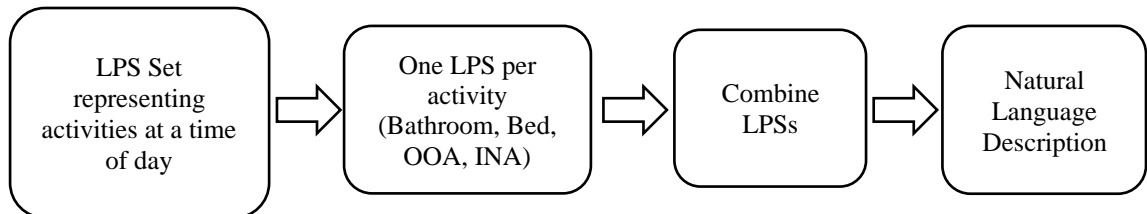


Figure 4.15: Overview of the process to generate natural language description of an LPS set

with highest truth-value, as shown in Table 4.3b. The next step is to combine the four LPSs representing the four activities into a natural language statement. For example, the LPSs of Table 4.3b convey that during the morning, the resident spent a small amount of time in bed, a small amount of time outside the apartment, and almost no time in the bathroom. The rest of the time was spent inside the apartment, but during this time, the resident was not in bed or in the bathroom. We can consider the time inside the apartment when the resident is neither in the bathroom nor in the bed as the time when the resident was in the living area, performing activities such as cooking or watching TV. Based on this reasoning, the LPSs of Table 4.3b can be expressed in the statement: *During the morning, the resident spent almost half of the time in the living area inside the apartment and a little time each in bed, and outside the apartment.* Note that we do not include any information about the time spent in the bathroom since the LPS corresponding to the bathroom activity has the quantifier as *almost none*.

Similarly, we can define natural language statements for different combinations of quantifiers. However, realistically, there are only a few combinations possible, as the LPSs with highest truth value for each activity are not independent of each other. For example, intuitively, it is very unlikely that all the LPSs that have the highest truth-values will have the quantifier *almost all*. To this end, we first enumerate the combinations of quantifiers that are likely to occur together and then define natural language templates for each of them.

Likely Combinations of Quantifiers that can Occur Together

In order to find the combinations of quantifiers that are likely to occur together in the LPSs, we assign a weight to each quantifier, based on its semantic meaning. The weights roughly represent the percentage of information quantified by the quantifier. For example, the quantifier *almost all* quantifies more information than the quantifier *some*. We set the weights as the centroids of the fuzzy sets representing the quantifiers, as shown in Figure 4.16. Intuitively, we can think of the weights as a rough estimate of the fraction of information quantified by the corresponding quantifier. Based on centroids, *almost none*, *few*, *some*, *many*, and *almost all* quantify about 8%, 25%, 50%, 75%, and 92% of the information being summarized.

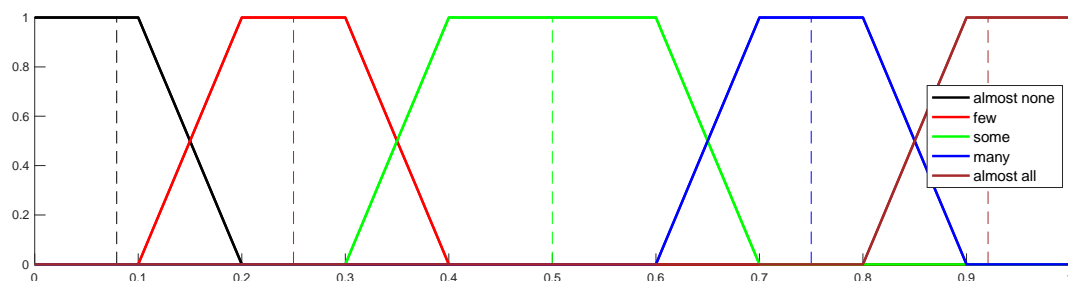


Figure 4.16: Fuzzy set definitions and their centroids for each quantifier

Considering the total information being summarized to be 100%, we find tuples of size 4 (for the four activities: bed, bathroom, INA, OOA) from the five quantifiers that will result in around 100% of the information. Since the weights assigned to the quantifier only roughly represent the fraction of information represented by them, we use a slack variable that relaxes the constraint of 100% information. For example, given a slack variable of

20%, we find tuples of size 4 that represent 80% to 120% of the information. Table 4.4 shows the possible combinations of the four quantifiers and the corresponding natural language templates when using the centroid as the weights, and the slack variable set as 20%. To apply this method to generate natural language summaries corresponding to the LPSs selected for each activity, as in Table 4.4, we first sort the LPSs in the descending order of the corresponding quantifier weights. Then, Quantifier 1 & ACT 1 of Table 4.4 are associated with the LPS with the highest weighted quantifier, Quantifier 2 & ACT 2 are associated with the next LPS and so on. ACT 1, ACT 2, ACT 3, and ACT 4 represent the four activities. To make the natural language statements intuitive, we realize the four activities as: *in the bathroom, in the bed, outside the apartment, in the living area inside the apartment*. The TIME-DAY variable is either morning, daytime or nighttime based on the time of day being summarized.

Table 4.4: Likely combinations of quantifiers and the corresponding natural language templates. ACT 1 is the activity corresponding to Quantifier 1, ACT 2 corresponds to Quantifier 2 and so on, where Quantifier 1 is the quantifier with highest weight based on the centroid

#	Quantifier 1	Quantifier 2	Quantifier 3	Quantifier 4	Natural Language Template
1	many	almost none	almost none	almost none	During the TIME-DAY, the resident spent most of the time ACT1
2	almost all	almost none	almost none	almost none	During the TIME-DAY, the resident spent almost all the time ACT1
3	some	few	almost none	almost none	During the TIME-DAY, the resident spent about half of the time ACT1 and a little time ACT2
4	many	few	almost none	almost none	During the TIME-DAY, the resident spent most of the time ACT1 and a little time ACT2
5	some	some	almost none	almost none	During the TIME-DAY, the resident spent about half of the time ACT1 and the other half ACT2
6	few	few	few	almost none	During the TIME-DAY, the resident spent a little time each at ACT1, ACT2, and ACT3
7	some	few	few	almost none	During the TIME-DAY, the resident spent almost half of the time ACT1 and a little time each ACT2 and ACT3
8	few	few	few	few	During the TIME-DAY, the resident spent a little time each ACT1, ACT2, ACT3, and ACT4

Natural Language Summary of the prototypes of synthetic routine

Based on the procedure described above, the two Linguistic Medoid prototypes of the synthetic routine of Section 4.2.3 is described by the natural language summaries shown in Figure 4.17. The summary rightly describes the change in outside apartment activity on the weekend routine (Pattern 2) as compared to the weekday routine (Pattern 1). The summary of the first pattern says that the resident spent almost half of the morning inside the apartment, while the summary of the second prototype says that the resident spent almost

Pattern 1:

- *During the morning, the resident spent almost half of the time in the living area inside the apartment and a little time each in the bed and outside the apartment*
- *During the daytime, the resident spent about half of the time outside the apartment and the other half in the living area inside the apartment*
- *During the night, the resident spent almost all the time in the bed*

Pattern 2:

- *During the morning, the resident spent almost half of the time outside the apartment and a little time each in the bed and in the living area inside the apartment*
- *During the daytime, the resident spent about half of the time outside the apartment and the other half in the living area inside the apartment*
- *During the night, the resident spent almost all the time in the bed*

Figure 4.17: Natural language summaries of the two linguistic medoid prototypes of the synthetic routine of Figure 4.11

half of the morning outside the apartment. The rest of the information in the natural language summaries is the same for the two prototypes.

4.2.5. Changes in Behavioral Routine

In this experiment, we focus on using LPSs to detect deviation in behavioral pattern, in numeric and linguistic terms. We use the activity pattern presented in the previous experiment as the baseline routine and modify it to come up with a changed activity pattern. Specifically, we assume a person suffering from Urinary Tract Infection (UTI) and modify the baseline routine in order to simulate that. Figure 4.18 shows the behavioral activity pattern of Mr. Jones for a period of 38 days starting from the 1st of January. In the figure, the month of January shows the baseline behavioral pattern of the resident, while the next 9 days, in February, show an increase in bathroom activity, to depict UTI symptoms. Similar to the previous section, we generate LPSs for each day shown in Figure 4.18. The truth-values of the LPSs including the days with increased bathroom activity is shown in Figure 4.19. We see that the truth-values reflect the higher number of bathroom visits during the morning, daytime and nighttime on days when the resident was suffering from UTI. For the bathroom activity, the truth-value of LPSs corresponding to quantifier *almost none* decreases while that for *few* increases. This is also reflected by less amount of time spent in bed throughout the day for the days depicting UTI.

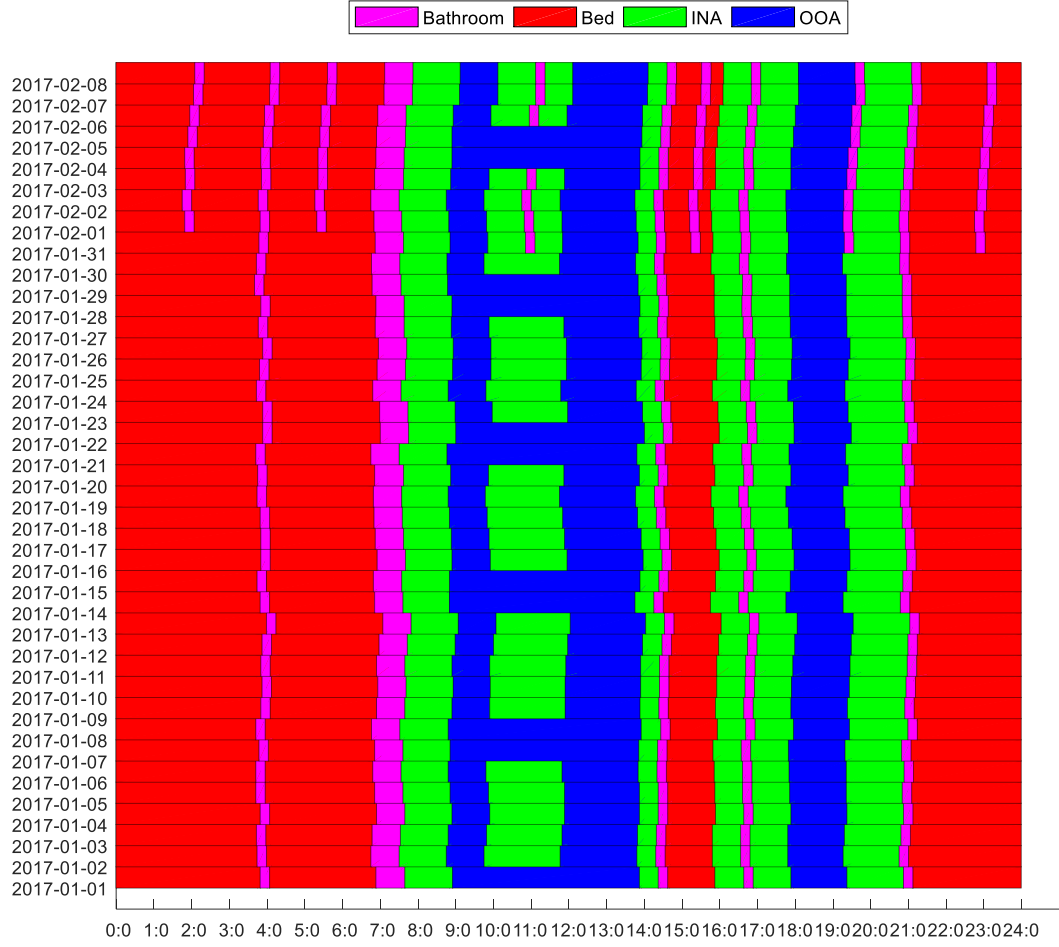


Figure 4.18: Location pattern showing days with increased bathroom activity along with the usual routine during the baseline time period

From the results of Section 4.2.3, we have the LPS sets describing the baseline month, in terms of weekend and weekday routines. To compare the induced routine pattern to the days with increased bathroom activity pattern, we compute dissimilarity between the LMPs and the LPSs sets representing the days shown in Figure 4.18.

Consider $\Pi = \{P_1, P_2 \dots P_C\}$ as the set of C LMPs and L as an LPS set, then the dissimilarity between L and Π is computed using Equation 4.5. That is, the dissimilarity between an

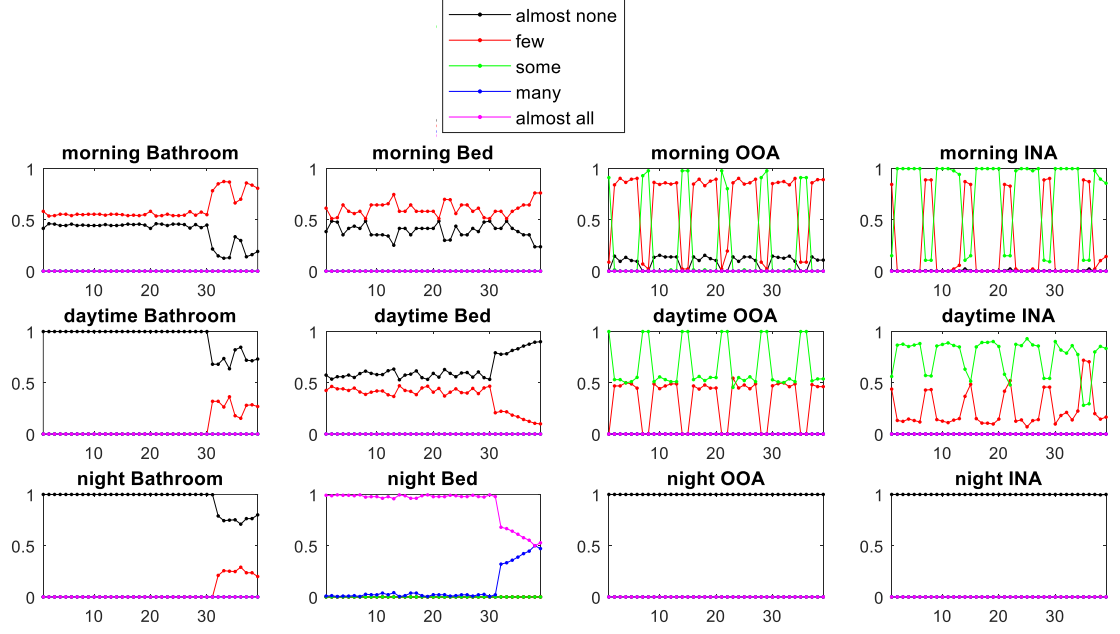


Figure 4.19: Variation of truth-values of LPSs, depicting activity pattern of Figure 4.18, of the form: *Q of the Δt slots in R were spent in P*, where $\Delta t = 5 \text{ min}$, Q is the quantifier, R is the time of day (morning, daytime, nighttime) and P is one of the four locations. Notice the significantly different truth-values on days with increased bathroom activity.

individual LPS set and the prototypes is its dissimilarity with the prototype that is closest to the LPS set.

$$\text{dissim}(L, \Pi) = \min(\text{dissim}(L, P_1), \text{dissim}(L, P_2), \dots, \text{dissim}(L, P_C)) \quad (4.5)$$

The $\text{dissim}(L, P_C)$ is the comparison between the LPS set L and the LMP set P_C , which was defined in Equation 4.2.

Figure 4.20 shows the dissimilarity between the LMPs and each of the 39 days shown in Figure 4.18, using the *maxMean* aggregation. The first 30 dissimilarity values show the comparison between the LMPs and the baseline month. Therefore, we expect the dissimilarity to be very small (very similar). The next 9 days denoted by U in the

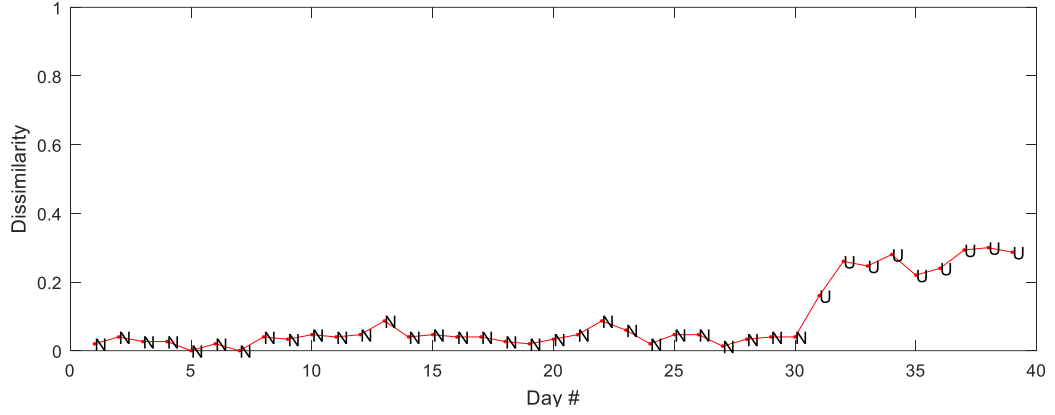


Figure 4.20: Dissimilarity between the Linguistic Medoid Prototypes and the location pattern of the days in Figure 4.18. N depicts days with usual location pattern, while U shows the days with increased bathroom activity.

dissimilarity plot are the days with increased bathroom activity. We observe that these are at a relatively higher dissimilarity than the days in the baseline month.

To further study the reason behind the dissimilarities of Figure 4.20, we plot the output of the Fuzzy Inference System (FIS) that is used to compare LPS sets. Note that, the dissimilarity is nothing but a scaled version of the value that is obtained after performing centroid defuzzification to the output of the FIS. Also, the linguistic dissimilarity is obtained by thresholding this value, as explained in Section 3.1.2. In Figure 4.21, the first subplot presents the output of the FIS for the Bathroom, In Bed, OOA and INA activity for the morning hours, while the next 2 subplots show the comparison for daytime and nighttime, respectively. In all the three subplots, for the first 30 days, the baseline period, the FIS output is close to 0.5 for all locations (which is equivalent to a dissimilarity close to zero). This suggests almost no deviation from the routine prototypes, which is also indicated by the overall dissimilarity plot in Figure 4.20. Table 4.5 shows the linguistic

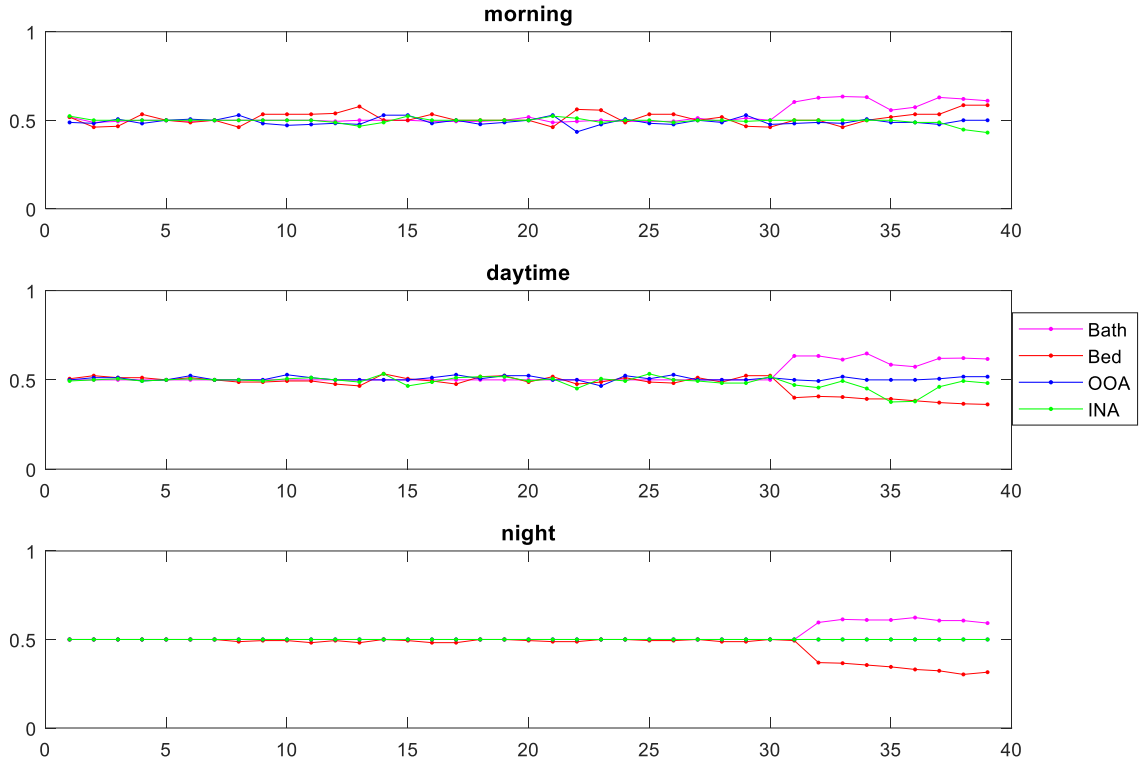


Figure 4.21: Output of the Fuzzy Inference System depicting dissimilarity between the LMPs and the days of Figure 4.18. A value of 0.5 represents zero dissimilarity, while a value above and below 0.5 means an increase and decrease, respectively. The first subplot compares the location pattern during morning, while the next two compares patterns during the daytime and the nighttime.

dissimilarity between the LMPs and the days with increased bathroom activity (31-39).

The first four columns show the dissimilarity between the time spent in each activity in the morning hours, while the next two sets of columns show the same for daytime and nighttime. To increase readability, we do not fill the cells corresponding to the activities that were found to be similar. We use a template based natural language generation methodology to express the changes in the routine linguistically. In Figure 4.22, we show the natural language description of change for February 8.

As compared to the baseline, on February 8:

- *During the morning, there was an increase in time spent in the bathroom*
- *During the daytime, there was an increase in time spent in the bathroom and a decrease in time spent in bed*
- *During the nighttime, there was an increase in time spent in the bathroom and a decrease in time spent in bed*

Figure 4.22: Natural language description of change on February 8 as compared to the baseline routine

Table 4.5: Comparison of the LMPs and the days of Figure 4.18 in linguistic terms. The first block of four columns shows the changes during morning, while the next two blocks depict changes during daytime and nighttime. The cells corresponding to location patterns where no change was detected are left empty for readability.

Day #	morning				daytime				nighttime			
	bath-room	bed	OOA	INA	bath-room	bed	OOA	INA	bath-room	bed	OOA	INA
02-01	inc				inc							
02-02	inc				inc					dec		
02-03	inc				inc				inc	dec		
02-04	inc				inc	dec			inc	dec		
02-05						dec		dec	inc	dec		
02-06						dec		dec	inc	dec		
02-07	inc				inc	dec			inc	dec		
02-08	inc				inc	dec			inc	dec		
02-09	inc				inc	dec				dec		

Using Figure 4.21 and Table 4.5, we draw the following observations:

- The first plot in Figure 4.21 suggests an increase in the bathroom activity during the morning for all the days in February, when the UTI symptoms start showing in Figure 4.18. This is also reported in the first column of Table 4.5, where an increase

in bathroom activity is shown on all the mornings except the weekends (Feb 5 & 6). This makes sense according to Figure 4.18 where the person made less visits to the bathroom on two weekend days when most of the morning was spent outside the apartment.

- The variations in the second plot indicate an increase in bathroom activity and a decrease in bed activity during the daytime for UTI days. This can also be observed in the daytime columns of Table 4.5 which shows an increase in the bathroom activity on all the days except the weekends of UTI symptoms. Again, on these weekends there was more time spent outside the apartment, hence the table does not report an increase on those days. Similarly, a decrease in INA activity is only reported for the weekends of UTI days.
- The third subplot and the columns of Table 4.5 pertaining to the nighttime activities suggest an increase in bathroom activity during the night concurrent with a decrease in time in bed. This is easiest to relate to the activity shown in Figure 4.18, since most of the time during nights is spent in bed during the baseline period, but during the UTI days, some of this bed time gets occupied by bathroom time. We note that for Feb 02 and Feb 09, even though the plot indicates a small increase in the bathroom activity, it is not reflected in Table 4.5. This is because, eventually in the end, we need to apply a crisp threshold to report a linguistic change, and the FIS output values just missed the threshold; hence, they are not reported in the table.

4.2.6. Retrospective Case Studies

In this section, we validate our behavioral routine modelling strategy with the help of retrospective cases studies where we model and detect changes in the behavioral routine and relate those changes to the resident's Electronic Health Records. The data for the case studies comes from apartments of the elderly living at aging in place facilities around Columbia, MO. The behavioral routine modelling approach described above requires an activity map of a resident containing activity information of the residents throughout the day. That is, the algorithm assumes that a separate method pre-processes the sensor data to label the resident's activity into one of the four categories: *in bathroom*, *in bed*, *out of apartment* and *inside apartment*.

4.2.6.1. Data Pre-processing

We employ the activity recognition algorithms described in (Wang 2011) to detect bathroom and out of apartment activities. In (Wang 2011), the sequence of motion sensor firings inside the apartment were used to identify times spent in the bathroom and outside the apartment. They used fuzzy inference systems to interpret motion sensor data at different locations inside the apartment in order to infer the start and end times of bathroom and out of apartment activities.

To recognize the time in bed activity, we use data from the bed sensor. The bed sensor system produces raw text files whenever it detects an activity on bed (based on weight). A file is generated every 10 minutes whenever a person is detected on the bed. Each file is

saved in the database with the time stamp of the 10-minute period included in its name. No data files are produced when the person is not detected on the bed. We use the time stamps included in these filenames to determine whether a resident is in bed.

