International Journal of Technology 10(7): 1276-1285
ISSN 2086-9614

ANOMALY DETECTION FOR HOME ACTIVITY BASED ON SEQUENCE PATTERN

Soon-Chang Poh[1*], Yi-Fei Tan[1], Soon-Nyean Cheong[1], Chee-Pun Ooi[1], Wooi-Haw Tan[1]

[1]*Faculty of Engineering, Multimedia University, Persiaran Multimedia, 63000, Selangor, Malaysia*

(Received: November 2018 / Revised: January 2019 / Accepted: September 2019)

ABSTRACT

In Malaysia, the elderly population continues to grow. At the same time, young adults are unable to take care of their elderly parents due to work commitments. This results in an increasing number of elderly people living in solitude. Therefore, it is crucial to monitor elderly people's behavior, especially the pattern of their daily home activities. Abnormal behaviors in carrying out home activities may indicate health concerns in elderly people. Past studies have proposed the use of complex machine learning algorithms to detect anomalies in daily sequences of home activities. In this paper, a simple, alternative method for detecting anomalies in daily sequences of home activities is presented. The experiment results demonstrate that the model achieved a test accuracy of 90.79% on a public dataset.

Keywords: Anomaly detection; Elderly; Home activities; Sequence pattern

1. INTRODUCTION

According to the World Health Organization (2018), the population of elderly people (60 years old and over) in the Southeast Asia region will likely increase to around 12% in 2025. In Malaysia, the Department of Statistics recorded an increase in the old-age population (65 years and over) from 2 million to 2.1 million from 2017 to 2018 (Mahidin, 2018). Due to work commitments, young adults are unable to care for their elderly parents all the time. One solution to this problem is to hire caretakers to look after elderly people. However, this solution has several limitations. First, a caretaker service might not be affordable for some families due to its high price. Second, it is difficult to hire caregivers to take care of elderly people all day and night throughout the week. Amid the development of Internet of Things (IoT) technology and machine learning, an alternative solution arises, known as activity recognition. By installing sensors at home and applying machine learning algorithms to classify sensor data, the activities of a person at home can be tracked and recorded (Lara & Labrador, 2012; Bux, et al., 2016). Another type of activity recognition employs wearable sensors such as an accelerometer for activity recognition (Dwiyantoro et al., 2016; Zainuddin et al., 2017). In a retirement town, activity recognition can be used to monitor a large number of elderly residents with a few human experts.

Behavioral changes in carrying out home activities may be related to an elderly person's health decline. For example, a sudden increase in sleeping duration during the daytime may imply that the person is not well. With activity recognition, the historical records of the elderly person's home activities can be obtained. By conducting data analysis on these records using an anomaly detection method, changes in behavior can be detected. In this paper, an anomaly detection

[*]Corresponding author's email: psoonchang@gmail.com Tel. +60173157637
Permalink/DOI: https://dx.doi.org/10.14716/ijtech.v10i7.3230

method to detect anomalies in patterns of the daily sequences of home activities is presented. Figure 1 illustrates an example of a sequence of home activities, consisting of all the activities that occurred in a day from 0:00 to 23:59. Each daily sequence is a data instance in this paper.

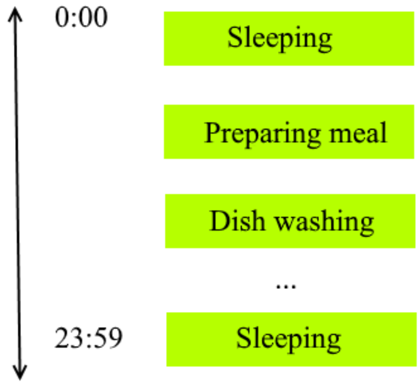

Figure 1 An example of daily sequence of home activities

Forkan et al. (2015) proposed an anomaly detection based on the Hidden Markov Model (HMM) to identify anomalies in a sequence pattern of home activities. For HMM, the number of hidden states is an unknown parameter. The researchers trained several models with a varying number of states on a synthetic dataset generated based on public datasets. The results showed the method has an average accuracy of 90%.

