ABSTRACT
Real-time human context recognition is one of the most exciting emerging technologies in sensing nowadays. Compared with most recognition problems in machine learning, the challenge lies in the complexity and incompleteness of labels, in other words, each sample can have several label concepts simultaneously but some of them could be missing. This poster proposes an effective approach for multilabel human context recognition with signals from sensors embedded in the wearable devices. The proposed algorithm demonstrates to be very robust to incomplete labels.

1 INTRODUCTION
The health industry brings a boom in human context recognition. A person’s context includes their location, activity and so on. Traditionally, such information are collected either by self-report or by the caregivers. Automatic recognition technology greatly reduces manpower requirements. More importantly, it can provide feedback at arbitrary time so that interventions could be taken in time.

In this work, we aim to solve the multilabel human context recognition problem. To reflect the real situation, daily-used wearable devices are used to obtain data. The dataset used in this paper was collected by Vaizman et al. [2]. We adopted signals from six core sensors and used the same input features to be consistent with the data preprocessing steps in [3].

The positive labels of each instance are selected from a large candidate label set by the user themselves. This could result in incomplete labeling, so in the poster we mitigate classification bias due to incomplete labels with a maximal correlation type regularization. Our work was compared with the original work in [3], i.e. Multiple Layer Perceptron (MLP).

2 PROPOSED METHOD
Low rank label transformation is usually adopted in the multilabel classification problems, in which labels are embedded into dense vectors with lower dimension than the original label space. While previous works implement label transformation using matrix decomposition or alternating optimization, both are limited in non-linear expressiveness and scalability for large datasets. To overcome these flaws, we integrate the idea of low rank label transformation into a network shown in Figure 1. We restrict the dimension of the last hidden layer to be smaller than the label cardinality, and the weights connected between the last hidden layer and the output layer could be regarded as embedding vectors for labels. For example, in our problem, the first unit in the output layer stands for the probability of lying down, and all the weights connected between this unit and the units in last hidden layer compose a vector denoted as $v_1$, then $v_1$ is the embedding vector for the label “Lying down”.

In the case with incomplete labels, the missing labels are mixed up with negative ones. This could bring noise to the training process. Consider the following scenario: a training instance is labeled with “Cooking”, but missed with a positive label “At home”. Then the training optimizer would treat the label “At home” as negative and compute the gradients in a totally wrong direction. However, the label “Cooking” and “At home” often co-occur in the same instance in the whole training data. In other words, these two labels are highly correlated. The correlations among labels can be utilized to mitigate the problem caused by missing labels. Based on the claim, we proposed a label similarity regularization on the embedding vectors for labels.

Huang et al. [1] proposed the Generalized Maximal Correlation (GMC) to maximize the total correlation among multiple variables. We transform the original optimization problem with constraints
Then the total loss function is

\[ \text{use a hyper-parameters } \alpha \text{ and ground truth } l \]

More precisely, \( l(y_j^{(i)}, \hat{y}_j^{(i)}) \) is the log-loss between prediction \( \hat{y}_j^{(i)} \) and ground truth \( y_j^{(i)} \). Since different losses have different scales, we use a hyper-parameters \( \alpha \) to weight label similarity regularization. Then the total loss function is

\[ \text{total loss} = \text{log-loss} + \alpha \Omega \]