THUIR at the NTCIR-14 Lifelog-3 Task: How does Lifelog help the user's status recognition*

Isadora Nguyen Van Khan[1], Pranita Shrestha[1], Min Zhang[1†], Yiqun Liu[1], and Shaoping Ma

Department of Computer Science & Technology, Institute for Artificial Intelligence, Beijing National Research Centre for Information Science and Technology, Tsinghua University, Beijing 100084, China
ydl17@mails.tsinghua.edu.cn z-m@tsinghua.edu.cn

Abstract. Automatically recognizing a user's status by using Lifelog data can be used to annotate a user's day and then make personal suggestions based on the previous status or as feature for others applications. However, this recognition is yet not well studied.
In this paper we present a method to automatically recognise a user's status. To achieve it we use two different set of features -a non-visual one and another one based on the semantic from pictures- and use supervised Machine Learning algorithms for the recognition. Then, we discuss the impact of the non-visual features on the different statuses we chose and try to find a smaller dataset of features for each status. Finally, we give some statistics and visual insights about the users.
We have obtained good results with the non-visual features: 0.89 accuracy for the Inside or Outside detection, 0.74 for Alone or not and 0.80 for Working or Not. The results are better when using the visual features: 0.95 accuracy.

Keywords: Lifelog· Status Recognition· Clusterisation.

Teamname: THUIR

Subtask: Lifelog Insight Task (LIT)

1 Introduction

Due to the rise of wearable sensors that are getting more affordable, more accurate and more powerful (both in terms of memory or processing) Lifelog and especially visual Lifelog are getting more attention. Lifelog data processing can

* This work is supported by Natural Science Foundation of China (Grant No. 61672311, 61532011) and the National Key Research and Development Program of China (2018YFC0831900).
† Corresponding Auhtor

have various aims such as multimedia memory [1], a way to enhance users' daily life [7], or health status [16].

Lifelog data can be used to give insights about the users: statistics on how many time the user spends on a particular activity or on physical exercice for instance. For this, not only automatic data such as biometrics ones, location or images can be used but also manually gathered data: activity annotation, mood, and so on.

Therefore, this paper proposes to give an automatic recognition of a user's status based on Lifelog data (both numerical and visual data) and to analyse what the detection main clues are. So the aim is to use the automatically gathered data to be able to annotate the user's status.

The paper is organised as follows: in section 2 we review the related work, section 3 presents how the automatic recognition has been done with the features and model presentation and the results for the chosen statuses. Section 4 will discuss the impact of the features on the result, section 5 will give statistics about the users' statuses. Finally section 6 will conclude and describe the future work.

2 Related Work

Most of researches based on Lifelog data are about user activities analysis. A lot of researches want to give statistics and insights on the users' daily life by automatically characterising daily activities [4], matching repeated events [17] or annotating periods of time [15]. Most of these works also give a visualisation of the user activities, sometimes by processing it (aggregation, clusterisation, comparison ...) in order to have more meaningful results for the user [15] or by grouping it by subject [4]. These results can also give insights about the lifestyle and behaviours of the subject [13] by classifying events and base concepts from Lifelog images with semantic data associated with the images.

Other resarches focused on automatic recognition of specific lifestyle traits (Indoors, View of horizon, Buildings, Tree ...) using visual lifelogging and image processing. By analysing the results (co-occuring traits, traits according to users' characteristics) they were able to find correlation between some lifestyle traits and users' characteristics [2].

Because Lifelog data can give behavioural insights, some researches have been done on psychological analysis through Lifelog data. So far, Lifelog data have been used in sleep quality prediction based on the user's activities [9] and in sleep quality impovement with a willingness to detect sleeping habits [7]. Lifelog data have also been used to automatically detect personnality traits based on the Big Five theory. Some work has been done on mood classification using Thayer's 2D model: detection of the music mood style according to the user mood [9] and mood detection and prediction using the user's activities [10].

These studies enable to give more informations that can further be used as automatically collected features for other applications. For instance, nowadays,

to have informations about the user's mood, it has to be manually collected from the user.

For visual Lifelog data, there are two ways to use the images: by extracting features that describe the image content: color, texture or shape using image processing or deep learning or by using semantic tags and clusterising them into concepts.

