Feature Encoding by Location-Enhanced Word2Vec Embedding for Human Activity Recognition in Smart Homes

Junhao Zhao[1], Basem Suleiman[1,2(✉)], and Muhammad Johan Alibasa[3]

[1] School of Computer Science, University of Sydney, Sydney, Australia
jzha5185@uni.sydney.edu.au, basem.suleiman@sydney.edu.au
[2] School of Computer Science and Engineering, University of New South Wales, Sydney, Australia
[3] School of Computing, Telkom University, Bandung, Indonesia
alibasa@telkomuniversity.ac.id

Abstract. Human Activity Recognition (HAR) in Smart Homes (SH) is the basis of providing automatic and comfortable living experience for occupants, especially for the elderly. Vision-based approaches could violate occupants' privacy and wearable sensors based approaches could be intrusive with their daily activities. In this study, we proposed an NLP-based feature encoding for HAR in smart homes by using the Word2Vec word embedding model and incorporating location information of occupants. We used the NLP approach to generate semantic and automatic features directly from the raw data that significantly reduced the workload of feature encoding. The results showed that both Word2Vec embedding and location-enhanced sequences can significantly improve the classification performance. Our best model which used both Word2Vec embedding and location-enhanced sequences achieved an accuracy of 81% and a weighted average F1 score of 77% on the test data with Sensor Event Windows (SEW) size of 25. This size is considered as a small SEW size which can be applied better to real-time classification due to the short latency.

Keywords: Human Activity Recognition · Smart Home · IoT · NLP · Feature Encoding

1 Introduction

With the development of the economy, the progress of medical care and the improvement of people's living standards, people's life expectancy is also getting longer than before. The World Health Organisation (WHO) reported that the elderly population (aged over 60) in the world is about to reach 2 billion by 2050 [19]. Although the smart home technology [16] is particularly attractive to young people, in recent years, an important application area of smart home is

S. Longfei and P. Bodhi (Eds.): MobiQuitous 2022, LNICST 492, pp. 191–202, 2023.
https://doi.org/10.1007/978-3-031-34776-4_11

to provide convenience for the life of the elderly such as health monitoring or ambient assisted living (AAL) [13]. The basis of such applications is so-called "Human Activity Recognition" (HAR) which requires smart home environments having the ability to represent context and characteristics of human activity.

There are two main categories of the system for HAR: vision-based and sensor-based. The sensor-based approach can further be divided into wearable sensors and ambient sensors. Vision-based systems use cameras to recognise human activity and environmental changes. However, there is generally some controversy surrounding this approach over privacy issues, especially "home" is considered a private place [3,11] Sensor-based systems solve this problem to some extent, because sensor data does not directly expose a person's life behaviour. Wearable HAR systems require users to wear smart devices like smart bands/watches with inertial measurement units to capture signals that are generated from different axes. Nevertheless, the use of body-worn devices may be uncomfortable and interfering with daily behaviour [2].

The ultimate goal of the HAR research field is to continuously improve recognition performance to the peak. Several studies [2,8,11] used different methods to incorporate features about occupants' location. By comparison, it can be concluded that after adding location information, the performance of the HAR recognition algorithm can be improved. Bouchabou et al. [4] was the first one to use word embedding for HAR. Due to the limitation of the dataset they used, it is impossible to capture the location features.

In summary, the contributions of this paper are:

1. improving activity recognition performance by using word embedding model with [13] as the baseline;
2. using NLP methods to further enhance the classification performance by introducing location information of user activities as inspired from the past studies [6,16,18];
3. using the combination of word embedding model and location-enhanced event generation method to achieve a model which uses SEW (sensor event window) size of 25 (a small SEW size);
4. processing the raw data directly to save effort on pre-processing data and feature selection compared to traditional methods.

