THUIR at the NTCIR-15 Micro-activity Retrieval Task

Jiayu Li*
DCST, Tsinghua University
jy-li20@mails.tsinghua.edu.cn

Ziyi Ye*
DCST, Tsinghua University
ye-zy20@mails.tsinghua.edu.cn

Weizhi Ma
DCST, Tsinghua University
mawz@tsinghua.edu.cn

Min Zhang†
DCST, Tsinghua University
z-m@tsinghua.edu.cn

Yiqun Liu
DCST, Tsinghua University
yiqunliu@tsinghua.edu.cn

Shaoping Ma
DCST, Tsinghua University
msp@tsinghua.edu.cn

ABSTRACT

Activity recognition is a general but important task in various scenarios of human behaviors, in which some pre-defined activities are recognized based on heterogeneous sensor data. Previous studies mainly focused on the problem of long-term physical state distinguish, helping researchers understand the behaviors of individuals. Differently, NTCIR-15 Micro-activity Retrieval Task (MART) concentrates on micro activity recognition, which aims to identify activities in minutes of daily behavior, requiring a deeper insight into the character of the activities.

In this paper, we present the methodologies that our team, THUIR, employed in the MART. Firstly, various feature engineering methods are applied to extract valuable features from multi-modal raw data, and feature selection methods are adopted to maintain useful features. Then, we try two different ways to handle this task: taking it as 1) a ranking problem or 2) a multi-label classification problem, two distinct approaches are proposed: a similarity-based approach for the ranking problem and tree-based classifiers for the classification problem. In two-fold cross-validation experiments, the combined model of correlation-based feature selection method and rule-based Gradient Boosting Decision Tree (GBDT) classifier outperforms other models, reaching mAP of 0.95 on the test set. And this method also achieves the best performance among all participants in the MART.

TEAM NAME

THUIR

SUBTASKS

Retrieval Task

*These authors contributed equally to this work and should be considered co-first authors.
†Corresponding author.

Conference'17, July 2017, Washington, DC, USA

ACM ISBN 978-x-xxxx-xxxx-x/YY/MM…$15.00
https://doi.org/10.1145/nnnnnnn.nnnnnnn

1 INTRODUCTION

The THUIR team participates in the Retrieval Task of the NTCIR-15 Micro-activity Retrieval Task (MART) [7], and this report describes our approach to solve the MART problem and discusses the official results.

With the development of wearable devices, researches on individual activity sensor data have become popular in the community of computer science, medicine, and psychology [5]. Based on various kinds of sensor data, researchers studied the processing, recognition, and analysis of human activities, in order to explore and comprehend how we live our life.

In the NTCIR-15 Micro-activity Retrieval Task (MART), we focus on the recognition of micro activities, which only last minutes. For dataset collection, participants were asked to complete twenty activities (described in Table 1) for 3 times, 90 seconds each time. During the experiment, their first-person perspective photos, biometric signals, and computer interactions were recorded in a data stream. Each repetition for participant is called an *instance*. The task is to construct a retrieval system that could return the relevance ranking list of all instances for a given query (i.e., a certain activity).

Compared with previous researches on activity recognition [13, 16, 23], there are several new challenges in the MART: 1) The multimodal data is collected from abundant kinds of sensors, requiring carefully-designed methods to extract and combine different types of records. 2) The activities collected in the MART occur over short time-scales rather than the traditional long-duration activity segmentation, which means valid features should be extracted from relatively time-limited data. 3) Many of the micro-activities are quite similar and hard to distinguish. Some activities may be recognized by human easily, but others are much more difficult to predict by observation.

To cope with these challenges, feature engineering and model construction are carefully arranged in our approach. For the multimodal raw data, we design various feature extraction methods adapting to different kinds of data, learning from previous works in sequence signal processing, computer version, and biology communities. Then feature selection proceeding is undertaken based on the selected features, for better use of the limited data over minutes. To distinguish similar activities, ranking methods and classification methods are both conducted for the retrieval model. Human-designed rules are combined with machine learning algorithms to utilize both human knowledge for the activities and algorithm understanding for digital data.

To summarize, our main contributions are as follows:

- Various methods for feature selection are applied for the dataset, which can handle the multi-modal and time-limited data.
- We show that the retrieval task can be solved as 1) a ranking problem or 2) a multi-label classification problem, and propose two distinct approaches: 1) similarity-based approach for the ranking problem, and 2) tree-based approach for the classification problem.
- Abundant experiments show promising results of all the methods we proposed. Among them, the rule-based classification model with feature selection method of correlation and PCA dimensionality reduction achieves an mAP of 0.950 on the test set, ranking first at the MART retrieval subtask.

2 BACKGROUND AND RELATED WORK

Activity recognition helps us understand the lifestyle and behaviors of individuals, and it is widely applied in entertainment, healthcare, and military [13, 18, 24]. With the development of multi-sensor technologies, image, screenshot, accelerometer, eye movement, electro-oculography, and electroencephalo-graph are applied to detect human behaviors. Based on the multi-modal records, it is a general but important topic to recognize and predict human activity. Previous activity recognition approaches include recognition of home behavior [21, 22], military-specific activity [24], online activity [18], and hand gesture [9]. Cook divided these human activity recognition tasks into video-based and sensor-based activity recognition [2], which analyses visual data (such as photos or videos) and smart-sensor data respectively. Since MART dataset mixes data from different sources and involves abundant micro activities, the task can be more challenging and interesting.

