AR-T: Temporal Relation Embedded Transformer for the Real World Activity Recognition

Hyunju Kim(✉) and Dongman

Korea Advanced Institute of Science and Technology, Daejeon, Korea
iplay93@kaist.ac.kr, dlee@kaist.ac.kr

Abstract. Activity recognition is a fundamental way to support context-aware services for users in smart spaces. Data sources such as video or wearable devices are used in many recognition approaches, but there are challenges in utilizing them in the real world. Recent approaches propose deep learning-based methods on IoT sensor data streams to overcome the issues. Since they only describe single user-based spaces, they are vulnerable to complex sequences of events triggered by multiple users. When multiple users exist in a space, various overlapping events occur with longer correlations than a single user situation. Additionally, ambient sensor-based events appear far more than actuator-based events, making it difficult to extract actuator-based events as important features. We propose a transformer-based approach to derive long-term event correlations and important events as elements of activity patterns. We also develop a duration incorporated embedding method to differentiate between the same type but different duration events and add a sequential manner to the transformer approach. In the experiments section, we prove that our approach outperforms the existing approaches based on real datasets.

Keywords: Activity recognition · Transformer · Temporal relation embedding · Multi-user smart spaces · Sensor data streams

1 Introduction

With the emergence of IoT (Internet of Things) technologies, smart objects around us generate their states with intelligence and turn our daily space into a smart space. One of key drivers for this is activity recognition that infers users' intentions to support their efficient and effective lives [3,25]. Various sources such as video or wearable devices are widely used to recognize activities [10,16] while they have limitations such as privacy issues or difficulties in collecting data reliably [2,9,24]. To avoid such issues, ambient sensing stream data such as user movement, sound or lighting from smart spaces is used.

Recent approaches [21–23,34,37,39] propose deep learning-based recognition models for inferring activities without hand-crafted feature engineering for each environment. In the activity recognition domain, major approaches are based

T. Hara and H. Yamaguchi (Eds.): MobiQuitous 2021, LNICST 419, pp. 617–633, 2022.
https://doi.org/10.1007/978-3-030-94822-1_40

on RNN-type structures that have abilities to find causal relations of context sequences in smart spaces. The *context* implies a value that abstractly expresses the situation that occurred over a certain period of time in the sensor stream data. Murad et al. [21] use LSTMs based model to capture long-term dependencies between contexts and Zhao et al. [39] leverage bidirectional LSTMs to tighten correlations overall context sequences by adding reverse temporal relations into the model. To extract major context sequence patterns of each activity in both spatial and temporal aspect, Ordóñez et al. [23] develop a hybrid deep learning model which combines both CNNs and LSTMs.

Despite of their outstanding outcomes, they have limitations in their application to a growing research field, a space where multiple users work together (e.g., smart seminar room or smart factory). In this paper, we define the space as a *Multi-user space*. The previous studies focus on single-user activities in which a user's actions occur sequentially. Multiple users perform multiple actions in the same time[4] without user identification [14] and it causes difficulty to define correlations among events in sequential manners as shown in Fig. 1. The issues pose three major challenges in recognizing activities in real-world, multi-user spaces.

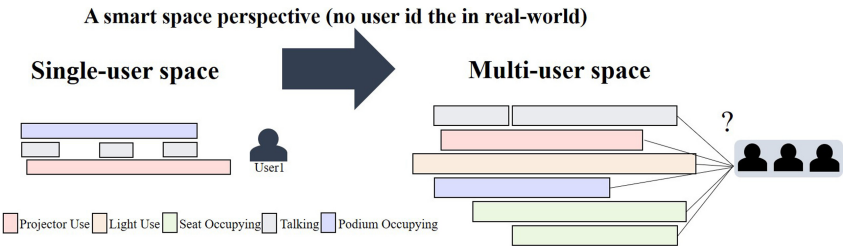

Fig. 1. Differences between single-user space and multi-user space in a smart space perspective

First, the existing models are sequence-based which lead to a gradient vanishing problem and has limitations in extracting long-term context relations. From a space perspective, the mix of events means that a significant amount of unrelated events occur consecutively between correlated events. It means that long-term correlated events may have more significant dependencies than events that occur immediately before or after relationships. In addition, multi-user activity generates relatively much longer context sequences than single-user activity, resulting in more long-term correlation events. Second, the existing studies are regarded all activities have the same importance. It makes infrequent but important events for recognizing certain activity (e.g. Stand in front of a podium in *Seminar*) are ignored when extracting activity patterns. Finally, previous research focus mainly on event types. Even if events of the same type are active in different duration, they are learned in the same contexts as they are the same event type.

In this paper, we propose a novel approach, AR-T (i.e. **A**ctivity **R**ecognition **T**ransformer), to complex contexts in multi-user spaces based on the transformer structure. AR-T employs the transformer [33] to discover long-term context correlations by computing attention scores between all context pairs. It also finds the importance weights of each context which makes infrequent but important contexts extracted as one of the activity patterns. To differentiate the same-type contexts with different duration, it embeds not only context type but also the duration and positional information as an input to the Transformer. We experiment with two datasets IoT sensor stream data which have been collected from multi-user based real-world testbeds, a seminar room testbed and a smart home testbed from CASAS group [31]. Compared to the existing approaches [17,21,39], experimental results show that AR-T improves the recognition accuracy by 12.57% in the seminar room testbed and 11.42% in the CASAS dataset.