A lot of studies have been conducted to define words' meanings according to concepts. It exists lexical systems such as Probase or WordNet that for one word give concepts information: synonyms according to concept, clusterisation into bigger concepts [14][8]. Other works try to labelize bag-of-words with a minimum number of concepts [11]. However, these works use syntactic patterns such as isA or isPropertyOf.

Another common way to deal with semantic concept is to use a graph with the nodes representing the words and the edges the relationships between them. This can employed to describe documents with keywords and find document similarity [5]. One work uses Markov Clustering in order to find the different meanings of an ambiguous word. The data used is extracted from books and articles where the relationship between words comes from the proximity of two words in an enumeration [3].

However, to the best of our knowledge, none of these studies have tried to clusterise keywords (thus words that already consist into concepts and without syntactic relationship between them) into bigger concepts.

3 Proposed Insights on Users' status recognition

3.1 NTCIR-14 Dataset

The NTCIR14 Lifelog dataset for the Lifelog-3 task [6] is composed of 42 days of data from 2 users with pictures (around 2 pictures per minute) and non-visual data given per minute (biometrics, semantic location, activity, location coordinates, music history and steps).With the pictures comes a document that gives some semantic concepts (picture's attribute or objects categories) per pictures.

3.2 Features and Model

Features

Original Features First the non-visual data were used to create the original features. The features are presented in Table 1. These features are normalized before being trained and tested.

Visual Concepts Features The status recognition has been done by automatically annotating the pictures with the status (inside or outside, alone or not alone, working or not working).

The automatic recognition was done using the Microsoft Vision API after having processed the pictures: because annotating dozens of thousands pictures

Table 1. Original features table

Name	Type	Description
UserID	int	ID of the user whose sample is from
Time	int	Time (HHMM) of the sample
Heart Rate	int	Heart Rate during the sample's minute
Calories	float	Calories during the sample's minute
Steps	int	Steps walked during the sample's minute
Activity	int	ID of the activity during the sample's minute
Latitude	float	Latitude during the sample's minute
Longitude	float	Longitude during the sample's minute
Location	int	ID of the semantic location during the sample's minute
City	int	ID of the city during the sample's minute

is time-costly, we compared the pictures using the pictures' histogram so we could group pictures in segmentations. Only one picture per segmentation was processed and the same status was given to the other pictures in the segmentation.

The Microsoft API gives categories, tags annotation and a description of the image. In addition, the NTCIR14 dataset contains a file with visual concepts for all the pictures. Therefore, after using the original non-visual features, we worked on semantically recognise the user's status inside or outside, using the clusterisation of the visual concepts from both NTCIR14 and the ones we extracted with the Microsoft API.

The same set of features has been used for the three insight tasks.

Classification Models We decided to try several ensemble and supervised algorithms to train our model. We trained our models with adaptive boosting combined with Random Tree (RT) and C4.5, bagging combined with C4.5 and Logistic model tree (LMT) and finally Random Forest (RF).

For that, we used the AdaboostM1, Bagging, J48, RandomTree LMT and RandomForest classifiers implemented in Weka workbench.

Clustering based Model From both the NTCIR14 visual concepts data and the Microsoft Vision API, some of the concept words have a confidence value. Therefore, to extract the words, only those whose confidence value is above a threshold are taken into account. For the words without confidence value, all are taken.

Then an undirected graph is constructed. A link between two words exists if the two words are describing the same segmentation. The value of the edge corresponds to the number of times the two words are encountered in the same segmentation.

Even if all the words are already concept words, because they come from two different sources, it is necessary to gather the variants of a same word (for instance "cell_phone" and "cellphone"). Then we apply some techniques that

have already been used discovering ambiguous words meaning [3]. To eliminate the weak links, we begin to delete the edges whose values are not above a certain threshold and we delete the edges that are not involved in a triangle. The latest technique is under the assumption that if an edge connects two words, these words belong to the same semantic cluster. So if another word is connected to at least one of the two previous words, if it also belongs to the same cluster, it should be connected to both first edges.