The basic process for an activity recognition task includes feature extraction, feature selection, and activity classification with the selected features.

At the first step, feature extraction determines the upper bound of overall model performance. Various methods can be applied to deal with the multi-sensor data. For feature extraction of photos and screenshots, the latest progress in computer version is available. Object detection methods such as ResNet101 model [6] can be applied to recognize the items in surrounding scene, which contributes to inferring human activity. OCR technique [8] for text detection is useful for feature extraction in screenshot data, especially in web links detection. As for time-series data, statistical characteristics in the time domain and frequency domain are the most common features. Other methods in sequence signal processing including wavelet transform [4], LPC cofficient [4], peak detection are also suitable for the raw data.

Then, the feature selection step is significant for excluding useless features and avoiding the influence of spurious correlation characteristics. To exclude irrelevant features, human knowledge is accurate but time-consuming for high-dimensional feature space. Thus, machine learning method that can automatically select better features is an alternative solution. Dash et al. [3] introduces the feature selection framework for classification tasks, which includes a generation procedure to produce the candidate feature set, an evaluation procedure to examine the subset, and a stopping procedure to decide when to stop. Filter-based, wrapper-based [12, 17], and embedded-based models [19] can be applied to this framework according to the specific task.

The approaches to construct activity recognition and prediction models are various. The most commonly-used classifiers are decision tree [14], support vector machine (SVM), naive Bayes, and hidden Markov models [11]. Deep learning models are also confirmed to be effective in recent studies [10, 23], including deep neural network [21], convolutional neural network [25], recurrent neural network, and ontological reasoning [15]. To choose the proper model, accuracy, efficiency, and explainability are often considered as evaluation metrics. Hybrid models combining the advantages of different models are also widely used.

Table 1: Description of twenty micro activities in MART.

Activity Group	Activity ID	Activity Description
Screen-relevant activity	1	Writing/replying to an email
	2	Reading text on screen.
	3	Editing a presentation.
	11	Watching a youtube video.
	12	Browsing news website.
Physical activity (static)	4	Zoning out while staring at a point.
	16	Close eyes, refrained from any movement.
Physical activity (partial)	6	Physical precision task with both hands.
	9	Counting physical currency.
	15	Drinking/eating.
	19	Use both hands to play a tennis ball.
Physical activity (general)	17	Cleaning.
	18	Repeatedly sit up-and-down.
	20	Walking/pacing around.
Document-relevant activity	5	Finance management.
	7	Document organization.
	8	Reading text on paper.
	10	Writing with pen on paper.
Communications	13	Having a conversation with another person.
	14	Making a telephone call.

3 DATASET DESCRIPTION AND FEATURE EXTRACTION

The dataset for MART task is a new record collection of daily micro-activities. Since the dataset contains rich multi-modal data from various sensor devices, numerical features should be extracted from the raw data records. In this section, we will give a brief introduction to the original dataset and the methods we use for feature extraction.

3.1 Dataset Overview

The MART task dataset contains multi-modal sensor records of 7 participants doing 20 micro activities, which are presented and grouped in Table 1. In the dataset, each participant repeats the 20 different activities for three times, one 90 seconds. Meanwhile, various biometric and movement information is collected, including

his/her first-person perspective photos, electrooculography data, heart rate, acceleration of the head and both arms, mouse movements, and screenshots of the computer. In total, 420 activities with length of 90 seconds are recorded, and the entire dataset is about 2G.

For the retrieval task, 280 of the instances (2 repetitions of each participant and activity) are released with labels as the training set, and the rest 140 activities are left as the test set.

3.2 Photo Feature Extraction

The photos in the dataset are taken automatically from the first-person perspective and contain rich information about the micro-activities. On the one hand, participants are doing different types of activities in different environment, so the similarity of photos in the same instance can be different. For example, *Reading text on paper* may generate similar images, but the photos taken while *Walking around* are more varying. Therefore, in each instance, the pairwise similarity of image histogram is calculated for all the photos, and the mean value, minimum, maximum, and variance of similarities are extracted as features.

On the other hand, we try to extract the semantic concepts in the photos to better understand the activity. In the original dataset, detection probability of 1000 labels is computed for each instance by a pre-trained resnet101 model [6]. Besides, *Densecap API* from deepai[1] is also used for object detection, where 294 concepts are detected and the mean value, minimum, maximum, and variance of photos in the same activity instance are computed.

Finally, we extract 4180 features from the photos for each instance.

3.3 Biometric Signal Features

In the dataset, we have abundant records of time-aligned biometric information, which reflects participants' physiological status and changes during the activities. Therefore, statistical analysis and time series analysis methods are used for extracting features from the biosignals.