The remaining of this paper is organized as follows. In Sect. 2, we introduce a related work of activity recognition methods focusing on deep learning and the previous studies based on the transformer. We describe key problems of the existing works in Sect. 3 and illustrates the key components of AR-T in Sect. 4 to solve those issues. In Sect. 5 and 6, we analyze the empirical evaluation results of AR-T and discuss them. We present the conclusion and future works in Sect. 7 .

2 Related Work

2.1 Existing Approaches for Activity Recognition in Smart Spaces

In order to recognize users' activities in a smart space, wearable sensors or video-based approaches are proposed [10,16]. However, those disciplines have privacy or accessibility issues to use in daily living, data streams generated by deployed smart objects (e.g. sound sensors, projectors, etc.) without user identification have been leveraged. Early approaches employ graphical models [6,35] or space-knowledge based models [5,28] that deriving semantic correlations among activities and events to represent the characteristics of complex activities, the schemes have difficulties in directly applying them to various spaces because they need to manually pre-define all sensors and probabilistic relations of activities.

Recent studies [21–23,37,39] use deep learning models for resolving the hand-crafted feature engineering problems [34]. The methods they use are classified into two categories: CNN and RNN. CNN-based approaches [22,23,37] compress complex patterns of activities to efficient representations using kernels and filters. Yang et al. [37] show that a CNN technology-based automated feature learning from raw sensor data has better performance than the traditional machine learning methods. In the activity recognition field, RNN based approaches [21,39] are mainstreams since RNN type models [7,13] are better to extract sequential patterns. Murad et al. [21] propose LSTM based recognition models for classifying activities from variable-length input sequences by capturing long dependencies between contexts. However, the LSTM model only contains forward information that loses event correlation information and Zhao et al. [39] use bidirectional LSTM to add reverse context information to overcome the issue. Ordóñez et al.

[23] propose a hybrid deep learning model which combines both CNN and LSTM layers. It is suitable for multimodal sensors in smart spaces since it trains patterns of sensor data in both spatial and temporal aspects and it outperforms the existing single models. However, these models still have limitations in extracting long-term correlations in a multi-user space and finding important but infrequent events of each activity.

2.2 Attention Mechanisms in Various Domains

In the NLP field, the seq2seq model called transformer [33] which consists of attention mechanisms becomes the mainstream, showing more effective performance for context understanding. Compared to the existing sequential deep learning models, it only utilizes attention mechanisms that reflect connections between all individual contexts in form of weights. Not only this enables finding important contexts but also results in greater parallelization. In addition, it is more appropriate for finding longer relations between contexts since it reduces the gradient vanishing problem compared to the previous techniques. Due to those advantages, many time series research domains try to recognize a specific context. Wu et al. [36] leverage the transformer for recognizing diseases like influenza and Yang et al. [38] propose a transformer-based approach for volatility recognition. A transformer is a promising method in the time-series domain, which means it is suitable for finding correlations in event streams in smart spaces, but only a small number of studies are conducted. Haresamudram et al. [12] introduce a pre-training method of the transformer to single-user smart spaces but focus less on how to deploy the transformer itself in a smart space. Ma et al. [17] utilize attention mechanisms with an RNN-type model (i.e. GRU [7]) and CNNs to capture correlations of multimodal sensor data from spatial and temporal perspectives. However, their approach is still difficult to long-term event correlations since it is RNN-based. Both approaches are single user-based environments that are vulnerable to real problems of multi-user spaces.

3 Problem Definition

This section describes issues of recognizing activities based solely on sensor streams in multi-user smart spaces.

3.1 Preliminaries

We represent a i^{th} sensor value in timestamp t as s_t^i. Each sensor generates a raw data stream $(s_1^i, s_2^i, s_3^i, ...)$. In a preprocessing step, each sensor stream is converted to an event stream denoted by $(e_1^i, e_2^i, e_3^i, ...)$.

Definition 1. *An **event** is a value that converts raw sensor values into a human-understandable value to reduce fluctuations and noises in the real world. Changed event values are generated when user **actions** (i.e. user behavior such as 'sitting') changes. Each event has its value and duration as attributes. E.g. raw sound sensor value '76' is converted to event 'Sound_Level2' and raw projector value '1' is converted to event 'Projector ON'.*

All event streams are split appropriately through a window sliding step and transformed into a sequence of contexts $(c_{w1}, c_{w2}, c_{w3}, ...)$. wj indicates that it is the jth order window. The sequence becomes one of the patterns of a certain activity A^k. k means the type of activity.

Definition 2. *A **context** c_{wj} represents a set of events $\{e^1_{wj}, e^2_{wj}, ..., e^i_{wj}, ...\}$ in the j^{th} window. It is an abstract value representing a specific situation. Similar to an event, it has a specific value and duration.*

Definition 3. *In this paper, an **activity** refers to a single user or multiple users performing a task in the same space. E.g. a 'seminar' is a task involving multiple users, but it is one activity as they perform it for the same one goal. An activity A^{k_1} consists of a sequence of contexts $(c^{k_1}_{w1}, c^{k_1}_{w2}, c^{k_1}_{w3}, ...)$.*

3.2 Problems

A multi-user activity has more complex activity patterns than a single-user activity since it involves interactions between users and various independent actions of each user [4]. In addition, in the real world, it is hard to collect and recognize identities of users, which raises problems of correlation between users and sensor data they generate [14]. In these aspects, we define four major problems that are not able to be solved by previous research.