After eliminating edges, we weight the links according to the log-likelihood score: for two words v and w O_{ij} where $(i, j) \in 0, 1$ with i (j) representing the presence (1) or absence (0) of v (w). So for instance $O_1 1$ is the number of times v and w are in the same segmentation. After we calculate:

$$E_i j = \frac{O_i O_j}{N} \tag{1}$$

where $O_i = O_{i0} + O_{i1}$, $O_j = O_{0j} + O_{1j}$ and $N = \sum_{i,j} O_{ij}$ And finally the log-likelihood is given by:

$$L = 2 \sum_{(i,j)\in 0,1} O_{ij} log \frac{O_{ij}}{E_{ij}} \tag{2}$$

Finally, we use the Markov Cluster algorithm [12] to clusterise the words. Some of the clusters obtained are in Table 2. These clusters (around 30) are finally manually annotated with the statuses inside, outside or unknown.

Table 2. Some clusters' results

cluster1	bed, cat, reflection, looking, laying, sleeping, head	inside
cluster2	clouds, far-away horizon, open area, asphalt, transporting, pavement, biking	outside
cluster3	sunny, trees, foliage, vegetation, shrubbery, leaves, grass, green, tree	outside
cluster4	remote, playing, video, game, wii, control	outside

3.3 Insight Task 1: Inside or Outside Detection

This task is to recognize if the user is indoor or outdoor. To train the model, the status inside or outside has been obtained with Microsoft API tags: if there is the tag "indoor", the status is inside, if there is the tag "outdoor", the status is outside. Table 3 shows the results obtained with the original features for all the models. The best result obtained is with Adaboost+RT, 0.886 accuracy calculated with the formula:

$$accuracy = \frac{1}{N_{classes}} \sum_{i\in classes} \frac{C^i_{correct}}{C^i_{incorrect} + C^i_{correct}} \tag{3}$$

where $N_{classes}$ represents the number of classes and *classes* the classification classes. In this case, there are two classes (inside or outside). This equation has been chosen to calculate the accuracy because for these binary tests, one class has more samples than the other.

Table 3. Results for models trained with original features for outside inside

Adaboost		Bagging		RF
RT	C4.5	C.45	LMT	
0.886	0.880	0.724	0.873	0.83

For the visual features, we represented each picture as a vector of words. For each word, we looked if it belonged to a cluster and if it did, we created a new vector with the clusters' values corresponding to the words. The classification decision is done by looking at the majority in the values (inside, outside or unknown). If it is unknown, the result is automatically considered as incorrect, otherwise, the result is compared to the real value. The accuracy result obtained is 0.948 which is better than the one obtained with the original features.

3.4 Insight Task 2: Alone or Not?

Task 2 is to get insights about the user's surrounding: is the user alone or not? Not alone means that the user is at least surrounded with people even without interaction. The training statuses for the model have been extracted from MS Vision API tags: if there is a mention that there is more than the user in the picture (such as "people", "crowd" or "person"), then it means the user is not alone, otherwise the user is alone. Table 4 shows the results obtained with the original features for all the models. The best accuracy (calculated with the equation (3)) is obtained with Adaboost+RT, 0.742.

Table 4. Results for models trained with original features for alone or not alone

Adaboost		Bagging		RF
RT	C4.5	C.45	LMT	
0.742	0.564	0.555	0.692	0.70

3.5 Insight Task 3: Working or Not?

Task 3 is to recognize the moments when the user is working. The working status is labeled from tags recovered from MS Vision API: if the user is using a laptop

but not for leisure, then the status is considered working, otherwise the status is not working. Table 5 shows the results obtained with the original features for all the models. The best accuracy (calculated with the equation (3)) is obtained with Adaboost+RT, 0.803.

Table 5. Results for models trained with original features for working or not working

Adaboost		Bagging		RF
RT	C4.5	C.45	LMT	
0.802	0.698	0.686	0.795	0.791

For the three tasks, the combination Adaboost+Random Tree gives the best result. The fact that boosting has been designed to improve the accuracy of other ensemble algorithms by assigning weights to the ensemble samples might explain the fact that boosting algorithms give better results in our case. For the weak classifier, except for task 1 where both Adaboost results are really close, C4.5 as a weak classifier does not seem to work even with the Bagging algorithm.

4 Discussion on the Impact of Features

To see the impact of features for the tasks, for each task we train a model with the best algorithm according to each task -Adaboost+Rand Tree for the three of them- where we removed one feature at a time or only kept one feature. The results for task 1, task 2 and task 3 are respectively in Table 6, Table 7 and Table 8. We also looked at the correlation matrix for the features (Figure 1).