3.3.1 EOG data. Electro-oculography (EOG) is a technology to record the voltage difference caused by eye movement. It is widely used in the detection of blink, glance, and vigilance. EOG up/down activity and EOG left/right activity are recorded in the MART dataset, calculated by re-referencing channels for electrodes placing vertically and horizontally. Record units are in micro Volts with a sampling rate of 100 HZ for the 90-seconds activity. We extract features for both directions of EOG records with the following steps.

Firstly, we select 8 time-domain features and 11 frequency-domain features. The time-domain features include mean value, variance, standard deviation, maximum value, minimum value, number of zero crossings, difference between maximum, minimum values, and mode value. The frequency-domain features include dc component and 5 statistic values (mean, variance, standard deviation, slope, and kurtosis) for both frequency-domain graph and amplitude. Secondly, we calculate 12 dimensional LPC coefficient [1] as our features according to the Levinson-Durbin's recursive algorithm [20]. Since there are two directions of EOG data, we get 62 EOG features for each instance.

3.3.2 Acceleration and heart rate. The accelerator is a common sensor to collect data in activity recognition. In the dataset, acceleration in three axes of head and both arms are recorded with a sampling rate of 100 Hz, generating a sequence of digital data. For each position and axis, we use the same methods to extract features.

First, the time-domain features from *PANDAS* file of the original dataset are retained, including minimum, maximum, median, mean value, variance, and length of every position and axis data sequence. Second, with the help of a toolkit for activity recognition[2], we get features in the frequency domain, such as dc component, characteristic of frequency spectrum, and other statistic features in magnitude spectra. Moreover, inspired by the recent work on activity recognition [13], the relative energy of short-time Fourier transform of the acceleration series in different frequency bands is appended. And correlations between each pair of the three axes are also computed.

The heart rate data is relatively simple, collected with a sampling rate of 1Hz for each instance. Hence, the time-domain features are extracted using the same method as acceleration. And frequency-domain features are not computed due to the low frequency of heart rate.

At last, 523 dimensions of acceleration features and 26 dimensions of heart rate features are extracted and saved.

3.4 Computer Interaction Features

Participants' interactions with computer are also recorded in some of the activities. These kinds of data are able to be used for distinguishing screen-relevant and screen-irrelevant activities, as well as classifying the instances with computer.

3.4.1 Mouse movements. Euclidean distance (in units of pixels) and time differences (in seconds) between successive mouse movements are recorded for our task and we can easily acquire the velocity of mouse movements with them. Considering the correlation between distance, time, and velocity, we only utilize distance and velocity data because they are more explainable for translation and warp of cursor. To be more specific, we utilize the recorded length, mean, median, standard deviation values of distance, and velocity data as our features (7 features in total). We also calculate the number of peaks in velocity data as our feature (peak means a recorded velocity which is greater than its n nearest neighbors, and n is 10 in our practice). This feature might be correlated to real wrap times in the activity. Finally, we extract 14 features for mouse movement data.

3.4.2 Screenshots. About 1 to 5 pieces of screenshot are recorded for each of some activity instances. For this kind of images, it's hard to extract semantic information with traditional object detection models. To utilize the data effectively, the top of the screenshot is snipped and detected with an OCR API provided by Baidu[3]. In this way, we get the URL for websites and the toolbars of applications (Called *URL* collectively in the following discussions), which helps identify different activities.

[1] https://deepai.org/

[2] https://github.com/jindongwang/activityrecognition

[3] https://ai.baidu.com/tech/ocr

Table 2: Feature dimension for each kind of data.

Feature	Original dimension	Selected dimension
Acceleration	523	337
EOG	62	45
Heart rate	26	14
Mouse	14	8
Photo	4180	150 (or depend on PCA dimension)
Screenshot	1	1
User id	1	1
All	4806	556

4 FEATURE SELECTION

With high-dimensional features and limited labels, suitable approaches for feature selection are in demand. Previous researches have shown lots of feature transformation and selection methods in feature engineering [26]. Filter-based and tree-based methods are commonly used, including correlation coefficient analysis, chi-square test analysis, and decision tree methods. Methods based on wrapper such as MDLM [17], LVF [12] are also verified to be effective. Since most of our selected features are ordinary and our labels are limited, we only attempt filter-based and tree-based methods. Meanwhile, we apply hybrid method by averaging the selected features' rank of different methods. Since the photo features are too detailed, and most of the labels from ResNet are detected in none of the instances, we only retain the Resnet-detected features with max probability score greater than 0.5. For better comparison of different methods, a fixed number of features is defined for each kind of data (shown in Table 2). Then the following methods are applied:

- **Chi2.** We calculate the variance of each feature, and features with relatively greater variances are remained, since they may be more informative to distinguish the labels. The threshold of variance filter is chosen according to the feature numbers in Table 2.
- **Correlation.** If two features are too correlated to one another, they may express similar information. Therefore, to simplify the feature set, one of them can be removed. Hence, we calculate the pair-wise correlation of features and randomly drop one feature of the pairs whose correlation is greater than threshold.
- **GBDT.** In this approach, features are selected according to the importance of each feature given by Gradient Boosted Decision Tree(GBDT). The global importance of a feature j is given by:

$$\hat{J}_j^2 = \frac{1}{M}\sum_{m=1}^{M}\hat{J}_j^2(T_m) \tag{1}$$

Where M is the tree number, $\hat{J}_j^2(T_m)$ is the importance of feature j in tree m, which is calculated by:

$$\hat{J}_j^2(T) = \sum_{t=1}^{L-1}\hat{i}_t^2\mathbb{I}(v_t = j) \tag{2}$$

Where L is the number of tree nodes, v_t is the features correlated with node t, and $\hat{i}_t^2$ is the square loss reduction when node t is split. The selected dimension for each type of features is given in Table 2.