Long-Term Correlations Between Contexts. In a multi-user environment, as shown in Fig. 1 which illustrates an activity *Seminar*, events are mixed from a space point of view since users perform actions at the same time or at overlapping times. It leads multiple, unrelated contexts to exist between pairs of related contexts, allowing them to have long-term relations. E.g. c_{w1} and c_{w10} are more relevant than c_{w1} and c_{w2}. Also, compared to a single-user activity, a multi-user activity creates a longer sequence of context. Not only does the duration of the activity increases, but changes in contexts also increase, resulting in a natural increase in a sequence length of contexts. However, the existing approaches assume that all events are performed sequentially in an activity and that a given context and its immediately preceding context are more relevant than the others. This hampers recognition performance of the existing approaches in multi-user spaces.

Unbalanced Frequencies Between Events. The number of occurrences between events in a space varies due to differences in the number of times users can change the state of smart objects. For example, when a *Seminar* activity occurs, users incur a *Light ON* event only once but make a *Sound Level2* event multiple times. The difference in the number of occurrences between events is even greater in a multi-user space . However, the existing approaches derive activity patterns based on frequency and give equal importance to all events. This makes it difficult to extract actuator-based events as elements of an pattern of specific activity, which is important for recognizing the activity.

Different Activities but Similar Context Sequences. The number of sensors installed and the number of events that can occur in one smart space is fixed. Thus, a context sequence consists of only a limited number of context types, resulting in many similar sequences are created between different activities. As the existing works do, methods only using the values of contexts are not able to distinguish them.

4 Proposed Approach: *A*ctivity *R*ecognition *T*ransformer

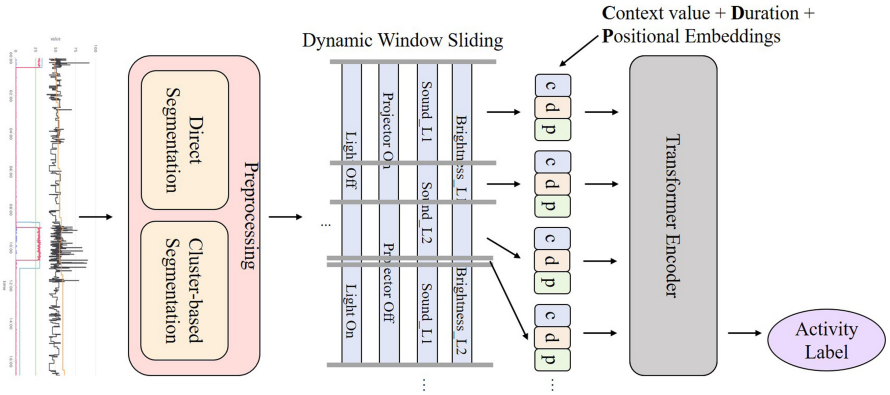

Fig. 2. Overall architecture and key components of AR-T

In this section, we details the key components of AR-T (i.e. **A**ctivity **R**ecognition **T**ransformer). Figure 2 illustrates the overall architecture of AR-T for recognizing activities solely using sensor streams generated in multi-user smart spaces. AR-T improves the recognition performance of activities through three steps: (1) AR-T transforms raw sensor values into events by direct and cluster-based segmentation. Then, AR-T obtains event streams of each sensor. (2) AR-T slides a window when any event changes in the entire event streams and converts the set of event values for each window into a single vectorized context embedding.

AR-T also constructs a duration embedding based on the actual duration length of each window and adds a positional embedding that represents the order of the window. (3) AR-T trains patterns of activities in smart spaces with the embeddings and transformer [33]. The transformer encoder of AR-T utilizes attention mechanisms to determine forward and reverse dependencies between contexts and calculate the importance weights of each context. Based on the trained attention values of each activity, AR-T infers the most probable activity in the recognition step.

4.1 Sensor Stream Preprocessing

In this step, AR-T transforms a raw sensor stream $(s_1^i, s_2^i, s_3^i, ...)$, s_t^i into an event stream $(e_1^i, e_2^i, e_3^i, ...)$. The result e_t^i consists of the duration and its event label. Sensor data generated in a smart space are classified into two types: Data from actuators (e.g. electronic lights or projectors) and data from ambient sensors (e.g. brightness or sound sensors).

Actuators are smart objects that users directly manipulate and they publish data when the state changes directly by users. Their values are discrete and AR-T translates them directly to event labels. The process of generating events based on the discrete sensor data type is called *Direct Segmentation*. On the other hand, ambient sensors publish constant observations of a space. They include a lot of noises (i.e. sensor data generated by some environmental factors rather than user action) and fluctuations that increase value changes of the sensors. AR-T reduces these problems by using a simple signal averaging [32] method. Then, it employs density-based clustering [15] to automatically find the optimal number of event levels for each ambient sensor and defines a range of sensor values for each cluster. Then, it gives each cluster a human-understandable level of event label and s_t^i that corresponds to a particular cluster is assigned the label of that cluster. This continuous sensor data type-based event generation method is called *Cluster-based Segmentation*.