Table 6. Results for one remove feature or only one feature at a time for task 1

Removed or kept Feature	Removed Accuracy	Kept Accuracy
Baseline (all features)	0.886	0.886
UserID	0.881	0.5
Time	0.831	0.5002
Latitude	0.884	0.56
Longitude	0.883	0.582
City	0.883	0.5
Location	0.876	0.512
Activity	0.882	0.5
Steps	0.885	0.549
Calories	0.877	0.542
Heart Rate	0.865	0.5001

Table 7. Results for one remove feature or only one feature at a time for task 2

Removed or kept Feature	Removed Accuracy	Kept Accuracy
Baseline (all features)	0.742	0.742
UserID	0.742	0.5
Time	0.5786	0.5002
Latitude	0.734	0.528
Longitude	0.741	0.529
City	0.737	0.5
Location	0.733	0.499
Activity	0.739	0.5
Steps	0.74	0.499
Calories	0.742	0.499
Heart Rate	0.676	0.5

Table 8. Results for one remove feature or only one feature at a time for task 3

Removed or kept Feature	Removed Accuracy	Kept Accuracy
Baseline (all features)	0.802	0.802
UserID	0.801	0.5
Time	0.671	0.524
Latitude	0.801	0.525
Longitude	0.801	0.526
City	0.799	0.5
Location	0.793	0.587
Activity	0.799	0.5
Steps	0.801	0.5001
Calories	0.785	0.502
Heart Rate	0.760	0.501

For all the tasks, even if the features Latitude and Longitude seem important according to the accuracy for only one feature used, when removed, the result is very close to the normal result. This can be explained with the correlation matrix: both features are really close: -0.99 of correlation. The feature City is also close to the Longitude and Latitude features but without bringing any useful information. The users do not travel a lot between cities, which can explain the low impact of this feature.

Likewise the feature Activity does not add any valuable information for the three tasks. The reason might be the fact that this feature contains a lot of missing values and is according to the correlation matrix close to steps (correlation of 0.58).

However the feature Location, even if close to the features Latitude, Longitude and City seems relevant for tasks 1 and 3.

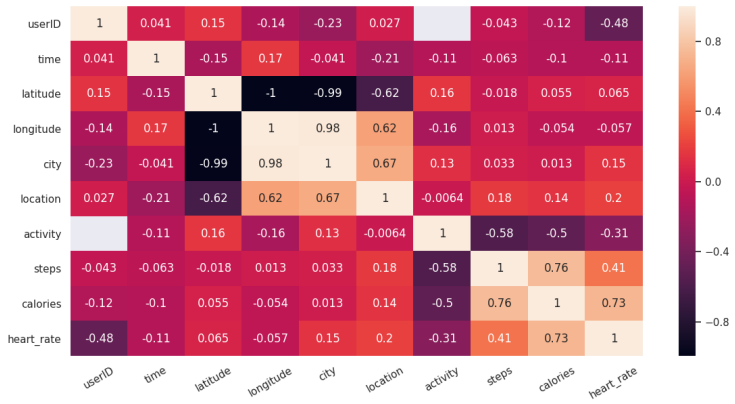

Fig. 1. Correlation Matrix (Pearson)

As well, the Time feature looks to be one of the most significant features for all the tasks, probably because one status depends a lot on the hour of the day: for instance, it is unlikely to begin to work before 8 a.m or after 7p.m.

The ID feature do not seem relevant either. The low number of users in the dataset and the similarity of statuses between the users (see Section 5) can be an explanation.

For task 1 the most important features appear to be Time, Steps, Latitude/Longitude, Location, Heart Rate and Calories. On the contrary, the features City, Activity and ID appear to be improper to this task.

For task 2, the main features are Time, Latitude/Longitude and Heart Rate and the less relevant are: ID, Activity, Steps and Calories.

Finally, for the Working/Not working task, the main features are Heart Rate, Time, Latitude/Longitude, Calories and Location and the minor ones are: ID and Activity.

When trying to remove the less significant features, we obtained the following results (Table 9). The results, even if close to the old ones, are not better unlike expected.

Table 9. Results for removed features based on the features' impact

Task	Features removed	New accuracy	Old accuracy
1	City, Activity, Latitude	0.882	0.886
2	Steps, Calories	0.709	0.742
3	ID, Activity	0.799	0.802

5 Statistics on User's statuses

Figures 2, 3 and 4 shows the variations of statuses per day for the three tasks. In total we have 34744 minutes of pictures for both users with an average of 2 pictures per minute for 42 days of data. On these 34744 minutes of pictures, we were only able to annotate 30631 of them according to task 1, 20309 from user 1 and 10322 from user 2.