The inside apartment activity is determined from the raw motion sensor firings around the apartment, barring the bathroom activity. The motion sensor fires every 7 seconds and we use the raw firings as the inside apartment activity to determine the complete activity map of a resident. Figure 4.23 shows the activity map of a resident over a 6 month period, where the times when the resident was in bed are shown in red, bathroom activity in magenta,

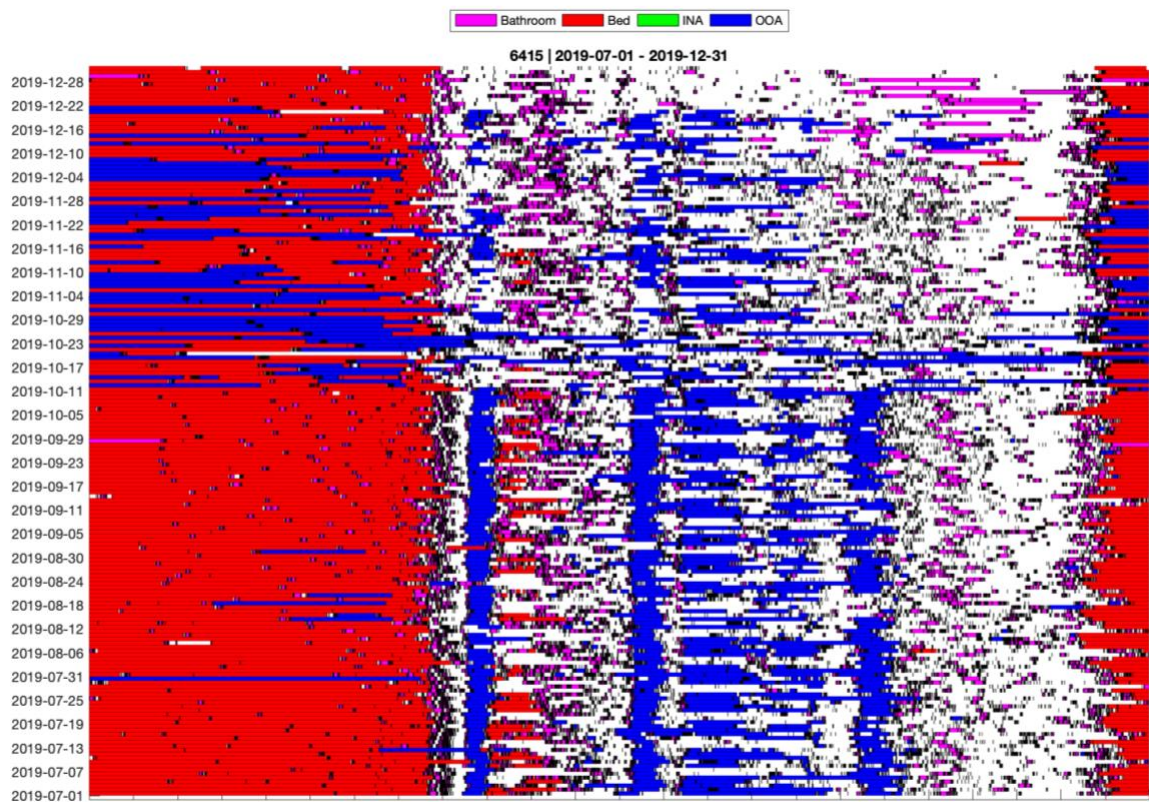


Figure 4.23: Activity map of a resident for a six month period using raw motion sensor firings to determine inside apartment activity

outside apartment activity in blue and the times when the resident was inside the apartment in green. We see that there is almost no green and a lot of white, which are times when no activity was observed (holes in the data). As we mentioned before, we use the raw motion sensor firings to determine inside apartment activity, therefore, we do not have the start and end times of when the resident was inside the apartment. Hence, we do not observe any green activity in the figure and a lot of white space. To fix this, we use the following heuristic: *If same activity is detected both immediately before and after a hole in the data, then fill the hole with that activity.* Using this heuristic, we obtain the activity pattern shown in Figure 4.24. We observe that Figure 4.24 shows a fairly regular activity pattern where the resident wakes up around 8 AM and goes to bed around 11 PM most of the days.

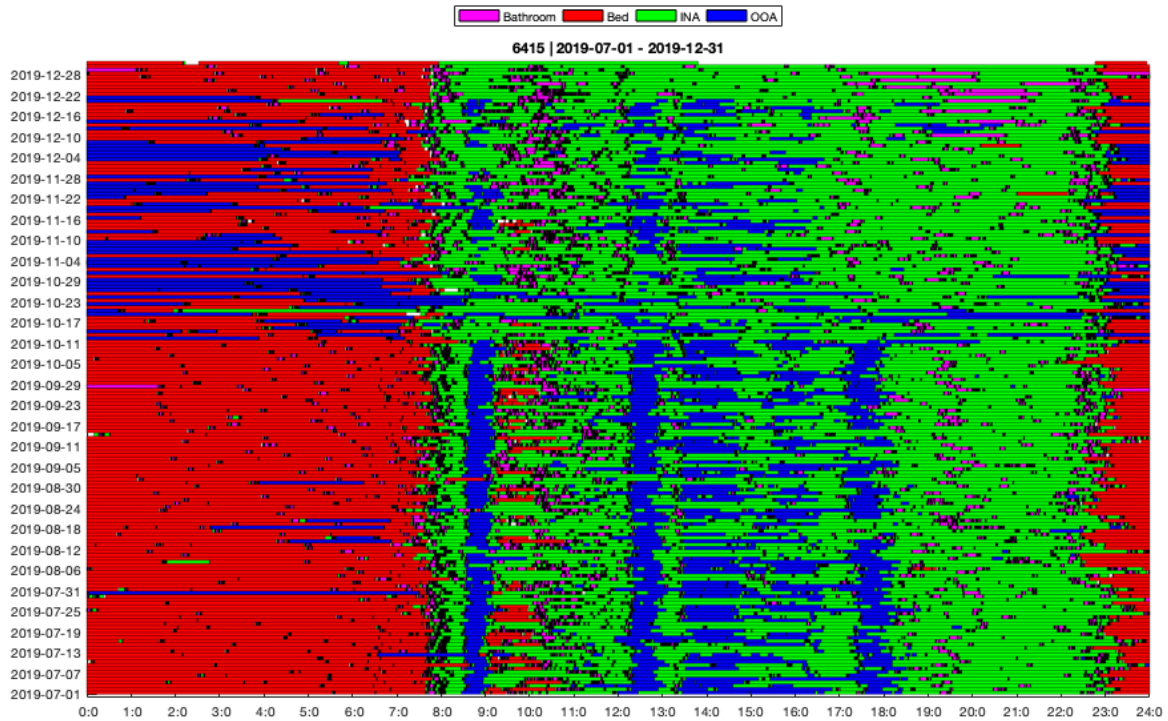


Figure 4.24: Activity map of a resident over a six month period, with overlaps

However, there is one issue that still remains; we see that on most of the nights there is bed activity as expected, but on some nights the figure suggests that the resident was out of the apartment, which seems counter intuitive. It turns out, that on these nights, there were overlapping activities. That is, both in bed activity and out of apartment activity was observed simultaneously. This can be due to a caretaker or some family member leaving the apartment at night when the resident is in bed. To resolve this, we define a priority list among the activities in the following order: Bed, Bathroom, Outside apartment and Inside

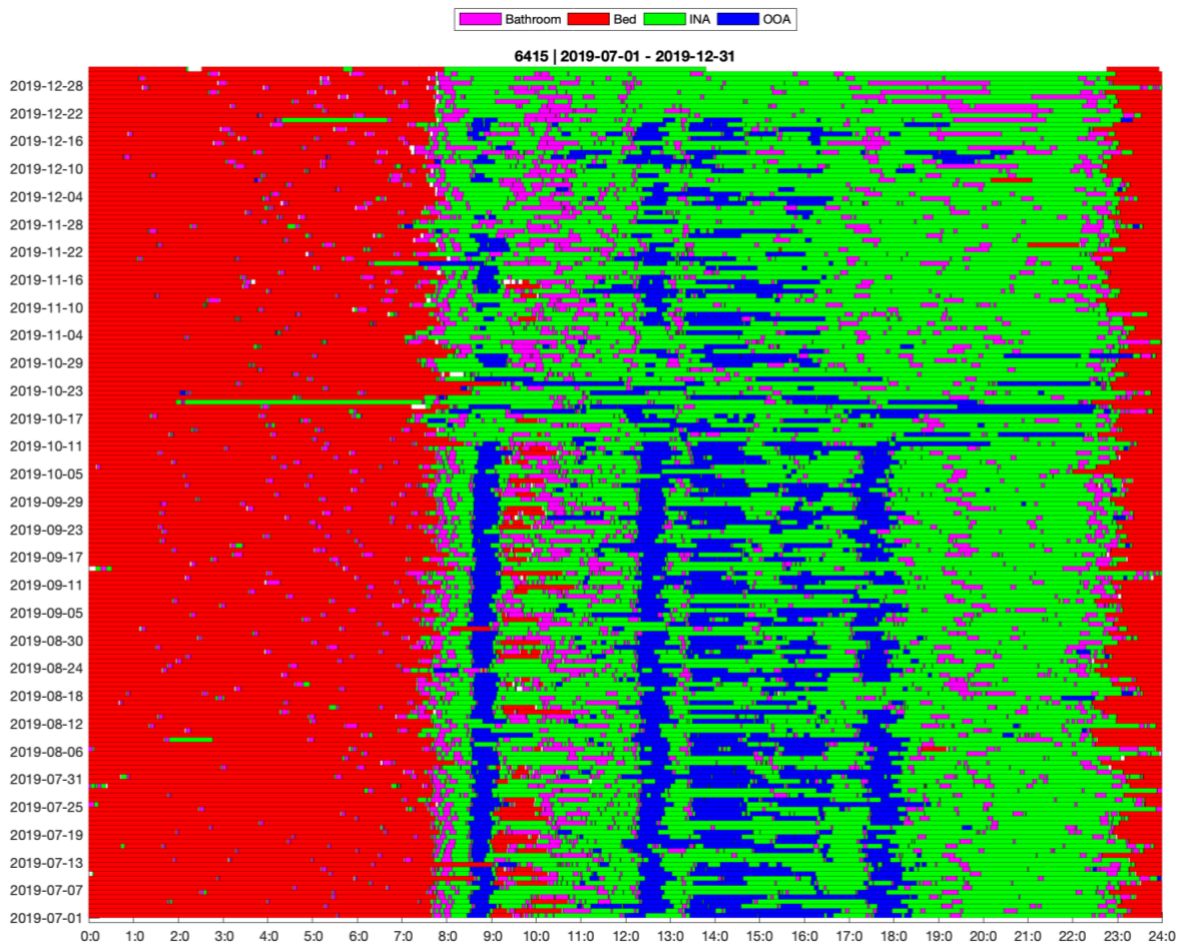


Figure 4.25: Activity map of a resident over a six-month period after data pre-processing

apartment. That is, given an overlapping set of activities, the bed activity is given highest priority, followed by bathroom activity and so on. Applying this heuristic on the data, we obtain the activity map in Figure 4.25. We see a regular nightly pattern throughout the six-month time period. We use the pre-processing steps to obtain the behavioral activity map for all the cases described next.

4.2.6.2. *Case I*

In this case we use our behavioral routine modelling strategy to study and analyze behavioral data pattern for a 97-year-old female resident living at an assisted living facility in Boonville, MO. Figure 4.26 show the behavioral data for this resident for a six-month period starting from July 2019 through December 2019.

The activity pattern of Figure 4.26 suggests that the resident had a fairly stable routine up until the month of October. On almost all days she gets up around 8:00 AM and after performing some personal hygiene activity, she goes out of the apartment around 8:30 AM. Then on some days she takes a nap around 9 AM and goes out around noon, most likely for lunch. Then during the daytime, on some days she goes out of the apartment, while on other days she spends most of her afternoon inside the apartment. She again goes out of the apartment for about an hour in the evening around 5:30 PM and then stays inside the apartment up until bedtime around 11:00 PM.

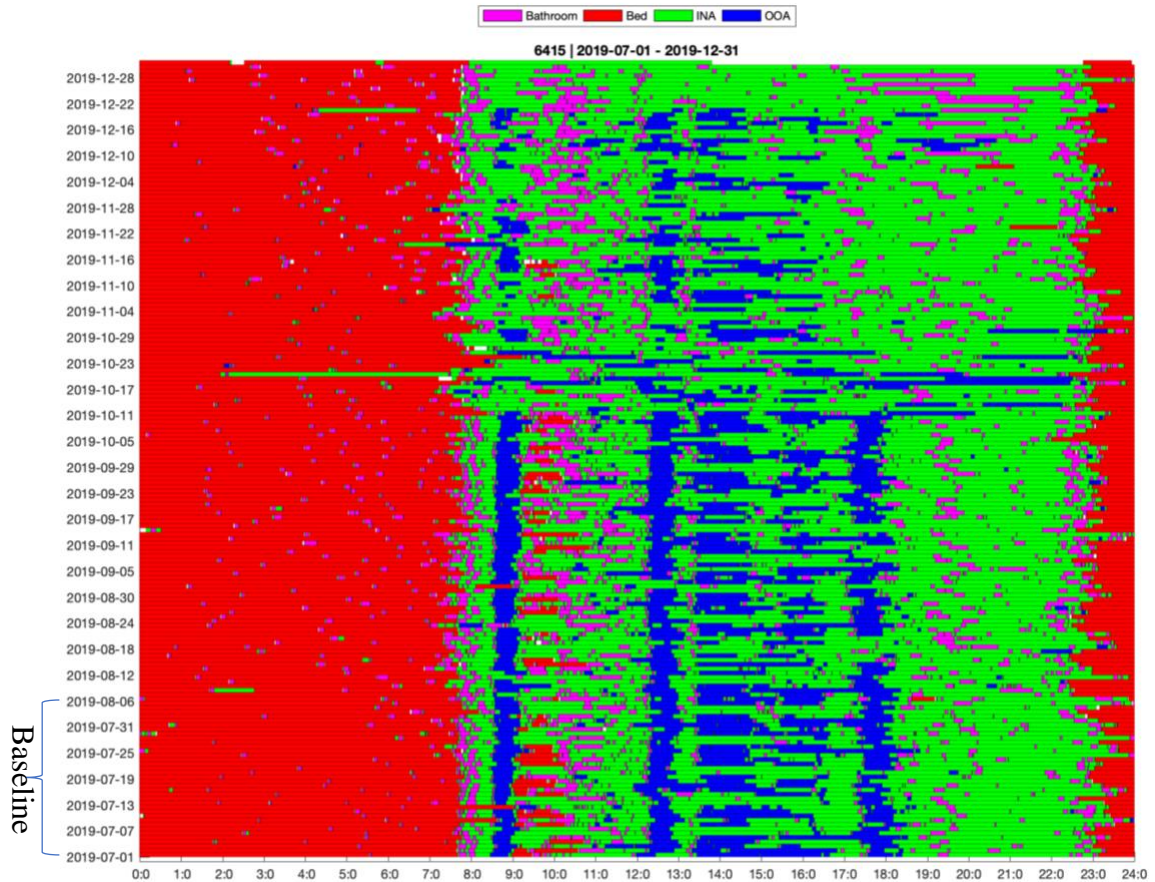


Figure 4.26: Activity map of a resident over a six month period (Case I)

During this six-month period, the resident had two adverse health events that were noted in her Electronic Health Record (EHR). The resident suffered from a Fall on August 12, as described by the following nursing note in the EHR (*verbatim*, some information is redacted due to privacy concerns):

“Resident page CMA . CMA found resident on floor in bathroom on her bottom resident stated she hit her head and lower back. CNA noticed that back on the right side has a scrape.By this afternoon to scrape turn into a bruise which also popped up a bruise on the

left side also across the lower back. Cma offered to some resident to ER to be checked out wasn't refused and said that she was not hurt. CMA notified DON and fax Dr. XXX."

Then on October 12, the resident reported significant pain, noted in the EHR in the following note (*verbatim*):

"Resident has a lot of pain in the right hip refuses to go out and be checked. Resident has been up and moving around in your room need some assistance with getting back in bed this am. Resident has eight both breakfast and lunch in her recliner in the room and has ate 100%. Resident was given aleave at 8 AM this morning."

Based on the activity pattern of the resident during the month of July and absence of any significant adverse health events that were recorded in the EHR during this period, we use this time period to model the behavioral routine of this resident. We then use our method to compare the routine model to the activity pattern over the entire six-month time period.

Behavioral Routine Modelling & Explanation

We follow the procedure described in Section 4.2.3 to model the behavioral routine of the resident based on the data obtained over the month of July. We divide the overall routine of the resident into morning, daytime and nighttime routine, where the boundaries between these three time periods are defined by fuzzy sets that were previously shown in Figure 4.3.

Next, we compute linguistic features representing the activity pattern of the resident on each day from the baseline period. Note that we use the same parameter definitions that were used with the synthetic data in Section 4.2.3. The process of linguistic feature extraction produces LPS sets with 60 LPSs in each set. Next, we compute distance between LPS sets to measure dissimilarity between activity patterns on each pair of days in the baseline time period. This results in a 31x31 distance matrix that is used to perform relational clustering. To visualize the clusters, we use the iVAT image (Havens and Bezdek 2012) and dendrogram plot, as shown in Figure 4.27.

We label each data point in both the iVAT image and the dendrogram plot with the first letter of the day of the week. Both the iVAT and the dendrogram plot show evidence that all the weekend days (Saturdays and Sundays) during the baseline period fall in one cluster, while the rest of the days fall in the other cluster. This suggests that the resident has different behavioral routine pattern on weekends and weekdays.

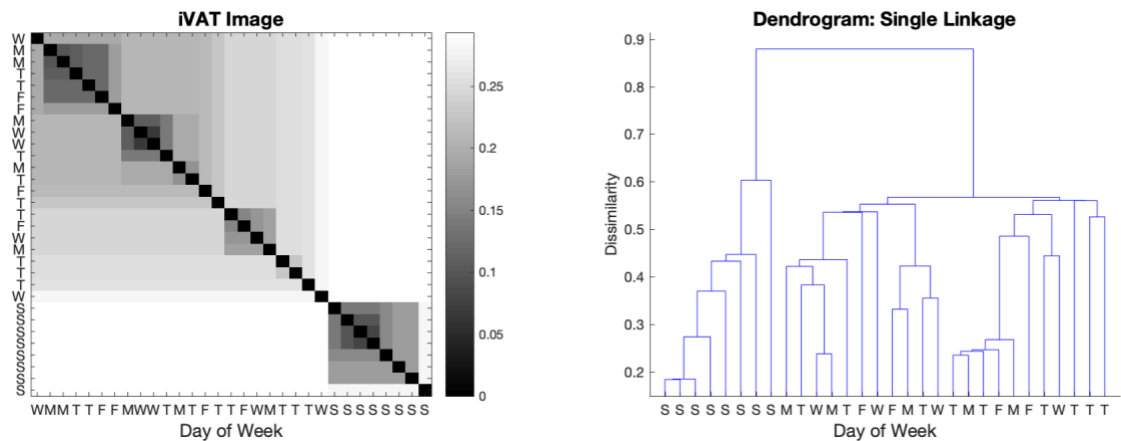


Figure 4.27: iVAT image & dendrogram plot to visualize clusters in the baseline time period (Case I). Each day is labeled by the first letter of the day of the week

Using the above plots as evidence for existence of two clusters, we use the fuzzy relational clustering algorithm, Non-Euclidean Relational Fuzzy c-means (NERF) (Hathaway and Bezdek 1994) to cluster the data into two clusters. Contrary to a crisp relational clustering algorithm like single linkage clustering, NERF provides a membership degree of each point in each of the specified number of clusters. Figure 4.28 shows the memberships of each day in the baseline period into the two clusters. We see that, the weekends have a relatively high membership value in Cluster 2, while the weekdays have a high membership value in Cluster 1.

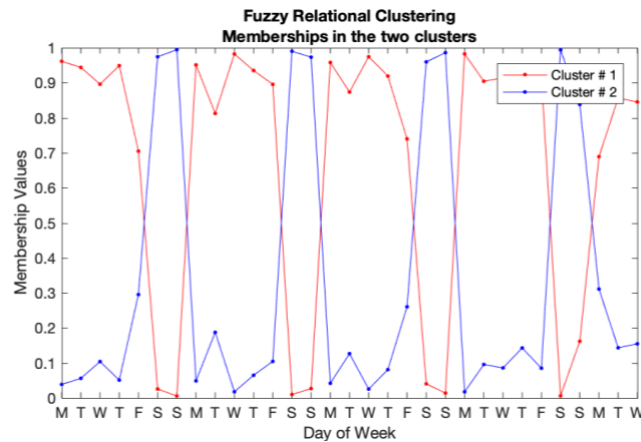


Figure 4.28 Membership of each day in the baseline period in the two clusters using NERF (Case I)

We use the membership values to separate the days into two clusters by assigning a data point to the cluster in which it has a higher membership value. Similar to the synthetic data example, we use the medoids of the two clusters as the cluster prototypes. We use our NLG process to linguistically describe the Linguistic Medoid Prototypes (LMPs) of the two clusters, shown in the box in Figure 4.29. Here pattern 1 corresponds to the weekdays, while pattern 2 corresponds to the weekends.

Pattern 1:

- *During the morning, the resident spent about half of the time in the bed and a little time in the living area inside the apartment*
- *During the daytime, the resident spent about half of the time outside the apartment and the other half in the living area inside the apartment*
- *During the night, the resident spent most of the time in the bed and a little time in the living area inside the apartment*

Pattern 2:

- *During the morning, the resident spent almost half of the time in the living area inside the apartment and a little time each in the bathroom and in the bed*
- *During the daytime, the resident spent most of the time in the living area inside the apartment*
- *During the night, the resident spent most of the time in the bed and a little time in the living area inside the apartment*

Figure 4.29: Natural language explanations of the cluster prototypes in the baseline period (Case I). Pattern 1 is the weekday prototype, while Pattern 2 corresponds to weekends

From the linguistic explanation of the two prototypes, we see that the resident has a slightly different morning routine on weekends than on weekdays, where on weekends the resident spends less time in bed than on weekdays. However, the significant change between the weekend and weekday routine is due to change in her daytime activity. On weekdays, the resident spends almost half of the time inside the apartment during the daytime and the

other half outside the apartment, while on weekends, she spends most of her daytime hours inside the apartment.

Deviation in behavioral routine

Here, we use our method to detect changes in the behavioral routine by comparing the routine model to the activity pattern throughout the six-month time period. We start by computing a numerical dissimilarity between the baseline routine and each day over the study period. Then we delve deeper to quantify the changes in the individual activities at different times of days. We follow this with the linguistic explanation of the changes in the routine.

Overall Dissimilarity

Figure 4.30 shows the overall dissimilarity between the behavioral routine model and each day in the six-month time period. In the figure, each data point in red shows the dissimilarity between the routine model and activity pattern on the corresponding day. In order to facilitate a better visualization of changes in routine, we also draw a smoothing line, which is the LOWESS fit (Cleveland 1979) of the raw dissimilarity value. We also show the baseline time period highlighted in green and the dates of the health events according to the EHR.

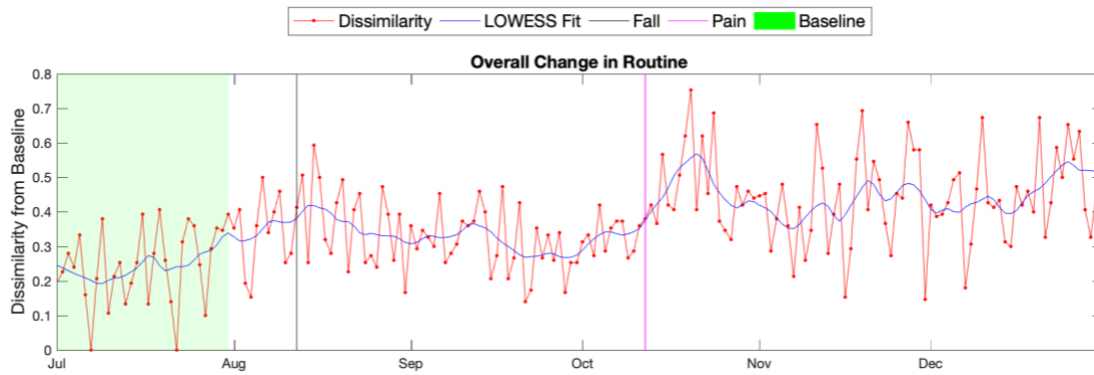


Figure 4.30: Dissimilarity between the routine model and each day in the six month time period (Case I)

We observe an increase in dissimilarity before and around the time when the resident suffered a fall, suggesting a change in routine. This implies a change in the routine of the resident even before the actual fall happened. After the fall, the dissimilarity value remains high temporarily until September, but gets back to the usual value after that, signaling a return to their baseline routine. The dissimilarity increases again around the second event when the resident reported severe pain. This increase in the dissimilarity value is more significant than the change during the event of fall, suggesting that the routine changed more drastically. After this event, the dissimilarity value increases sharply and then drops a little. However, the value is still higher than the baseline dissimilarity value, suggesting a permanent change in the routine after the second health event.

Individual Dissimilarity

In Figure 4.31, we investigate changes in the individual activities performed at different times of day. In the figure, using stacked bar plots, we present three plots showing the

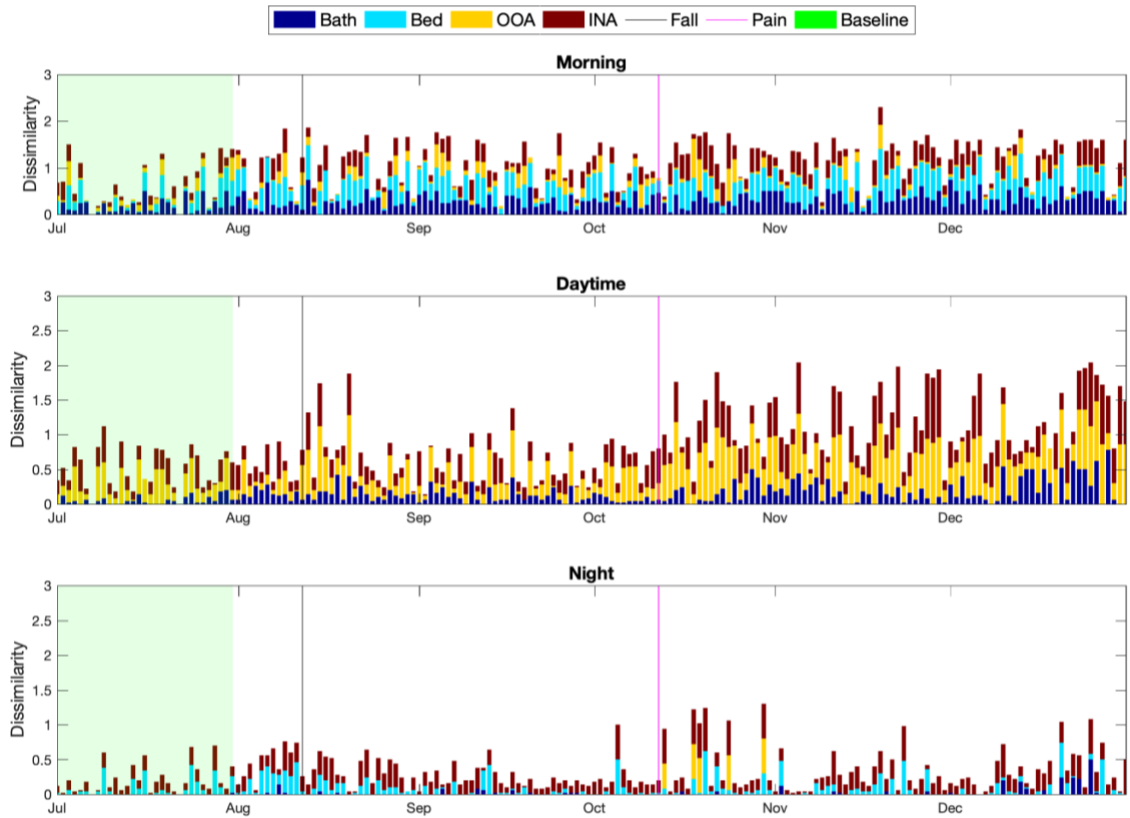


Figure 4.31: Dissimilarity of each of the four activities from the baseline time period for morning, daytime, and night time over the 6 month study period (Case I). Note that each bar represents dissimilarities for the four activities stacked on top of each other

changes in the morning, daytime and nighttime activity patterns of the resident as compared to the baseline routine model. Each stacked bar shows the contribution of each activity in the overall dissimilarity from the routine model, where the colors blue, cyan, yellow, and brown represent dissimilarity in the bathroom activity, bed activity, out of apartment activity, and inside apartment activity, respectively.