In this study, we use the dataset that is obtained from ambient PIR motion sensors, door/temperature sensors and light switch sensors that are installed in a smart home environment. Such smart home environment architecture will make occupants feel "seamless" and unobtrusive. Compared with vision-based method, sensor-based methods do not enable monitoring of the actual activities of occupants, and these activities are out of the defined scope.

2 Related Works

To recognise human Activities of Daily Living (ADLs), traditional machine learning algorithms are used in the past studies. Avgoustinos et al. [2] compared the activity recognition performance among KNN, RF, LR and SVM. They used wrist-worn smart devices to capture 3-axis accelerometer data when participants

perform different activities and further used BLE beacons to track participants' location. Their experiments showed that SVM outperformed other classifiers and was the most bootstrapped classifier by using beacon data. However, data collected from accelerometers were only applied to recognise ambulatory movements like running, walking, and failing. Diane J. Cook and Parisa Rashidi [6] experimented with SVM, HMM, CRF and NB on 3 different CASAS datasets and the SVM had the best performance across the 3 datasets (91.52% average accuracy across 3 datasets.). They further proposed the Activity Discovery (AD) algorithm to discover patterns of different activities which can assist Activity Recognition (AR) algorithms achieve better performance because the activity labels are not always annotated correctly in the raw dataset.

The traditional HAR always employs handcrafted feature extraction methods that require lots of pre-processing steps (feature selection, validation, etc.) and domain expert knowledge while deep learning methods can enable automatic feature extraction which is much more efficient. Munkhjargal Gochoo et al. [7] processed the raw data directly and generated an "Activity Image" for each sequence of sensor events and used 2D Convolutional Neural Networks (CNNs) for HAR whose best model achieved 0.951 F1 score for the eight activities. As an extension, Gochoo et al. [17] proposed a RGB activity image conversion method which achieved 95.2% accuracy on the sensor-based dataset. Their method mapped sensor events to the corresponding coordinates on the activity image as a pixel whose colour is dependent on the time of the event and the first two lines of the activity image refer to the two previous activities.

Enabling the NLP idea for HAR is a novel attempt in this area. Bouchabou et al. [4] used frequency-based encoding to encode sequences of events for automatic feature extraction and further used embedding layers in the Fully Convolutional Network (FCN) and LSTM model for gaining context knowledge. They used the Aruba dataset and treated each sensor event as a word so that it formed a "sentence" in each sliding window. Their experiments illustrated that gaining context knowledge can improve model performance and LSTM outperformed FCN. For further improvement, we can use embedding algorithms like Word2Vec for word embedding and sequence encoding because word embedding methods take into account the context of words in each sequence of events. In addition, as we discussed before, there are some studies that tried to bootstrap HAR algorithms by incorporating location information. Therefore, we can incorporate location "word" in each sequence of events for further improvement.

3 Proposed Methods

The problem is a classification task which classifies human activity in a smart home environment. There are k sensors $S = s_1, s_2, ..., s_k$ in the dataset that generate events $e_i \in E$. An event records the date, time, sensor id, room location, more detailed location, sensor value, sensor type and activity label when the sensor is activated. In this study, we focus on the sensor id, room location and sensor value so for each event we have $e_i = (s_i, L_i, v_i)$where s_i is the sensor id,

L_i is the sensor location and v_i is sensor value. We use Sensor Event Windows (SEW) to segment the time series data so that each sequence of events will have a fixed length which equals the SEW size. Each sequence of events $Seq_i = (e_i, ..., e_k)$ is associated with one activity instance $a_i \in A$. The two timestamp indicators date and time are not taken into account because we consider that different occupants may have different habits for the same activity and the same activity may happen at any time during the day.

3.1 Data Pre-processing

In the raw dataset, each row represents one event. SEW segments the data into intervals with the same number of sensor events. Quigley et al. [14] illustrates that SEW is the second best sliding window approach and it can classify more activities than the Time Window which segments the data into intervals with the same time duration. The reason for using SEW is that we would like to use CNN which requires a fixed input size. By contrast, TW can generate different lengths for windows that have different counts of events.