When events with the same level occur consecutively, AR-T merges those events into one event. If an event with any other level appears, the merge process stops and that time point is considered the end time of the event. AR-T derives event streams from all types of sensors.

4.2 Temporal Relation Embedding

Multivariate event streams $\{(e_1^1, e_2^1, ...), ..., (e_1^i, e_2^i, ...), ...\}$ enter the embedding step as inputs. Compared to windows that are split into fixed sizes that the existing approaches do, AR-T slides a window according to changes in values of events. When e_{t-1}^i is changed to e_t^i, the point at which the event value changes, AR-T is regarded as the time when the window is split.

$\{e_{t-1}^1, e_{t-1}^2, ..., e_{t-1}^i, ...e_{t-1}^n\}$ is transformed into a single context embedding $c_{w(t-1)}$ by a popular word embedding method, Word2Vec[19]. The embedding is called *context embedding* as each embedding represents the entire contextual situation of a smart space. To differentiate situations even if types of context

embeddings are the same, AR-T embeds temporal information using duration embedding and positional embedding methods. To generate a duration embedding $d_{w(t-1)}$, AR-T also employs the Word2Vec [19] method based on duration number of e^i_{t-1}. Since the transformer structure does not contain the sequential order information that LSTM [13] or GRU [7] have, AR-T incorporates a positional embedding $p_{w(t-1)}$ to represent the order of context embeddings $\{c_{w1}, c_{w2}, ..., c_{w(t-1)}, ...\}$. *sin* and *cos* functions are leveraged to calculate the position values of each context. Finally, AR-T creates an embedding output of each window E_{wj} by concatenating three embedding states c_{wj}, d_{wj} and p_{wj}.

4.3 Transformer [33] Based Activity Recognition

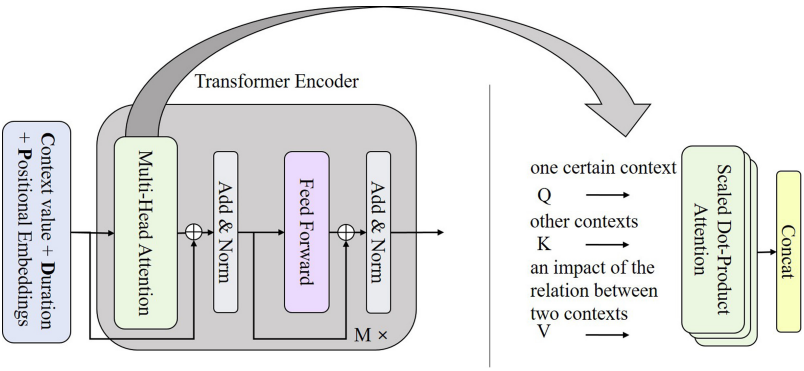

Fig. 3. Overall architecture of the transformer [33] and the detailed attention mechanism

When an embedding sequence $(E_{w1}, E_{w2}, E_{w3}, ...)$ comes in as an input, based on an attention mechanism, AR-T assigns importance weights to each context to preserve important contexts that appear in low frequency and extract long-term correlations between contexts. Figure 3 describes an overall architecture of the transformer [33] and the detailed attention mechanism.

Encoder Structure of AR-T. Each transformer encoder is composed of one multi-head attention element which calculates the importance weights of contexts and a feed-forward element as shown in the left one in Fig. 3. The attention element will be explained in the below section **Attention mechanism**. By connecting the M encoder, AR-T learns various aspects of context correlations and captures long-term dependencies better than the existing approaches. The result of a preceding encoder is input to the next encoder. The encoder utilizes

a feed-forward element to avoid loss of temporal information between contexts by linear activation, expressed by the following formula:

$$FF(x) = Relu(0, xW_1 + b_1)W_2 + b_2 \tag{1}$$

Using Add & Norm elements, AR-T maintains the same embedding shape between inputs and outputs of all encoders and normalizes the values.

Attention Mechanism. The attention mechanism computes the correlation weights between contexts and implicitly assigns the importance weights of each context. AR-T calculates the attention score representing correlation importance weights of a specific context using the following scaled dot product-based formula:

$$Attention(Q, K, V) = softmax(QK^T/\sqrt{d_k})V \tag{2}$$

Referring to the right figure of Fig. 4, Q represents a specific context, K means other contexts and V stands for the importance of those correlations to recognize an activity. AR-T calculates attention scores for all events by the scaled dot product between Q and K and compresses them using a softmax method and dot product with V. For example, as shown in Fig. 4, if the context containing event *Podium Occupying* correlates with the context containing event *Projector Use*, their relations have a high attention score. As the name of multi-headed attention suggests, AR-T proceeds the attention process multiple times and allows covering context correlations in many aspects. AR-T concatenates outcomes from each attention process and the final result represents holistic context correlations that are important for recognizing activities.