- **Hybrid**. The features' rank is calculated and averaged for three methods: Correlation, CHi2, and GBDT. Then the top rank features were selected.
- **Correlation+PCA**. Correlation method is applied to select features according to a settled threshold 0.95, and we use Principal Component Analysis (PCA) to reduce the dimensions in Photo data. The reduction dimension for Photo data is selected from 50, 150, 250.

Table 3: Similarity of feature sets generated from different feature selection methods.

Methods	Correlation and Chi2	Correlation and GBDT	Chi2 and GBDT
Similarity	0.6219	0.5159	0.5318

The selected feature sets of different proposed models are partially different. Table 3 summarizes the similarity of our feature sets, where similarity denotes the fraction of the same selected features from different methods. The comparison of results from different methods are discussed in Section 6, and it is shown that all of these feature selecting methods work well in our task.

5 METHODS

In the original task setting, MART is a retrieval task to retrieve the most likely instances. Given a query of an activity description, all instances are required to be ranked by their relevance to the query, which can be considered as a ranking problem. On the other hand, as queries are limited in the twenty activity types, the task can be taken as a multi-label classification problem. Firstly, we prove the ranking problem is equivalent to the classification problem.

In a multi-label classification problem, for a given instance I, the possibility that I belongs to class A is $P(I \in A) = P(A|I)$, which satisfy that $\sum_A P(A|I) = 1$. As for the ranking problem, taken an activity as a query, the relevance $P(I|A)$ is used to generate rank for instances. Considering the ranking of instances for a specific query A_i, the Bayes's formula gives the ranking relevance of instance I_j:

$$P(I_j|A_i) = \frac{P(A_i|I_j)P(I_j)}{\sum_{k=1}^{n}P(A_i|I_k)P(I_k)} \tag{3}$$

Where n is the number of instance. $P(I_k)$ denotes the prior probability of I_k, which is equal to $\frac{1}{n}$ for any k. Furthermore, $\sum_{k=1}^{n}P(A_i|I_k)$ is equal for all instance I_k with the same A_i.

As a result, $P(I_j|A_i)$ is proportional to $P(I_j|A_i)$ for a given A_i and any instance I_j, so the retrieval task can be solved by optimizing the ranking problem as well as the classification problem.

Therefore, from these two aspects, we propose two different approaches. As a ranking problem, we develop similarity-based methods to calculate the relevance between instances and each of the 20 activities. As for the classification problem, multi-level classifiers are designed, where prediction probability for every class is used for ranking in the final retrieval task. We will introduce details of the methods in the following subsections.

5.1 Methods for Ranking Problem

The original idea to calculate relevance between instance and activity is to calculate the similarity between the present instance and previous instances of this activity, because features are expected to be similar in the same activity. For a given instance I and an activity A, we calculate their similarity score in the following steps.

Firstly, we normalize the features and segment an instance into five feature vectors $I = [i_1, i_2, i_3, i_4, i_5]$ according to their data sources: Acceleration, EOG, Heart rate, Photo, and Mouse movement. The segmentation is helpful because different features may have different importance in similarity prediction.

Secondly, the relevance score $S_{I,I'}$ for a given instance I and instance I' is calculated by

$$S_{I,I'} = \sum_{j=1}^{5} \alpha_j * i_j \bigotimes i'_j \tag{4}$$

where $\bigotimes$ denotes similarity functions for two vectors, such as cosine similarity and Euclidean distance.

Then the similarity score between instance I and activity A is given by

$$S_{I,A} = \sum_{I' \in A} S_{I,I'} + \beta * \sum_{I'' \in A_u} S_{I,I''} \tag{5}$$

where I, I', and I'' belongs to activity A, while I and I'' belongs to the same user u. In our dataset, there is only one instance in the latter sum equation.

In practice, we tune the parameters α and β, and investigate different similarity functions. According to experiments, we choose Euclidean distance as our similarity function. And the final vector α is $[0.5, 3.0, 1.0, 1.5, 0.4]$, and value β is set as 14. The parameters are explainable, as α shows the importance of each data source and β suggests that the similarity score of the same user's instance is more useful for prediction. The parameter tuning result also inspires us to design user-specific models in the future.

5.2 Methods for Classification Problem

Based on the selected 556 features in Section 4, we propose an auto-clustering two-level classifier and a multi-level classifier with human defined rules.

Firstly, we try some traditional basic classifiers on the selected features directly and find that several activities are often confused with each other. Therefore, we conduct a two-level classifier. At first level, we attempt to partition the activities into subgroups based on activity similarity, and predict the group label with a basic classifier. Then second-level classification is implemented within the group. Because confusing activities are classified respectively at the second level and we could try different feature combinations in different basic classifiers, this two-level classifier is expected to have better performance. A framework with one possible group partition pattern is shown in Figure 1.