Damla and Bouchachia (2017) presented an anomaly detection method using Recurrent Neural Network (RNN) to detect abnormal behavior for elderly people with dementia with a true positive rate of 91.43% and a false positive rate of 40.96%. In this work, daily activity sequences of a public dataset were used as normal data, whereas abnormal data were artificially generated by injecting some abnormal activity sequences into the normal data. The researchers trained the RNN model on a training set of activity sequences. For each activity sequence, the RNN outputted a confidence value that ranged from 0 to 1. The average of the training set confidence values was used as the threshold for anomaly detection. Given an activity sequence and a trained RNN model, if the confidence value outputted by the RNN is higher than the threshold, then the activity sequence is classified as normal and vice versa.

Hoque et al. (2015) used a sequential pattern mining algorithm called PrefixSpan and a statistical method for anomaly detection. First, they retrieved home activity sequences frequent in the dataset using PrefixSpan. Then, they modelled the duration and interval between activities with Gaussian distribution for anomaly detection.

On the other hand, Riboni et al. (2016) introduced a rule-based anomaly detection method to detect anomalies in the home activity of patients with mild cognitive impairments. They defined sequences of activity that are unique to dementia patients.

Zhao et al. (2014) introduced a method using 2 Markov chains to detect anomalies in an elderly person’s location sequences at home with a high detection ratio of 92.539%. For this method, a Markov chain was trained on normal data, while another was trained on abnormal data. Given an unseen location sequence, a ratio of probabilities calculated with these trained Markov chains is used to classify whether it is anomalous or not.

2. METHODS

The flowchart in Figure 2 illustrates the methodology for building the anomaly detection model to identify anomalies in a daily sequence of home activities. In subsection 2.1, the details of the dataset used for the experiment are presented. In subsection 2.2, the process for cleaning the raw data instances and processing them into the required format is described. In addition, the processed data instances were split into training, validation and test sets. In subsection 2.3, the anomaly detection method is described. The anomaly detection method needs a threshold for classifying the data instances as normal or anomalous. The threshold can be obtained through threshold sampling and model selection. In subsection 2.4, the method for evaluating model performance is given.

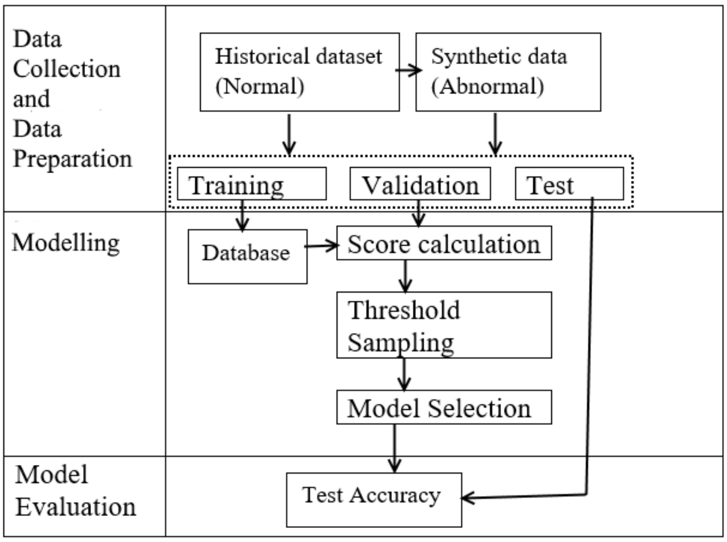

Figure 2 Flow of methodology

2.1. Data Collection

The dataset (Cook, 2010) used in this research is the “Aruba” dataset downloaded from website of Centre of Advanced Studies in Adaptive System (CASAS), Washington State University. It is a daily life record for 220 days of an adult volunteer living in a smart home installed with multiple kinds of sensors, such as motion sensors for activity recognition research. The dataset consists of the volunteer’s daily home activities and their respective timestamps and sensor states. Table 1 displays all 11 types of activity that appear in the dataset and their corresponding number of occurrences. The most frequent activity type is preparing meals and the rarest is Resperate.

Table 1 Type of activities and number of occurrences

Type of activity	Number of occurrences
Preparing meal	1606
Relaxing	2910
Eating	257
Working	171
Sleeping	401
Dish washing	65
Toileting	157
Entering home	431
Leaving home	431
Housekeeping	33
Resperate	6

The daily sequences of home activities in the dataset were used as normal data in this research. On the other hand, abnormal data were generated by randomly inserting multiple toileting and sleeping activities during the nighttime of a normal daily sequence of home activities. This simulates the abnormal activities of a diabetes symptom called nocturia, which involves waking up many times during sleep time to urinate.