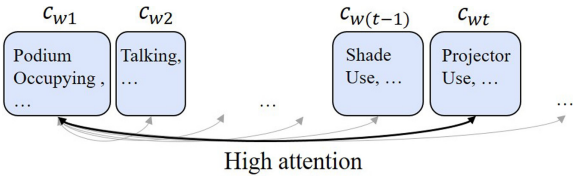

Fig. 4. An example of how to assign attention scores to correlated contexts

5 Experiments and Results

In this section, we explain comparison results between AR-T and the existing works [17,21,39] with accuracy, f1-score, recall and precision metrics. Unless otherwise specified, the representative of *recognition performance* mainly represents f1-score. Existing public datasets [27,29] for activity recognition approaches have several limitations. They are generated by controlled experiments in which well-defined instructions are given to experimental users. It implies that specific

user and specific sensor values are coupled, which makes it unsuitable to verify recognition performance in real multi-user spaces. Second, they do not involve enough types of smart objects that covering various patterns of the real world smart spaces. We conduct experiments in a testbed in our university, a seminar room where we install various smart objects for capturing everyday usages. We also experiment with a public data set created by CASAS group [31] to evaluate applicability in various smart spaces.

5.1 Evaluation Description

We conduct our experiments on Intel Xeon Gold 5215 CPU (2.5GHz), 256GB RAM, and Ubuntu 18.04.5 LTS os. For recognition comparative experiments, we implement AR-T and the three existing approaches [17,21,39] in Python and Keras [11]. The comparison approaches are defined as follow:

LSTM-based (baseline): [21] A recognition approach with LSTMs
BiLSTM-based: [39] A recognition approach with bidirectional LSTMs
AttnSense: [17] A recognition approach with RNN type models and an attention mechanism
AR-T: A proposed temporal relation representable transformer-based approach

We obtain the final result as the average of the experimental results of 5 cross-validations. We apply the same values for parameters such as the number of hidden states, the dropout rate and the number of epochs for all approaches. To analyze the experimental results in detail, we use Matplotlib [18] to display confusion matrices of each approach. The code is publicly available to reproduce experimental results of AR-T[1].

5.2 Case 1: Real Seminar Room Dataset

Environment Setup. Figure 5 describes the installation configuration of smart objects in our testbed and the detailed generated types of events from them. We install twenty-one smart objects in our testbed which publish seven different types of events. They are developed by Raspberry Pi 3 and commercial sensor frameworks (e.g. Phidgets [26], Monnits [20] and DigiKeys [8]). The description of four major activities in our testbed is as follows:

Group Chatting: A pair of people sit down nearby and have a casual conversation
Seminar: One or more speakers make presentations, others discuss topics of the presentations
Technical Discussion: Using a projector to display discussion topics, many people discuss them
Group Study: Some people study together for a long time

We experiment with 111, 129, 52, and 40 examples of each activity. The maximum length of a context sequence for each activity is 31, 146, 141 and 106.

[1] We will release github later due to anonymity.

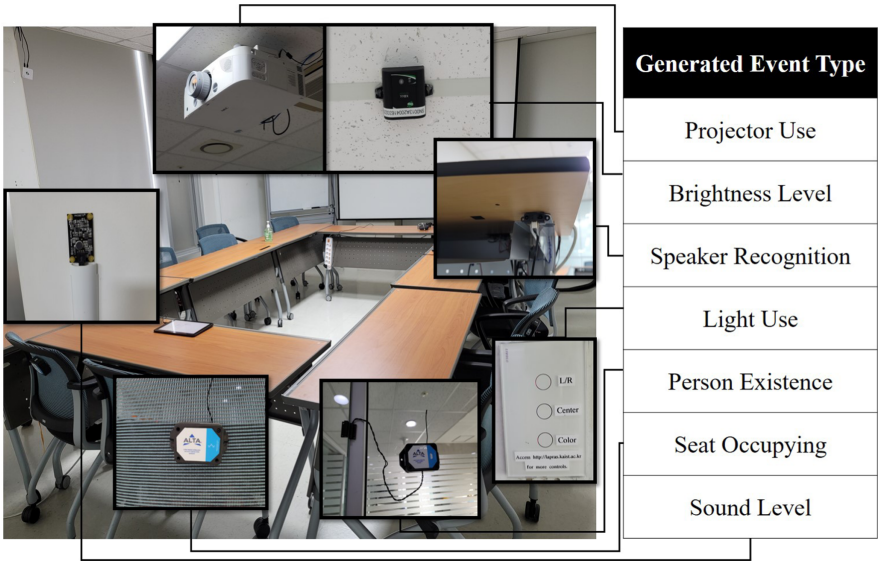

Fig. 5. Multi-user smart space testbed construction and deployed smart objects

Table 1. Recognition performance comparison in the seminar room testbed

Evaluation metrics	AR-T	AttSense	BiLSTM	LSTM(baseline)
Precision	**0.889** ± 0.044	**0.713** ± 0.164	**0.842** ± 0.059	**0.284** ± 0.228
Recall	**0.882** ± 0.038	**0.746** ± 0.105	**0.825** ± 0.060	**0.412** ± 0.068
F1-score	**0.879** ± 0.049	**0.717** ± 0.149	**0.828** ± 0.046	**0.274** ± 0.110
Accuracy	**0.882** ± 0.038	**0.746** ± 0.105	**0.825** ± 0.060	**0.412** ± 0.068

Recognition Performance Comparison. AR-T improves recognition performance by resolving the problems of previous approaches. As shown in Table 1, the AR-T performs better than the existing ones in terms of all performance metrics. The proposed scheme improves accuracy by 18.23%, 6.91%, and 114.08% compared to AttSense [17], BiLSTM-based approach [39], and baseline model [21], respectively. In addition, as expressed in Table 1, the performance values of AR-T are more stable than other approaches since AR-T discovers context correlation in various aspects. Figure 6 displays the confusion matrix of our testbed for detailed analysis at activity levels.