Table 4: Activity groups of three partition methods. The numbers in the table indicate IDs of activity.

Partition Method	Group 1	Group 2	Group 3	Group 4
Impurity	1,2,11,12	4, 16	others	-
Similarity	1,3	2, 11, 12	4, 16	others
Cluster	1,3,5,6,8,10	2,4,11,12,16	7,9,13,14,15	17,18,19,20

Table 5: Performance on basic linear classifiers and tree classifiers.

Classifier	Accuracy	mAP(classify)	mAP(ranking)
LR	0.825	0.899	0.898
SVM	0.821	0.875	0.848
MLP	0.811	0.890	0.910
Random Forest	0.779	0.869	0.927
XGboost	0.826	0.882	0.921
GBDT	**0.836**	**0.901**	**0.947**

To partition the activities, we consider the following methods:

- *Impurity partition.* Classification results of a basic 20-label classifier helps define which activities are easily confused. Hence, with the results of a 20-label classifier, we recursively merge the activities into groups (two at a time), with the goal of minimizing impurity of the partition. Impurity of partition is defined by the entropy:

$$Impurity(\pi) = -\sum_{i=1}^{n(\pi)} \sum_{j=1}^{n(\pi)} p(C_{\pi,ij}) \cdot log_2 p(C_{\pi,ij}) \tag{6}$$

  Where π represents the partition pattern, $n(\pi)$ means the number of groups under π, and $p(C_{\pi,ij})$ is the probability that activities in group i is classified into group j in the 20-label classifier. At first, each activity is a single group, and $n(\pi)$ is 20. The merging process continues until $n(\pi)$ is no greater than a predefined K. For better prediction accuracy, the unmerged activities are partitioned in one subgroup, as the example in the framework in Figure 1.
- *Similarity partition.* Inspired by the methods in Section 5.1, similarity can also be used for activity partition. The group similarity is defined as the average similarity of each pair of instances between the two groups. Starting from 20 separate groups of activities, a pair of groups with the highest similarity will be merged recursively until K groups are left. The similarity is calculated with equation 4.
- *Cluster partition.* More straightforward, the activities can be clustered by their features directly. In this way, the K-means algorithm is used to cluster all the instances into K groups. Then, the activity is labeled with the group where most of its instances belong to.

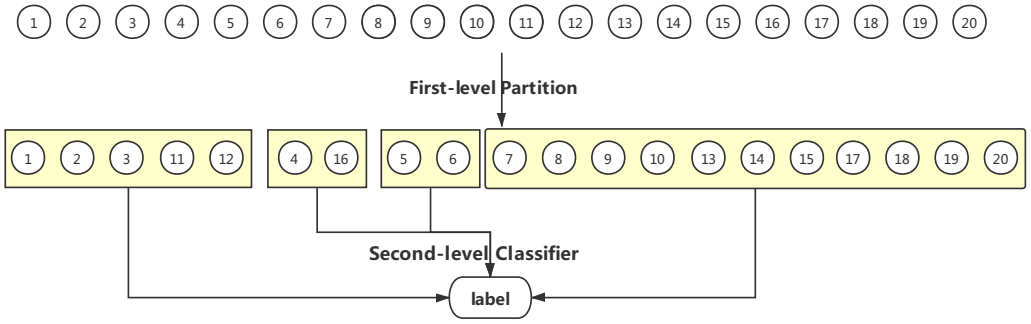

Figure 1: Framework of the two-level classifier with one possible group partition pattern.

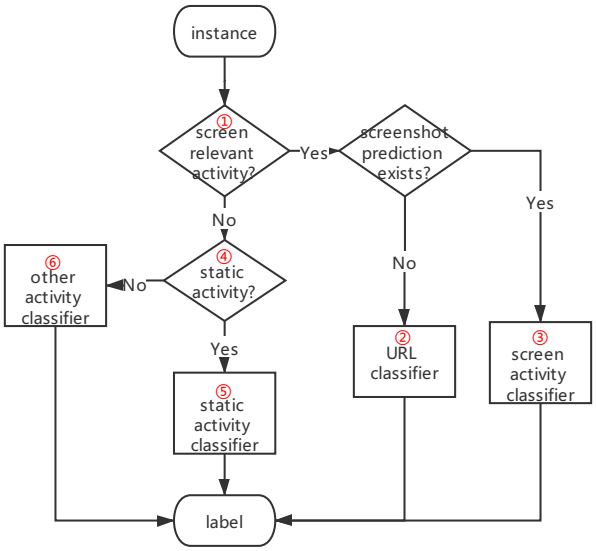

Figure 2: Architecture of the rule-based multi-level classifier.

The groups generated from all partition methods are shown in Table 4. It is observed that results of impurity partition and similarity partition are much the same. The 4th and 16th activities are grouped together in both methods, and they are both static activities. And the screen-relevant activities 1,2,3,11,12 also belongs to the same group(s). These findings illustrate that our partition methods are helpful, and inspire us to develop methods with these rules directly.