Comparing the dissimilarities during the baseline period to the dissimilarities during the rest of the time, we see that the morning routine changes around August and stays this way

until the end of December. The length of the yellow and brown bars in the daytime routine after both the health events suggests that the change in the daytime routine is mainly due to the change in the inside apartment and outside apartment activity. The nighttime routine of this resident stays fairly constant with only some small changes around the two health events. From looking at the individual dissimilarities during morning, daytime and nighttime, we conclude that most of the changes in the overall dissimilarity values around the health events in Figure 4.30 were due to the change in the daytime outside apartment and inside apartment activity pattern of the resident.

Linguistic Dissimilarity

From Figure 4.30 and Figure 4.31 we can conclude that the health events lead to deviations in the normal routine of the resident and moreover, the significant amount of change was due to the change in the daytime routine. However, the direction of change and whether it was significant or not is still unknown. Next, we show that our method is able to detect the direction of the change in linguistic terms. In Figure 4.32, we show the linguistic change as computed by our method during the daytime period. Our method quantifies the change in the following linguistic terms: *decrease*, *significant decrease*, *similar*, *increase*, and *significant increase*. We see that some days have a decrease in the daytime bathroom

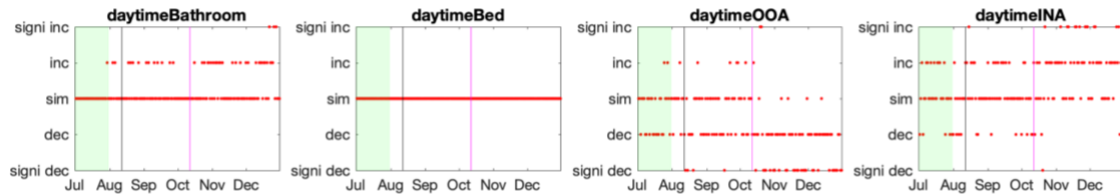


Figure 4.32: Linguistic change in the four activities during daytime (Case I)

activity throughout the study period and a significant increase at the end of December. The most significant change is in the daytime outside apartment and inside apartment activity, where the method detects a significant decrease in the daytime outside apartment and a significant increase in the daytime inside apartment, after the second event. This was also conveyed in EHR notes, where it was noted that: “...*Resident has a lot of pain in the right hip refuses to go out and be checked...*”.

Natural Language Descriptions of Changes in Routine

In Figure 4.33, we provide explanation of changes in the behavioral routines as determined by our method around the end of the study period, on December 25. We see that in the natural language description, along with reporting a change in the activities, the method provides information about their significance in the inside and outside apartment activities during daytime on December 25.

Compared to the baseline routine, on December 25:

- *During the morning, there was an increase in time spent in the bathroom, an increase in time spent in the living area inside the apartment and a decrease in time spent in the bed*
- *During the daytime, there was an increase in time spent in the bathroom, a significant decrease in time spent outside the apartment and a significant increase in time spent in the living area inside the apartment*
- *During the night, there was an increase in time spent in the bathroom and a decrease in time spent in the living area inside the apartment*

Figure 4.33: Natural language description of change in behavioral routine (Case I)

4.2.6.3. Case II

In this case we use our method to analyze 8 months of behavioral activity data for a 95 year old female resident living at an assisted living facility in Columbia, MO. Figure 4.34 presents data for this resident from January to September 2019. The resident was admitted to the hospital on April 27, 2019 and was diagnosed with UTI, as per the following EHR note (*verbatim*, some information is redacted due to privacy concerns):

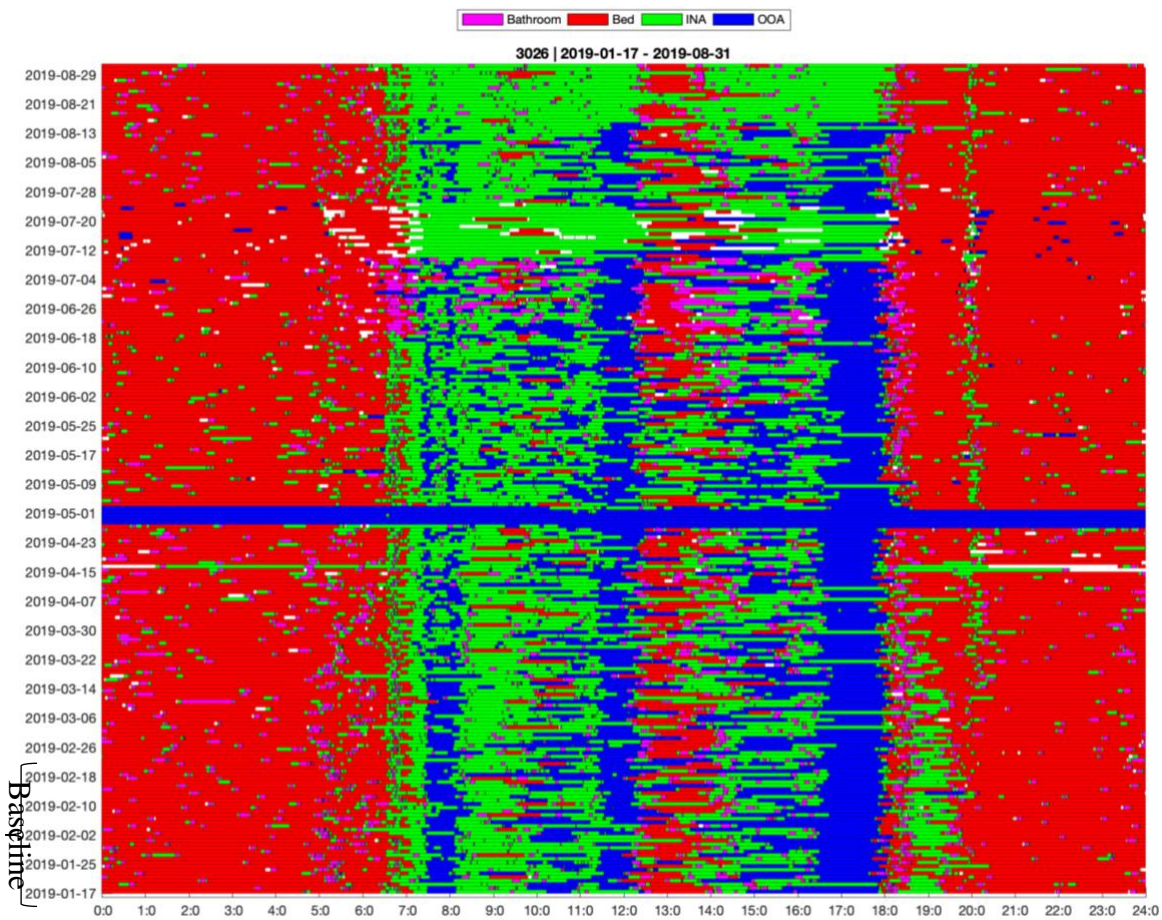


Figure 4.34: Activity map of a resident over an eight month period (Case II)

Resident is very lethargic. Oriented only to name. Resident is denying any pain at the moment. Vitals WNL. XXX (emergency contact) informed of residents condition. Spoke to Dr. XXX and advised to send Resident to the ER for further evaluation. Resident transferred to UHC via EMS for further evaluation, instructed by Dr. XXX. Resident admitted to 4W with an admitting diagnosis of UTI.

The resident suffered from another UTI related episode on June 20, 2019, which was described in the following EHR note (verbatim):

This RN called to resident's room by activity personnel because resident refused activity due to shortness of breath. Upon entering resident room resident denies shortness of breath, but states was short of breath earlier so she thought she would take it easy this afternoon. Vitals WNL see vitals flowchart. Pulse ox 96percent on room air, respirations 18. Resident did state that yesterday she had two episodes of urine release while in hairdressers. This was quite unusual for her. Resident has just returned from physician visit. Will re-evlauate.

Figure 4.34 suggests that the resident had a fairly stable pattern until the UTI diagnosis on April 27 and there were no adverse health events that were noted in their EHR. Based on this, we consider the period from January 17 to Feb 17 as the baseline period for this resident. During this period, the resident usually gets up in the morning around 6:30 AM and then goes out around 7:45 AM. After returning at around 8:15, the resident spends time inside the apartment until going out again at noon. The resident spends some time in bed

during the afternoon, and then goes out again at 5 PM. The resident returns to the apartment at around 6 PM and then retires for the night at 7 PM.

Behavioral Routine Modelling & Explanation

We employ the analysis pipeline used in Case Study I to model the behavioral routine of the resident. To this end, we describe each day's activity with LPSs and then find the distance between LPS representation of each pair of days to find clusters of activity patterns. Figure 4.35 shows the iVAT image and dendrogram plot for the activity pattern during the baseline time period. We see that the iVAT image as well as the dendrogram plot does not provide evidence of clusters in this data. That is, according to our method, the resident does not have multiple well-defined sets of behavioral activity pattern during the baseline period. Therefore, we use the medoid of data over the entire baseline time period as the prototypical routine of this resident.

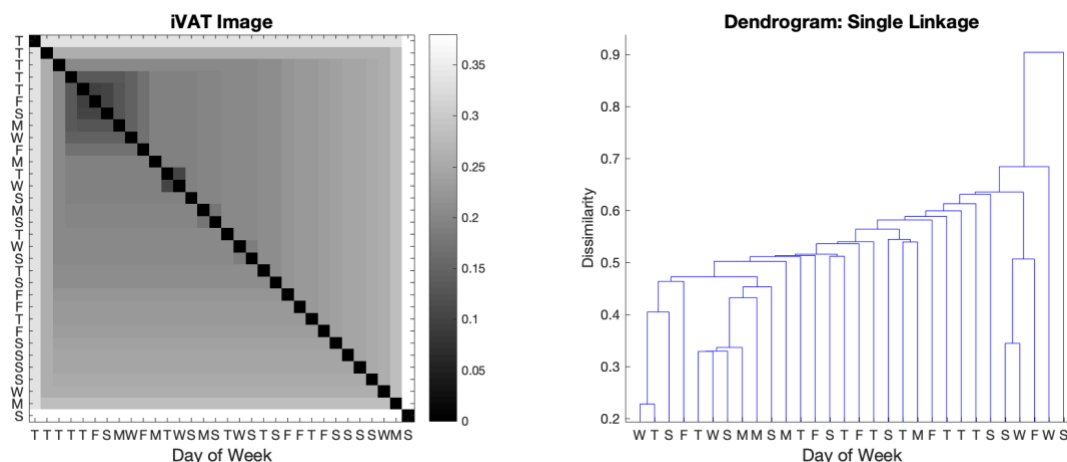


Figure 4.35: iVAT image & dendrogram plot to visualize clusters in the baseline time period (Case II). Each day is labeled by the first letter of the day of week

In Figure 4.36, we show the natural language summary of the obtained LMP. The summary suggests that typically the resident spends most of the morning active inside the apartment, with little time in the bed and outside. During the daytime she sleeps for some time, spends some time outside and inside the apartment, while during the nighttime, most of her time is spent in bed.

- *During the morning, the resident spent almost half of the time in the living area inside the apartment and a little time each in the bed and outside the apartment*
- *During the daytime, the resident spent a little time each in the bed, outside the apartment, and in the living area inside the apartment*
- *During the night, the resident spent almost all the time in the bed*

Figure 4.36: Natural language summary of the prototypical behavioral routine of a resident (Case II)

Deviation in behavioral routine

Using the prototypical routine computed in the previous section, we compare it to the LPS representation of the activity pattern on each day of the entire 8-month study time period. Similar to Case Study I, we first compute the overall numerical dissimilarity and then delve further into the changes in specific activity patterns.

Overall Dissimilarity

Figure 4.37 presents the daily overall dissimilarity from the baseline routine. We see that the dissimilarity increases gradually leading to the UTI diagnosis and hospital admission on April 27, suggesting a gradual change in the resident's routine. Following this, the

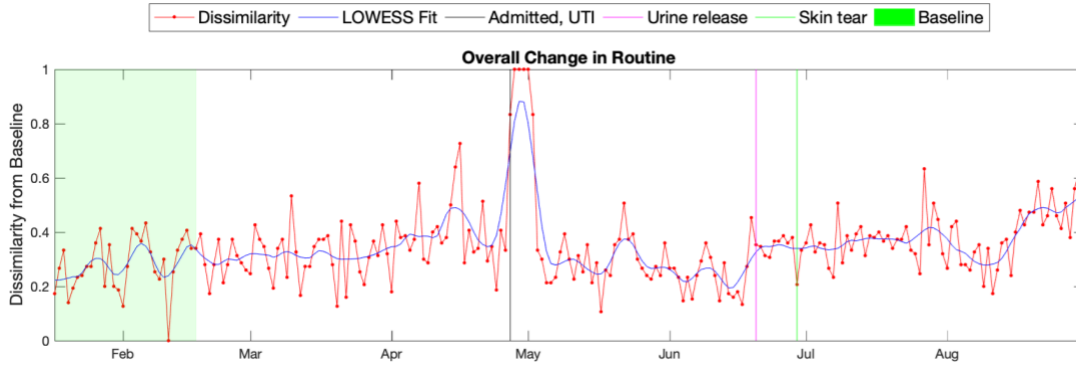


Figure 4.37: Dissimilarity between the routine model and each day in the eight month time period (Case II) dissimilarity drops back to the usual value but increases again around the second UTI related event. No significant change in the dissimilarity was observed around the time when the resident suffered from a skin tear on her left leg at the end of June.

Individual Dissimilarity

In Figure 4.38 we show the changes detected by our method during the morning, daytime and nighttime for each of the four activities, using a stacked bar plot. We see that leading to the UTI diagnosis, a change in the bed and the inside apartment activity during the morning. The method also detects a large change in the inside apartment, outside apartment, and bed activity when the resident was hospitalized, showing that they were in the hospital during early May. A change in the morning bathroom activity is observed for a few days after the second health event when an episode of urine release was noted in mid-June. After this second health event, a long-lasting change in the inside and outside apartment activity is observed during the morning. Similarly, during the daytime, a short-term change in the bathroom activity and a longer lasting change in inside and outside



Figure 4.38: Dissimilarity of each of the four activities from the baseline time period for morning, daytime, and night time over the 8 month study period (Case II). Note that each bar represents dissimilarities for the four activities stacked on top of each other

apartment activity is observed after the second health event. A change in the resident's nighttime activity pattern is also observed due to a changed bed activity, as suggested by the bottom subplot. The plot also suggests a large change in the nighttime bed activity leading to the UTI diagnosis in April, signaling a changed bed routine.

Linguistic Dissimilarity

In Figure 4.39, we present the linguistic change in each of the four activities during morning, daytime, and nighttime over the entire 8-month study period. We observe that there is an increase in the bathroom activity in the morning and daytime after the second UTI related event. During the nighttime, there is a consistent decrease in the bed activity until the second health event. On some mornings after the second health event, a significant increase in the inside apartment activity is reported by the method suggesting that the resident is spending significantly more time inside the apartment.

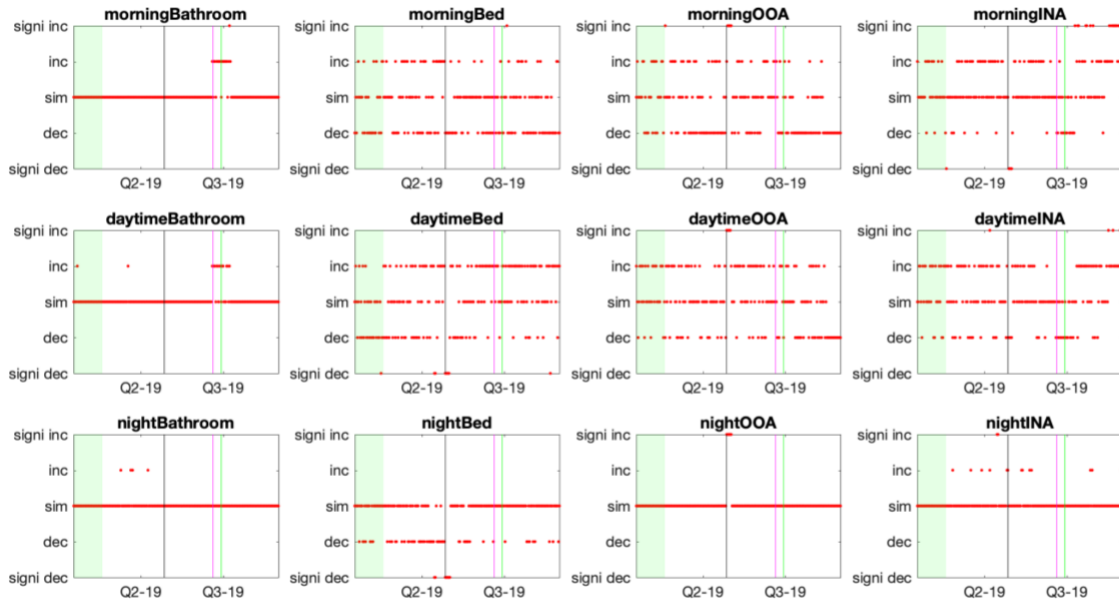


Figure 4.39: Linguistic change in the morning, daytime, and nighttime routine for each of the four activities over the 8-month duration (Case II)

Natural Language Descriptions of Changes in Routine

Next, in Figure 4.40 we provide the natural language description of change on the day of the second health event, when urine release was reported. We see that, as compared to the baseline, the resident spent less time in the living area inside the apartment during both the morning and daytime. Also, an increase in the bathroom activity is observed during daytime. The method reports no significant change in the nighttime activity pattern.

Compared to the baseline routine, on June 20:

- *During the morning, there was an increase in time spent outside the apartment and a decrease in time spent in the living area inside the apartment*
- *During the daytime, there was an increase in time spent in the bathroom, a decrease in time spent in the living area inside the apartment and an increase in time spent in the bed*
- *No change in night routine*

Figure 4.40: Natural language description of change on the day of the second UTI related health event in Case II

4.2.6.4. Case III

In this case, we use our method to study the impact of COVID-19 on the behavioral activity pattern of a 95-year old female resident living at an assisted living facility in Columbia, MO. Figure 4.41 shows about 3 months of data for this resident starting from February through April, 2020. We see that the resident's activity pattern changes around the middle of March when she starts spending less time outside and more time inside the apartment.

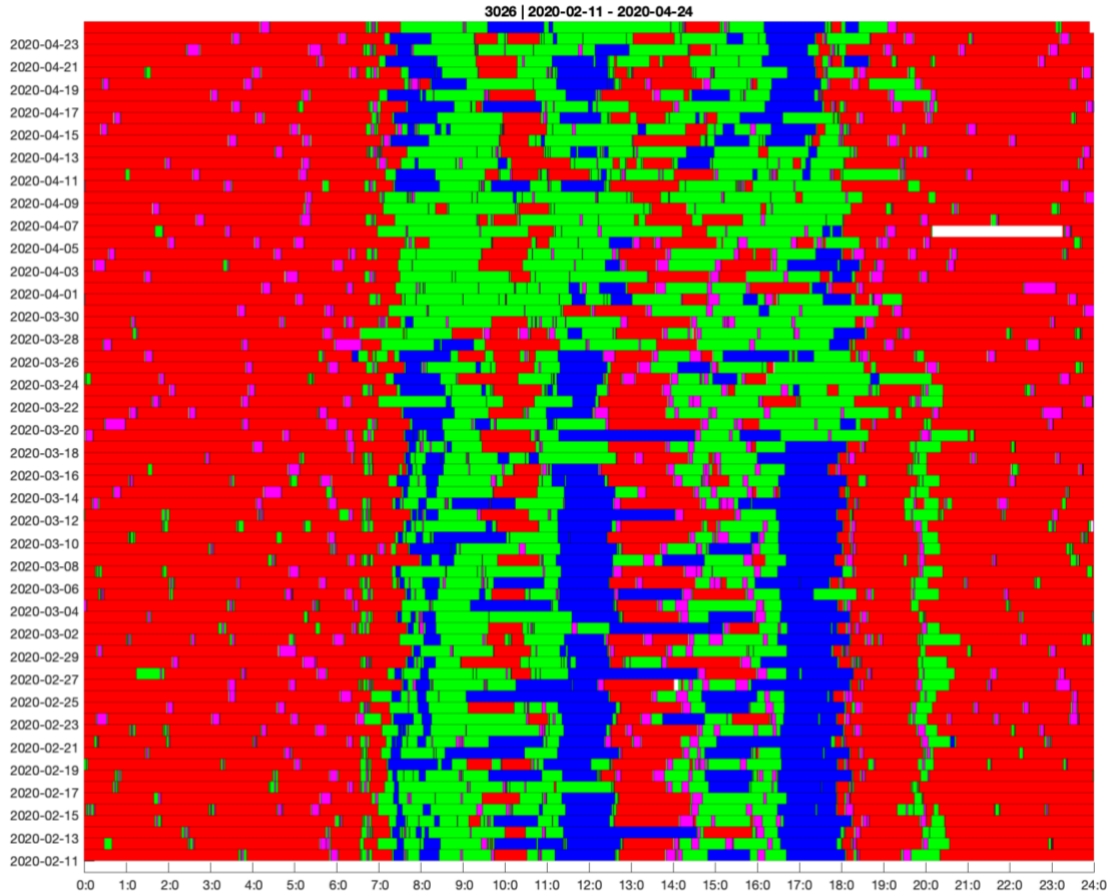


Figure 4.41:Activity map of a resident over a 3 month period, showing the change in the acidity pattern after COVID-19 lockdown on March 13 (Case III)

Behavioral Routine Modelling & Explanation

We model the resident's normal routine using the data obtained during the month of February and then use it analyze changes in the behavioral pattern due to the COVID-19 lockdown, which took place on March 13. We follow our behavioral routine modelling and deviation detection pipeline that was used previously. Figure 4.42 shows the iVAT image

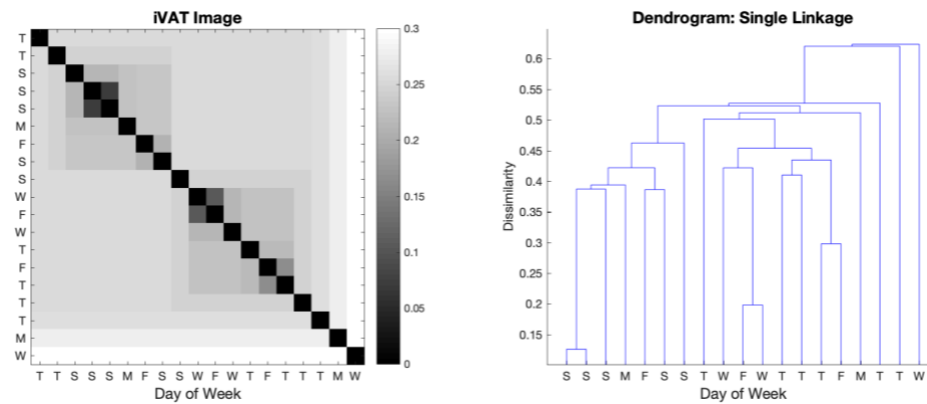


Figure 4.42: iVAT image & dendrogram plot to visualize clusters in the baseline time period (Case III). Each day is labeled by the first letter of the day of week

and the dendrogram plot obtained over the month of February. We see that the iVAT image, as well as the dendrogram plot does not present a clear evidence of clusters in the data. Hence, we use the medoid of the data over the entire month of February as the prototypical routine of this resident during this time period. Figure 4.43 shows the natural language description of the obtained Linguistic Medoid Prototype. We see that in her normal routine, the resident spends the majority of her morning active inside the apartment, while about

- *During the morning, the resident spent almost half of the time in the living area inside the apartment and a little time each in the bed and outside the apartment*
- *During the daytime, the resident spent almost half of the time in the bed and a little time each outside the apartment and in the living area inside the apartment*
- *During the night, the resident spent almost all the time in the bed*

Figure 4.43: Natural language summary of the prototypical behavioral routine of a resident during the baseline period (Case III)

half of her daytime hours are spent in bed. The resident goes outside the apartment for a small duration of time during both the morning and the daytime.

Deviation in behavioral routine

Here, we use our method to quantify and describe changes in the resident's behavioral routine after the COVID-19 lockdown.

Overall Dissimilarity

Figure 4.44 presents the dissimilarity between the model of the behavioral routine and the activity pattern of the resident on each day over the entire 3-month time period. The plot shows an increase in the dissimilarity value about a week after the COVID-19 lockdown on March 13, suggesting a change in the resident's behavioral activity pattern. We observe that the dissimilarity value remains at this level until the middle of April, when it seems to

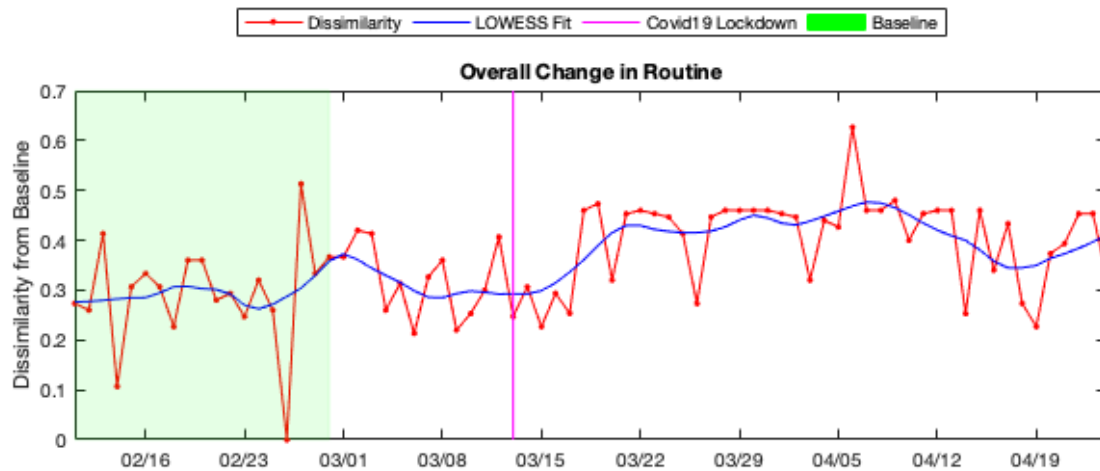


Figure 4.44: Dissimilarity between the routine model and each day in the three month time period (Case III)

be dropping down to its usual value, signaling that the resident is getting back to their routine after about a month of lockdown.