2.2. Data Preparation

In this subsection, the transformation of the dataset into the model's desired format is detailed. The process consists of three steps, which are data cleaning, data processing and data partitioning.

Data cleaning aims to remove noise in the dataset. There are three types of noise in this dataset. The first is the occurrence of activity sandwich, which is a unique characteristic of the Aruba dataset where some activities happened (were sandwiched) between the beginning and end of another activity. This type of data needed to be removed because it may affect the model's performance. The second type of noise is daily sequences of home activities whose length varied too much from that of the rest of the sequences. Figure 3 shows a histogram that indicates the lengths of home activity sequences. Some daily sequences have a length that varies too much from the rest. During the noise removal process, any sequences with a length less than 10 or more than 30 were removed. The third type of noise is the existence of rare activities. As shown in Table 1, "Respirate" only occurred six times in the entire dataset, making it only about 0.0099% of the dataset. Therefore, it was removed from the dataset. The size of the remaining data is 189.

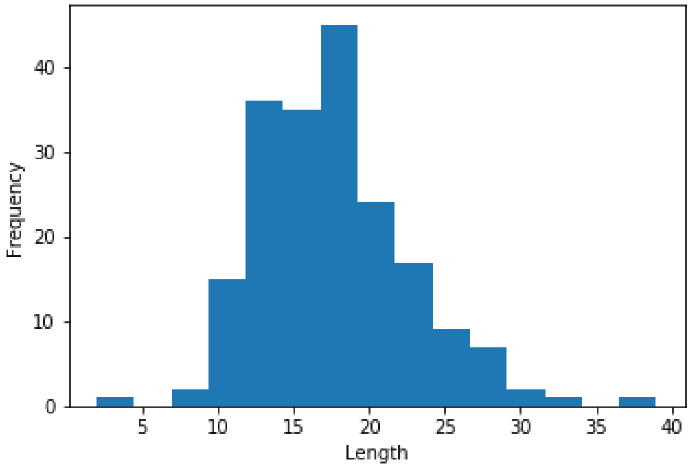

Figure 3 Histogram of home activity sequence length for Aruba dataset

The second step for the data preparation is data processing, which converts the dataset into the format required by the anomaly detection model. In this step, each daily sequence of activity was segmented into shorter sequences by applying a sliding window of size 3 and a step size of 1. Thus, each segmented sequence consists of three activities. For example, given a daily sequence of activities of length 5 (*Sleeping*, *Preparing meal*, *Eating*, *Dish washing*, *Relaxing*), we can split it into three segmented sequences, which include (*Sleeping*, *Preparing meal*, *Eating*), (*Preparing meal*, *Eating*, *Dish washing*) and (*Eating*, *Dish washing*, *Relaxing*).

Table 2 Data partitioning

	Small	Medium	Large
Training	30	60	113
Validation	76	76	76
Test	76	76	76

The final step for the data preparation is data partitioning. During data partitioning, data instances were split into training, validation and test sets. The training set is used for model building. The validation set is used to evaluate many different models and select the best one. The test set is used to evaluate the performance of the selected model. This is to ensure the consistency of the selected model's performance on an unseen dataset. To investigate the effects of the training set size on the performance of the anomaly detection method, three different training set sizes were used and labelled as small, medium and large as listed in Table 2. The training set consists of only normal data instances split from 189 normal data instances. The size of the validation and test sets are 76 respectively (each consists of 38 normal data instances and 38 abnormal data instances), regardless of the training set size.

2.3. Modelling

The anomaly detection model is trained using only normal data instances. A score metric is used as a measure of the degree of abnormalities. Given a new data instance, the trained model is applied and a score can be calculated. Then, the calculated score is compared with the threshold to classify the data instance as anomalous or not. Figure 4 illustrates the components of the anomaly detection model, which include the database and anomaly detector. Subsection 2.3.1 discusses the anomaly detection method and its training procedure. There is no method to obtain the exact threshold. The solution is to select a few threshold choices and choose the one that works best. Subsection 2.3.2 presents the method for sampling the threshold choices. Subsection 2.3.3 discusses the model selection step for choosing the model with the most suitable threshold.