Accordingly, we further design a rule-based multi-level classifier, including more human knowledge and predefined rules into the design. Framework of the rule-based classifier is shown in Figure 2.

The main modifications of this classifier focus on screen-relevant activities and static activities (defined in Table 1). As we have computer interaction features in dataset, a binary classifier (No.1) is first imposed to distinguish screen-relevant instances. To be specific, instances without mouse movement and screenshot data will be directly determined as screen-irrelevant, and the other instances will be predicted with a basic binary classifier. For the screen-relevant instances, if the exact activity class can be estimated by URL information detected in section 3.4.2, the label of instance will be determined in classifier No.2. Otherwise, another basic 5-label classifier No.3 predicts which screen-relevant activity the instance belongs to.

For the screen-irrelevant instances, we further separate another special kind of activities, the static activities. Specifically, No.4 is a binary classifier for static activity distinguish. Then No.5 is a binary classifiers for identifying static activities 4 and 16, and No.6 is a 13-label classifier for the rest of the activities.

In Section 6, the choice for basic classifiers, feature groups, and partition methods will be discussed in detail.

6 EXPERIMENTS AND RESULTS

In this section, we conduct extensive experiments to inspect all the methods proposed in Section 4 and Section 5.

6.1 Experiment Settings

We perform two-fold cross-validation for experiment. Since the instances with labels contain the first and second repetition of activities, we pick one as the training set and the other as the validation set for one time, perform two experiments, and report the average performance.

Besides, feature group selection is conducted for the multi-level classifiers. For each basic classifier, subsets of features shown in Table 2 are used for classification, and the feature subgroup with the best performance is chosen.

As for the metrics, we report accuracy for activity prediction for classification problem, and mAP for both problems. Because there is only one correct result for classification, mAP for classification is indeed the average of position reciprocal. For ranking mAP, there are 7 relevant results for each query (7 participants), and we performed normalized mAP:

$$mAP(ranking) = \frac{1}{n}\sum_{i=1}^{n}\frac{\sum_{j=1}^{7} 1/pos(r_{i,j})}{\sum_{i=1}^{7} 1/i} \tag{7}$$

Where n is the number of queries, and $pos(r_{i,j})$ is the position of the j_{th} relevant result for the i_{th} query.

Table 6: Overall results of accuracy, classification and ranking mAP of two-fold cross-validation, as well as ranking mAP on the test set (submission results).

Classifier	Accuracy	mAP (classify)	mAP (ranking)	Submission results
Basic GBDT Classifier	0.836	0.901	0.947	0.895
Similarity-based method	0.789	0.843	0.836	0.782
Two-level Classifier (Impurity partition)	0.875	0.921	0.971	0.901
Two-level Classifier (Similarity partition)	0.875	0.926	0.970	0.928
Two-level Classifier (Cluster partition)	0.796	0.880	0.931	0.886
Rule-based Classifier	**0.889**	**0.933**	**0.974**	**0.950**

Table 7: Accuracy, mAP (classify), and mAP (ranking) for feature selection methods.

Method	Accuracy	mAP (classify)	mAP (ranking)
None	0.711	0.820	0.879
Chi2	0.879	0.928	0.971
Correlation	0.868	0.923	0.961
GBDT	0.897	**0.940**	0.973
Hybrid	**0.900**	0.937	0.972
Correlation +PCA(50)	0.872	0.920	0.960
Correlation +PCA(150)	0.889	0.933	**0.974**
Correlation +PCA(250)	0.878	0.926	0.973

6.2 Classifier Selection

To start with, a basic classifier should be chosen for the construction of all models for the classification problem in Section 5.2. Therefore, we test several common traditional classifiers on 556 features selected by correlation and PCA dimensionality reduction of 150 dimensions. The results are shown in Table 5.

The tested classifiers can be divided into two types, linear models and tree models. They gain similar performances on accuracy and classification mAP, but tree models perform better on the ranking metric. This indicates that tree models learn the relevance between activity and instance better, which suggests that tree-based model is a better composition for the complete multi-level classifier. The experiment result also shows that GBDT (Gradient Boosting Decision Tree) Classifier performs the best on all three metrics. Therefore, we choose GBDT as the basic classifier for all classification-related methods in the following experiments.

6.3 Feature Selection Method

Different feature subsets generated from selection methods in Section 4 are tested on the rule-based classifier (Table 7).

The results reveal that different feature selection methods all perform much better than the features from original dataset (the method *None*). However, these methods have no consistent results on the three metrics, in which *GBDT*, *Merge*, *Correlation+PCA(150)*, and *Correlation+PCA(250)* methods perform much the same. This demonstrates that feature selection methods have relatively less influence on the final results.

Considering that the final task is a retrieval task and feature dimension should be as few as possible, the *Correlation+PCA(150)* method with best mAP(ranking) is used for further explorations.

6.4 Overall Performance

The overall performance of all the models conducted in Section 5 are presented in Table 6. In all, the *rule-based classifier* performs the best on all three metrics, which illustrates the validity of our approaches.