Individual Dissimilarity

Figure 4.45 shows the changes detected by our method during the morning, daytime, and nighttime for each of the four activities, using a stacked bar plot. During the morning hours, we observe that the bed activity changes significantly after the month of February, even more so after the COVID-19 lockdown. A change in the inside apartment and outside apartment activity is also observed after March 13. Similarly, a very high change is the

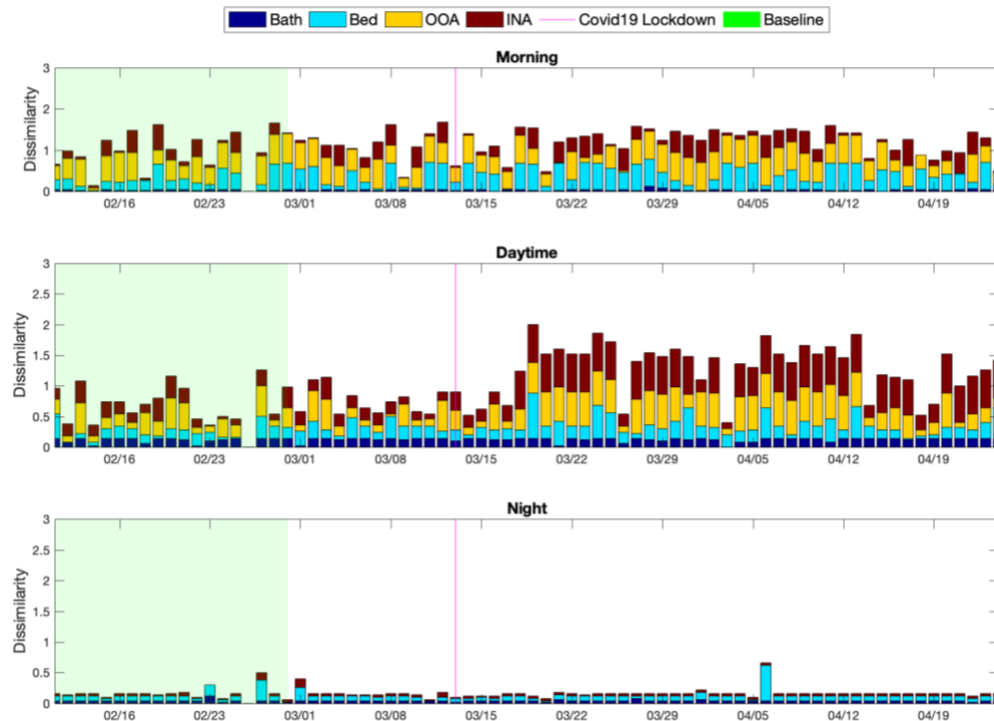


Figure 4.45 : Dissimilarity of each of the four activities from the baseline time period for morning, daytime, and night time over the 3 month study period (Case III). Note that each bar represents dissimilarities for the four activities stacked on top of each other

daytime inside and outside apartment activity is observed about a week after the lockdown. No significant change in the nighttime routine is observed throughout the time period.

Linguistic Dissimilarity

Here, we present the change detected by our method in linguistic terms. In Figure 4.46, we display the linguistic change in each of the four activities during the morning, daytime, and nighttime. The figure shows a significant increase in the time spent in bed on many mornings after the COVID-19 lockdown. Also, a significant decrease in the time spent outside the apartment is observed on many mornings. Relating this to the patch plot of Figure 4.41 shows that the method was able to detect that after the lockdown, the resident starts getting up later than usual. During the daytime, many days show a significant increase in the time spent inside the apartment during the lockdown, while a decrease in both the time spent in bed and time spent outside the apartment is observed.

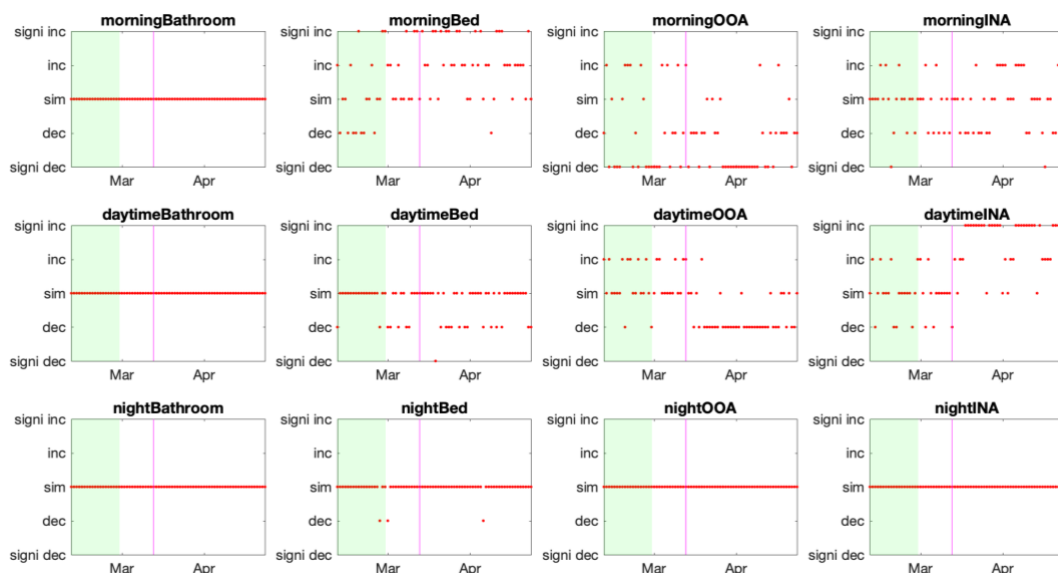


Figure 4.46: Linguistic change in the four activities during morning, daytime, and nighttime (Case III)

Natural Language Descriptions of Changes in Routine

In Figure 4.47, we present the natural language description of change in the behavioral routine, as produced by our method a week after the lockdown, on March 21. We see that the description notes a significant increase in the time spent in bed in the morning and significant increase in the time spent active inside the apartment during the daytime.

As compared to the baseline, on March 21:

- *During the morning, there was a decrease in time spent in the living area inside the apartment and a significant increase in time spent in the bed*
- *During the daytime, there was a decrease in time spent outside the apartment, a significant increase in time spent in the living area inside the apartment and a decrease in time spent in the bed*
- *No change in night routine*

Figure 4.47: Natural language description of change a week after the COVID-19 lockdown in Case III

4.3. Physiological Routine

The sensors installed in the apartments are capable of providing continuous measurements of physiological parameters such as the pulse rate and the respiration rate. Modelling and detecting changes in the usual patterns of these measurements can help assess important health changes in the elderly. To this end, we use our routine modelling approach presented in Section 4.1, to model the physiological routine of the residents.

4.3.1. Using LPSs to Represent Physiological Measurements

To illustrate our physiological routine modelling approach, we use a running example comprising of pulse rate data obtained from an apartment of an elderly living in a sensor equipped facility. As mentioned before, the apartments of elderly used in this study are equipped with various types of sensors, like motion sensors to monitor activity, bed sensors to monitor their sleep and depth sensors to monitor their walking parameters (Rantz, et al. 2013). The sensors installed on the bed of the residents can measure their pulse rate at 15-

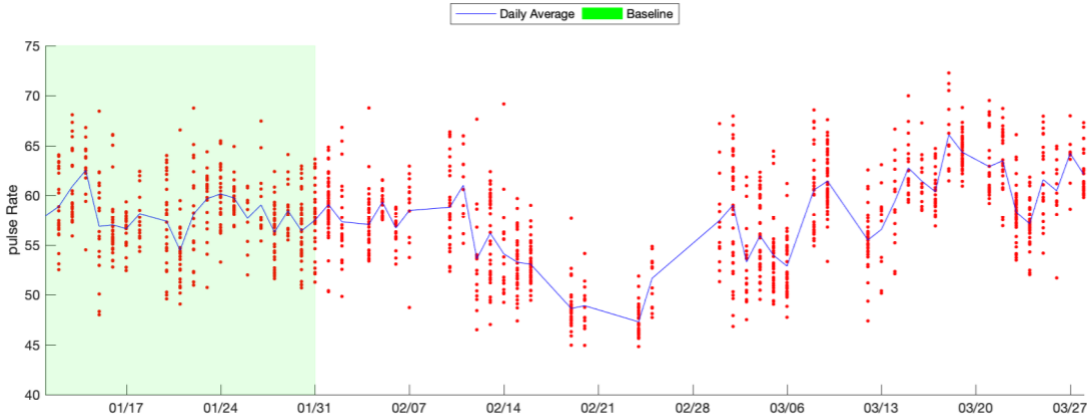


Figure 4.48: Daily Pulse Rate measurements of a resident for a period of three months. The blue line shows the daily average pulse rate

minute intervals. Various methods to measure pulse rate from the bed sensor data have been introduced in the literature. Here, we use a technique that employs Hilbert Transform and was published in (Su, et al.). Figure 4.48 shows 15-minute interval pulse rate measurements of a 90-year old male resident for a period of 3-months starting from January through March 2016. Each column in the figure shows all the pulse rate measurements recorded on that day. The blue line in the plot is the average daily pulse rate. The resident was diagnosed with Congestive Heart Failure (CHF) on February 25. If we compare the measurements over the month of January and late February, then we see that during January, most of the days have measurements in the range of 60 BPM, while during late February the majority of the measurements are around 50 BPM. Based on this observation, we model the resident's regular pulse rate pattern over the month of January and compare it with the measurements over the entire three-month time period. Next, we describe the use of Linguistic Protoform Summaries to model the pulse rate pattern or the pulse rate routine over the baseline period. We then use our FIS based dissimilarity method to measure deviations from this pulse rate routine and explain the reason behind the changes in natural language.

Linguistic Protoform Summaries of the form, ***Few** pulse rate measurements are **high***, are well suited at summarizing each days pulse rate measurements. First, this form of LPS can record the distribution of the pulse rate measurements over a day in detail. Second, the terms *high/low/normal* pulse rates are better represented by fuzzy sets instead of crisp thresholds. Unlike the behavioral routine where we used type II LPS to represent location

patterns at different times of day, to model physiological routines, we use type I LPS with the protoform: ***Q** pulse rate/respiration rate measurements are **P*** where *Q* and *P* are the quantifier and the summarizer, respectively. Since the pulse rate and respiration rate are obtained from the bed sensor while the person is in bed, the substantial number of patterns are recorded during the nighttime. Hence, we do not model the patterns separately for morning, daytime and nighttime; instead, we construct the model using all the measurements taken together. The definitions of the quantifiers are the same as those used in previous experiments (Figure 4.5). The summarizers are composed of fuzzy sets used to represent *low*, *normal* and *high* pulse rate and respiration rate, as shown in Figure 4.49 & Figure 4.50, respectively.

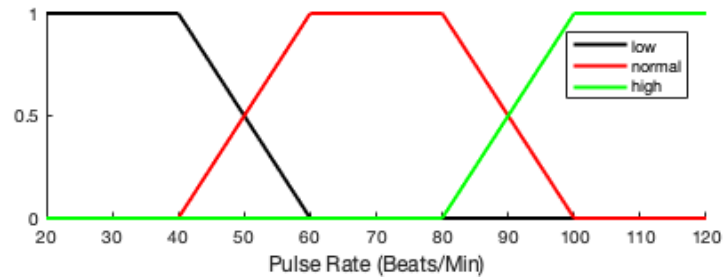


Figure 4.49: Fuzzy set definitions of low, normal, and high pulse rate

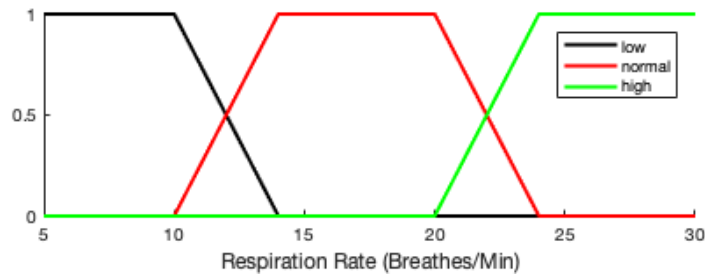


Figure 4.50: Fuzzy set definitions of low, normal, and high respiration rate

4.3.2. Prototype Pulse Rate Routine

In order to find the pulse rate routine over a baseline time period (the month of January in Figure 4.48), we summarize the pulse rate measurements for each day with the LPSs of the form mentioned previously. That is, for the measurements recorded over each day, we use the semantic truth-value method for type I summaries to generate LPSs comprising of all the combinations of the summarizers and the quantifiers. For five quantifiers of Figure 4.5 and 3 summarizers of Figure 4.49, measurements over each day are represented by $5 \times 3 = 15$ LPSs. Similar to the behavioral routine modelling, we use our FIS based dissimilarity measure to perform NERF c-means over LPS sets of each day and use the Linguistic Medoid Prototypes as the cluster representatives. To estimate the number of clusters in this dataset, we draw the iVAT image and dendrogram plot, shown in Figure 4.51. The iVAT image presents 6 clusters, 4 of which have very few elements and the rest of the two clusters are large. Based on this, we use the NERF c-means to cluster the data into two clusters.

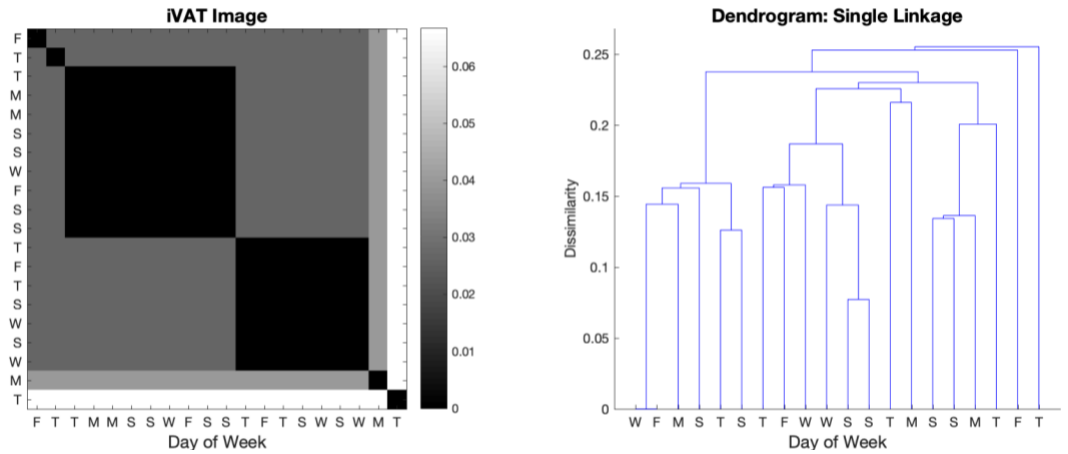


Figure 4.51: iVAT image and dendrogram plot to visualize clusters in pulse rate pattern during the baseline period (Case I)

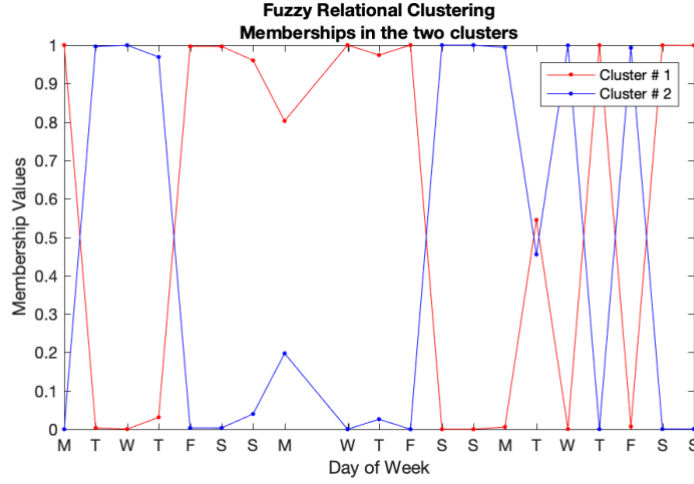


Figure 4.52: Membership of each point in the baseline period into the two clusters obtained by using NERF

Figure 4.52 shows the membership of each point in the two clusters. Table 4.6 lists the prototypes for the two clusters. We see that the truth-values of the LPSs in the two

Table 4.6: Linguistic Medoid Prototypes obtained using NERF

#	Quantifier	Summarizer	Prototype 1 Truth-Values	Prototype 2 Truth-Values
1	almost none	low	0.6	0.8
2	Few	low	0.4	0.2
3	some	low	0.25	0.05
4	many	low	0	0
5	almost all	low	0	0
6	almost none	normal	0	0
7	few	normal	0	0
8	some	normal	0.25	0.05
9	many	normal	0.4	0.2
10	almost all	normal	0.6	0.8
11	almost none	high	1	1
12	few	high	0	0
13	some	high	0	0
14	many	high	0	0
15	almost all	high	0	0

prototypes are very similar. Based on the truth-values, the second prototype represent the days that have a slightly fewer low pulse rate measurements than prototype 1. Both the LMPs suggest that there were almost no days in this period with high pulse rate recordings.

4.3.3. Natural Language Explanation of Routine

In this section, we describe the procedure to produce natural language explanation of the physiological routine. We follow a method similar to the behavioral routine where we convert each Linguistic Medoid Prototype to natural language, and the summaries of all the LMPs together form the natural language summary of the physiological pattern. We use the LMPs shown in Table 4.6 to illustrate our natural language generation process.

In order to describe the LMPs in natural language, we select the LPS with highest truth-value for each summarizer in a given LMP, as shown in Table 4.7. We then use templates to convert the LMPs into natural language. For both the LMPs, the LPS with quantifiers *almost none*, *almost all*, and *almost none* have the highest truth value for summarizers *low*, *normal*, and *high*, respectively. Therefore, we describe the baseline pulse rate pattern of

Table 4.7: LMPs of Table 4.6 with one LPS per summarizer (LPS with highest truth-value for each summarizer)

	Quantifier	Summarizer	TV
LMP # 1	almost none	low	0.6
	almost all	normal	0.6
	almost none	high	1
LMP # 2	almost none	low	0.8
	almost all	normal	0.8
	almost none	high	1

Figure 4.48 with the natural language expression: *Almost all instances had normal pulse rate.*

4.3.1. Changes in Pulse Rate Routine

We compare the LMPs obtained during the month of January to each day in the three-month period using our FIS based dissimilarity algorithm. We follow Equation 4.5 to compute the dissimilarity between the LMPs and the LPS sets representing each day's pulse rate data. There, we compare the LPS representation of each day's pulse rate to both the LMPs and select the smaller value as the distance between the day and the LMPs. Figure 4.53 shows the distance value between the LMPs and each day in the three-month period of Figure 4.48. We observe that the method is able to detect a sharp drop in the pulse rate measurements around the end of February (increase in dissimilarity). The dissimilarity increases gradually until a few days before the CHF event and then rises sharply immediately prior to the event. The dissimilarity then drops back to a lower value;

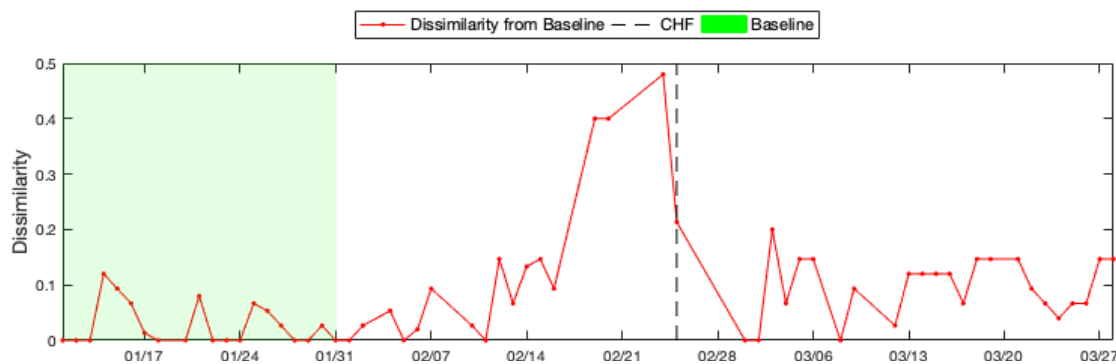


Figure 4.53: Dissimilarity between the baseline and the pulse rate data over the entire 3 month period

however, we see fluctuations, suggesting that the resident’s pulse rate does not return to their normal pattern after suffering from CHF.

We also use our method to produce natural language summary of the change in pulse rate pattern just before the CHF diagnosis. The following summary was produced on the day before the CHF diagnosis, February 24: Compared to the baseline, on February 24, *there was a significant increase in the number of low pulse rate measurements.*

4.3.2. Retrospective Case Studies

4.3.2.1. Case I

In Section 4.2.6.2 we analyzed the behavioral routine pattern of a 97 year old female resident for a six-month period starting from July through December 2019. Here, we use our physiological routine modelling approach to study the respiration rate measurements pattern for this resident during the same time period, as shown in Figure 4.54. Recall that this resident suffered a fall on August 12 and reported severe pain in October. Earlier, we

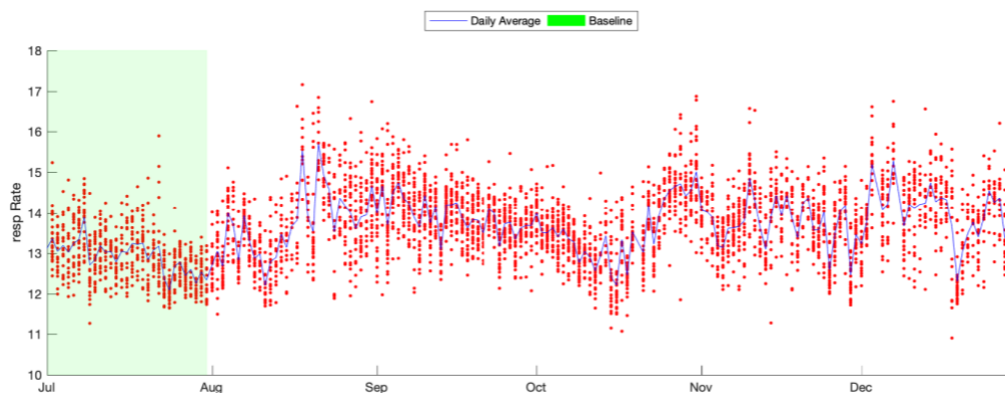


Figure 4.54: Daily Respiration Rate measurements of a resident for a period of six months (Case I)

saw that our method detected a change in the behavioral routine of the resident around the two health events. Similarly, we see a change in the respiration rate measurements of the resident around these events. We see a sharp increase in the respiration rate after the fall on August 12. The respiration rate gradually drops until mid-October, after which it again increases.

Physiological Routine Modelling

We follow the procedure described in the previous section to model the respiration rate measurements pattern. We summarize the daily respiration measurements by LPSs of the form: Q of the instances has P respiration rate, where Q are the quantifiers: *almost none*, *few*, *some*, *many*, and *almost all*, and P are the summarizers, *low*, *normal*, and *high*, defined previously in Figure 4.50.

Similar to the behavioral routine for this resident, we model the respiration pattern of the resident by the measurements taken over the baseline month of July. Figure 4.55 shows the

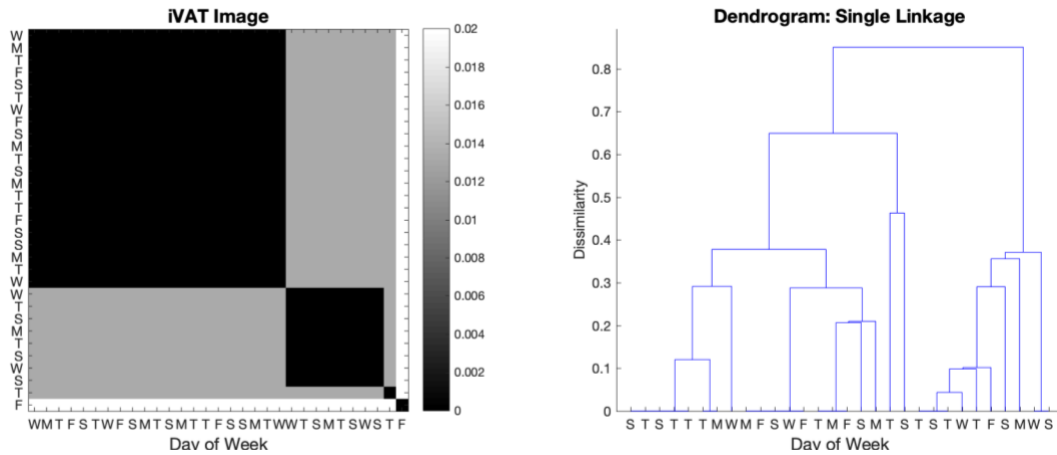


Figure 4.55: iVAT image and dendrogram plot to visualize clusters in the baseline period (Case II)

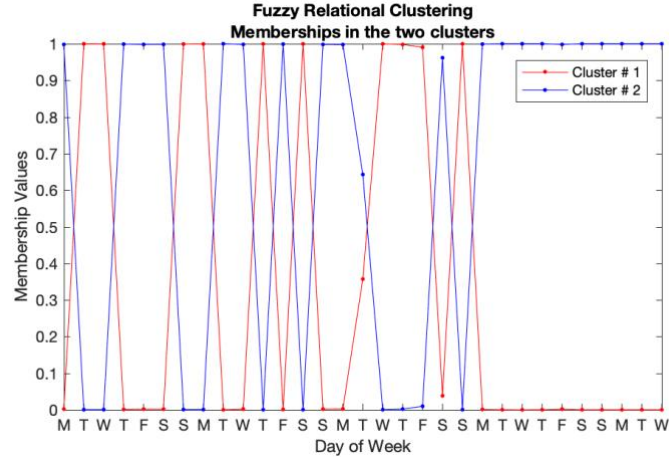


Figure 4.56: Memberships of each day in the baseline period into the two clusters

iVAT image and dendrogram plot obtained after converting measurements over each day during the baseline period to LPSs and then computing pairwise distance between them. The iVAT image shows 4 clusters with two large clusters. Based on this, we cluster the respiration rate measurements over the baseline period into two clusters using the NERF c-means. Figure 4.56 shows the membership of each day during the baseline into the two clusters. We use the membership values to assign each data point into one of the two clusters, which results in one cluster of size 10 & the other of size 21. We then use the medoids of the two clusters as their prototypes. This results in two typical respiration rate patterns for this resident, for which the natural language summaries are shown in Figure 4.57. The summaries suggest that for some days, the resident had almost all of the respiration rate measurements as normal, while on some days only many of the measurements were in the normal range.

Pattern 1:

Almost all instances had normal respiration rate

Pattern 2:

Many instances had normal respiration rate

Figure 4.57: Natural language description of the cluster prototypes

Deviation in Physiological Routine

Here we compute changes in the respiration rate pattern of the resident through the six-month time period. Figure 4.58 shows the dissimilarity. We see that the dissimilarity during the baseline days is close to zero for almost all the days, while it increases around the health events. Specifically, the respiration rate of the resident changed around the time when she fell, and it remains high until mid-September when it went back to the usual pattern as suggested by the almost zero dissimilarity from mid-September to mid-October. The dissimilarity again increased after the second health event, and remained changed until the

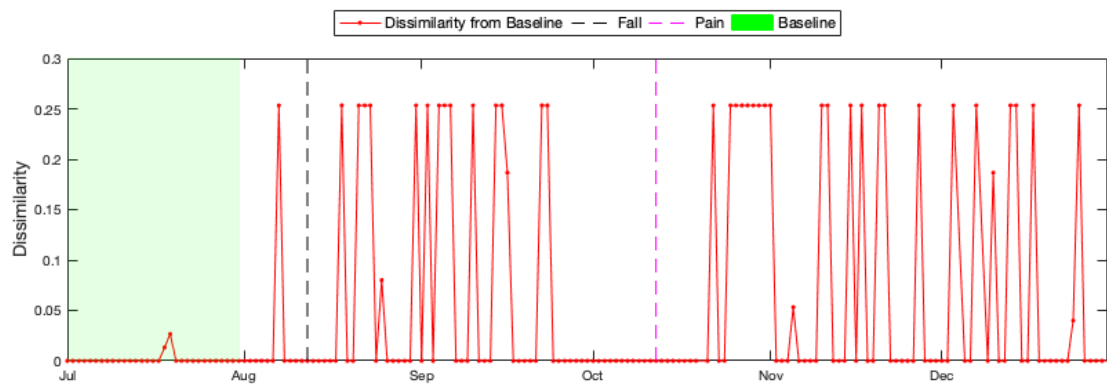


Figure 4.58: Dissimilarity between the baseline and the respiration rate data over the entire 6 month period

end of December. Next, we generate the linguistic explanation of the change just before the resident had the fall: *There was a decrease in the number of low respiration rate measurements*. We see that the method reports fewer low pulse rate measurements than usual, which is also observed in Figure 4.54, where more number of high respiration rate measurements were recorded before the resident had a fall.