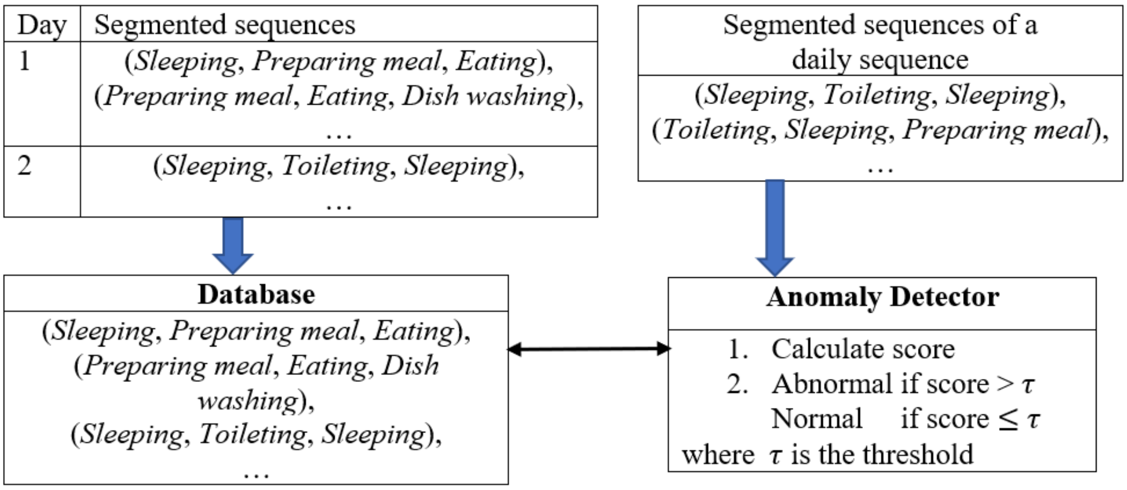
Day
Segmented sequences
1
(*Sleeping*, *Preparing meal*, *Eating*),
(*Preparing meal*, *Eating*, *Dish washing*),
…
2
(*Sleeping*, *Toileting*, *Sleeping*),
…
Segmented sequences of a daily sequence
(*Sleeping*, *Toileting*, *Sleeping*),
(*Toileting*, *Sleeping*, *Preparing meal*),
…
Database
(*Sleeping*, *Preparing meal*, *Eating*),
(*Preparing meal*, *Eating*, *Dish washing*),
(*Sleeping*, *Toileting*, *Sleeping*),
…
Anomaly Detector
1. Calculate score
2. Abnormal if score > τ
Normal if score ≤ τ
where τ is the threshold

Figure 4 Overview of anomaly detection system

2.3.1. *Model description and training process*

The first component of the proposed model is the database. The database models a subject's sequence pattern of home activities. During the training process, the segmented sequences of the training set were saved into the database. For this method, each unique segmented sequence was only saved once. In the experiment, three different databases were trained using three training sets with varying sizes.

The second component is the anomaly detector. Basically, the anomaly detector decides whether a new data instance (a daily sequence of home activities) is anomalous or not. Given a data instance, the first step is to split it into segmented sequences by applying a sliding window of size 3 and step size 1 as described in the data processing step. Then, we check whether each segmented sequence is recorded in the database or not. A segmented sequence of a data instance that is not saved in the database is called an abnormal segmented sequence. After checking all the segmented sequences, a score can be calculated for that data instance using Equation 1. The score is actually the percentage of the abnormal segmented sequence out of all the segmented sequences of the given data instance. In an anomaly detection task, the threshold is the cutting point between the scores of normal and abnormal data instances. During anomaly detection, if the score computed for a data instance is higher than the threshold, then that data instance is classified as anomalous. Conversely, if the score is less than the threshold, it is classified as a normal data instance.

$$\text{score} = \frac{ASS}{TSS} \times 100\% \tag{1}$$

where ASS is the number of abnormal segmented sequences and TSS is the total number of segmented sequences.

2.3.2. Threshold sampling

Because the optimal threshold is unknown, the threshold becomes one of the parameters in the model. Therefore, we have to sample some threshold choices systematically, build and evaluate different models with varying threshold choices and choose the best one. In this section, we discuss how to sample threshold choices. In subsection 2.3.3, the method for evaluating and selecting a model is described. To sample threshold choices, the first step is to compute the score for every data instance in the validation set. Then, *n* threshold choices are linearly sampled from the range between the minimum and maximum scores using Equation 2. In the experiment, we used *n* = 10, which resulted in 30 models with different databases and varying thresholds.

$$\text{Threshold}(i) = \frac{\text{max score} - \text{min score}}{n-1} \times i \tag{2}$$

where $i = 0,1,\ldots,n-1$.