Inspecting other results, we have the following observations. Firstly, the two-level classifiers based on impurity and similarity partition have good performance on two-fold cross-validation, but perform not that well on the test set, which may be the consequence of overfitting on the training set during the feature group selection process. On the other hand, the activity partition based on clustering of features makes the classifier even worse than a basic GBDT, demonstrating that similarity of original features is not a great indicator for activity similarity in GBDT-based classification.

In addition, the similarity-based method has poor performance on both the training set and test set. We inspect the queries with low accuracy, and find that the main reason is the repetition of activity might not be all the same. For instance, when *Having a conversation with another person*, the participant could be sitting or pacing, facing the person or backing to the person. Hence, the similarity-based relevance score can not describe features of some activities accurately.

Finally, we submitted ranking lists of 20 activities generated from all the methods on the test set, and the submission results (ranking mAP on the test set) are displayed in the last column of Table 6. The best *rule-based classifier* also achieves the highest score of 0.95 for submission, ranking first among all participant groups.

7 CONCLUSIONS AND FUTURE WORK

In this report, we present our approach for Micro-activity Retrieval Task. Based on various kinds of feature engineering approaches, we propose ranking models and classification models for the task.

We first apply various methods to extract features from the multi-modal data. As the extracted features are too many and diverse,

feature selection is conducted to reduce the dimension of feature. Then with the proof that ranking problem and classification problem are equal in our task, we construct a similarity-based ranking model and a series of classification models.

Abundant experiments are implemented on basic model selection, feature selection methods, and overall performance. The GBDT is chosen as basic classifier and *Correlation+PCA* is chosen for feature selection. Overall performance shows promising results of our models. On the test set, the ranking model shows an mAP of 0.782, and the classification models achieve mAP of at least 0.88. Among them, the rule-based classifier reaches an mAP of 0.95, ranking first of all the models. The results indicate the utility of our feature engineering methods and classification models, and also illustrate the possibility to detect human micro activities with sensor data.

Although the idea that instance of one activity should be similar is intuitive, the performance of our similarity-based model was worse than the tree-based model. Since the activities are too similar and the recorded time is too short, our simple model can't solve the ranking problem well. For future work, models based on this idea can be further studied since it is more intuitive and explainable.

Moreover, detailed explanations of classification results (or retrieval relevance) can be a direction. Understanding the prediction procedure may help improve the performance of model, as well as help people comprehend the character of human activity. Finally, the cold-start participants can be introduced into the experiments. It is an important application to predict activities of a user without much knowledge about the user.

8 ACKNOWLEDGEMENTS

This work is supported by the National Key Research and Development Program of China (2018YFC0831900, 2018YFC0831700), Natural Science Foundation of China (Grant No. 61672311, 61532011), Beijing Academy of Artificial IntelligenceBAAIand Tsinghua University Guoqiang Research Institute. Dr Weizhi Ma has been supported by Shuimu Tsinghua Scholar Program.

REFERENCES

[1] Rajeev Agarwal, Tomoka Takeuchi, Suzie Laroche, and Jean Gotman. 2005. Detection of rapid-eye movements in sleep studies. *IEEE Trans. Biomed. Eng.* 52, 8 (2005), 1390–1396. https://doi.org/10.1109/TBME.2005.851512

[2] Diane J. Cook, Kyle D. Feuz, and Narayanan Chatapuram Krishnan. 2013. Transfer learning for activity recognition: a survey. *Knowl. Inf. Syst.* 36, 3 (2013), 537–556. https://doi.org/10.1007/s10115-013-0665-3

[3] Manoranjan Dash and Huan Liu. 1997. Feature selection for classification. *Intelligent data analysis* 1, 3 (1997), 131–156.

[4] Jerome Gilles. 2013. Empirical wavelet transform. *IEEE transactions on signal processing* 61, 16 (2013), 3999–4010.

[5] Cathal Gurrin, Alan F Smeaton, Aiden R Doherty, et al. 2014. Lifelogging: Personal big data. *Foundations and Trends® in information retrieval* 8, 1 (2014), 1–125.

[6] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. 2016. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*. 770–778.

[7] Graham Healy, Tu-Khiem Le, Hideo Joho, Frank Hopfgartner, and Cathal Gurrin. [n.d.]. Overview of NTCIR-15 MART. In *Proceedings of the NTCIR-15 Conference, Tokyo, Japan(2020)*.

[8] Md Rabiul Islam, Chayan Mondal, Md Kawsar Azam, and Abu Syed Md Jannatul Islam. 2016. Text detection and recognition using enhanced MSER detection and a novel OCR technique. In *2016 5th International Conference on Informatics, Electronics and Vision (ICIEV)*. IEEE, 15–20.