NLP has many similarities with processing sequences of sensor events. Sensor events can be processed as the "words" and sequences of events can be processed as the "sentence" in natural language. Each "sentence" describes the features for each activity label. In addition, events in each sequence also have the contextualised relationship which also shows the parallel between NLP and sensor events processing. For each sensor event, we encode them by two approaches. The first approach is extracting the sensor ID and sensor value and combining them to be a "word". The second approach which is location bootstrapping, we further concatenate the sensor location into the word. For the location bootstrapping approach, the location information is recorded every other event. Each word is indexed when the vocabulary is being generated so that each word has an index (Fig. 1).

After index sequences are generated, SEW is used to generate sliding windows. In the literature, a SEW size 20–30 is always selected [1,3]. [4] experimented with even larger SEW sizes (50,75...) and their results showed that larger SEW sizes can improve the model's performance. Therefore, we select SEW sizes of 25, 50 and 75 to experiment. There are two different encoding approaches following the sliding windows generation step. For the frequency-based encoding approach, the length of each sequence is dependent on the size of the vocabulary. For the Word2Vec embedding approach, we need to pad the sequences whose length is less than the defined size of the SEW window to make sure they have the same length.

3.2 Our Model

Word2Vec can learn the similarities between words and capture a sense of word in the training corpus. There are 2 different training approaches for it. One is Continuous Bag of Words (CBOW) which predicts centre words from context words while another one is Skip-gram which predicts context words by a

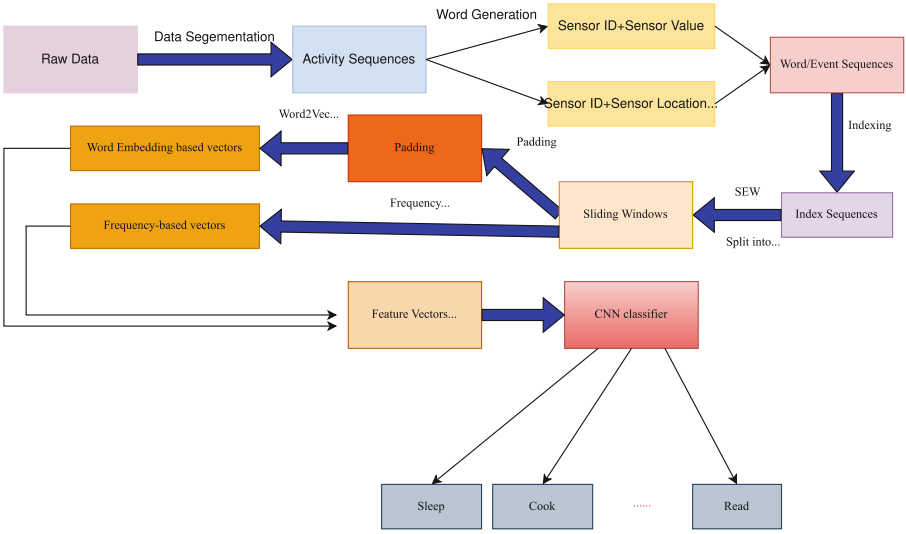


Fig. 1. Framework of the proposed method

given centre word. After generating the sequences of events and tokenizing each "word", the input for the Word2Vec model is ready. Skip-gram is selected for training the Word2Vec model because it works well with small datasets and can better represent less frequent words [12]. The word representation will be the inputs for CNN classifiers.