2.3.3. Model selection

The model selection is a process for choosing the best model from all the existing models. In model selection, the data in the validation set and a performance metric called the *F*1 score are used to measure the performance of each model. Each mode has an associated database and threshold that can be used to classify data instances in the validation set. In an anomaly detection task, positive refers to abnormal, while negative refers to normal. True positive and true negative are correctly classified data instances. On the other hand, false positive and false negative are misclassified data instances.

Table 3 Confusion matrix

		Predicted	
		Positive	*Negative*
Actual	*Positive*	True positive, *TP*	False negative, *FN*
	Negative	False positive, *FP*	True negative, *TN*

From the confusion matrix in Table 3, two performance metrics can be derived, namely ‘precision’ in Equation 3 and ‘recall’ in Equation 4.

$$\text{Precision} = \frac{TP}{TP+FP} \times 100\% \tag{3}$$

$$\text{Recall} = \frac{TP}{TP+FN} \times 100\% \tag{4}$$

Moreover, the *F*1 score in Equation 5 is further derived using precision and recall where the *F*1 score is a balanced measure of precision and recall. The *F*1 score for each of the models is calculated, and the model with the highest *F*1 score is selected as the best.

$$F1\,\text{score} = \frac{2 \cdot \text{precision} \cdot \text{recall}}{\text{precision+recall}} \tag{5}$$

2.4. Model Evaluation

Model evaluation evaluates the performance of the selected model on an unseen test set using the performance metric 'accuracy.' Accuracy measures the number of correctly classified data instances over the total number of data instances. The accuracy can be computed using Equation 6.

$$\text{Accuracy} = \frac{TP+TN}{TP+TN+FP+FN} \times 100\% \tag{6}$$

3. RESULTS AND DISCUSSION

In this section, the results of the experiment are discussed. Subsection 3.1 discusses the suitability of the score metric as given in Equation 1. Subsection 3.2 presents the effects of the training set size on the performance of the model. Subsection 3.3 discusses the threshold selection and overall best model. In subsection 3.4, the performance of the proposed method is compared to the state-of-the-art method using HMM.

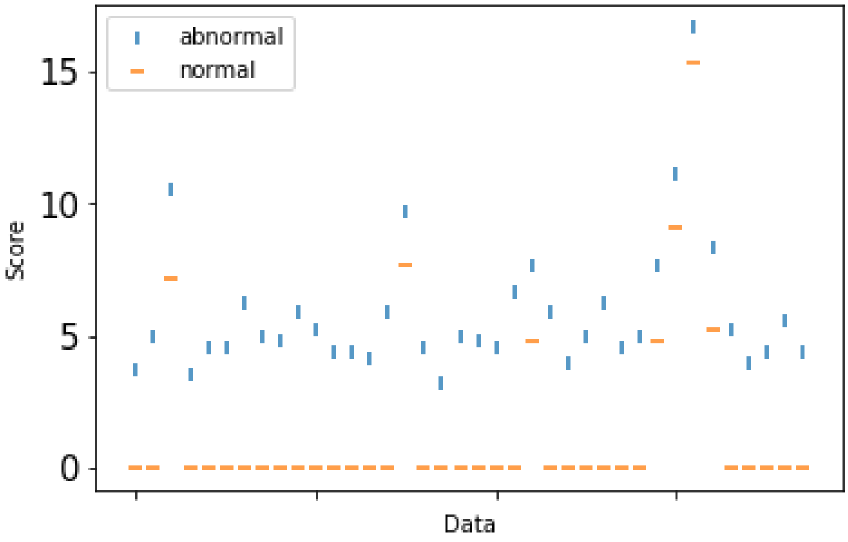

Figure 5 Scores of the data instance in the validation set

3.1. Suitability of Score Metric

Figure 5 visualizes the scores calculated for the data instances in the validation set using Equation 1. Visually, the scores generally formed two clusters. The majority of the scores of normal data instances clustered around 0%, whereas the abnormal scores clustered around 5%. There is a considerable distance between the cluster of normal data instances scores and the cluster of abnormal data instances scores. Therefore, it is possible to use a threshold value to differentiate normal and abnormal data instances based on these scores. Out of 76 normal data instances, only seven normal data instances have very high scores (≥ 5%) compared to the rest of the normal data instances. Only these seven normal data instances will be classified wrongly as abnormal. All in all, the score metric is effective in distinguishing normal data instances from abnormal data instances and can be used for the classification of a data instance as normal or anomalous.