The multi-user space has a lot of long-term context correlations due to longer context sequences and complex events mixed by multiple users. Accordingly, the baseline model, which is the most vulnerable to the vanishing gradient problem, is difficult to capture these long-term relations and shows overwhelmingly low performance. As shown in Fig. 6, AR-T shows much better accuracy in *Group Study* (0.59) than the AttSense (0.16) and BiLSTM based approach (0.40). *Group Study* has many contexts that are relatively long-term related compared

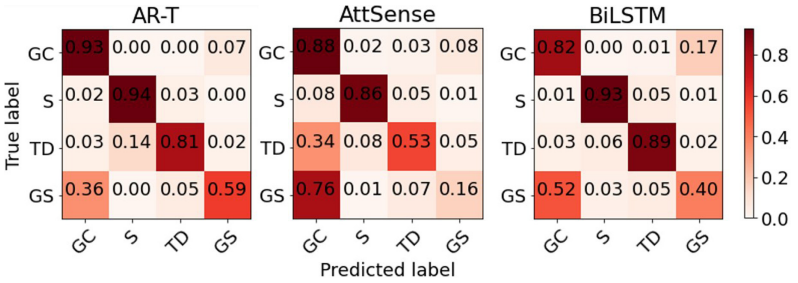

Fig. 6. Normalized confusion matrix comparison in the testbed

to other activities. It implies AR-T has better capability to capture long-term context correlations in longer sequences.

AR-T also aims to increase recognition performance by preserving important contexts which infrequently occur. According to *Seminar* in Fig. 6, AR-T shows the highest accuracy (0.94) since it assigns high importance scores to contexts containing important events such as *Projector On*. However, as a side effect, *Technical Discussion* is recognized as *Seminar* which also regarded *Projector On* as an important event.

The same type of contexts with different durations usually happen in the multi-user space. *Group Chatting* and *Group Study* have the same type of events as elements of activity patterns, however, their durations are very different. The context sequences generated by *Group Study* are much longer than by *Group Chatting* and duration embeddings assist to differentiate them. In this aspect, as represented in Fig. 6, AR-T recognizes *Group Chatting* (0.93) and *Group Study* (0.59) better than the existing approaches. The effects of incorporating durations in embeddings are described in Sect. 6.

5.3 Case 2: Multi-user CASAS Dataset

Dataset Description. To validate the applicability of AR-T to various environments, We conduct an additional experiment using the CASAS activity dataset [31] without using users' identification. It is a smart home testbed based on 51 motion sensors and 15 cabinet sensors and two users performed collaborative tasks together. The sensor configuration of this implies that the recognition approach should exploit context sequences representing users' movement paths for identifying activity patterns. The CASAS activity dataset contains 26 examples of four multi-user activities: *Move furniture*, *Play a game*, *Prepare for dinner* and *Pack a picnic*. The maximum length of a context sequence for each activity is 21, 46, 57 and 143.

Recognition Performance Comparison. As shown in Table 2 and Fig. 7, AR-T outperforms accuracy by 17.90%, 4.95%, and 179.24% compared to AttSense, BiLSTM-based approach, and baseline model, respectively. Since the

Table 2. Recognition comparison of the existing approaches and AR-T in the CASAS dataset [31]

Evaluation metrics	AR-T	AttLSTM	BiLSTM	LSTM(baseline)
Precision	**0.962** ± 0.029	**0.828** ± 0.168	**0.925** ± 0.058	**0.186** ± 0.147
Recall	**0.955** ± 0.033	**0.810** ± 0.162	**0.910** ± 0.065	**0.342** ± 0.210
F1-score	**0.954** ± 0.034	**0.794** ± 0.196	**0.909** ± 0.066	**0.208** ± 0.181
Accuracy	**0.955** ± 0.033	**0.810** ± 0.162	**0.910** ± 0.065	**0.342** ± 0.210

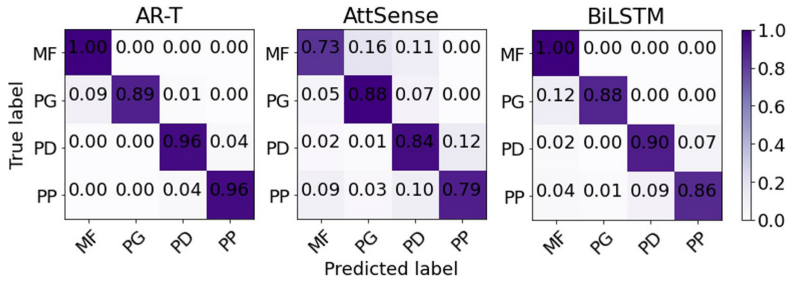

Fig. 7. Normalized confusion matrix comparison in the CASAS dataset

dataset consists of only two users' activities and the activities consist of simpler sequences of events, the overall recognition performance of all approaches are higher than those of the testbed. AR-T also shows reliable performance than others in this dataset as shown in Table 2.