There could be some sequences with fewer events than the window size we decide. To make sure the length of every input stream is the same, value 0 is used for sequence padding. Our classifier is based on the Convolutional Neural Networks for sentence classification [10] and slightly modified during the experiment. The first layer of the CNN architecture is an Embedding layer which is used to extract the embedding matrix for mapping each word in the input sequence on its embedding vector. There are 3 1D convolutional layers. Each convolutional layer has a rectified linear unit activation which can help make the convergence faster. After the feature extraction, there is a Global Max Pooling layer followed by a flatten layer. Global Max Pooling (GMP) layer is frequently used for text classification tasks. One advantage of GMP is it can be used to reduce the dimensionality of feature maps and even replace Flattening or Dense layers [5]. Another advantage is there are not any parameters to be tuned for GMP. Before the output dense layer, there is one Batch Normalisation [9] layer and one dropout layer. Batch Normalisation can tackle the internal covariate shift problem so that it has a regularising effect. Dropout layer also solves the overfitting problems by randomly setting the outgoing edges of hidden units to 0 at each update of the training phase. The last layer is the output layer which is activated by softmax to perform the final classification.

3.3 Evaluation Method

A study [15] suggests that 80%/20% train/test split ratio can provide the best training performance and reveal the model performance. Therefore, to evaluate the performance of the classifier, the raw data is split into two parts where 80% for training purposes and 20% for testing. The random shuffle is used to prevent non-random assignment to train and test set so that the generalisation of the model can be improved. We adopt the following measurement metrics: recall, precision, F1-score and accuracy. To avoid the effect of unbalanced labels, we will also look at the weighted metrics. During the training phase, the early stop method integrated in the keras framework is adopted to avoid overfitting. The model's accuracy on the test data provides evidence for this method. Once the model's performance on the test dataset does not improve after n (n=10 in our experiment) epochs since the last, the training will be stopped.

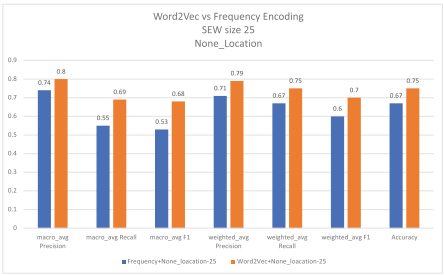
4 Experiment Results

4.1 Comparison Between Frequency Encoding and Word2Vec Embedding Encoding

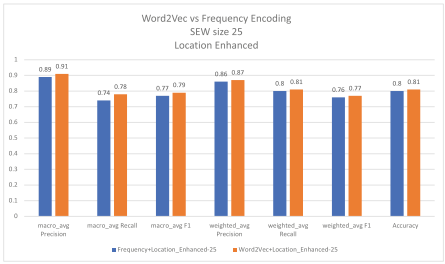
To validate the effect of Word2Vec embedding on the classifier, we need to control the SEW sizes and the approach of generating event sequences (Location-Enhanced vs None-Location). Based on Fig. 2a, 2c, 2e, it can be concluded that Word2Vec embedding can significantly improve the model's performance when we use the event sequences without location information. The two most affected indicators are macro average recall and macro average F1, which both have an increase equal to or more than 0.1 after using the Word2Vec word embedding model. To avoid the effect of imbalance labels on the evaluation metrics, the weighted average metrics should be focused on. Compared to the model using frequency encoding, the weighted average F1 scores increase by 0.1, 0.08 and 0.05 with SEW size 25, 50 and 75 respectively. As the SEW size increases, the gain brought by the Word2Vec model on the F1 score indicator decreases but the improvement was still significant. In terms of the accuracy, Word2Vec embedding can improve the accuracy by 0.08, 0.07, and 0.05 for SEW size 25, 50 and 75 respectively. Based on Fig. 2b, 2d, 2f, the 3 models that adopt the event sequences with location information, it can be concluded that Word2Vec embedding can still improve the model's performance. However, the magnitude of improvement is much smaller than that of the method which does not use location information. With regard to the accuracy, the Word2Vec embedding increases the accuracy by 0.01 for all the 3 models with different SEW sizes.