3.2. Effects of Training Set Size

In this experiment, three different training set sizes, including 30, 60 and 113, were used. The data sizes of 30 and 60 correspond to one month and two months of data, respectively. Compared to a machine learning dataset, the largest training set size (113) is relatively small. This is because the model takes in an activity sequence for a whole day as a single data instance. Therefore, collecting a large dataset for this problem would take a very long time. The training set size of 113, which is almost four months of data, is reasonably large.

Table 5 *F*1 score of the best model for each training set size

Training set size	*F*1 score
30	79.17%
60	82.61%
113	88.37%

Ten threshold choices were sampled for each category of training set size, resulting in 30 models with varying databases and thresholds. In Table 5, the *F*1 score of the best model for each category of the training set size is listed. The results in Table 5 shows that the *F*1 score of a model increases with the size of the training set used to build the model. The *F*1 score of the database built using 113 data instances is 88.37%. This evidenced that the performance of the anomaly detection method improves as the training set used to build the database increases.

3.3. Overall Best Model

Ten threshold choices were linearly sampled from the minimum and maximum of the scores computed for every data instance in the validation set. Figure 6 shows the *F*1 score calculated for each of the 10 different models with varying threshold choices using a database trained with a training set size of 113.

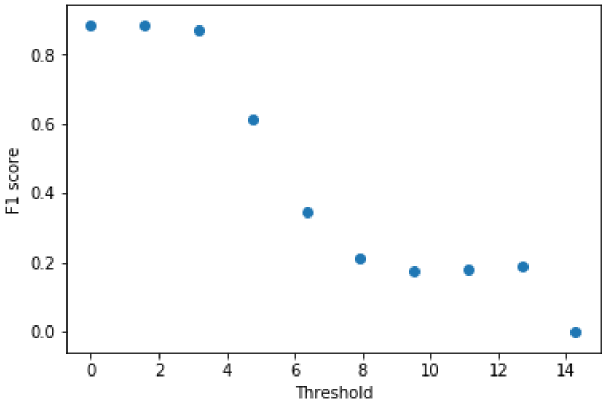

Figure 6 *F*1 scores of 10 models with varying thresholds

The plot shows that the *F*1 score decreases as the threshold increases. The model with a threshold choice of 0% has the highest *F*1 score of 88.37% and was selected as the best model.

Table 6 lists the performance metrics of the selected model. The model has a test accuracy of 90.79% with a 3.95% difference from the validation accuracy of 86.64%. It shows that the performance of the model can generalize well to unseen testing data.

Table 6 Performance metrics of the overall best model

	Validation	Test
Precision	79.17%	84.44%
Recall	100%	100%
Accuracy	86.84%	90.79%
*F*1 score	88.37%	91.57%

3.4. Comparison with HMM

HMM is the state-of-the-art for detecting anomalies based on a sequence pattern. Forkan et al. (2015) used an artificially generated dataset instead of a real dataset to experiment and evaluate the method. However, in this paper, a real dataset, Aruba, was used to evaluate the performance of the model. We tried to replicate Forkan et al.'s method as accurately as possible with a Python library for HMM called *hmmlearn*. The maximum number of training iterations is set to 200. The *F*1 scores of HMM with varying numbers of states clustered around 67.5% and 70%, whereas the test accuracy of different HMM models are around 55%. The proposed anomaly detection method using the database performs better than HMM.