4.3.2.2. Case II

In this case we use our method to analyze pulse rate data of a 32-year-old male resident of Columbia, MO. The pulse rate measurements were obtained from his Apple Watch. Throughout the time period, the subject wore the device only during the daytime before bedtime. The subject's health data was exported to a JSON file using the Apple Health App and the resting pulse rate measurements were extracted from this file. Note that in this study we focus only on the resting pulse rate measurements and do not include the readings that were taken when the subject was active or performing workouts. Figure 4.59 shows 9 months of resting heart rate measurements starting from January 2019 to September 2019. We observe that broadly, the daily pulse rate measurements are stable and vary from 60 BPM to 120 BPM, with a slight increase during the months of May, June, and August. The person travelled to Ohio for a wedding in the end of May, while a friend was visiting him in the end of July. The person was visiting family in India from July 23 to August 23.

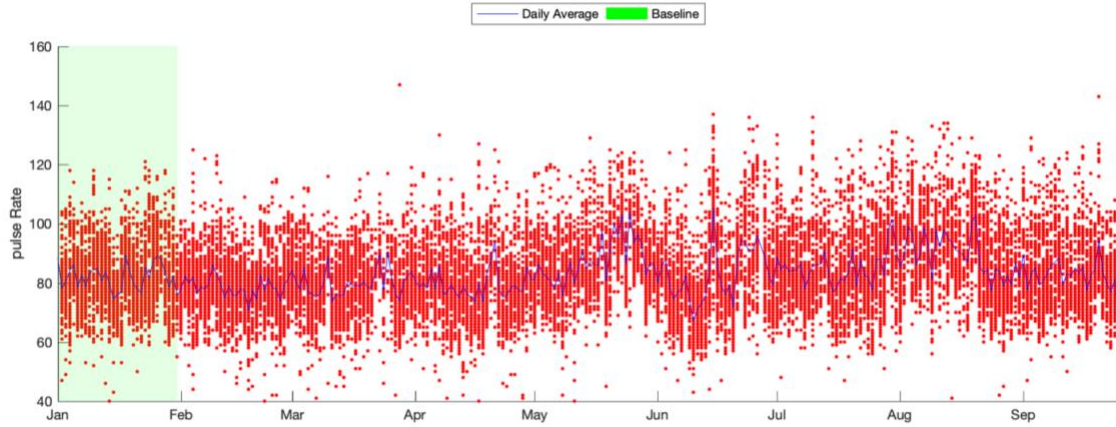


Figure 4.59: 9 months of resting pulse rate data of a subject obtained from their Apple Watch (Case II)

Pulse Rate Routine

Similar to the above case studies, we define a baseline time period and model the usual pulse rate pattern of this person using the data obtained in that period. We then compare the usual pulse rate pattern with data over the entire time period. In this case, we use the data obtained over the month of January as the baseline.

Figure 4.61 shows the iVAT image and dendrogram plot to examine any clusters in data. The plots were obtained after summarizing and computing distance between each day's pulse rate measurements using LPS of the form: Q pulse rate measurement were P .

The iVAT image suggests three big and two relatively small clusters. Based on this, we cluster the data into three clusters using NERF c-means. Figure 4.60 shows the memberships of the baseline data points into the three clusters. We use the membership

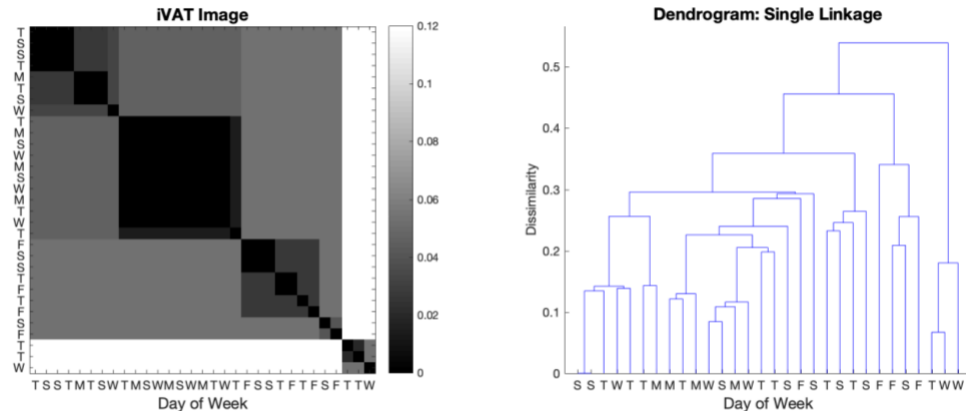


Figure 4.61: iVAT image and dendrogram plot for the baseline period (Case II)

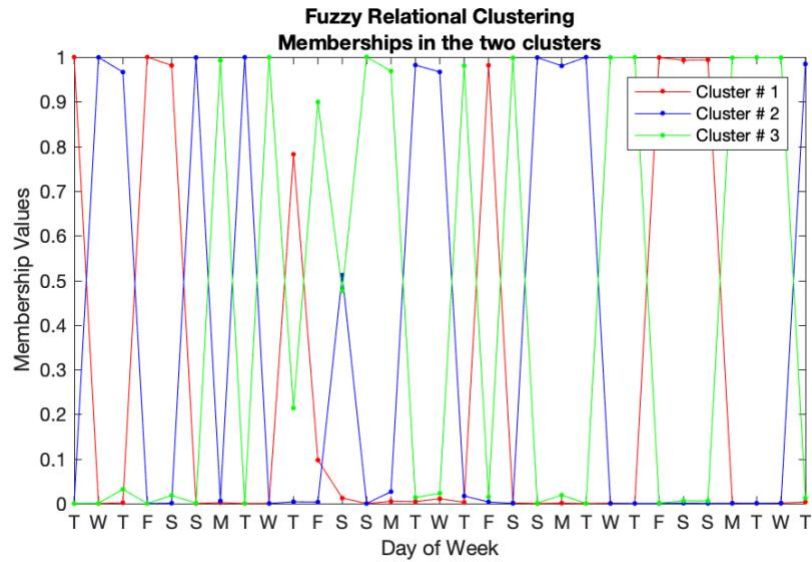


Figure 4.60: Memberships of each day in the baseline period into the three clusters

values to assign each data point into one of the clusters, and use their medoids as the cluster prototypes. The natural language summaries of the obtained prototypes is shown in Figure 4.62.

We see that the person has three typical daily pulse rate patterns and they differ mostly in the number of daily high pulse rate measurements. On some days there are almost no high pulse rate measurements, while on some days there are a few measurements that are high and the rest are in normal range. There are some measurements in the *high* and some in the *normal* range on the rest of the days. Typically, the subject does not have any significant number of low pulse rate measurements.

Pattern 1:

Some instances had normal pulse rate and some instances had high pulse rate

Pattern 2:

Almost all instances had normal pulse rate

Pattern 3:

Many instances had normal pulse rate and few instances had high pulse rate

Figure 4.62: Natural language description of the typical pulse rate patterns during the baseline period (Case II)

Deviation in Pulse Rate Routine

Here, we compare the entire 9-month time period with the baseline pulse rate pattern in Figure 4.63. We see that our method produces low dissimilarity when the subject is in his regular routine, while a high dissimilarity is always correlated with some unusual event in his day to day life, for example, travelling. A high dissimilarity is obtained when the subject was travelling to Minnesota and Rocheport on 25 March and 23 June, respectively and when he participated in a Table Tennis tournament. An even higher dissimilarity value was obtained on many of the days during the time when the subject was taking trips to Ohio, New Orleans, and India. In Table 4.8, we present natural language summaries of change in their pulse rate data on a few of the high dissimilarity days from Figure 4.63. We see that for March 25, April 21, and June 23 there is a small increase in the number of high pulse

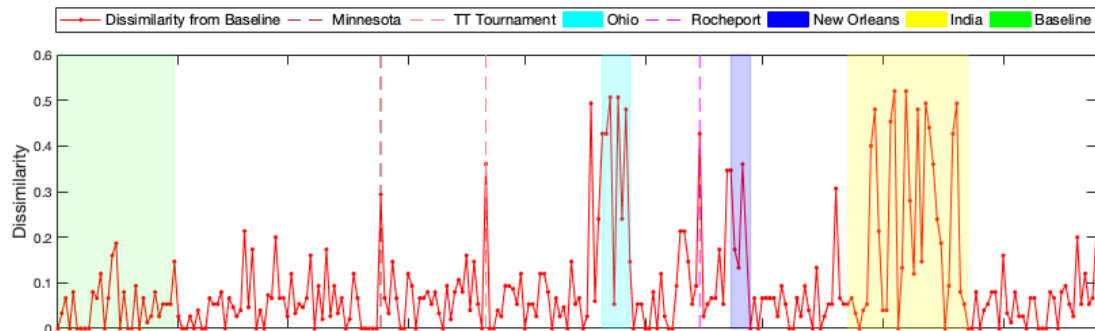


Figure 4.63: Dissimilarity between the baseline and the pulse rate data over the entire 9 month period (Case II)

rate measurements, while on May 21, June 15, and August 19, there is a significant increase in the number of high pulse rate measurements.

Table 4.8: Natural language description of change on days with high dissimilarity from baseline pattern (Case II)

Date	Natural Language Description of Change
March 25, April 21, June 23	<i>There is an increase in the number of high pulse rate measurements.</i>
May 21, June 15, August 19	<i>There is a significant increase in the number of high pulse rate measurements.</i>

4.4. Conclusions

We presented our explainable routine modelling and change detection approach to recognize patterns in both behavioral and physiological data obtained from sensor-equipped homes of the elderly. We made use of Linguistic Protoform Summaries to summarize daily activities and used our method to compute dissimilarity between LPS sets to model and detect changes in the regular patterns. The use of LPSs allowed us to express the results produced by our method in natural language. After rigorously testing our method with the help of synthetic data, we presented multiple case studies to validate our method on real data obtained from homes of the elderly. We showed that our technique is able to detect changes in the behavioral and physiological activity patterns that are correlated with changes in the health of the elderly using Electronic Health Records. Lastly, we produced natural language description of both their regular activity pattern and deviations in those patterns.

5. Linguistic Summarization of In-Home Sensor Data

5.1. Overview

In the previous chapter, we introduced a method to use in-home sensor data to monitor the general behavioral and physiological routines of the elderly. Although identifying changes in routine can provide important insights towards changes in the health of the elderly, the in-home sensor data is believed to contain much richer information that can be of assistance in monitoring the residents' wellbeing. In this chapter, we present a Natural Language Generation system that summarizes important health relevant information from the in-home sensor data generated in the apartments of elderly. This system is deployed as part a National Library of Medicine (NLM) funded project (Popescu, NIH-NLM #R01LM012221).

In Section 5.2, we provide an overview of the existing infrastructure that is already setup at the sites that are part of this project. Section 5.3 lists different components of the summarization process and describes each individual part in detail. In Section 5.4, we present the results of multiple surveys that we conducted to validate different components of our summarization pipeline. Next, in Section 5.5 we present results of our NLM project and discuss a few case studies to validate and illustrate the functioning of our linguistic summarization system. We conclude with a discussion on the utility and future prospects of this system.

5.2. Existing Infrastructure

We summarize the data generated from the Passive InfraRed (PIR) motion sensors, which are strategically placed to monitor activities of the residents inside the apartments; hydraulic bed sensors, which are used to measure the quality of sleep and other physiological parameters while the person is in bed; depth cameras, which are placed in the living rooms to measure gait parameters, such as walking speed, stride time and stride length. The raw data obtained from the sensors is processed in two stages in order to extract meaningful behavioral and clinical information. The first stage involves converting raw data to various useful measurements, which in this work we call *health concepts*. Health concepts are the general measurements that can be extracted from the raw sensor data. Table 5.1 presents the health concepts obtained from the sensors installed at various sites that are part of our study. Most of the sites are equipped with a bed sensor, 4 to 6 motion sensors and a depth camera sensor. The raw analog signal obtained from the bed sensor is fed to different algorithms to obtain measurements like restlessness, pulse rate, respiration rate and time spent in bed. While the other measurements are self-explanatory, restlessness quantifies the amount of motion a person makes while lying in bed. The motion sensors produce ‘hits’ (time stamped events) at different locations inside the apartment. These are

Table 5.1 : Health concepts extracted from the sensors

#	Health Concept	Obtained from	Description
1	Restlessness	Bed sensor	Movements made by person while on bed
2	Pulse Rate	Bed sensor	Measured while the person is on bed
3	Respiration	Bed sensor	Measured while the person is on bed
4	Time in Bed	Bed sensor	Times when the person is in bed
5	Inside apartment activity	Motion sensors	Monitors activity made by the person when inside apartment
6	Bathroom activity	Motion sensors	Monitors bathroom activity of the person
7	Walking speed	Depth sensor	The speed with which the person walks inside the apartment
8	Stride Time	Depth sensor	Time between two consecutive footsteps
9	Stride Length	Depth sensor	Distance travelled in each walking step

aggregated time-wise to assess the overall activity of a person, while the motion sensors placed inside the bathroom are used to measure the time spent in the bathroom.

The health concepts are further processed to compute what we call *health features*. The health features make the health concepts more understandable and each health concept can result in multiple health features. For example, the health concept *restlessness* can be further processed to obtain *restlessness* during the nighttime or average *restlessness* per hour and so on. Table 5.2 lists the health features used. We aggregate the *restlessness*, *time*

Table 5.2 : Health features extracted from the health concepts

#	Health feature name	Obtained from health concept	Description	Unit
1	Total Daily/Nightly Restlessness	Restlessness	Total restlessness during the night-time and day-time	minutes
2	Average Daily/Nightly Pulse rate	Pulse Rate	Average pulse rate during the night-time and day-time	beats/min
3	Average Daily/Nightly Respiration rate	Respiration Rate	Average respiration rate during the night-time and day-time	breaths/min
4	Total Daily/Nightly Bathroom Motion	Bathroom Activity	Total time spent in the bathroom during night-time and day-time	minutes
5	Total Daily/Nightly Apartment Motion	Inside Apartment Activity	Total time spent inside the apartment during night-time and day-time	minutes
6	Total Daily/Nightly Time in Bed	Time in bed	Total time spent in bed during night-time and day-time	minutes
7	Average daily walking speed	Walking speed	Average walking speed over a 24 hr period	cm/sec
8	Average daily stride time	Stride time	Average stride time over a 24 hr period	sec
9	Average daily stride length	Stride length	Average stride length over a 24 hr period	cm
10	Pulse rate details	Pulse rate	Average pulse rate over each 15 min slot when the person is in bed	beats/min
11	Respiration rate details	Respiration rate	Average respiration rate over each 15 min slot when the person is in bed	breaths/min

in bed, inside apartment activity and bathroom activity to daily and nightly measurements, while *walking speed, stride time and stride length* are aggregated over 24-hour periods. The *restlessness, amount of motion inside the apartment and bathroom motion* are obtained as an overall measurement over a given course of time, while the physiological parameters like *pulse rate, respiration rate, walking speed, stride time and stride length* are measured as an average over a duration. *Pulse rate and respiration rate* measurements over each 15-

minute period, while the person is in bed, are used to focus deeper on the variation of these health concepts.

In the linguistic summarization framework, the health features are monitored daily to look for unusual patterns in the data by an alert mechanism described in (Rantz, et al. 2013). An alert is sent to the clinical personnel if an abnormality is detected, which, in turn, initiates our summarization system. Currently, the alerts are uni-modal in nature (each parameter has its own alert), which may cause multiple alerts per resident, per day. The clinicians are notified about the alerts via email. In the standard system, the email lists the number of alerts that have been generated for a resident. A screenshot of an email is shown in Figure 5.1. Here, the email heading displays the id of the resident along with their apartment number. The heading is followed by the number of alerts that have been generated since the last time an alert was sent for this resident. The email also contains a link that takes the user to a visual interface containing the resident's data. As mentioned before, one of the aims of our NLM project is to describe the data related to this alert in natural language. To

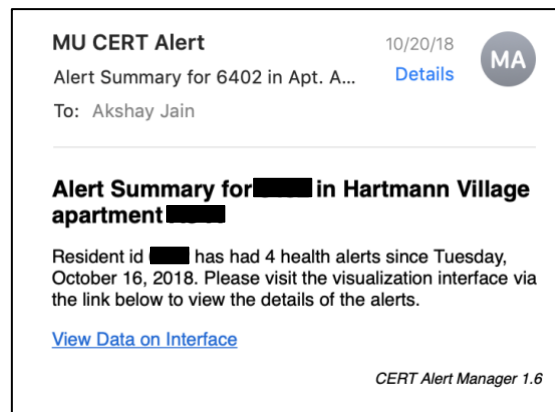


Figure 5.1: Screenshot of the standard alert email notification (non-linguistic)

linguistically describe the sensor data related to an alert, we start by summarizing the reason behind the alert and the sensor feature(s) that caused it.

We envision to employ the linguistic summaries presented in this chapter along with the routine modelling and change detection framework of Chapter 4. Specifically, we propose to use the technique to detect significant change in the routine as a trigger to produce linguistic explanations that have information about the routine as well as other important sensor parameters. In Figure 5.2, Figure 5.3, and Figure 5.4 we replicate the dissimilarity plots of Section 4.2.6 along with the alerts generated by the current alert generation system that was described above.

In Figure 5.2, we see multiple gait alerts (decrease in walking speed) around the time the person fell in August. We also observe multiple increase in restlessness alerts between the two health events. An increase in living room activity alert is observed right after the resident reported severe pain. We also see a living room (increase) and a restlessness (increase) alert about a month after the resident reported pain. For Case II in Figure 5.3, a

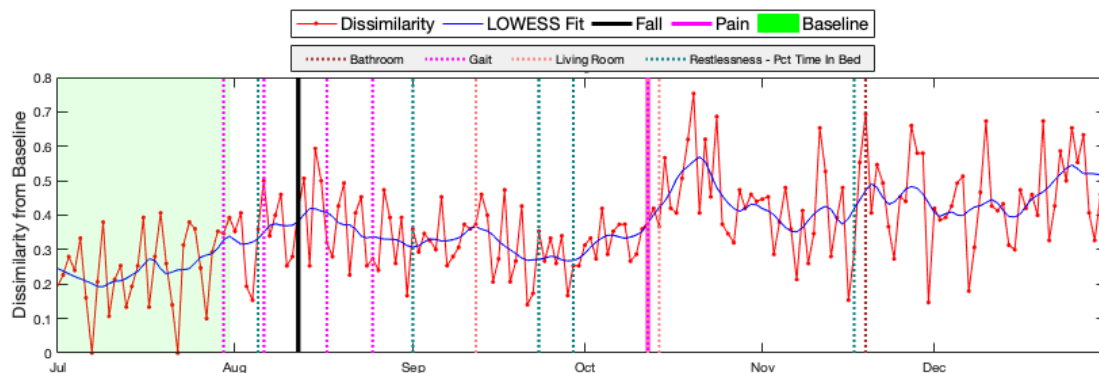


Figure 5.2: Dissimilarity plot showing deviation in behavioral routine (Case I, Section 4.2.6.2) along with alerts triggered by the current alert generation algorithm

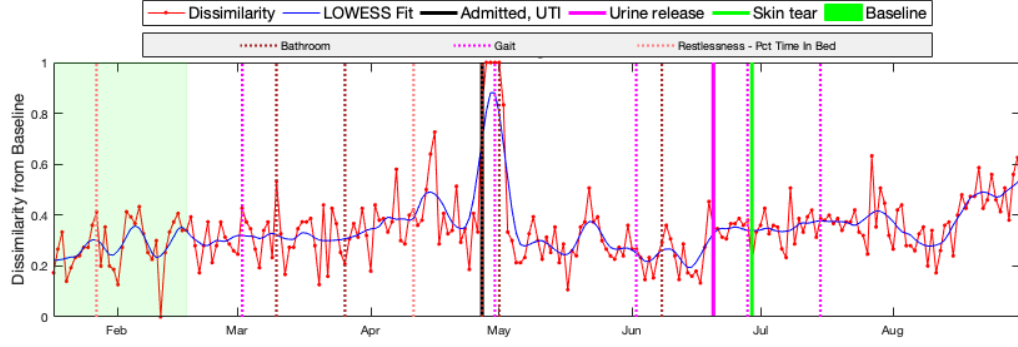


Figure 5.3: Dissimilarity plot showing deviation in behavioral routine (Case II, Section 4.2.6.3) along with alerts triggered by the current alert generation algorithm

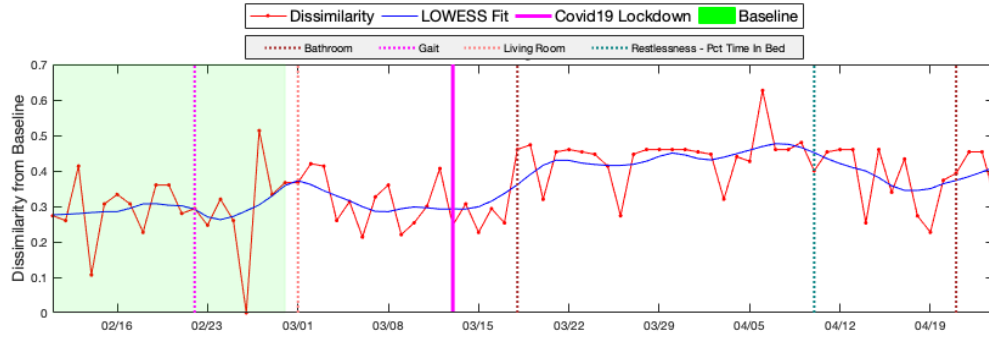


Figure 5.4: Dissimilarity plot showing deviation in behavioral routine (Case III, Section 4.2.6.4) along with alerts triggered by the current alert generation algorithm

gait (decrease in walking speed) and a bathroom alert (decrease) is triggered after the resident was admitted to hospital in April due to UTI. We also observe a gait alert around the time when the resident suffered from a skin tear. For the last case study shown in Figure 5.4, we see a bathroom alert (decrease) about a week after the COVID-19 lockdown. We also observe a restlessness (decrease) and a bathroom (decrease) alert after a month of the lockdown.

Since the routine modelling framework is still in the early stages of development, here we describe the proposed linguistic summarization system with the sensor alert framework

that was described above. In the following, we describe the mechanism used to linguistically summarize different types of alerts implemented in the online monitoring system.

5.2.1. Alerts Based on Significant Changes from Baseline

This type of alert compares the current values of the health features to a baseline, which is defined by the measurements recorded in the past two weeks. Note that, in a previous study, it was determined that a health event (e.g. hospitalization or fall) can be indicated by changes in health features two weeks prior to the event (Rantz, et al. 2012). An alert is generated if the current reading deviates significantly from the baseline. The significance of the deviation is quantified by computing the standard deviation of the current value from the baseline distribution. The reader is referred to (Rantz, et al. 2013) for a detailed description of the alert generation mechanism. We describe this alert linguistically by listing the health features that caused it. For example, if an alert is generated because of a significant increase in the living room motion during the night, then it is summarized as: *Yesterday, nighttime living room activity was significantly higher than usual.*

5.2.2. Alerts Based on Hard Limits

These types of alerts are generated if a health feature crosses a user defined absolute threshold. They are only applicable for physiological parameters since it is not possible to put a meaningful absolute threshold on the behavioral health features. To this end, the

health features that generate such alerts are pulse rate and respiration rate. An example of linguistic description of an alert of this kind is: *Yesterday, pulse rate exceeded the threshold of 120 bpm.*

5.2.3. Gait Alerts

There are two types of alerts implemented to monitor the gait of a resident. The first alert is based on the study conducted in (Phillips, et al. 2017), where the authors found that a cumulative decrease of more than 5.1 cm/sec in walking speed over a week suggests an increased risk of fall. The second type of gait alert is generated if average walking speed over a week deviates significantly from a pre-defined baseline (Stone, et al. 2014). We design the summary of these alerts to concisely explain the underlying mechanism that generated the alerts. The summaries of the gait alert of the first and second kind respectively are: *Over the last week, walking speed decreased more than the threshold of 5.1 cm/sec*, and, *the average walking speed over the last week was significantly lower than usual*. Summarization Process

5.3. Summarization Process

The alert generation algorithm is run for each resident, every morning. If there are one or more alerts generated for a resident, the sensor data leading to the current day is summarized. After describing the sensor parameter that caused the alert, we broaden our scope towards all the health features. The reasoning behind this is that if there is an unusual

change in any one of the sensor parameter, then a clinician might be interested in looking at other sensor readings before making any judgements. Hence, irrespective of the sensor feature that generated the alert, we look for and report important patterns in all the sensor streams. We call these patterns data features. All the significant data features along with the alert trigger form the linguistic alert summary. Figure 5.5 an overview of our alert summarization system and the data features we look for. Next, we describe the rationale behind each of the data features and their generation mechanism in detail.

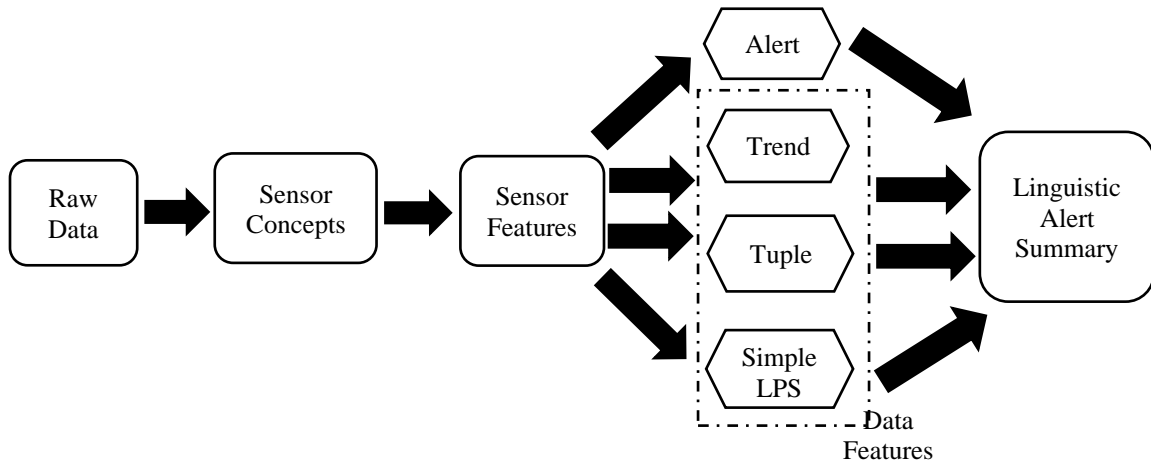


Figure 5.5: Overview of the linguistic alert summarization system

5.3.1. Tuple Summary

In a survey (Popescu, et al. 2017), we asked several clinicians in our team to list health features which might correspond to specific health conditions in the elderly. The findings suggested that changes in health conditions are often reflected in more than one parameter. For example, an onset of a Urinary Tract Infection (UTI) may be reflected in bathroom activity as well as bed restlessness. Inspired from these results, we designed a data feature

to detect and summarize concurrent abnormal variations in pairs of health features. We call this data feature *Tuple Summary*. For a given set of health feature measurements over a two-week period, we find the pairs for which the following statement is applicable: *There were many days with high/low health feature A and high/low health feature B*. That is, we investigate the data to find pairs of features that had significant number of instances when they were high together, or low together, or when one was low and the other was high. The applicability of this statement is determined according to the algorithm described in the following.