4.2 Comparison Between None-Location Models and Location-Enhanced Models

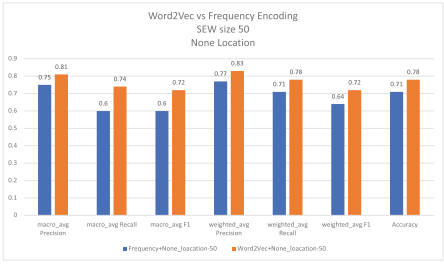
To validate the effect of Word2Vec embedding, we need to control the SEW size and the approach to encode the event sequences (Frequency encoding and



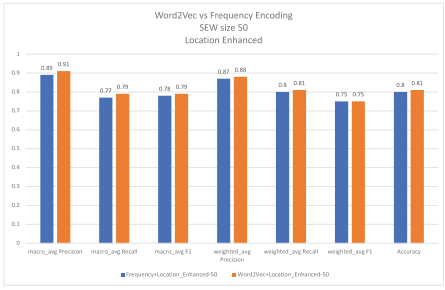
(a) SEW 25, None-location sequences



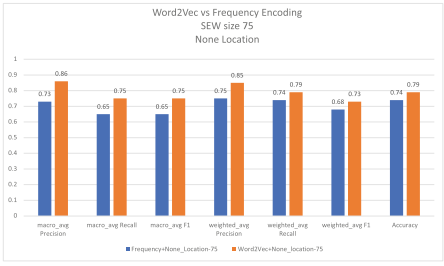
(b) SEW 25, Location-enhanced sequences



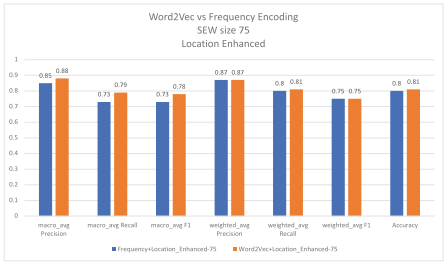
(c) SEW 50, None-location sequences



(d) SEW 50, Location-enhanced sequences



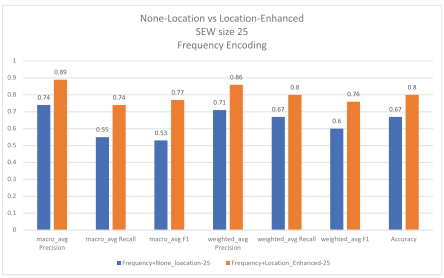
(e) SEW 75, None-location sequences



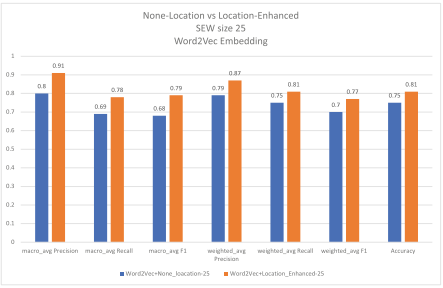
(f) SEW 75, Location-enhanced sequences

Fig. 2. Evaluation metrics comparison between Word2Vec approach and Frequency Encoding approach

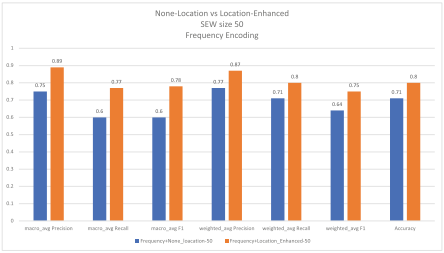
Word2Vec embedding). Based on Fig. 3a, 3c, 3e, it can be concluded that using location information during the event sequence generating phase can significantly improve model's performance. Under the circumstance of frequency encoding for event sequences, location information can boost the weighted F1 score by 0.16, 0.11 and 0.07 for models with SEW size 25, 50 and 75 respectively. For the models with SEW size 25 and 50, most of the macro average metrics can achieve the improvement over 0.15. The magnitude of the improvement of the model with SEW size 75 is smaller than the other 2 models but the effect of the boost brought by location information is still significant. The accuracy is improved by 0.13, 0.09 and 0.06 for the models with SEW 25, 50 and 75 respectively with the bootstrap of location information. Based on Fig. 3b, 3d, 3f, the conclusion