In Fig. 7, ability of AR-T to correlate long-term relations of contexts makes recognizing *Prepare for dinner* (0.96) and *Pack a picnic* (0.96) better than others. Using the importance weights of contexts, AR-T supports the accuracy improvement of *Prepare for dinner* by leaving events that are important but infrequently occurring as elements of *Prepare for dinner* patterns. The duration embeddings in AR-T distinguish between *Prepare for dinner* and *Pack a picnic* sharing the same types of contexts as the elements of patterns of the activities.

6 Discussion

6.1 Effects of Embedding Types

In the embedding step of AR-T, we embed not only the context value but also the duration and positional information. We try to explore performance changes according to embedding methods. In our testbed, the duration incorporated embeddings-based method improves recognition performance by 5.65% for context only embeddings-based, and by 4.52% for context and positional embeddings-based. In the CASAS dataset, duration included embeddings-based

method outperform by 4.26% compared to context only embeddings-based, and by 4.15% compared to context and positional embeddings-based. This means that the duration is a good distinction factor for distinguishing between activities composed of the same context types.

6.2 Recognition Performance of BiLSTMs with an Attention Mechanism

From the results shown in Table 1 and Table 2, one may guess that BiLSTMs with an attention mechanism (i.e. AttBiLSTM), which no approach has tried, could perform as good as the transformer. To figure out whether this assertion stands, we conduct an experiment on BiLSTM with attention. The result shows that AR-T outperforms by 9.06% in our testbed and 13.84% in the CASAS dataset. Compared to other approaches [17,21], it derives long-term context relations well, however, the fact that it is a sequential model with a gradient vanishing problem makes performance worse than AR-T.

6.3 Effects of Window Sliding Methods

We try to analyze how window size affects recognition performance. We compare a dynamic window sliding method of AR-T and a fixed window sliding. By the experiments, the dynamic window sliding method of AR-T improves recognition performance around 1%∼2% than the fixed window sliding. We expect the dynamic window sliding approach to further improve recognition performance, but other sliding approaches are needed to find the optimal window size for a multi-user space.

7 Conclusion

In this paper, we propose AR-T, an accurate and applicability approach for activity recognition in multi-user smart spaces of the real world. The existing studies have difficulties in (1) correlating long-term relations between contexts, (2) finding important contexts that are infrequent, and (3) distinguishing activities that have the same types of contexts. AR-T utilizes the attention mechanism of the transformer [33] to find long-term context correlations. In addition, AR-T allows important but infrequent contexts to remain as elements of activity patterns, based on the importance weight of each context. AR-T uses duration embeddings to differentiate between the same type of context with different durations. This supports to differentiate between activities that share the same context sequences.

To recognize multiple activities in multi-user spaces, we plan to extend AR-T with multi-task learning [30]. We also plan to develop a sparse attention method [1] for sensor streams to build a more efficient and scalable structure in various spaces while retaining important contextual information.

Acknowledgement. This work was partly supported by Institute of Information & communications Technology Planning & Evaluation(IITP) grant funded by the Korea government(MSIT) (No.2019–0-01126, Self-learning based Autonomic IoT Edge Computing) and the ICT RD program of MSIT/IITP (No.2020–0-00857, Development of Cloud Robot Intelligence Augmentation, Sharing and Framework Technology to Integrate and Enhance the Intelligence of Multiple Robots).

References

1. Ainslie, J., Ontanon, S., Alberti, C., Cvicek, V., Fisher, Z., Pham, P., et al.: ETC: encoding long and structured inputs in transformers. In: Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP), pp. 268–284 (2020)
2. Al Ameen, M., Liu, J., Kwak, K.: Security and privacy issues in wireless sensor networks for healthcare applications. Journal of medical systems **36**(1), 93–101 (2012)
3. Benmansour, A., Bouchachia, A., Feham, M.: Multioccupant activity recognition in pervasive smart home environments. ACM Comput. Surv. (CSUR) **48**(3), 1–36 (2015)
4. Campion, M.A., Medsker, G.J., Higgs, A.C.: Relations between work group characteristics and effectiveness: implications for designing effective work groups. Personnel Psychol. **46**(4), 823–847 (1993)
5. Chen, L., Nugent, C.: Ontology-based activity recognition in intelligent pervasive environments. Int. J. Web Inf. Syst. (2009)
6. Chen, R., Tong, Y.: A two-stage method for solving multi-resident activity recognition in smart environments. Entropy **16**(4), 2184–2203 (2014)
7. Cho, K., Van Merriënboer, B., Gulcehre, C., Bahdanau, D., Bougares, F., Schwenk, H., Bengio, Y.: Learning phrase representations using RNN encoder-decoder for statistical machine translation. arXiv preprint arXiv:1406.1078 (2014)
8. DigiKey. www.digikey.com/. Accessed 31 Mar 2021
9. Foresti, G.L., Mähönen, P., Regazzoni, C.S. (eds.): Multimedia video-based surveillance systems: requirements, issues and solutions. Springer Science and Business Media, New York (2012)
10. Gavrilyuk, K., Sanford, R., Javan, M., Snoek, C.G.: Actor-transformers for group activity recognition. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 839–848 (2020)
11. Gulli, A., Pal, S.: Deep learning with Keras. Packt Publishing Ltd (2017)
12. Haresamudram, H., Beedu, A., Agrawal, V., Grady, P.L., Essa, I., Hoffman, J., Plötz, T.: Masked reconstruction based self-supervision for human activity recognition. In: Proceedings of the 2020 International Symposium on Wearable Computers, pp. 45–49 (2020)
13. Hochreiter, S., Schmidhuber, J.: Long short-term memory. Neural Comput. **9**(8), 1735–1780 (1997)
14. Hsu, C. C., Wang, L. Z.: A smart home resource management system for multiple inhabitants by agent conceding negotiation. In: 2008 IEEE International Conference on Systems, Man and Cybernetics, pp. 607–612. IEEE (2008)
15. Kriegel, H.P., Kröger, P., Sander, J., Zimek, A.: Density-based clustering. Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery **1**(3), 231–240 (2011)