For two features A and B, the input to the algorithm are two time series of the same length, containing measurements of two health feature recorded daily for a period of two weeks. Also, for both the features, along with the time duration for which the tuple summary is to be calculated, measurements for a pre-defined length of baseline period also needs to be supplied.

Preprocessing: The tuple summary quantifies the number of occasions when two features are high or low simultaneously or when one is high and the other is low. In a strict sense, the term simultaneous would require the two health feature measurements to occur on the exact same date. However, a loose definition of the word can include events occurring in some user-defined neighborhood. For example, when considering a night-time health feature, it is not clear whether to associate the feature value to the date on which the night started (say, 10 PM), or the date on which it ended (for example 7 AM the next day). To this end, we pre-process the series of the two sensor features separately, using the following

computation. For each data series: $\tau = x_1, x_2, x_3, \dots, x_n$, where x_i is the sensor feature value at time t_i , we compute two new series by using the *max* and the *min* operator: $x_{max}'_i = \max_{j=i-w:i+w} x_j$ and $x_{min}'_i = \min_{j=i-w:i+w} x_j$, where, w is the window size and $\tau'_{max} = x_{max}'_1, \dots, x_{max}'_n$ and $\tau'_{min} = x_{min}'_1, \dots, x_{min}'_n$ are the two pre-processed series. For example, if $w = 1$, then this pre-processing step basically replaces each value in the time series by the highest or the lowest value in its neighborhood of size 1. To fix ideas, we illustrate this process with the help of a time series of length 7 in Table 5.3. We can see that each observation in the processed series is replaced by the highest / lowest value in its neighborhood in the original series. We use the series obtained by using the max operator for finding features that are *high*, while using the one obtained with min operator when finding *low* features. The next step is to compute the distance of the current

Table 5.3:Pre-processing step in Tuple Summaries

τ	284	426	352	333	328	525	273
τ'_{max}	426	426	426	352	525	525	525
τ'_{min}	284	284	333	328	328	273	273

measurements from the baseline. The baseline is comprised of data samples over some time duration prior to the current time-period being summarized. For each health feature series, we compute the Mahalanobis distance of each measurement from the baseline distribution. The computation is shown in Equation 5.1 & 5.2, where b is the set of baseline measurements.

$$\tau''_{max} \equiv x_{max_i}'' = \frac{x_{max_i}' - mean(b)}{std(b)} \quad (5.1)$$

$$\tau''_{min} \equiv x_{min_i}'' = \frac{x_{min_i}' - mean(b)}{std(b)} \quad (5.2)$$

Linguistic Protoform Summaries: The new time series, τ''_{max} and τ''_{min} are then used to compute the applicability of the summary: *There were many days with high/low health feature A and high/low health feature B*. The validity of this statement can be calculated using Linguistic Protoform Summaries (LPSs) methodology developed in Section 2.5. If we treat the summary *There were many days with **high/low** health feature A and **high/low** health feature B* as a type I LPS, then the terms *many* and *high/low* are called the quantifier and the summarizer, respectively, and are defined by fuzzy sets. Figure 5.6 shows the fuzzy sets representing the terms *high/low* and *many*, which are used to compute the tuple summaries. In the fuzzy set for the summarizer *high and low*, the x-axis is the Mahalanobis distance between baseline and the current point, while the y-axis is the membership for each distance value. The quantifier *many*, is defined as a standard relative quantifier used in Linguistic Protoform Summaries.

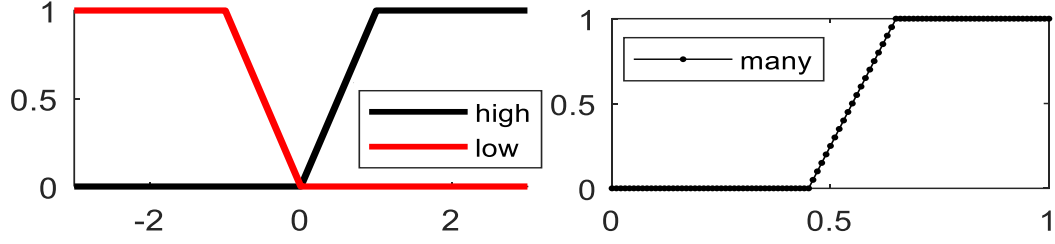


Figure 5.6 Fuzzy sets representing the Summarizer (left) & Quantifier (right)

To compute the truth-value of the tuple summaries, we use the semantic truth-value computation method for type I summaries, which was described in Section 2.5.1. Note that the LPS defined above is a slightly modified form of type I LPS mentioned earlier. The exact procedure to compute their truth-values was described in (Jain and Keller 2015).

The statement: *There were **many** days with **high/low** feature A and **high/low** feature B*, is deemed worthy of being reported if it meets a user defined truth-value threshold. A threshold of 0.7 is empirically selected for this study. In case there are more than one pairs of sensor features that meet this criterion, we include all of them in the same summary sentence, separated by conjunction operators. For example, if there are 3 pairs, A & B, A & X, and X & Y that meet the truth-value threshold, they are reported as: *There were many days with high A & high B, high A & low X, low X & low Y*.

5.3.2. Trend Summary

Looking for increasing or decreasing trends in time stamped data is, arguably, the most common technique to extract information from time series. While the tuple summaries look for sporadic changes in the health feature time series, we use the trends to find gradual

changes. In our long collaboration with nurses monitoring the sensor information, we have found that they often look for trends in the data. As we demonstrate below, detecting/interpreting trends in data streams is not a trivial task, for humans as well as machines. The first step in expressing trends in linguistic summaries is to detect trends in the data with good accuracy. Below we describe the trend detection algorithm used in this work.

Normalization: The most widely used techniques to detect increasing/decreasing trends is to fit a line to the data and use the slope of this line to determine the type of trend. The sign of the slope of the best fit line does not depend on the scale of x and y axis values, however, its magnitude does depend on this scale. Therefore, in order to avoid the dependence of the magnitude of slope on the range of the points on which we fit the line, we first normalize the data. The normalization is done so that the resultant data points have zero mean and a standard deviation of one. Note that we do not change the x-axis values since they are of the same scale for all sensor features.

Best fit: After normalization, we use a least squared error methodology (implemented as Matlab's *fitlm*) to find the slope. Suppose the given data points are modelled by Equation 5.3 where β and α are the slope and the intercept of the line, respectively, and μ_i is the random noise. Then, the slope of the least square fit line is given by Equation 5.4, where, x_i and y_i are the coordinates of i^{th} point, and \bar{x} and \bar{y} is the mean value of the two coordinates.

$$y_i = \beta x_i + \alpha + u_i \quad (5.3)$$

$$\hat{\beta} = \frac{\sum (x_i - \bar{x})(y_i - \bar{y})}{\sum (x_i - \bar{x})^2} \quad (5.4)$$

The *fitlm* function also returns a p value as an indicator of whether the fit is good or not. A small p value suggests that there is a small probability that the slope of the least square fit line is 0, that is, a higher chance of the slope being either positive or negative.

Fuzzy Inference System (FIS) to refine trend reporting: The p value obtained by the above method is a good indicator of trend in the underlying data. However, in our experiments, we found it to be very sensitive to small changes in the data. To alleviate this problem, we design a Mamdani fuzzy rule-based system (Keller, et al. 2016) with the p value forming one of the inputs. The other input to the system is the slope of the fitted line. Table 5.4 shows the rules used in the FIS. Here, the membership functions used to represent the antecedents (p value and gradient) and consequents (report trend) are shown in Figure 5.7. We call the final defuzzified value of the FIS the trend score and use it as the final indicator of goodness of fit.

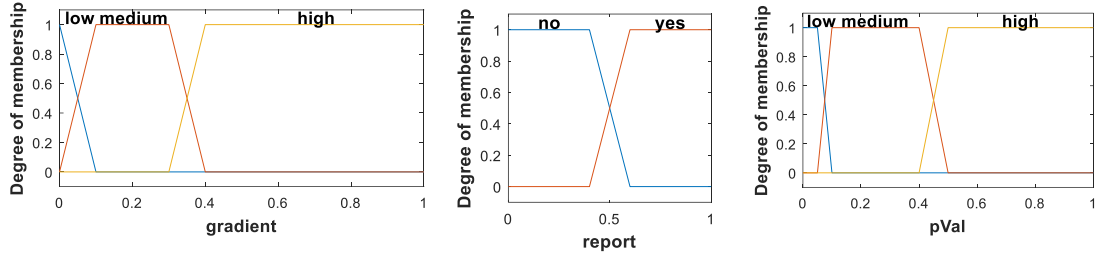


Figure 5.7: Fuzzy sets representing the antecedents (gradient-left and p value -middle) and the consequent (report -right)

Table 5.4: Fuzzy Rules used to refine trend detection

Rule #	Fuzzy Rules
1	If the p value is low then report trend is yes
2	If the p value is high then report trend is no
3	If the p value is medium and gradient is low then report trend is no
4	If the p value is medium and gradient is medium then report trend is yes
5	If the p value is medium and gradient is high then report trend is yes

We use the process mentioned above to determine whether there is a trend in a given time series. However, trend in data leading to a certain day can be reported in several ways. For example, we can use a trend-based time series segmentation algorithm to break the series into different parts, and then report the trends in all the segments. Applying this process to multiple health features might result in too many reportable trends, which would be too much detail for a summarization system. An alternative approach is to find the overall trend in the two weeks of data. This would work for the cases where the change in the health feature is gradual over the two weeks period, but we won't be able to find the trend reliably if the change is only over the last few days. To avoid this problem, we made the trend

detection algorithm to be dynamic in terms of duration of the trend. To this end, for each health feature we start with 5 days leading to the alert day, and then go back up to two weeks, one day at a time, computing trend scores at each step. Note that, in a previous study, it was determined that a health event (e.g. hospitalization or fall) can be indicated by changes in health features two weeks prior to the event (Rantz, et al. 2012). The trend with the highest score is selected as the best trend. If this score of the best trend is greater than a threshold of 0.5, then the trend is considered worthy of being included in the alert summary. Note that we fix the minimum length to 5, because we need at least a few points to find a reliable trend.

After making a decision about the existence of a trend in a given health feature time series, we require it to fulfil one more criterion before it can be reported. Sometimes, we observe that even though there is a trend in a sensor feature, the change it causes is very small with respect to changes in health. For example, if there is an increasing trend in the average pulse rate over 5 days but increase is just 3 beats per minute, this might not be worth reporting. In such cases, when the change is below a given threshold, we do not report the trend. To obtain these thresholds, we conducted a focus group session among clinicians in our group. For each health feature, we presented them with questions like the one shown in Figure 5.8. The focus group setting allowed the experts to have a discussion before reaching a consensus about the thresholds. Even though the experts are quite familiar with the data, they were not very confident about the definite thresholds. At the end of the focus group, they ended up assigning a minimum 20% change required to most of the sensor

What is the smallest absolute and/or relative change in **average pulse rate (beats/minute) over 24 hours** you would determine necessary to report during a period of:

Time period	Absolute	Relative
One week		
Two weeks		
One month		

Figure 5.8: Sample question used in focus group survey with clinicians

features for all the time periods, except the gait parameters, *time in bed* and *time spent in the bathroom* during night-time. For the gait parameters, they believed that a change of 2% is enough for it to be reportable, while for *time in bed* they required a change of at least 50%. They suggested that the time spent in the bathroom during the daytime does not carry significance and hence need not be reported, while for the nighttime, a change of 30 minutes or 10% would be needed for it to be of any significance. At the end, they also suggested that these thresholds are not definite, and we should monitor the sensor data and the alerts and adjust them as we gain more experience with the system.

In cases where our algorithm detects reportable trends in more than one health feature, we can either list each trend separately or combine the trends using some heuristic. In the former case, each detected trend will have its own sentence with the health feature name and the length of the trend as part of the summary. For example, consider two features X and Y in which an increasing trend has been detected, with the length of trend equal to 7 for Feature X and 14 for Feature Y. When listing each trend separately, the summary would

be something like: *Feature X has been increasing for 7 days and Feature Y has been increasing for 14 days*. This summary would get wordier very quickly with the increase in the number of trends being reported. In our experience in showing these summaries to clinicians, we found that they do not like such verbose statements. Also, the exact length of the trend is less important to them than just the fact that there is an increase or decrease in the sensor feature measurements. To this end, we combine the trends with the same direction with the following heuristic: In case of more than one health features having trend in the same direction, report the length of the shortest trend along with the names of all health features. With this heuristic, the summary of the above example would be: *Feature X and Feature Y have been increasing for 7 days*. We apply this heuristic separately to both increasing and decreasing trends, that is, we do not combine trends of different direction. There is some loss of information in this type of summary but having a short and concise summary is more desirable as we do not want to overwhelm the clinician with a lot of not very useful information.

In our conversations with the clinicians on our team, they suggested that for the physiological sensor features, along with the trend, they would also like to know the absolute change the trend induced. Accordingly, we include the exact change in the measurements for pulse rate, respiration rate, walking speed, stride time and stride length. In the summary, the change is included in parenthesis besides the name of the health feature. For example, in the above example, assume that Feature X was pulse rate and Feature Y was time in bed, and the pulse rate at the start of the 7 day trend was 60 BPM

and at the end was 75 BPM. Then the summary becomes: *Pulse rate (60 to 75 BPM) and time in bed have been increasing for the past 7 days*. Note that, irrespective of the length of the trend reported in the summary, we include the change in the feature over the entire duration of trend.

5.3.3. Simple LPS Summary

Physiological parameters like pulse rate and respiration rate measure the health condition of the elderly in a more direct sense as compared to behavioral parameters like *time in bed* or amount of *activity inside the apartment*. An abnormal measurement in pulse rate or respiration rate might be of immediate concern to the clinicians and they would not want to wait until there is a trend in these parameters. To this end, we design this data feature, called simple LPS summary, to detect and report finer changes in the physiological parameters. These summaries are of the form: *There were **few** instances with **low** pulse rate*. The instances are the small time periods over which the measurements are being made. Their duration can be defined by the designer, for example, 15 minutes, 30 minutes etc. Basically, these summaries quantify the number of times the feature in focus had a certain property in some given interval of time. For example, the above summary quantifiers the number of times the pulse rate was low. The mechanism to produce these summaries is described below.

As the name suggests, these summaries are computed with the help of LPSs which was described previously in this section in relation to Tuple Summaries. The protoforms used

in this case are: *There were Q instances of P pulse rate*, and *There were Q instances of P respiration rate*, where Q and P are the quantifiers and summarizers, respectively, both defined by fuzzy sets over suitable domains. Figure 5.9 presents the definitions of the fuzzy sets representing the quantifiers. Figure 5.10a displays the definition of *low*, *normal* and *high* pulse rate, while Figure 5.10b presents that of *low*, *normal* and *high* respiration rate. Note that the pulse rate and respiration rate definitions were created with the help of clinicians in our team.

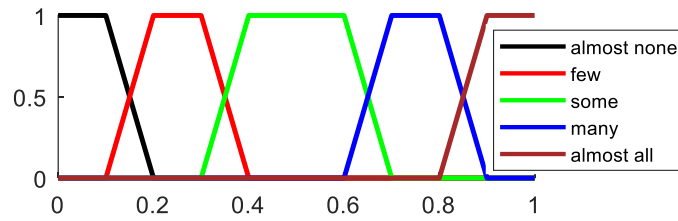


Figure 5.9: Fuzzy Sets representing quantifiers used in Simple LPS summaries

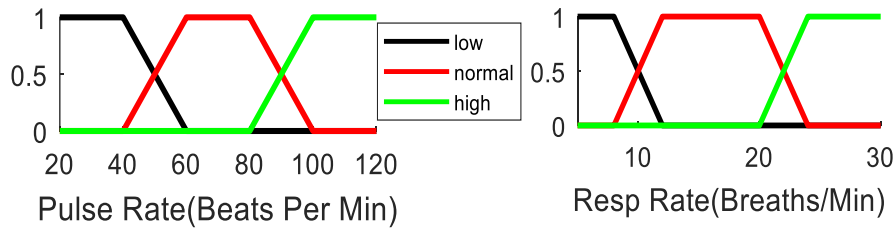


Figure 5.10a & 10b: Fuzzy sets representing pulse rate (left) and respiration rate (right)

Similar to the Tuple summaries, we use the truth-value computation method for type I LPS defined in Section 2.5. For a given pulse rate and respiration rate data, the truth-value is calculated using the fuzzy sets defined in Figure 5.9 and Figure 5.10. For both pulse rate and respiration rate, we compute the truth-value of all the LPSs formed by each

combination of the summarizer and the quantifier. Therefore, given 3 summarizers for both pulse rate and respiration rate (Figure 5.10), and 5 quantifiers (Figure 5.9), we compute the truth-value of $5 \times 3 = 15$ LPS to describe the pulse rate and 15 LPSs to describe the respiration rate measurements. Including all the 15 LPSs in the final alert summary would be unnecessary information. Hence, from these 15 LPSs for each of the health feature, we select the summary with the highest truth-value for each summarizer. This gives us a total of 3 LPSs, one per summarizer. Now from these 3 LPSs, we select the one which contains the most amount of information about the underlying data. By amount of information, we mean the quantifier which corresponds to most number of objects (the data points being summarized). For example, the quantifier *almost all* encompasses higher number of objects than the quantifier *few*. By this logic, the order of quantifiers with decreasing amount of information is: $\{almost\ all, many, some, few, almost\ none\}$. After this comparison, if the summarizer of the LPS to be reported in the summary is '*normal*', then it is considered not worthy of reporting since it is basically saying that the feature value is normal. However, if the summarizer is any other than '*normal*', then that LPS is included in the alert summary.

To fix ideas, we illustrate this process with the help of an example in Figure 5.11. Consider the pulse rate measurements over a night as shown in the plot on the top-left of Figure 5.11. For these sets of measurements, we compute the truth-value of all the 15 LPSs as shown in the table on the right. From these 15 LPSs, for each of the three summarizers: *low*, *normal* and *high*, we select the one with the highest truth-value. This results in three LPSs, as

shown in the Table on the left. The table suggests that there were *some* instances of *low* pulse rate, *some* instances of *normal* pulse rate and *almost no* instances of *high* pulse rate. Out of these three statements, we select the statement saying: There were *some* instances of *low* pulse rate, since this is the one with most useful information.

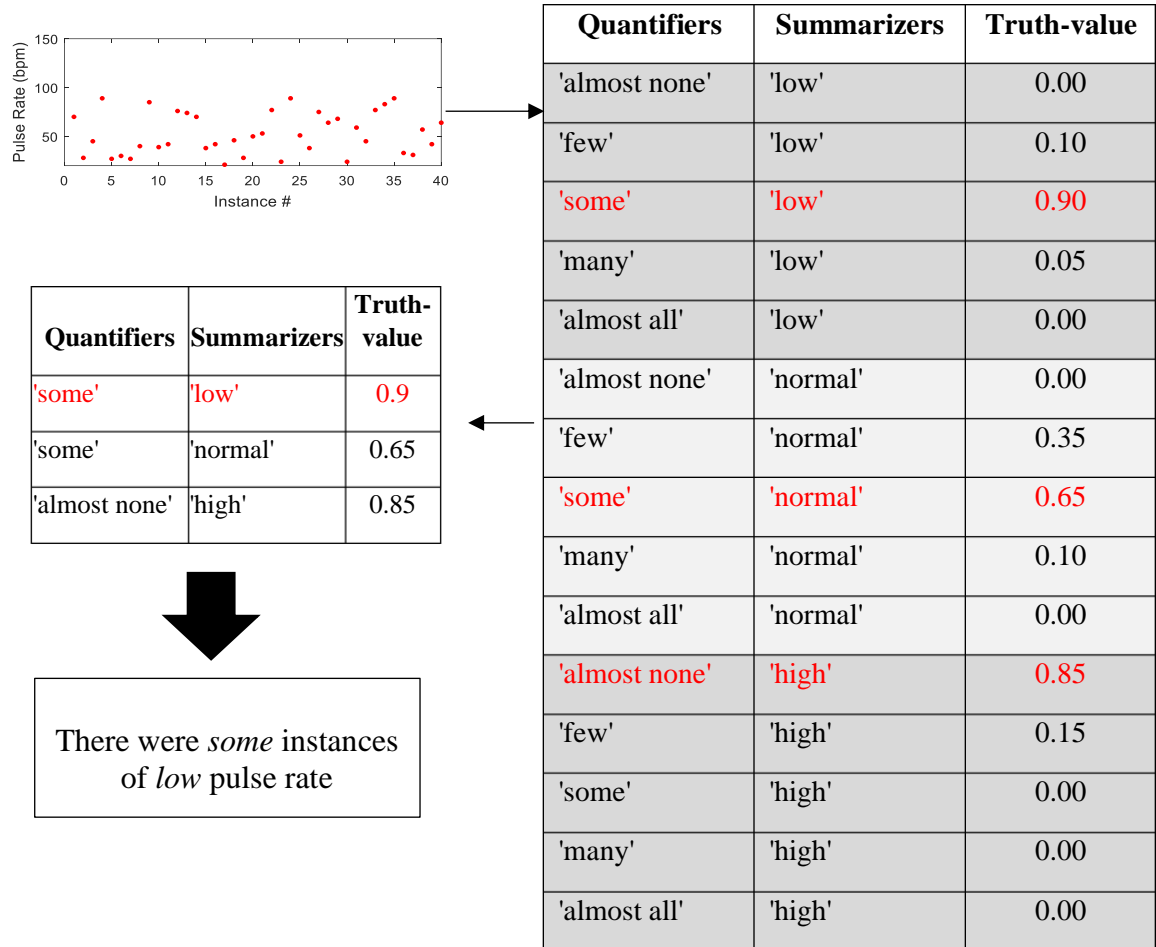


Figure 5.11: Illustration of the process to produce Simple LPS summary of data

5.4. Results

Here, we present the design and results of multiple surveys that we conducted to validate different components of our summarization pipeline. For each survey, we follow the guidelines of (Hripcsak and Wilcox 2002), where a minimum of six experts were recommended to effectively evaluate medical informatics systems.

5.4.1. Validation of the Tuple Summary

The goal of this experiment is to compare the tuple summaries produced by the LPS algorithm to the responses by humans when presented with the same data that produced the tuple summaries. To this end, we generated tuple summaries for five synthetic data examples. We then showed this data to six members of our engineering team who are familiar with the general nature of this research but do not have knowledge of the exact topic at hand. Figure 5.12 shows a sample question that was presented to the audience. The questionnaire asked the people to treat the two graphs as some imaginary features obtained from the apartments of the residents. For both the features, the first two weeks form the baseline (left of the green line in Figure 5.12) against which to compare the next two weeks (to the right of the green line). Based on this information, we asked them to assess the validity of the statement that quantifies the number of instances when both features A and B were high concurrently. Note that in order to not complicate the experiment, we only asked the audience to look for features that were high concurrently (and not when they were low, or one of them was low and other was high).

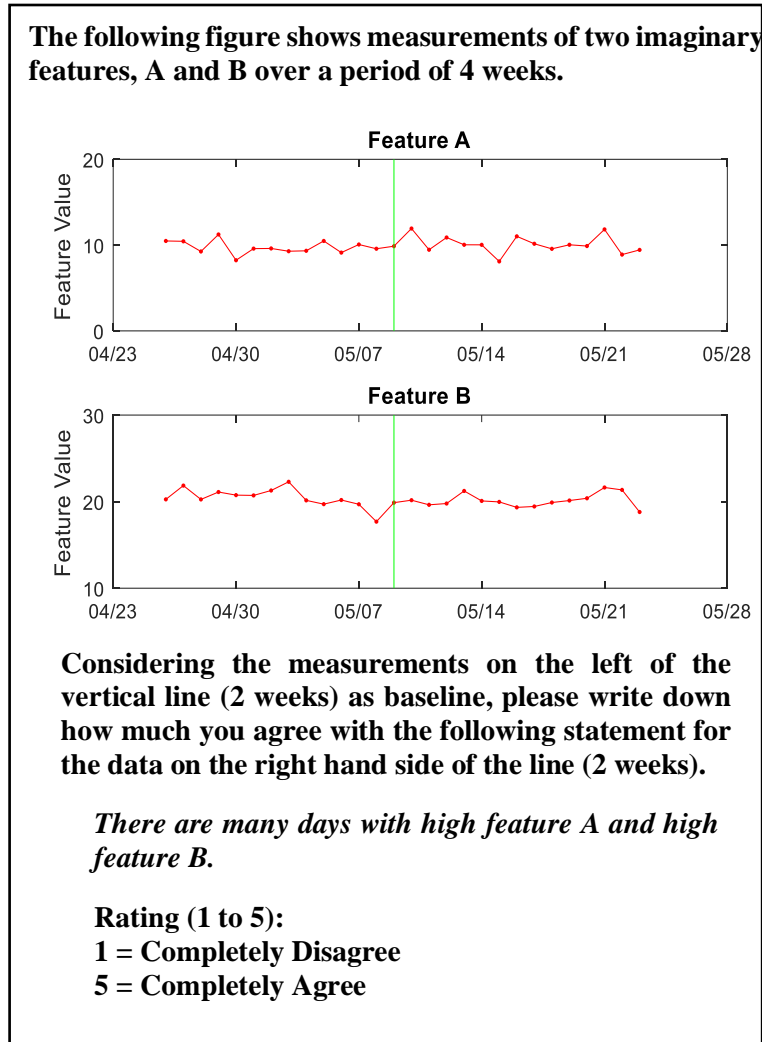


Figure 5.12: Sample question used in survey to validate tuple summaries

The truth-value of the LPS is used to compare the audience responses with that of the algorithm. The human responses range from 1 to 5, while the truth-value varies from [0,1]. To bring the two responses in the same scale, the truth-value range of [0, 1] is divided into five equal intervals, with truth-value between [0.0, 0.2) interpreted as confidence rating of 1 and [0.8, 1.0] as having confidence of degree 5. Table 5.5 shows the confidence ratings provided by all the five participants along with the confidence of the LPS algorithm. The

Table 5.5: Results comparing tuple summary algorithm with participant responses

Case #	LPS	P1	P2	P3	P4	P5	P6	Average (P1:P5)
1	1	1	2	1	3	1	1	1.5
2	2	1	2	3	4	2	1	2.2
3	4	3	3	2	5	4	1	3.0
4	1	3	1	2	5	4	1	2.6
5	5	4	2	5	5	5	4	4.2

first column shows the ratings produced by the tuple summary algorithm, while the cells in yellow in the subsequent columns contain participant ratings that were closer to the algorithm (within a neighborhood of 1). The last column shows the average of the ratings provided by the six participants.