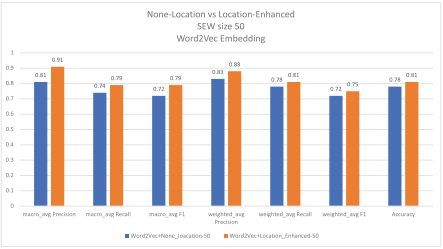
(a) SEW 25, Frequency Encoding



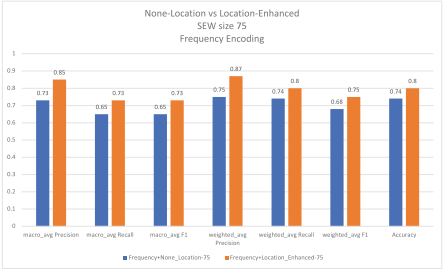
(b) SEW 25, Word2Vec Embedding



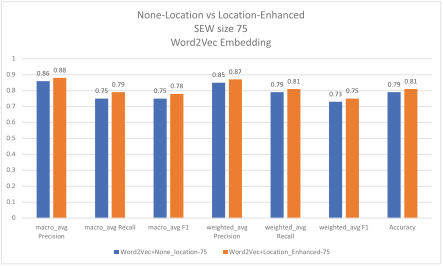
(c) SEW 50, Frequency Encoding



(d) SEW 50, Word2Vec Embedding



(e) SEW 75, Frequency Encoding



(f) SEW 75, Word2Vec Embedding

Fig. 3. Evaluation metrics comparison between None-Location approach and Location-Enhanced approach

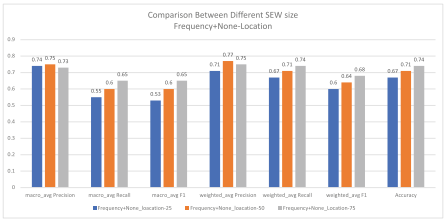
is that under the circumstance of using Word2Vec embedding to encode event sequences, the location information can still significantly improve the model's classification performance. Compared with the None-Location aware model, the weighted average F1 score increases by 0.07, 0.03 and 0.02 for the models with SEW size 25, 50 and 75 respectively. Compared to the weighted average metrics, the magnitude of the improvement on macro average metrics is more significant. For all the three conditions, the accuracy is improved by 0.06, 0.03 and 0.02 with the help of location information for the models with SEW size 25, 50 and 75 respectively.

4.3 Comparison Between Different SEW Size

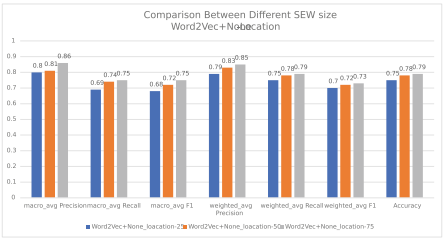
Based on Fig. 4a and 4b, the general trend is that with the increase of SEW, the model shows an upward trend in most metrics except the precision (Fig. 4a). Compared with the models with SEW size 25 and 50, the macro average precision of the model with SEW size 75 is reduced, but the magnitude is small. The weighted average precision also decreases for the model with SEW size 75 when it is compared with the model with SEW size 50. It can be concluded that increasing the SEW size can improve the overall performance for None-Location aware models, regardless of whether the model uses Word2Vec embeddings or frequency encoding for event sequences. However, a large SEW size could sometimes have a bad effect on some metrics. Figure 4c and 4d illustrate that, if we use Location-Enhanced method to generate event sequences, increasing the size of SEW cannot help to improve the classification performance. It can even decrease the performance, especially for the approach of using frequency encoding to encode sequences. Compared with SEW size 25 and 50, under the premise that the weighted average precision and recall are almost unchanged, the model with SEW size 75 significantly decreases the macro average precision and recall. The conclusion is that, once we use the Location-Enhanced approach for event sequence generation, small SEW size can also achieve good results and larger SEW size could sometimes have negative effect on the classifier performance.