16. Lara, O.D., Labrador, M.A.: A survey on human activity recognition using wearable sensors. IEEE communications surveys & tutorials **15**(3), 1192–1209 (2012)
17. Ma, H., Li, W., Zhang, X., Gao, S., Lu, S.: AttnSense: multi-level attention mechanism for multimodal human activity recognition. In: IJCAI, pp. 3109–3115 (2019)
18. Matplotlib. matplotlib.org/. Accessed 31 Mar 2021
19. Mikolov, T., Chen, K., Corrado, G., Dean, J.: Efficient estimation of word representations in vector space. arXiv preprint arXiv:1301.3781 (2013)
20. Monnit. www.monnit.com/. Accessed 31 Mar 2021
21. Murad, A., Pyun, J.Y.: Deep recurrent neural networks for human activity recognition. Sensors **17**(11), 2556 (2017)
22. Nweke, H.F., Teh, Y.W., Al-Garadi, M.A., Alo, U.R.: Deep learning algorithms for human activity recognition using mobile and wearable sensor networks: state of the art and research challenges. Expert Syst. Appl. **105**, 233–261 (2018)
23. Ordóñez, F.J., Roggen, D.: Deep convolutional and lstm recurrent neural networks for multimodal wearable activity recognition. Sensors **16**(1), 115 (2016)
24. Pantelopoulos, A., Bourbakis, N. G.: A survey on wearable sensor-based systems for health monitoring and prognosis. IEEE Trans. Syst. Man Cybern. Part C Appl. Revi. **40**(1), 1–12 (2009)
25. Papagiannidis, S., Marikyan, D.: Smart offices: a productivity and well-being perspective. Int. J. Inf. Manage. **51**, 102027 (2020)
26. Phidget. www.phidgets.com/ Accessed 31 Mar 2021
27. Reiss, A., Stricker, D.: Introducing a new benchmarked dataset for activity monitoring. In: 2012 16th International Symposium on Wearable Computers, pp. 108–109. IEEE (2012)
28. Riboni, D., Sztyler, T., Civitarese, G., Stuckenschmidt, H.: Unsupervised recognition of interleaved activities of daily living through ontological and probabilistic reasoning. In: Proceedings of the 2016 ACM International Joint Conference on Pervasive and Ubiquitous Computing, pp. 1–12 (2016)
29. Roggen, D., Calatroni, A., Rossi, M., Holleczek, T., Förster, K., et al.: Collecting complex activity datasets in highly rich networked sensor environments. In: 2010 Seventh International Conference on Networked Sensing Systems (INSS), pp. 233–240. IEEE (2010)
30. Ruder, S.: An overview of multi-task learning in deep neural networks. arXiv preprint arXiv:1706.05098(2017)
31. Singla, G., Cook, D.J., Schmitter-Edgecombe, M.: Recognizing independent and joint activities among multiple residents in smart environments. J. Ambient Intell. Human. Comput.**1**(1), 57–63 (2010)
32. Trimble, C.R.: What is signal averaging. Hewlett-Packard J.**19**(8), 2–7 (1968)
33. Vaswani, A., et al.: Attention is all you need. arXiv preprint arXiv:1706.03762 (2017)
34. Wang, J., Chen, Y., Hao, S., Peng, X., Hu, L.: Deep learning for sensor-based activity recognition: a survey. Pattern Recogn. Lett. **119**, 3–11 (2019)
35. Wang, L., Gu, T., Tao, X., Chen, H., Lu, J.: Recognizing multi-user activities using wearable sensors in a smart home. Pervasive Mobile Comput. **7**(3), 287–298 (2011)
36. Wu, N., Green, B., Ben, X., O'Banion, S.: Deep transformer models for time series forecasting: the influenza prevalence case. arXiv preprint arXiv:2001.08317 (2020)
37. Yang, J., Nguyen, M.N., San, P.P., Li, X., Krishnaswamy, S.: Deep convolutional neural networks on multichannel time series for human activity recognition. In: Ijcai, pp. 3995–4001 (2015)

38. Yang, L., Ng, T. L. J., Smyth, B., Dong, R.: Html: hierarchical transformer-based multi-task learning for volatility prediction. In: Proceedings of The Web Conference 2020, pp. 441–451 (2020)
39. Zhao, Y., Yang, R., Chevalier, G., Xu, X., Zhang, Z.: Deep residual bidir-LSTM for human activity recognition using wearable sensors. Math. Problems Eng. (2018)