We see that, overall, there are more cells in yellow than white, which implies that the rating of the algorithm and the participants matches more often than not. The mean participant ratings are also close to the algorithm ratings for all cases but Case 4. We observe that for this case, the ratings are a lot farther apart amongst the participants themselves. Participant 2 & 6 rated it as a 1 while Participant 4 rated it as a 5. This suggests that this is a peculiar case and the algorithm is no better or worse than the participants are. If we focus on individual participants, we see that Participant 4 is in most disagreement with the algorithm, with only two close matches. However, if we look closely, we observe that this participant always provides relatively higher ratings.

We note that we could have placed the two plots together on one graph to facilitate the human participants in viewing the two features together. The two features might have different scales on the y-axis, but this could be resolved by normalizing them between 0 and 1. The more important reason that we did not present the two features together is that in doing so, we did not want the participants to perform actual mathematics to come up with a response. This is because we intend to use these tuple summaries in the real care coordination environment, and the clinicians looking at these features would not have enough time and/or ability to look at and analyze each graph. In fact, this is the basic motivation behind producing these types of summaries in the first place.

We are aware of the fact that this is a small sample size, both in terms of the number of examples in the survey as well as the number of participants. Nevertheless, the limited results that we presented suggest that the algorithm produces meaningful results and gives us confidence to use it in a real-world environment.

5.4.2. Validation of the Trend Summary

To validate our trend detection algorithm, we created a survey where we presented a number of line-graphs to users and asked them to express their confidence about the presence of a trend in the graphs. We then compared the ratings obtained from the survey to the confidence produced by our algorithm. Before we discuss the results of the survey, we would like to mention a few factors concerning line-graphs that influence the way people perceive trends in them.

Figure Aspect Ratio: The aspect ratio of the figure containing the line-graph of a time series can have a significant impact on the trend perception. For example, Figure 5.13a and b show the same data presented in the form of a line-graph with two images having different aspect ratios. We see that in Figure 5.13a, a clear increasing trend is evident. However, this trend does not appear to be as evident and significant in Figure 5.13b where the aspect ratio of the figure is changed to have a larger width and smaller height. Mindful of this impact of the aspect ratio, we set it arbitrarily to one fixed value in all the graphs presented in the survey.

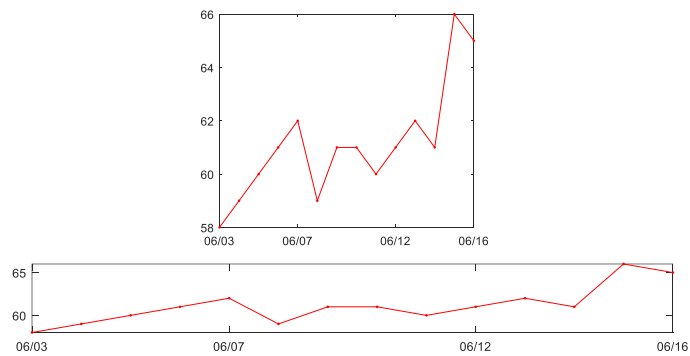


Figure 5.13a and b: Same data displayed in figures having different aspect ratios

Scale of axis: Another factor that might influence the way people perceive trends in line-graphs is the scale of the y axis. Similar to the aspect ratio, making the range of the y axis scale too large or too small can impact the perceived trend. For instance, Figure 5.14a and Figure 5.14b show the same data but one with a scale that just fits the data while the other with a much higher range. It is clear from the two figures that the trend is much more apparent in Figure 5.14a as compared to Figure 5.14b. Aware of this effect, we set the scale to fit the measurements (like in Figure 5.14a) in all the graphs presented in the survey.

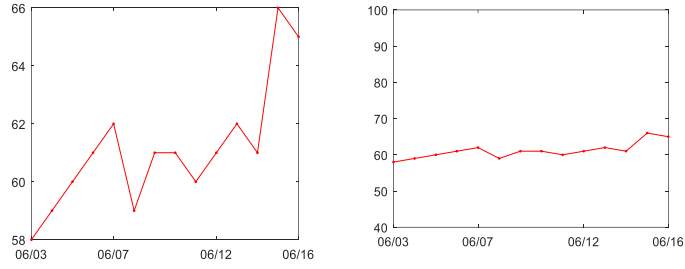


Figure 5.14a & b: Same data displayed in graphs with different scales

Survey description: To find candidate graphs for this survey, we selected 24 days distributed over 8 residents at TigerPlace, all of which were flagged to have abnormal activity by the alert generation algorithm. We then extracted two weeks of data leading to each of these 24 days. For each day, we extracted data for all the health features mentioned in Table 5.2 and ran our trend detection algorithm on the time series composed of these features. We selected 27 time series from these using stratified random sampling. Note that we did not perform fully random sampling, as we wanted to cover as much diversity as possible in the limited 27 examples. The examples were forced to contain at least one case for each of the health features. Also, the random process was forced to contain a few cases for which our algorithm detected an increasing trend, a few for which it detected a decreasing trend and a few cases where no trend was detected at all.

We asked six people in our Engineering team to provide a rating from 1 through 10 about the existence of a trend in each of the 27 cases, 1 being the lowest confidence that the line-graph contains a trend and 10 being the highest. Recall that our fuzzy rule-based trend detection algorithm produces confidence on the scale of 0 to 1. We linearly scaled its output

Table 5.6: Results comparing the two Fuzzy Rule Based trend detection systems

	ICC (p value & Gradient)	ICC (just p value)
P1-FIS	0.76	0.69
P2-FIS	0.85	0.66
P3-FIS	0.62	0.55
P4-FIS	0.87	0.70
P5-FIS	0.84	0.66
P6-FIS	0.78	0.61
Average ICC between Participants 1 to 6	0.74	
Average ICC between system and all participants	0.79	0.64

from 0 to 1 to 1 to 10 to match the scale of human responses. To quantify the agreements among the participants and the trend detection algorithm, we computed Inter Class Coefficient (ICC) (Hallgren 2012). An ICC of 1 denotes complete agreement between the participants while that of 0 suggests complete disagreement. Table 5.6 presents the results. P1 to P6 are the six participants, while FIS is the fuzzy inference system. Column 2 shows the results of the FIS described in Section 5.3.2, which was composed of both the p value and the gradient inputs. We also wanted to verify whether including the gradient in the FIS produced any positive impact in terms of detecting trends. Therefore, in Column 3, we show the results when the FIS was only composed of the p value. The fuzzy rules for the system with only the p value are: 1) If the p value is low, then report trend is yes, 2) if the p value is medium or high, then report trend is no. That is, this system is equivalent to the system with only a crisp p value threshold. We see that the FIS with both p value and

gradient inputs produces much higher agreement with the participants (0.79) than the FIS with only the p value (0.64). Also, the average agreement between the FIS and the participants (0.79) is higher than the average agreement between the participants themselves (0.74). This shows that our p value and gradient based FIS performs at par or even better than the participants in terms of detecting trends in line graphs.

5.4.3. Validation of the Simple LPS Summary

As described before, the simple LPS summaries are designed to look at the physiological parameters in more detail. To test the algorithm computing the simple LPS summaries, we created five synthetic data examples for summarizing pulse rate and five examples for respiration rate. Figure 5.15 shows a sample question where we show pulse rate measurements over a day, measured in 15-minute chunks. The x-axis of the plot lists the 15-minute chunk number whose pulse rate is shown on the y-axis. We presented this questionnaire to clinicians in our team, where they were asked to quantify, in linguistic terms, the number of instances of *low pulse rate*, *normal pulse rate* and *high pulse rate*. For each level of pulse rate, the participants were asked to choose one of the quantifiers out of $\{almost\ none, few, some, many\}$ and *almost all*. Note that unlike the tuple and trend summaries validation, the audience of this survey are clinicians who are adept at interpreting medical parameters like pulse rate and respiration rate.

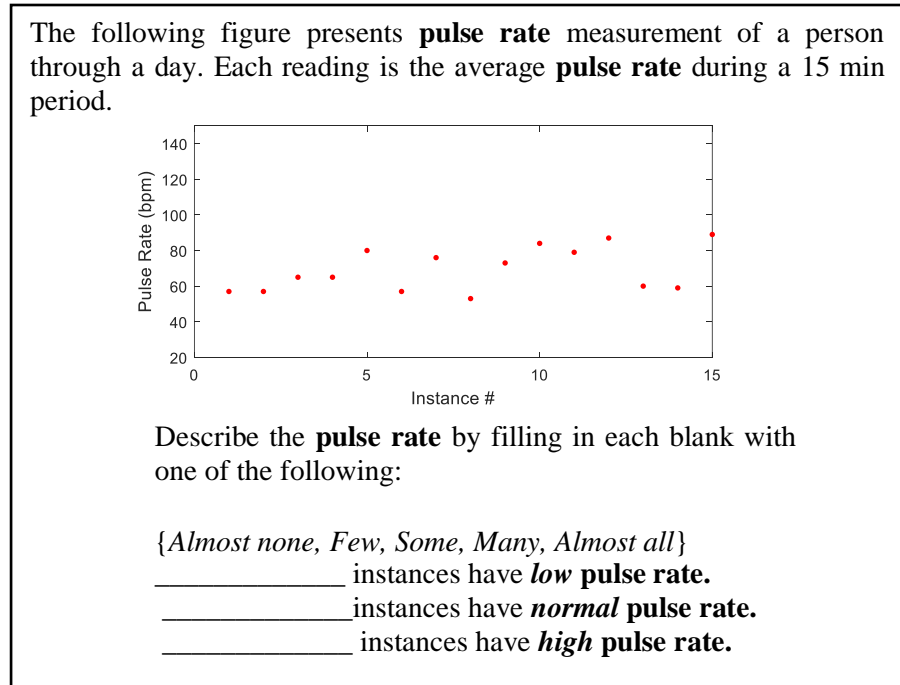


Figure 5.15: Sample question used in survey to validate LPS summaries

We presented the 10 examples (5 for pulse rate and 5 for respiration rate), like the one shown in Figure 5.15, to six clinicians. To facilitate the comparison of the responses, we converted the linguistic terms to numerical terms by representing *Almost none* as 1, *Few* as 2, *Some* as 3, *Many* as 4 and *Almost all* as 5. The linguistic quantifiers are ordered and so are the numbers. Table 5.7 presents the results. For each statement, we show the quantifier chosen by the algorithm and the mean participant response. The cells in yellow show the questions where the mean response is close to the output of the algorithm. We see that the mean responses and the algorithm output are close to each other most of the times. Most of the cases when they are further apart have one thing in common. The participants use the quantifier *many* differently than the way we define it in Figure 5.10. For example, in Case 5 for the pulse rate, P2 reported that *almost all* of the instances have

Table 5.7a (left – Pulse Rate) & 5.7b (right – Respiration Rate): Clinician responses to simple LPS survey. Participants were asked to fill in each blank with one of the quantifiers: Almost None (1), Few

	LPS	Algo-rithm	Mean Response (P1-P6)
Case 1	_____ instances have low pulse rate.	4	4.3
	_____ instances have normal pulse rate.	2	2.8
	_____ instances have high pulse rate.	1	1.0
Case 2	_____ instances have low pulse rate.	1	1.2
	_____ instances have normal pulse rate.	2	3.8
	_____ instances have high pulse rate.	4	3.5
Case 3	_____ instances have low pulse rate.	1	1.7
	_____ instances have normal pulse rate.	5	4.8
	_____ instances have high pulse rate.	1	1.0
Case 4	_____ instances have low pulse rate.	3	3.8
	_____ instances have normal pulse rate.	3	3.8
	_____ instances have high pulse rate.	1	1.2
Case 5	_____ instances have low pulse rate.	1	1.7
	_____ instances have normal pulse rate.	3	4.3
	_____ instances have high pulse rate.	3	3.3

	LPS	Algo-rithm	Mean Response (P1-P6)
Case 1	_____ instances have low respiration rate.	4	4.3
	_____ instances have normal respiration rate.	2	3.3
	_____ instances have high respiration rate.	1	1.7
Case 2	_____ instances have low respiration rate.	1	1.7
	_____ instances have normal respiration rate.	5	4.0
	_____ instances have high respiration rate.	1	2.2
Case 3	_____ instances have low respiration rate.	1	1.5
	_____ instances have normal respiration rate.	2	2.2
	_____ instances have high respiration rate.	4	4.8
Case 4	_____ instances have low respiration rate.	3	2.8
	_____ instances have normal respiration rate.	3	4.0
	_____ instances have high respiration rate.	1	2.5
Case 5	_____ instances have low respiration rate.	1	1.0
	_____ instances have normal respiration rate.	3	3.0
	_____ instances have high respiration rate.	3	4.5

normal pulse rate (5) and *many* instances have high pulse rate (4). This is not possible with the fuzzy set definitions of Figure 5.10, where *many* is designed to represent about 75% of the objects while *almost all* covers about 100% of the objects. Similarly, for this case,

participants P3 and P4 reported that that *many* instances have normal and *many* have high pulse rate. Again, this is unlikely, since according to our definition of *many*, there cannot be *many* instances of normal, as well as *many* instances of high pulse rate for the same set of objects. This explains the disagreement between the mean response and the algorithm for most of the cases. Similar to the validation of tuple summaries, this is a very small sample size to comment on the absolute validity of these types of summaries, but at the same time, it suggests that the algorithm produces meaningful results.

5.5. Discussion

Our monitoring system has been installed in six Americare, Inc facilities around Columbia, MO during the last year, including Tiger Place. We started with 30 apartments in August 2017 (systems from Tiger Place were installed during a previous project (Rantz, et al. 2017) and we reached the projected number (110 apartments) in October 2018. During this time, the monitoring system issued 699 health alerts, 377 of which were in linguistic format. Next, we present a few examples of linguistic summaries to show our summarization system in action. The summary is a composite of the individual data feature summaries, presented as a bulleted list. Also, in the final summary, we italicize the name of the health feature for emphasis. We first present an example that provides a screenshot of a linguistic alert email to demonstrate our alert notification system. Then, we present five retrospective case studies in which the summaries produced by our system correlates with changes in health conditions in five elderly residents.

Figure 5.16 shows a screenshot of the alert email that was sent to the nursing staff at the facility, where the alert was triggered due to an increase in the time spent in bed by the

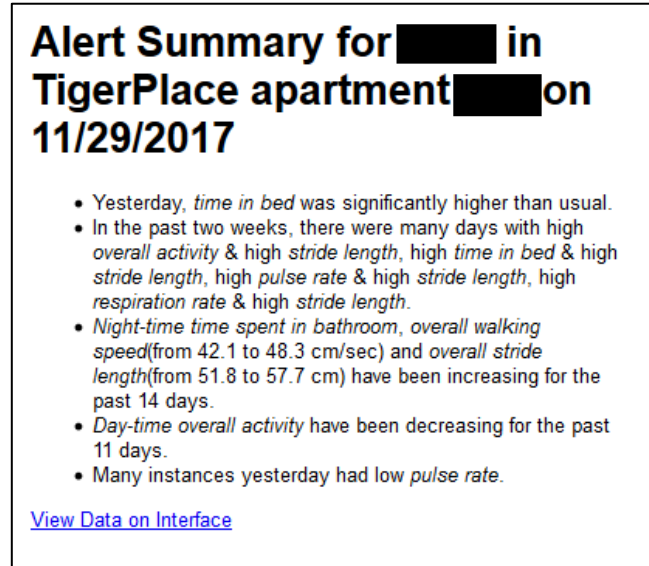


Figure 5.16: Screenshot of alert summary email corresponding to Example 1

resident. We present this example in order to illustrate our alert notification mechanism where the information is provided in linguistic format. Also, for this alert, the conditions were such that all the different parts of the alert summary were deemed worthy of being reported. As seen in the email screenshot, the summary starts with a description of the sensor feature that caused the alert. This is followed by the tuple summary listing the features that were high/low concurrently in the last two weeks. Next, all the increasing and decreasing trends in the data are reported. The last part of the summary describes the pulse rate measurements on the day of the alert.

We present five retrospective case studies in Table 5.8, where the summaries describe the change in sensor data that occurred around different health events for five residents living

Table 5.8: Five retrospective case studies. For each case, we show the health event and the date it occurred, along with the summary produced by our method on that date.

Resident ID	Date	Symptom / Health Event	Summary
1	11/14/2017	Not sleeping well	<ul style="list-style-type: none"> • Yesterday, <i>pulse rate</i> was significantly higher than usual. • <i>Night-time bed restlessness, night-time time in bed, overall stride time</i>(from 1 to 1.1 sec) and <i>overall stride length</i>(from 76.9 to 79.1 cm) have been increasing for the past 7 days. • <i>Day & night-time overall activity</i> have been decreasing for the past 6 days.
2	8/27/2017	Pneumonia	<ul style="list-style-type: none"> • Yesterday, <i>time in bed</i> was significantly higher than usual. • <i>Day & night-time bed restlessness</i> have been increasing for the past 8 days.
3	09/19/2018	UTI	<ul style="list-style-type: none"> • <i>Night-time time spent in bathroom, night-time time in bed</i> have been increasing for the past 13 days. • <i>Day & night-time overall activity, day-time bed restlessness, day-time time in bed</i> have been decreasing for the past 10 days.
4	2/25/2016	Congestive Heart Failure (CHF)	<ul style="list-style-type: none"> • In the past two weeks, there were many days with high <i>time spent in bathroom</i> & low <i>pulse rate</i>, high <i>bed restlessness</i> & low <i>pulse rate</i>, low <i>time spent in bathroom</i> & low <i>pulse rate</i>. • <i>Day-time overall activity, night-time bed restlessness, night-time time in bed</i> have been increasing for the past 8 days. • <i>Night-time time spent in bathroom, day-time bed restlessness, day-time time in bed, day & night-time pulse rate</i>(from 62.16 to 46.58 beats/min) have been decreasing for the past 10 days.
5	3/30/2019	Weakness / Hospitalization	<ul style="list-style-type: none"> • <i>Day & night-time overall activity, day & night-time bed restlessness, day & night-time time in bed</i> have been decreasing for the past 8 days.

at various sensor equipped facilities. For each case study, we list the diagnosis and the summary produced by our system on the day of the event. In the following, we list the health history of the resident around the time of the diagnosis, along with the two-week variation of the feature that is most directly related to the health event.

5.5.1. Case I

In this case, the sensors recorded a higher than usual pulse rate during the night-time for a resident. The email informs about the pulse rate increase that was the reason for the alert to begin with. It also informs about an increase in *time in bed* and *restlessness* during the night, which was also found to be the true according to the conversation between the nurse and the resident. After receiving the alert, one of the nurses checked on the resident and found that the resident was not sleeping well for the past couple of days. The lab work revealed that the resident's sodium levels were high, and she was in renal failure. In Figure 5.17, we show the restlessness of the resident during the nighttime for the past 14 days. We see a gradual increase over this time period (also shown by the trend line in blue). Note

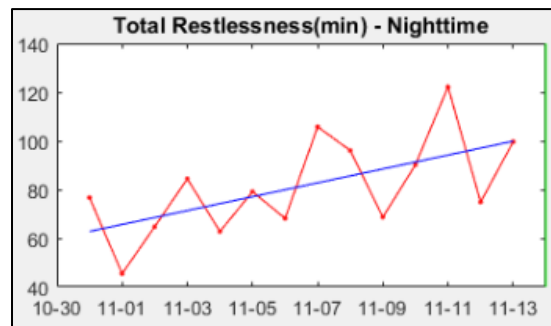


Figure 5.17: Nighttime restlessness of Resident 1 for the past two weeks leading to the day of health event. The blue line shows the least square fit line

that although the restlessness was found to be increasing for the past 14 nights, in the summary it is only reported to be increasing for the past 7 days. This is because when reporting multiple trends, we take the feature with smallest duration of trend and include only that in the summary (as described in Section 2.2.2).

5.5.2. Case II

In this example, an alert was generated by an increase in the time spent in bed. The summary starts by describing the reason for the alert. It also informs about an increase in the restlessness during the daytime as well as nighttime. This alert and the corresponding summary were generated on August 27, and the resident was diagnosed with pneumonia in early October. Since pneumonia takes a good amount of time to take full effect, there is a strong reason to believe that the increase in *time spent in bed* and *restlessness* were symptoms of the disease and were caught by the sensors much earlier than the actual diagnosis. Figure 5.18 shows the amount of time the resident was restless during the past 14 nights. According to the American Lung Association [106], symptoms of pneumonia

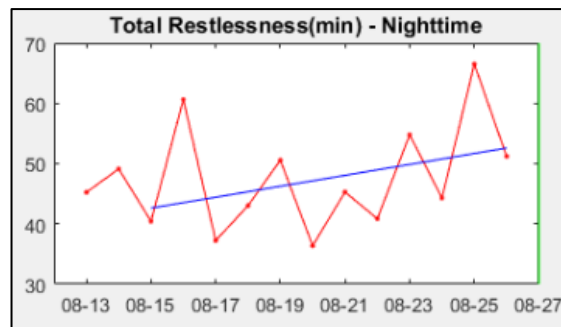


Figure 5.18: Nighttime restlessness of Resident 2 for the past two weeks leading to the day of health event. The blue line shows the least square fit line

include, but not limited to: low energy, fatigue, chest pain, and confusion. These symptoms could have contributed to the resident's restlessness and time in bed.

5.5.3. Case III

In this case, the resident was diagnosed with UTI on September 19. On the day of the diagnosis, the following entry was made into the EHR: “...resident has had c/o burning sensation with urination and has had 1-2 episodes of chills...”. The alert generation algorithm did not trigger an alert due to the gradual increase in the nighttime bathroom activity. However, the summarization method detected an increasing trend in the time spent in the bathroom over the past two weeks. The overall increase in the bathroom activity during the nighttime suggests that the summarization process was able to catch the symptoms of the underlying health condition. Figure 5.19 shows the bathroom activity of the resident during the nighttime for the two-week period. We see that the bathroom activity has been gradually increasing over this time period.

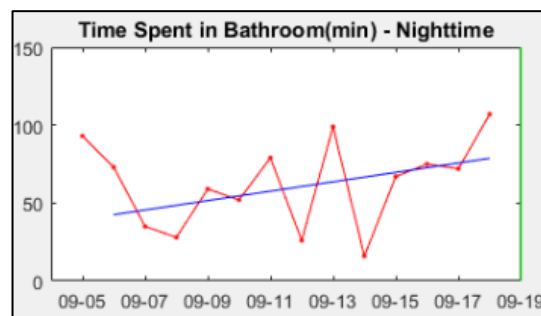


Figure 5.19 Bathroom activity of Resident 3 for the past two weeks leading to the day of health event. The blue line shows the least square fit line

5.5.4. Case IV

In this case, the resident suffered from Congestive Heart Failure (CHF) on the mentioned date. Although the alert generation algorithm did not issue an alert, we ran our summarization process to summarize the data leading to the day of CHF. Both the tuple summary and the trend summary suggested a change in the pulse rate. The trend detection algorithm detected a gradual decrease in pulse rate for the past 14 days during both nighttime and daytime, where the pulse rate decreased from 62 Beats per Minute to 46 Beats per Minute, as shown in Figure 5.20. This is a significant drop in pulse rate, which can be attributed to the change in the cardiovascular health of the resident leading to the day of CHF. Moreover, according to the American Heart Association, a resting heart rate of fewer than 60 beats per minute, as this resident exhibited, qualifies as bradycardia, which can lead to CHF.

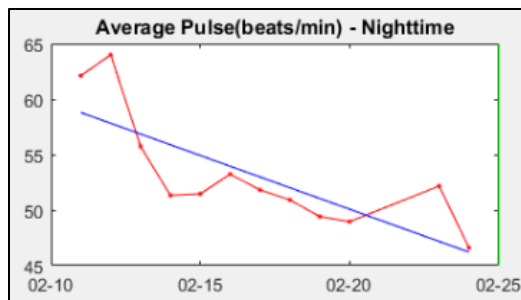


Figure 5.20: Nightly Pulse rate of Resident 4 for the past two weeks leading to the day of health event. The blue line shows the least square fit line

5.5.5. Case V

In this case, the nurse wrote the following note in the EHR: “*Res was sent to hospital this morning by ambulance res was unable to sit even with assistance she would just fall right back over*” Similar to the Case IV, the alert generation algorithm was not able to find any significant changes in the sensor data, hence it did not produce an alert. However, the summary produced on the day of hospitalization informed about the decrease in the overall activity, restlessness and time in bed over the past 8 days. This drop in activity can be attributed to the general weakness, which was reported by the nurse in the EHR. Figure 5.21 shows the overall activity of the resident inside the apartment for the past two weeks. We see over the eight days leading to the health event, the activity decreased from about 175 minutes to about 100 minutes.

Approximately one month leading up to the resident’s hospitalization and prominent weakness, nurses noted in EHR that the resident was “*sent out to the hospital for excessive diarrhea and vomiting, she return with no order changes ER gave prescription for lomotil.*”

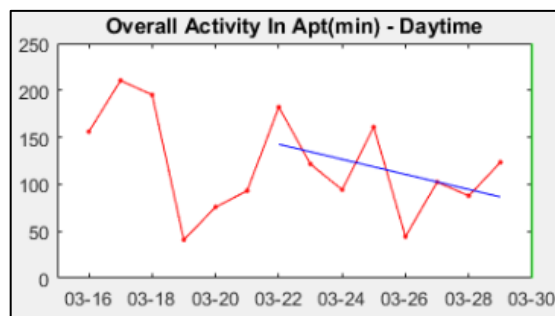


Figure 5.21: Overall activity of Resident 5 for the past two weeks leading to the day of health event. The blue line shows the least square fit line

The hospital called to inform facility that resident has Norovirus in her stool, universal precautions in place". This illness would have played in a significant role in the resident's weakness.

5.6. Conclusions & Future Work

Although the smart home sensor technology has a lot of potential to help the elderly remain independent, the amount and complexity of the data generated by the sensors make it difficult and time consuming for the clinicians to exploit the data to its fullest. We presented an NLG system that extracts important information from the sensor data and summarizes it in natural language. We carefully selected the information features that are most helpful to the clinicians and developed algorithms to compute these features from the data. We validated our algorithms with the help of multiple clinician surveys and focus groups using synthetic and real data. The outcome of the surveys showed that our algorithms have similar performance to the human subjects in terms of extracting the said features. We installed the in-home sensors in 110 elderly residents' apartments as part of an NIH-NLM project and implemented our summarization system to produce summaries of the sensor data leading to health alerts. We presented five retrospective case studies to illustrate the workings of our summarization system and show that the linguistic summaries can help detect change in the health of the elderly before the actual event happens.

Our immediate next step is to investigate whether the linguistic summarization framework can help improve the efficiency of the nurses in monitoring the in-home sensor data. To

this end, we aim to compare the time taken by the nurses to monitor the linguistic alerts versus the traditional graph-based alerts. As a long-term goal, we would like to see if describing the in-home sensor data in natural language can help reduce decline in the health of the elderly residents. Recently, hybrid techniques that employ both textual and visual formats have been found to be effective in communicating information contained in the data (Mahamood, et al. 2014, Gkatzia, et al. 2017). In line with these works, we would also like to explore the coupling of linguistic summaries and visualization techniques in way that they complement each other in presenting the in-home sensor data.