[9] Youngwook Kim and Brian Toomajian. 2016. Hand Gesture Recognition Using Micro-Doppler Signatures With Convolutional Neural Network. *IEEE Access* 4 (2016), 7125–7130. https://doi.org/10.1109/ACCESS.2016.2617282

[10] Nicholas D. Lane, Petko Georgiev, and Lorena Qendro. 2015. DeepEar: robust smartphone audio sensing in unconstrained acoustic environments using deep learning. In *Proceedings of the 2015 ACM International Joint Conference on Pervasive and Ubiquitous Computing, UbiComp 2015, Osaka, Japan, September 7-11, 2015*, Kenji Mase, Marc Langheinrich, Daniel Gatica-Perez, Hans Gellersen, Tanzeem Choudhury, and Koji Yatani (Eds.). ACM, 283–294. https://doi.org/10.1145/2750858.2804262

[11] Oscar D. Lara and Miguel A. Labrador. 2013. A Survey on Human Activity Recognition using Wearable Sensors. *IEEE Commun. Surv. Tutorials* 15, 3 (2013), 1192–1209. https://doi.org/10.1109/SURV.2012.110112.00192

[12] Huan Liu and Rudy Setiono. 1996. A Probabilistic Approach to Feature Selection - A Filter Solution. In *Machine Learning, Proceedings of the Thirteenth International Conference (ICML '96), Bari, Italy, July 3-6, 1996*, Lorenza Saitta (Ed.). Morgan Kaufmann, 319–327.

[13] Saeed Mehrang, Julia Pietilä, and Ilkka Korhonen. 2018. An activity recognition framework deploying the random forest classifier and a single optical heart rate monitoring and triaxial accelerometer wrist-band. *Sensors* 18, 2 (2018), 613.

[14] IVK Nguyen, P Shrestha, M Zhang, Y Liu, and S Ma. 2019. THUIR at the NTCIR-14 lifelog-3 task: how does lifelog help the user's status recognition. In *The Fourteenth NTCIR Conference (NTCIR-14)*.

[15] Mohd Halim Mohd Noor, Zoran Salcic, I Kevin, and Kai Wang. 2016. Enhancing ontological reasoning with uncertainty handling for activity recognition. *Knowledge-Based Systems* 114 (2016), 47–60.

[16] Juha Parkka, Miikka Ermes, Panu Korpipaa, Jani Mantyjarvi, Johannes Peltola, and Ilkka Korhonen. 2006. Activity classification using realistic data from wearable sensors. *IEEE Transactions on information technology in biomedicine* 10, 1 (2006), 119–128.

[17] Jacob Sheinvald, Byron Dom, and Wayne Niblack. 1990. A modeling approach to feature selection. In *[1990] Proceedings. 10th International Conference on Pattern Recognition*, Vol. 1. IEEE, 535–539.

[18] Muhammad Shoaib, Stephan Bosch, Özlem Durmaz Incel, Hans Scholten, and Paul J. M. Havinga. 2015. A Survey of Online Activity Recognition Using Mobile Phones. *Sensors* 15, 1 (2015), 2059–2085. https://doi.org/10.3390/s150102059

[19] Si Si, Huan Zhang, Sathiya Keerthi, Druv Mahajan, Inderjit Dhillon, and Cho-Jui Hsieh. 2017. Gradient boosted decision trees for high dimensional sparse output. In *International conference on machine learning*.

[20] Alpo Värri, Kari Hirvonen, Veikko Häkkinen, Joel Hasan, and Pekka Loula. 1996. Nonlinear eye movement detection method for drowsiness studies. *International journal of bio-medical computing* 43, 3 (1996), 227–242.

[21] Praneeth Vepakomma, Debraj De, Sajal K. Das, and Shekhar Bhansali. 2015. A-Wristocracy: Deep learning on wrist-worn sensing for recognition of user complex activities. In *12th IEEE International Conference on Wearable and Implantable Body Sensor Networks, BSN 2015, Cambridge, MA, USA, June 9-12, 2015*. IEEE, 1–6. https://doi.org/10.1109/BSN.2015.7299406

[22] Aiguo Wang, Guilin Chen, Cuijuan Shang, Miaofei Zhang, and Li Liu. 2016. Human Activity Recognition in a Smart Home Environment with Stacked Denoising Autoencoders. In *Web-Age Information Management - WAIM 2016 International Workshops, MWDA, SDMMW, and SemiBDMA, Nanchang, China, June 3-5, 2016, Revised Selected Papers (Lecture Notes in Computer Science, Vol. 9998)*, Shaoxu Song and Yongxin Tong (Eds.). 29–40. https://doi.org/10.1007/978-3-319-47121-1_3

[23] Jindong Wang, Yiqiang Chen, Shuji Hao, Xiaohui Peng, and Lisha Hu. 2019. Deep learning for sensor-based activity recognition: A survey. *Pattern Recognition Letters* 119 (2019), 3–11.

[24] Thomas Wyss and Urs Mäder. 2010. Recognition of military-specific physical activities with body-fixed sensors. *Military medicine* 175, 11 (2010), 858–864.

[25] Ming Zeng, Le T Nguyen, Bo Yu, Ole J Mengshoel, Jiang Zhu, Pang Wu, and Joy Zhang. 2014. Convolutional neural networks for human activity recognition using mobile sensors. In *6th International Conference on Mobile Computing, Applications and Services*. IEEE, 197–205.

[26] Honglei Zhuang, Xuanhui Wang, Michael Bendersky, and Marc Najork. 2020. Feature transformation for neural ranking models. In *Proceedings of the 43rd International ACM SIGIR Conference on Research and Development in Information Retrieval*. 1649–1652.