4.4 Discussion

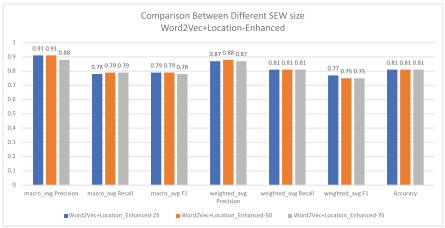
Based on the experiment result, both Word2Vec and location information model can increase the classifier's performance significantly. Compared to the model using frequency encoding to encode each event sequence, the model using Word2Vec can gain more semantics from context. Incorporating location information is another way to gain more semantics. For example, if an occupant triggers an environmental sensor installed in the kitchen, then we can at least determine that he/she must not be bathing, and other activities which certainly do not occur in the kitchen. Location information cannot help to distinguish those activities which occur in the same room but is beneficial for the classifier to distinguish those activities that happen in different rooms. The comparison between different SEW sizes shows that increasing the size of SEW can significantly increase the model performance for the models using the None-Location approach regardless of the encoding approach. Small SEW size is beneficial for real-time activity recognition because the delay is shorter. The analysis in the previous subsection illustrates that using larger SEW size has the potential to make the model unstable during the training phase.



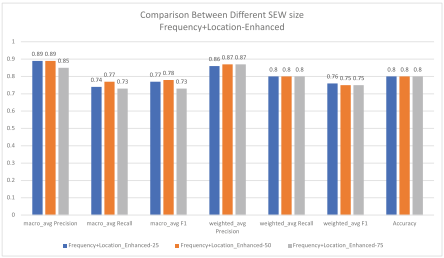
(a) Frequency+None-Location



(b) Word2Vec+None-Location



(c) Word2Vec+Location-Enhanced



(d) Frequency+Location-Enhanced

Fig. 4. Evaluation metrics comparison between different SEW sizes with a specific condition

5 Conclusion and Future Work

Our study extends the literature by using the Word2Vec embedding model for feature generation and location information to further bootstrap the classifier. Our results show that the best model can achieve a weighted average F1 score of 0.77 and an accuracy of 0.81 on the test dataset (16 predefined activity labels). Both Word2Vec embedding and Location-Enhanced approach can significantly improve the classification performance. The best model uses the Location-Enhanced approach to generate the event sequences and Word2Vec embedding for feature encoding with SEW size 25. It shows that even the small SEW size can achieve the best performance by using these two approaches together which is good for real-time activity recognition due to the shorter delay.

Vocabulary size and single representation for each word are the limitations of Word2Vec model. In addition, the comparison of different parameter combinations for Word2Vec has not been conducted. In the future, we can examine the effectiveness of contextualised embedding models such as ELMo and methods such as byte pair encoding (BPE) or WordPiece to split words into subwords to tackle the limitations of Word2Vec. Tuning the parameters of the Word2Vec model could also be one direction for performance improvement. The accuracy of the data recorded by the sensors also needs to be further verified. In the future, more detailed positions of sensors could be recorded during the collection process of sensor data which could help the classifier improve the ability to classify these activities that occur in the same room. Due to the time limitation, comparison with the state-of-the-art will also be conducted in the furture.