List of Publications

- **A. Jain**, M. Popescu, J. Keller, M. Rantz, and B. Markway, "Linguistic summarization of in-home sensor data," *Journal of biomedical informatics*, vol. 96, p. 103240, 2019.
- **A. Jain**, J. Keller, and M. Popescu, "Explainable AI For Dataset Comparison," in *2019 IEEE International Conference on Fuzzy Systems (FUZZ-IEEE)*, 2019, pp. 1-7.
- V. A. Shaffer, P. Wegier, K. Valentine, J. L. Belden, S. M. Canfield, S. J. Patil, M. Popescu, L. M. Steege, **A. Jain**, and R. J. Koopman, "Patient Judgments About Hypertension Control: The Role of Variability, Trends, and Outliers in Visualized Blood Pressure Data," *Journal of medical Internet research*, vol. 21, p. e11366, 2019.
- **A. Jain**, M. Popescu, J. Keller, J. Belden, R. Koopman, S. Patil, S. Canfield, L. Steege, V. Shaffer, and P. Wegier, "A decision support system for home BP measurements," in *Proceedings of the 11th EAI International Conference on Pervasive Computing Technologies for Healthcare*, 2017, pp. 231-234.
- **A. Jain**, J. M. Keller, and J. C. Bezdek, "Quantitative and qualitative comparison of periodic sensor data," in *Biomedical and Health Informatics (BHI), 2016 IEEE-EMBS International Conference on*, 2016, pp. 37-40.
- **A. Jain**, T. Jiang, and J. M. Keller, "Impact of the Shape of Membership Functions on the Truth Values of Linguistic Protoform Summaries," in *Information Processing and Management of Uncertainty in Knowledge-Based Systems*, Cham, 2016, pp. 204-213.
- **A. Jain** and J. M. Keller, "Textual summarization of events leading to health alerts," in *Engineering in Medicine and Biology Society (EMBC), 2015 37th Annual International Conference of the IEEE*, 2015, pp. 7634-7637.
- **A. Jain** and J. M. Keller, "On the computation of semantically ordered truth values of linguistic protoform summaries," in *Fuzzy Systems (FUZZ-IEEE), 2015 IEEE International Conference on*, 2015, pp. 1-8.

Bibliography

American Heart Association. (2016). *Bradycardia: Slow heart rate*. Available: <https://www.heart.org/en/health-topics/arrhythmia/about-arrhythmia/bradycardia--slow-heart-rate>

American Lung Association. (2019). *Pneumonia symptoms and diagnosis*. Available: <https://www.lung.org/lung-health-and-diseases/lung-disease-lookup/pneumonia/symptoms-and-diagnosis.html>

G. D. Abowd, A. F. Bobick, I. A. Essa, E. D. Mynatt, and W. A. Rogers, "The aware home: A living laboratory for technologies for successful aging," in *Proceedings of the AAAI-02 Workshop "Automation as Caregiver*, 2002, pp. 1-7.

M. A. Ahmad, C. Eckert, and A. Teredesai, "Interpretable Machine Learning in Healthcare," in *Proceedings of the 2018 ACM International Conference on Bioinformatics, Computational Biology, and Health Informatics*, 2018, pp. 559-560.

G. Alexander, A. Wilbik, J. Keller, and K. Musterman, "Generating sensor data summaries to communicate change in elder's health status," *Applied clinical informatics*, vol. 5, pp. 73-84, 2014.

J. M. Alonso, C. Castiello, and C. Mencar, "A Bibliometric Analysis of the Explainable Artificial Intelligence Research Field," in *International Conference on Information Processing and Management of Uncertainty in Knowledge-Based Systems*, 2018, pp. 3-15.

M. Alwan, D. C. Mack, S. Dalal, S. Kell, B. Turner, and R. A. Felder, "Impact of passive in-home health status monitoring technology in home health: Outcome pilot," in *Distributed Diagnosis and Home Healthcare, 2006. D2H2. 1st Transdisciplinary Conference on*, 2006, pp. 79-82.

D. Anderson, R. H. Luke, J. M. Keller, M. Skubic, M. Rantz, and M. Aud, "Linguistic summarization of video for fall detection using voxel person and fuzzy logic," *Computer Vision and Image Understanding*, vol. 113, pp. 80-89, 2009.

H. Banaee, M. U. Ahmed, and A. Loutfi, "A framework for automatic text generation of trends in physiological time series data," in *Systems, Man, and Cybernetics (SMC), 2013 IEEE International Conference on*, 2013, pp. 3876-3881.

- J. B. Begole, J. C. Tang, and R. Hill, "Rhythm modeling, visualizations and applications," in *Proceedings of the 16th annual ACM symposium on User interface software and technology*, 2003, pp. 11-20.
- J. C. Bezdek, J. Keller, R. Krisnapuram, and N. Pal, *Fuzzy models and algorithms for pattern recognition and image processing* vol. 4: Springer Science & Business Media, 1999.
- O. Biran and C. Cotton, "Explanation and justification in machine learning: A survey," in *IJCAI-17 Workshop on Explainable AI (XAI)*, 2017, p. 8.
- O. Biran and K. McKeown, "Generating Justifications of Machine Learning Predictions," in *1st International Workshop on Data-to-text Generation*, Edinburgh, 2015.
- F. E. Boran, D. Akay, and R. R. Yager, "An overview of methods for linguistic summarization with fuzzy sets," *Expert Systems with Applications*, vol. 61, pp. 356-377, 2016.
- F. Cardinaux, S. Brownsell, M. Hawley, and D. Bradley, "Modelling of behavioural patterns for abnormality detection in the context of lifestyle reassurance," in *Iberoamerican Congress on Pattern Recognition*, 2008, pp. 243-251.
- R. Caruana, Y. Lou, J. Gehrke, P. Koch, M. Sturm, and N. Elhadad, "Intelligible models for healthcare: Predicting pneumonia risk and hospital 30-day readmission," in *Proceedings of the 21th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, 2015, pp. 1721-1730.
- A. J. Cawsey, B. L. Webber, and R. B. Jones, "Natural language generation in health care," ed: BMJ Group BMA House, Tavistock Square, London, WC1H 9JR, 1997.
- S.-H. Cha, "Comprehensive Survey on Distance/Similarity Measures between Probability Density Functions," *International Journal of Mathematical models and Methods in Applied Sciences*, vol. 1, pp. 300-307, 2007.
- W. S. Cleveland, "Robust locally weighted regression and smoothing scatterplots," *Journal of the American statistical association*, vol. 74, pp. 829-836, 1979.
- D. J. Cook, A. S. Crandall, B. L. Thomas, and N. C. Krishnan, "CASAS: A smart home in a box," *Computer*, vol. 46, pp. 62-69, 2013.
- P. N. Dawadi, D. J. Cook, and M. Schmitter-Edgecombe, "Modeling patterns of activities using activity curves," *Pervasive and mobile computing*, vol. 28, pp. 51-68, 2016.

- M. Delgado, M. D. Ruiz, D. Sánchez, and M. A. Vila, "Fuzzy quantification: a state of the art," *Fuzzy Sets and Systems*, vol. 242, pp. 1-30, 2014.
- M. Delgado, D. Sánchez, and M. A. Vila, "Fuzzy cardinality based evaluation of quantified sentences," *International Journal of Approximate Reasoning*, vol. 23, pp. 23-66, 2000.
- B. Di Eugenio, A. D. Boyd, C. Lugaresi, A. Balasubramanian, G. M. Keenan, M. Burton, *et al.*, "PatientNarr: Towards generating patient-centric summaries of hospital stays," *INLG 2014*, p. 6, 2014.
- D. Doran, S. Schulz, and T. R. Besold, "What does explainable AI really mean? A new conceptualization of perspectives," *arXiv preprint arXiv:1710.00794*, 2017.
- F. Doshi-Velez, M. Kortz, R. Budish, C. Bavitz, S. Gershman, D. O'Brien, *et al.*, "Accountability of AI under the law: The role of explanation," *arXiv preprint arXiv:1711.01134*, 2017.
- N. Eagle and A. S. Pentland, "Reality mining: sensing complex social systems," *Personal and ubiquitous computing*, vol. 10, pp. 255-268, 2006.
- N. Eagle and A. S. Pentland, "Eigenbehaviors: Identifying structure in routine," *Behavioral Ecology and Sociobiology*, vol. 63, pp. 1057-1066, 2009.
- S. N. Friel, F. R. Curcio, and G. W. Bright, "Making sense of graphs: Critical factors influencing comprehension and instructional implications," *Journal for Research in mathematics Education*, pp. 124-158, 2001.
- C. Galambos, M. Skubic, S. Wang, and M. Rantz, "Management of dementia and depression utilizing in-home passive sensor data," *Gerontechnology: international journal on the fundamental aspects of technology to serve the ageing society*, vol. 11, p. 457, 2013.
- M. Galesic and R. Garcia-Retamero, "Graph literacy: A cross-cultural comparison," *Medical Decision Making*, vol. 31, pp. 444-457, 2011.
- A. Gatt and E. Krahmer, "Survey of the State of the Art in Natural Language Generation: Core tasks, applications and evaluation," *Journal of Artificial Intelligence Research*, vol. 61, pp. 65-170, 2018.
- A. Gatt, F. Portet, E. Reiter, J. Hunter, S. Mahamood, W. Moncur, *et al.*, "From data to text in the neonatal intensive care unit: Using NLG technology for decision support and information management," *Ai Communications*, vol. 22, pp. 153-186, 2009.
- L. H. Gilpin, D. Bau, B. Z. Yuan, A. Bajwa, M. Specter, and L. Kagal, "Explaining Explanations: An Overview of Interpretability of Machine Learning," in *2018 IEEE 5th*

International Conference on Data Science and Advanced Analytics (DSAA), 2018, pp. 80-89.

D. Gkatzia, O. Lemon, and V. Rieser, "Natural Language Generation enhances human decision-making with uncertain information," *arXiv preprint arXiv:1606.03254*, 2016.

D. Gkatzia, O. Lemon, and V. Rieser, "Data-to-text generation improves decision-making under uncertainty," *IEEE Computational Intelligence Magazine*, vol. 12, pp. 10-17, 2017.

A. Goldstein and Y. Shahar, "An automated knowledge-based textual summarization system for longitudinal, multivariate clinical data," *Journal of biomedical informatics*, vol. 61, pp. 159-175, 2016.

B. Goodman and S. Flaxman, "European Union regulations on algorithmic decision-making and a right to explanation", " *arXiv preprint arXiv:1606.08813*, 2016.

J. Gross, "A grass-roots effort to grow old at home," *The New York Times*, vol. 14, 2007.

S. Guillaume, "Designing fuzzy inference systems from data: An interpretability-oriented review," *IEEE transactions on fuzzy systems*, vol. 9, pp. 426-443, 2001.

D. Gunning, "Explainable artificial intelligence (xai)," *Defense Advanced Research Projects Agency (DARPA)*, nd Web, 2017.

K. A. Hallgren, "Computing inter-rater reliability for observational data: an overview and tutorial," *Tutorials in quantitative methods for psychology*, vol. 8, p. 23, 2012.

R. J. Hathaway and J. C. Bezdek, "NERF c-means: Non-Euclidean relational fuzzy clustering," *Pattern recognition*, vol. 27, pp. 429-437, 1994.

T. C. Havens and J. C. Bezdek, "An efficient formulation of the improved visual assessment of cluster tendency (iVAT) algorithm," *Knowledge and Data Engineering, IEEE Transactions on*, vol. 24, pp. 813-822, 2012.

D. Heise, L. Rosales, M. Skubic, and M. J. Devaney, "Refinement and evaluation of a hydraulic bed sensor," in *2011 Annual International Conference of the IEEE Engineering in Medicine and Biology Society*, 2011, pp. 4356-4360.

S. Helal and C. Chen, "The Gator Tech Smart House: enabling technologies and lessons learned," in *Proceedings of the 3rd International Convention on Rehabilitation Engineering & Assistive Technology*, 2009, p. 13.

- L. A. Hendricks, Z. Akata, M. Rohrbach, J. Donahue, B. Schiele, and T. Darrell, "Generating visual explanations," in *European Conference on Computer Vision*, 2016, pp. 3-19.
- A. Holzinger, C. Biemann, C. S. Pattichis, and D. B. Kell, "What do we need to build explainable AI systems for the medical domain?," *arXiv preprint arXiv:1712.09923*, 2017.
- G. Hripcsak and A. Wilcox, "Reference standards, judges, and comparison subjects: roles for experts in evaluating system performance," *Journal of the American Medical Informatics Association*, vol. 9, pp. 1-15, 2002.
- D. Huk Park, L. Anne Hendricks, Z. Akata, A. Rohrbach, B. Schiele, T. Darrell, *et al.*, "Multimodal explanations: Justifying decisions and pointing to the evidence," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2018, pp. 8779-8788.
- T. Huynh, M. Fritz, and B. Schiele, "Discovery of activity patterns using topic models," in *Proceedings of the 10th international conference on Ubiquitous computing*, 2008, pp. 10-19.
- O. A. Ibrahim, J. Keller, and M. Popescu, "An Unsupervised Framework for Detecting Early Signs of Illness in Eldercare," in *2019 IEEE International Conference on Bioinformatics and Biomedicine (BIBM)*, 2019, pp. 1043-1050.
- S. S. Intille, K. Larson, and E. M. Tapia, "Designing and evaluating technology for independent aging in the home," in *International Conference on Aging, Disability and Independence*, 2003.
- A. Jain and J. M. Keller, "On the computation of semantically ordered truth values of linguistic protoform summaries," in *Fuzzy Systems (FUZZ-IEEE), 2015 IEEE International Conference on*, 2015, pp. 1-8.
- A. Jain and J. M. Keller, "Textual summarization of events leading to health alerts," in *Engineering in Medicine and Biology Society (EMBC), 2015 37th Annual International Conference of the IEEE*, 2015, pp. 7634-7637.
- A. Jain, J. M. Keller, and J. C. Bezdek, "Quantitative and qualitative comparison of periodic sensor data," in *Biomedical and Health Informatics (BHI), 2016 IEEE-EMBS International Conference on*, 2016, pp. 37-40.
- A. Jain, M. Popescu, J. Keller, J. Belden, R. Koopman, S. Patil, *et al.*, "A decision support system for home BP measurements," in *Proceedings of the 11th EAI International Conference on Pervasive Computing Technologies for Healthcare*, 2017, pp. 231-234.

- S. Jiang, J. Ferreira, and M. C. González, "Clustering daily patterns of human activities in the city," *Data Mining and Knowledge Discovery*, vol. 25, pp. 478-510, 2012.
- Y. Jin, "Fuzzy modeling of high-dimensional systems: complexity reduction and interpretability improvement," *IEEE Transactions on Fuzzy Systems*, vol. 8, pp. 212-221, 2000.
- J. Kacprzyk, A. Wilbik, and S. Zadrozny, "Linguistic Summaries of Time Series via a Quantifier Based Aggregation Using the Sugeno Integral," in *FUZZ-IEEE*, 2006, pp. 713-719.
- J. Kacprzyk, A. Wilbik, and S. Zadrozny, "An approach to the linguistic summarization of time series using a fuzzy quantifier driven aggregation," *International Journal of Intelligent Systems*, vol. 25, pp. 411-439, 2010.
- J. A. Kaye, S. A. Maxwell, N. Mattek, T. L. Hayes, H. Dodge, M. Pavel, *et al.*, "Intelligent systems for assessing aging changes: home-based, unobtrusive, and continuous assessment of aging," *Journals of Gerontology Series B: Psychological Sciences and Social Sciences*, vol. 66, pp. i180-i190, 2011.
- J. M. Keller, D. Liu, and D. Fogel, *Fundamentals of Computational Intelligence: Neural Networks, Fuzzy Systems, and Evolutionary Computation*, : Wiley/IEEE Press, 2016.
- C. Lacave and F. J. Díez, "A review of explanation methods for Bayesian networks," *The Knowledge Engineering Review*, vol. 17, pp. 107-127, 2002.
- F. M. Ludwig, "How routine facilitates wellbeing in older women," *Occupational Therapy International*, vol. 4, pp. 215-230, 1997.
- S. Mahamood, W. Bradshaw, and E. Reiter, "Generating annotated graphs using the nlq pipeline architecture," in *Proceedings of the 8th International Natural Language Generation Conference (INLG)*, 2014, pp. 123-127.
- D. Martens, J. Huysmans, R. Setiono, J. Vanthienen, and B. Baesens, "Rule extraction from support vector machines: An overview of issues and application in credit scoring," in *Rule extraction from support vector machines*, ed: Springer, 2008, pp. 33-63.
- B. D. Mittelstadt, P. Allo, M. Taddeo, S. Wachter, and L. Floridi, "The ethics of algorithms: Mapping the debate," *Big Data & Society*, vol. 3, p. 2053951716679679, 2016.
- J. M. Ortman, V. A. Velkoff, and H. Hogan, *An aging nation: the older population in the United States*.

L. J. Phillips, C. B. DeRoche, M. Rantz, G. L. Alexander, M. Skubic, L. Despins, *et al.*, "Using embedded sensors in independent living to predict gait changes and falls," *Western journal of nursing research*, vol. 39, pp. 78-94, 2017.

M. Popescu, "Early illness detection in elderly using sensor networks: a review of the TigerPlace experience," in *E-Health and Bioengineering Conference (EHB)*, 2015, 2015, pp. 1-6.

M. Popescu, A. Craver, L. Phillips, R. Koopman, G. Alexander, L. Despins, *et al.*, "Linking Resident Behavior to Health Conditions in an Eldercare Monitoring System," in *AMIA Fall Symposium*, Washington DC, 2017, pp. 161-162.

C. I. QuietCare, "Intel-GE Care Innovations LLC," URL: <http://www.careinnovations.com/quietcare>.

A. Ramos-Soto, A. J. Bugarin, S. Barro, and J. Taboada, "Linguistic descriptions for automatic generation of textual short-term weather forecasts on real prediction data," *IEEE Transactions on Fuzzy Systems*, vol. 23, pp. 44-57, 2015.

M. Rantz, L. J. Phillips, C. Galambos, K. Lane, G. L. Alexander, L. Despins, *et al.*, "Randomized trial of intelligent sensor system for early illness alerts in senior housing," *Journal of the American Medical Directors Association*, vol. 18, pp. 860-870, 2017.

M. Rantz, M. Skubic, S. Miller, and J. Krampe, "Using technology to enhance aging in place," in *International Conference On Smart homes and health Telematics*, 2008, pp. 169-176.

M. J. Rantz, K. D. Marek, M. Aud, H. W. Tyrer, M. Skubic, G. Demiris, *et al.*, "A technology and nursing collaboration to help older adults age in place," *Nursing Outlook*, vol. 53, pp. 40-45, 2005.

M. J. Rantz, S. D. Scott, S. J. Miller, M. Skubic, L. Phillips, G. Alexander, *et al.*, "Evaluation of health alerts from an early illness warning system in independent living," *Computers, informatics, nursing: CIN*, vol. 31, p. 274, 2013.

M. J. Rantz, M. Skubic, R. J. Koopman, G. L. Alexander, L. Phillips, K. Musterman, *et al.*, "Automated technology to speed recognition of signs of illness in older adults," *Journal of Gerontological Nursing*, vol. 38, pp. 18-23, 2012.

M. J. Rantz, M. Skubic, S. J. Miller, C. Galambos, G. Alexander, J. Keller, *et al.*, "Sensor technology to support aging in place," *Journal of the American Medical Directors Association*, vol. 14, pp. 386-391, 2013.

- M. T. Ribeiro, S. Singh, and C. Guestrin, "Model-agnostic interpretability of machine learning," *arXiv preprint arXiv:1606.05386*, 2016.
- L. Rosales, B. Y. Su, M. Skubic, and K. Ho, "Heart rate monitoring using hydraulic bed sensor ballistocardiogram 1," *Journal of Ambient Intelligence and Smart Environments*, vol. 9, pp. 193-207, 2017.
- D. Sanchez-Valdes, A. Alvarez-Alvarez, and G. Trivino, "Dynamic linguistic descriptions of time series applied to self-track the physical activity," *Fuzzy Sets and Systems*, vol. 285, pp. 162-181, 2016.
- D. Scott, C. Hallett, and R. Fettiplace, "Data-to-text summarisation of patient records: Using computer-generated summaries to access patient histories," *Patient Education and Counseling*, vol. 92, pp. 153-159, 2013.
- J. H. Shin, B. Lee, and K. S. Park, "Detection of abnormal living patterns for elderly living alone using support vector data description," *IEEE Transactions on Information Technology in Biomedicine*, vol. 15, pp. 438-448, 2011.
- S. Sripada, E. Reiter, and I. Davy, "SumTime-Mousam: Configurable marine weather forecast generator," *Expert Update*, vol. 6, pp. 4-10, 2003.
- E. E. Stone and M. Skubic, "Unobtrusive, continuous, in-home gait measurement using the Microsoft Kinect," *IEEE Transactions on Biomedical Engineering*, vol. 60, pp. 2925-2932, 2013.
- E. E. Stone and M. Skubic, "Fall detection in homes of older adults using the Microsoft Kinect," *IEEE journal of biomedical and health informatics*, vol. 19, pp. 290-301, 2014.
- E. E. Stone, M. Skubic, and J. Back, "Automated health alerts from kinect-based in-home gait measurements," in *Engineering in Medicine and Biology Society (EMBC), 2014 36th Annual International Conference of the IEEE*, 2014, pp. 2961-2964.
- B. Y. Su, K. Ho, M. Skubic, and L. Rosales, "Pulse rate estimation using hydraulic bed sensor," in *2012 annual international conference of the IEEE engineering in medicine and biology society*, 2012, pp. 2587-2590.
- Y. van Kasteren, D. Bradford, Q. Zhang, M. Karunanithi, and H. Ding, "Understanding Smart Home Sensor Data for Ageing in Place Through Everyday Household Routines: A Mixed Method Case Study," *JMIR mHealth and uHealth*, vol. 5, 2017.
- G. Virone, M. Alwan, S. Dalal, S. W. Kell, B. Turner, J. A. Stankovic, *et al.*, "Behavioral patterns of older adults in assisted living," *IEEE Transactions on Information Technology in Biomedicine*, vol. 12, pp. 387-398, 2008.

- S. Wang, "Change detection for eldercare using passive sensing," *University of Missouri-Columbia*, 2011.
- S. Wang, M. Skubic, and Y. Zhu, "Activity density map visualization and dissimilarity comparison for eldercare monitoring," *Information Technology in Biomedicine, IEEE Transactions on*, vol. 16, pp. 607-614, 2012.
- R. J. White, "Using topic models to detect behaviour patterns for healthcare monitoring," *University of Reading*, 2018.
- A. Wilbik, "Linguistic summaries of time series using fuzzy sets and their application for performance analysis of investment funds," in *Ph. D. dissertation, Syst. Res. Inst., Polish Academy Sci.*, 2010.
- A. Wilbik and R. M. Dijkman, "On the generation of useful linguistic summaries of sequences," in *2016 IEEE International Conference on Fuzzy Systems (FUZZ-IEEE)*, 2016, pp. 555-562.
- A. Wilbik, U. Kaymak, J. Keller, and M. Popescu, "Evaluation of the Truth Value of Linguistic Summaries – Case with Non-monotonic Quantifiers," in *Intelligent Systems'2014*. vol. 322, P. Angelov, K. T. Atanassov, L. Doukovska, M. Hadjiski, V. Jotsov, J. Kacprzyk, *et al.*, Eds., ed: Springer International Publishing, 2015, pp. 69-79.
- A. Wilbik and J. M. Keller, "A distance metric for a space of linguistic summaries," *Fuzzy Sets and Systems*, vol. 208, pp. 79-94, 2012.
- A. Wilbik and J. M. Keller, "Anomaly detection from linguistic summaries," in *Fuzzy Systems (FUZZ), 2013 IEEE International Conference on*, 2013, pp. 1-7.
- A. Wilbik, J. M. Keller, and G. L. Alexander, "Linguistic summarization of sensor data for eldercare," in *Systems, Man, and Cybernetics (SMC), 2011 IEEE International Conference on*, 2011, pp. 2595-2599.
- A. Wilbik, J. M. Keller, and J. C. Bezdek, "Linguistic Prototypes for Data From Eldercare Residents," *Fuzzy Systems, IEEE Transactions on*, vol. 22, pp. 110-123, 2014.
- R. R. Yager, "A new approach to the summarization of data," *Information Sciences*, vol. 28, pp. 69-86, 1982.
- M. Yefimova, "Using in-Home Monitoring Technology to Identify Deviations in Daily Routines Preceding Changes in Health Trajectory of Older Adults," *UCLA*, 2016.

J. Yin, Q. Zhang, and M. Karunanithi, "Unsupervised daily routine and activity discovery in smart homes," in *Engineering in Medicine and Biology Society (EMBC), 2015 37th Annual International Conference of the IEEE*, 2015, pp. 5497-5500.

J. Yu, E. Reiter, J. Hunter, and C. Mellish, "Choosing the content of textual summaries of large time-series data sets," *Natural Language Engineering*, vol. 13, pp. 25-49, 2007.

L. A. Zadeh, "A computational approach to fuzzy quantifiers in natural languages," in *Computational linguistics*, ed: Elsevier, 1983, pp. 149-184.

L. A. Zadeh, "A prototype-centered approach to adding deduction capability to search engines-the concept of protoform," in *2002 Annual Meeting of the North American Fuzzy Information Processing Society Proceedings. NAFIPS-FLINT 2002 (Cat. No. 02TH8622)*, 2002, pp. 523-525.

A. Zisberg, N. Gur-Yaish, and T. Shochat, "Contribution of routine to sleep quality in community elderly," *Sleep*, vol. 33, pp. 509-514, 2010.

VITA

Akshay Jain grew up in Kota, Rajasthan, India. After finishing primary schooling in his hometown at St. Paul's Senior Secondary School, he went to college at Maharashtra Institute of Technology, Pune in Maharashtra, India in the year 2006. He graduated with Bachelors in Electronics and Telecommunication Engineering in the year 2010 after which he worked for two years as a DSP Engineer at Wavelet Group, Pune. He was admitted to the Master of Science program at the University of Missouri, Columbia in 2012 where he started working at the Vision Guided and Robotics Laboratory as a Graduate Student. In 2014, he joined the Center for Eldercare and Rehabilitation Technology as a Graduate Research Assistant, where he worked on two federally funded grants and published his work at various peer reviewed venues. He finished his Master of Science thesis titled, 'Textual Summarization of Sensor Data', in 2015 under the supervision of Dr. James Keller with whom he continued to work on his doctoral degree. During his doctoral degree, he worked on developing machine-learning methods to summarize patterns in data generated from smart-homes of the elderly in natural language. He successfully defended his doctoral dissertation titled 'Explainable Pattern Modelling and Summarization in Sensor Equipped Smart Homes of Elderly' in May 2020. His research interests include machine learning, computational intelligence, data science and sensor based systems that improve general wellbeing. He enjoys playing table tennis, listening to music, and travelling.