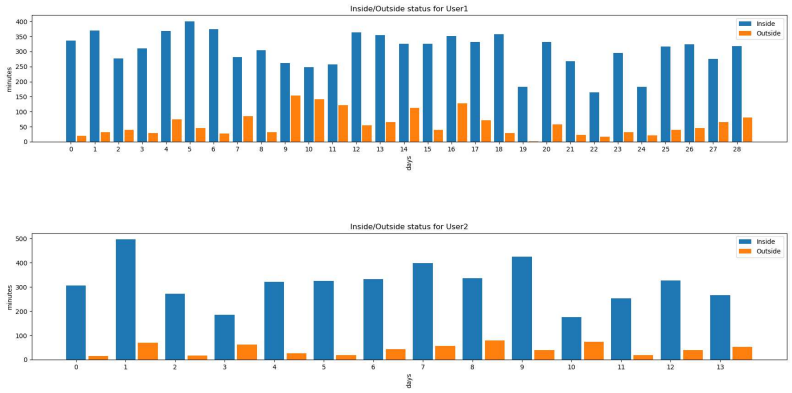

Fig. 2. Statistics for both users on task 1

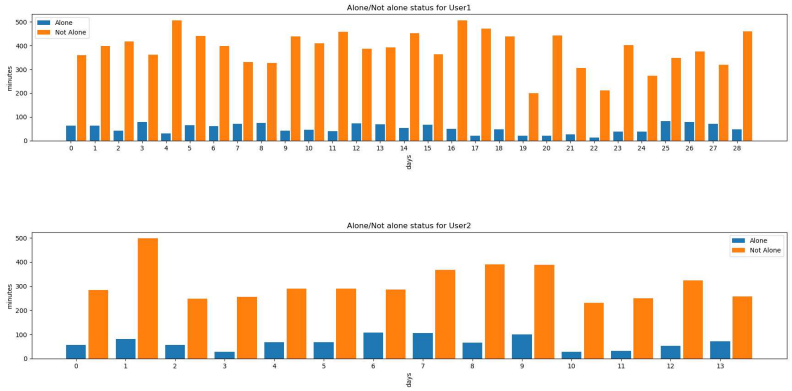

Fig. 3. Statistics for both users on task 2

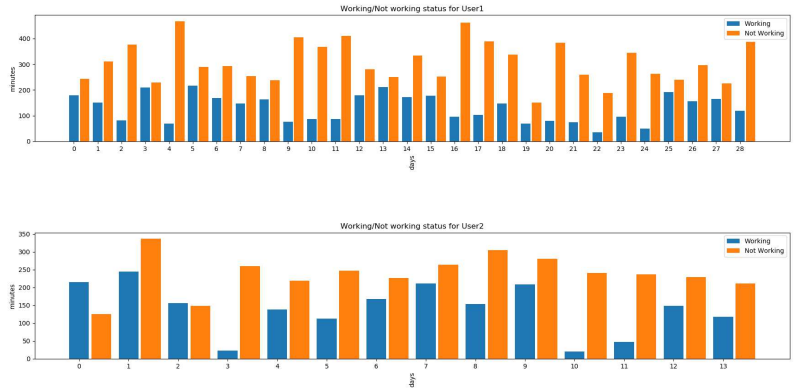

Fig. 4. Statistics for both users on task 3

User 1 spent 17716 minutes indoor and 2593 minutes outdoor in 29 days, so more than an average of 10 hours per day inside only for the moments when the user wore the camera (around 14 hours per day) for an average of 1h30 outside. User 2 on the other spent a total of 9582 minutes indoor and 740 minutes outdoor so an average of 11 hours indoor and less than an hour outdoor per day. Given that the users wear their cameras from the breakfast to the moment they go to bed, they spend only around 5% of their day outdoor.
For task 2, user 1 spent an average of 102 minutes per day alone and 12 hours not alone. For user 2, there is an average of 2 hours alone and 11 hours not alone. Therefore, through the day, both users spend most of their time not alone.