References

1. Aminikhanghahi, S., Cook, D.J.: Enhancing activity recognition using CPD-based activity segmentation. Pervasive Mob. Comput. **53**, 75–89 (2019)
2. Avgoustinos, F., William, O., Babak, T., George, L.: Location-enhanced activity recognition in indoor environments using off the shelf smart watch technology and BLE beacons. Sensors **17**(6), 1230 (2017)
3. Bouchabou, D., Nguyen, S.M., Lohr, C., Leduc, B., Kanellos, I.: A survey of human activity recognition in smart homes based on IoT sensors algorithms: taxonomies, challenges, and opportunities with deep learning. Sensors **21**(18), 6037 (2021)
4. Bouchabou, D., Nguyen, S.M., Lohr, C., LeDuc, B., Kanellos, I.: Fully convolutional network bootstrapped by word encoding and embedding for activity recognition in smart homes. In: Li, X., Wu, M., Chen, Z., Zhang, L. (eds.) DL-HAR 2021. CCIS, vol. 1370, pp. 111–125. Springer, Singapore (2021). https://doi.org/10.1007/978-981-16-0575-8_9
5. Christlein, V., Spranger, L., Seuret, M., Nicolaou, A., Maier, A.: Deep generalized max pooling. In: 2019 International Conference on Document Analysis and Recognition (ICDAR) (2019)
6. Cook, D.J., Krishnan, N.C., Rashidi, P.: Activity discovery and activity recognition: a new partnership. IEEE Trans. Syst. Man Cybern. Part B Cybern. Publ. IEEE Syst. Man Cybern. Soc. **43**(3), 820–828 (2013)
7. Gochoo, M., Tan, T.H., Liu, S.H., Jean, F.R., Alnajjar, F., Huang, S.C.: Unobtrusive activity recognition of elderly people living alone using anonymous binary sensors and DCNN. IEEE J. Biomed. Health Inf. **23**(2), 693–702 (2018)
8. Hardegger, M., Roggen, D., Calatroni, A., Troester, G.: S-smart: A unified bayesian framework for simultaneous semantic mapping, activity recognition, and tracking. ACM Trans. Intell. Syst. Technol. **7**(3), 1–28 (2016)
9. Ioffe, S., Szegedy, C.: Batch normalization: Accelerating deep network training by reducing internal covariate shift. In: International Conference on Machine Learning, pp. 448–456. PMLR (2015)
10. Kim, Y.: Convolutional neural networks for sentence classification. Eprint Arxiv (2014)
11. Lu, C.H., Fu, L.C.: Robust location-aware activity recognition using wireless sensor network in an attentive home. IEEE Trans. Autom. Sci. Eng. **6**(4), 598–609 (2009)
12. Mikolov, T., Chen, K., Corrado, G., Dean, J.: Efficient estimation of word representations in vector space. In: Bengio, Y., LeCun, Y. (eds.) 1st International Conference on Learning Representations (ICLR 2013), Scottsdale, Arizona, USA, 2–4 May 2013, Workshop Track Proceedings, pp. 1–12 (2013). http://arxiv.org/abs/1301.3781
13. Ni, Q., García Hernando, A., Pau, I.: The elderly's independent living in smart homes: a characterization of activities and sensing infrastructure survey to facilitate services development. Sensors **15**, 11312–11362 (2015). https://doi.org/10.3390/s150511312
14. Quigley, B., Donnelly, M., Moore, G., Galway, L.: A comparative analysis of windowing approaches in dense sensing environments. In: Proceedings, vol. 2, no. 19 (2018)
15. Rácz, A., Bajusz, D., Héberger, K.: Effect of dataset size and train/test split ratios in GSAR/GSPR multiclass classification. Molecules **26**(4), 1111 (2021)
16. Satpathy, L.: Smart housing: technology to aid aging in place-new opportunities and challenges (2006). https://scholarsjunction.msstate.edu/cgi/viewcontent.cgi?article=4966

17. Tan, T.-H.: Multi-resident activity recognition in a smart home using RGB activity image and DCNN. IEEE Sens. J. **18**(23), 9718–9727 (2018)
18. Tran, S.N., Zhang, Q., Smallbon, V., Karunanithi, M.: Multi-resident activity monitoring in smart homes: a case study. In: 2018 IEEE International Conference on Pervasive Computing and Communications Workshops (PerCom Workshops) (2018)
19. WHO: 10 facts on ageing and health (2017). https://www.who.int/news-room/fact-sheets/detail/10-facts-on-ageing-and-health