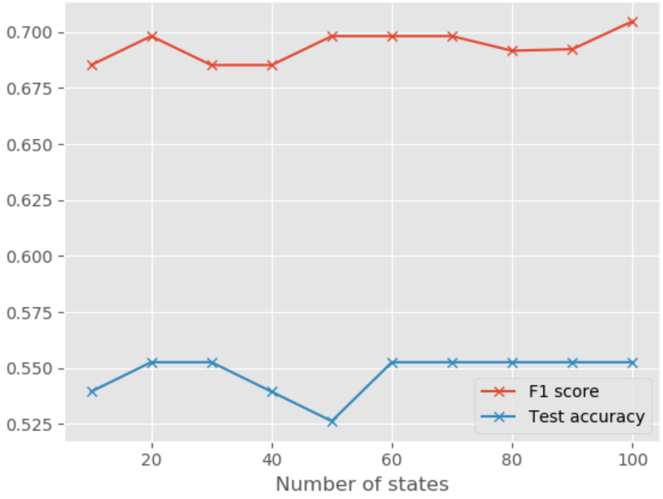

Figure 7 The *F*1 score and test accuracy of HMM with a varying number of states

4. CONCLUSION

This paper has presented a method for detecting anomaly in a person's routine based on his/her usual daily home activities pattern. The experiment was carried out using a public dataset, and the results demonstrated that the method performs well in terms of precision, recall and accuracy. However, there are a few areas that can be investigated further in the future, including: (1) The size of the sliding window. Current sliding window size is set as 3 but the efficiency of the sliding window size could be further studied; and (2) Dependence on the training set size. Hopefully, the method can be enhanced further in the future to reduce the anomaly detection method's reliance on the training set size.

5. ACKNOWLEDGEMENT

This work was funded by a Telekom Malaysia Research & Development (TM R&D) Grant in 2018. The dataset was obtained from the website of CASAS, Washington State University.

6. REFERENCES

Bux, A., Angelov, P., Habib, Z., 2016. Vision Based Human Activity Recognition: A Review. *Advances in Computational Intelligence Systems*, Volume 153, pp. 341–371

Cook, D., 2010. Learning Setting-generalized Activity Models for Smart Spaces. *IEEE Intelligence Systems*, Volume 27(1), pp. 32–38

Damla, A., Bouchachia, A., 2017. Activity Recognition and Abnormal Behavior Detection using Recurrent Neural Networks. *Procedia Computer Science*, Volume 110, pp. 86–93

Dwiyantoro, A., Nugraha, I., Choi, D., 2016. A Simple Hierarchical Activity Recognition System using a Gravity Sensor and Accelerometer on a Smartphone. *International Journal of Technology,* Volume 7(5), pp. 831–839

Forkan, A., Khalil, I., Tari, Z., Foufou, S., Bouras, A., 2015. A Context-aware Approach for Long-term Behavioural Change Detection and Abnormality Prediction in Ambient Assisted Living. *Pattern Recognition*, Volume 48(3), pp. 628–641

Hoque, E., Dickerson, R., Preum, S., Hanson, M., Barth, A., Stankovic, J., 2015. Holmes: A Comprehensive Anomaly Detection System for Daily In-home Activities. *In*: 2015 International Conference on Distributed Computing in Sensor Systems, IEEE, Fortaleza, Brazil, pp. 40–51

Lara, O., Labrador, M., 2012. A Survey on Human Activity Recognition using Wearable Sensors. *IEEE Communications Surveys & Tutorials*, Volume 15(3), pp. 1192–1209

Mahidin, M., 2018. Selected Demographic Indicators. Retrieved from Department of Statistics Malaysia. Available Online at: https://www.dosm.gov.my/v1/index.php?r=column/pdfPrev&id=RmsrQVZMVEh1SDR3Yng0cFRXNkxPdz09, Accessed on 12 December, 2018

Riboni, D., Bettini, C., Civitarese, G., Janjua, Z.H., Helaoui, R., 2016. SmartFABER: Recognizing Fine-grained Abnormal Behaviours for Early Detection of Mild Cognitive Impairment. *Artificial Intelligence in Medicine*, Volume 67, pp. 57–74

World Health Organization, 2018. Health Situation and Trend Assessment. Retrieved from World Health Organization Regional Office for Southeast Asia. Available Online at http://www.searo.who.int/entity/health_situation_trends/data/chi/elderly-population/en/, Accessed on 12 December, 2018

Zainuddin, M., Sulaiman, M., Mustapha, N., Perumal, T., Mohamed, R., 2017. Recognizing Complex Human Activities using Hybrid Feature Selections based on an Accelerometer Sensor. *International Journal of Technology*, Volume 8(5), pp. 968–978

Zhao, T., Ni, H., Zhou, X., Qiang, L., Zhang, D., Yu, Z., 2014. Detecting Abnormal Patterns of Daily Activities for the Elderly Living Alone. *In*: Health Information Science 2014, Lecture Notes in Computer Science, Volume 8423, Springer, Cham, pp. 95–108