Finally, for task 3, for both users, the proportion through the day between working and not working moments are more balanced. User 1 spent around 4 hours per day working and 9h30 not working. User 2 spent an average of 4h40 working per day and 8 hours not working. Therefore, user 1 is working around 30% of the awake time and user 2 37% of the waking period (the awake time is consider as being the time when the camera is worn).

For the three tasks, both users have the same proportions of time for the statuses: 5% of the day outdoor, around 1h30 alone through the day and around 34% of waking period working. Besides, in figure 4, we can see periodic low working days (especially for user 2), which can indicate the week-ends. The same variations can be observed for task 2: it seems the users are less alone during the week-ends.

6 Conclusion and Future Work

We succeded in recognize users' statuses according to the three insight tasks. For the first task, the result obtained with the visual features is better than the one

obtained with the original features. Using the visual features for the recognition of tasks 2 and 3 could be a future work.

We enhanced the impact of features for each task by looking at the relation of the features between them and the relation of a particular feature with the task. However, we could not find a smaller and better dataset of features.

Finally, these tasks enabled us to make insights on the dailies statuses of the user. A potential future work is to predict these statuses.

References

1. Bush, V.: As we may think. The Atlantic Monthly (July 1945)
2. Doherty, A.R., Caprani, N., Conaire, C.O., Kalnikaite, V., Gurrin, C., Smeaton, A.F., O'Connor, N.: Passively recognising hman activities through lifelogging. Computers in Human Behavior (September 2011)
3. Dorow, B.: A Graph Model for Words and their Meanings. Master's thesis, Stuttgart University (March 2006)
4. Duane, A., Gupta, R., Zhou, L., Gurrin, C.: Visual insights from personal lifelogs. 12th NTCIR Conference (June 2016)
5. Galdos, L.C., Guillén, G.D., Alamo, C.L.D.: A new graph-based approach for document similarity using concepts of non-rigid shapes. IMMM 2017 (2017)
6. Gurrin, C., Joho, H., Hopfgartner, F., Dang-Nguyen, D.T., Zhou, L., Ninh, V.T., Le, T., Albatal, R., Healy, G.: Overview of the NTCIR-14 lifelog-3 task. Proceedings of the 14th NTCIR Conference on Evaluation of Information Access Technologies (2019)
7. Iijima, S., Sakai, T.: Slll at the NCTIR-12 Lifelog task: Sleepflower and the LIT subtask. 12th NTCIR Conference (June 2016)
8. Miller, G.A.: Wordnet: A lexical database for english. Communications of the ACM Vol. 38, No. 11: 39-41. (1995)
9. Soleimaninejadian, P., Yewen Wang, H.T., Feng, Z., Zhang, M., Liu, Y., Ma, S.: THIR2 at the NCTIR-13 Lifelog-2 task: Bridging technology ad pyschology through the Lifelog personality, mood and sleep quality. 13th NTCIR Conference (2018)
10. Soleimaninejadian, P., Zhang, M., Liu, Y., Ma, S.: Mood detection and prediction based on user daily activities. Asian Conference on Affective Computing and Intelligent Interaction (2018)
11. Sun, X., Xiao, Y., Wang, H., Wang, W.: On conceptual labeling of a bag of words. Proceedings of the Twenty-Fourth International Joint Conference on Artificial Intelligence (IJCAI 2015) (2015)
12. Van Dongen, S.: Graph Clustering by Flow Simulation. Ph.D. thesis, University of Utrecht (2000)
13. Wang, P., Smeaton, A.F.: Using visual lifelogs to automatically characterise everyday activities (January 2013)
14. Wu, W., Li, H., Wang, H., Zhu, K.Q.: Probase: A probabilistic taxonomy for text understanding (2012)
15. Xu, Q., Subbaraju, V., del Molino, A.G., Lin, J., Fang, F., lim, J.H.: Visualizing personal Lifelog data for deeper insights at the NTCIR-13 Lifelog-2 task. 13th NTCIR Conference (2018)
16. Y, K., K, K., C, B., JH, C.H.K.: Lifelog agent for human activity pattern analysis on health avatar platform. Healthc Inform Res (July 2014)

17. Yamauchi, K., Akiba, T.: Repeated event discovery from image sequences by ysing segmental dynamic time warping: Experiment at the NTCIR-12 Lifelog task. 12th NTCIR Conference (June 2016)