Energy-Efficient Eating Detection Using a Wristband

Simon Stankoski
Department of Intelligent Systems
Jožef Stefan Institute
Jožef Stefan International
Postgraduate School
Ljubljana, Slovenia
simon.stankoski@ijs.si

Mitja Luštrek
Department of Intelligent Systems
Jožef Stefan Institute
Jožef Stefan International
Postgraduate School
Ljubljana, Slovenia
mitja.lustrek@ijs.si

ABSTRACT

Understanding people's dietary habits plays a crucial role in interventions that promote a healthy lifestyle. For this purpose, a multitude of studies explored automatic eating detection with various sensors. Despite progress over the years, most proposed approaches are not suitable for implementation on embedded devices. The purpose of this paper is to describe a method that uses a wristband configuration of sensors to continuously track wrist motion throughout the day and detect periods of eating automatically. The proposed method uses an energy-efficient approach for activation of a machine learning model, based on a specific trigger. The method was evaluated on data recorded from 10 subjects during free-living. The results showed a precision of 0.84 and a recall of 0.75. Additionally, our analysis shows that by using the trigger, the usage of the machine learning model can be reduced by 80%.

KEYWORDS

Eating detection, wristband, energy efficient, activity recognition

1 INTRODUCTION

Understanding people's dietary habits plays a crucial role in interventions that promote a healthy lifestyle. Obesity, which is a consequence of bad nutritional habits and excessive energy intake, can be a major cause of cardiovascular diseases, diabetes or hypertension. Latest statistics indicate that obesity prevalence has increased substantially over the last three decades [1]. More than 600 million adults (13% of the total adult population) were classified as obese in 2014 [2]. In addition, the prevalence of obesity is estimated to be 23% in the European Region by 2025. Also, in 2017, it was reported that poor diet has contributed to 11 million deaths worldwide. Monitoring eating habits of overweight people is an essential step towards improving nutritional habits and weight management.

Another group of people that require monitoring of their eating behavior are people with mild cognitive impairment and dementia. They often forget whether they have already eaten and, as a result, eat lunch or dinner multiple times a day or not at all. It might cause additional health problems. Proper treatment of these issues requires an objective estimation of the time the meal takes place, the duration of the meal, and what the individual eats.

Wristband devices and smartwatches are increasingly popular, mainly because people are accustomed to wearing watches, which makes the wrist placement one of the least intrusive body placements to wear a device. Additionally, the cost of these devices is relatively low, which makes them easily accessible to everyone. However, these devices offer limited computing power and battery life, which makes the implementation of a smart feature as eating detection on such a device a challenging task.

This paper describes a method for real-time eating detection using a wristband. The proposed method detects periods and duration of eating. The output from the method can be used to track frequency of eating and could serve to start methods for counting food intakes.

The work done in this study is important for the following reasons. We developed a trigger that can reduce the usage of the machine learning procedure, meaning that our method will not greatly affect the battery life of the device. Additionally, we evaluated different machine learning algorithms in terms of accuracy and model size. The method was evaluated on data recorded in real-life from 10 subjects.

2 RELATED WORK

Recent advancements in wearable sensing technology (e.g., commercial inertial sensors, fitness bands, and smartwatches) have allowed researchers and practitioners to utilize different types of wearable sensors to assess dietary intake and eating behavior in both laboratory and free-living conditions. A multitude of studies for the detection of eating periods have been proposed in the past decade. Mirtchou et al. [3] explored eating detection using several sensors and combining real-life and laboratory data. Edison et al. [4] proposed a method that recognizes intake gestures separately, and later clusters the intake gestures within 60-minute intervals. The method was evaluated on real-life data. Dong et al. [5] proposed a method for eating detection in real-life situations based on a novel idea

that meals tend to be preceded and succeeded by periods of vigorous wrist motion. Amft et al. [6] presented an accurate method for eating and drinking detection using sensors attached to the wrist and upper arm on both hands. Navarathna et al. [7] combined sensor data from a smartwatch and a smartphone, which resulted in improved eating detection accuracy compared to only using smartwatch data. Kyritsis et al. [8] proposed a deep learning based method that recognizes bite segments, which are used for construction of eating periods.

The work presented in this paper is an extension of our previous work [9], and the main novelty is an energy efficient approach for real-time eating detection.

3 METHOD

The proposed eating detection method consists of two parts, namely: a threshold-based trigger, used for activation of an eating detection machine learning procedure, and a machine-learning method that predicts whether eating took place.

3.1 Energy-Efficient Trigger

The recent advancements in the technological development and accessibility of wearable devices bring new opportunities in the field of human activity recognition (HAR). However, the limited battery life and computational resources remain a challenge for real-life implementation of advanced HAR applications. Using a machine learning based model for eating detection that is working all the time results in a rapid battery drain. Therefore, we designed a threshold-based trigger that activates the machine learning model only when specific criteria are met. The main concept behind the trigger is to only select moments when the human is making a movement with his hand towards the head.

For this purpose, we used data from an accelerometer. This sensor provides information about the wristband's orientation from which we can see whether the hand is oriented towards the head. The recent accelerometers that are used in battery-limited devices can store acceleration values in their internal memory without interacting with the main chip of the microcontroller.

The first step of trigger implementation is to define the buffer size in the sensor's internal memory and the sensor's sampling frequency. Based on these two parameters, we enable the accelerometer to collect data for a specific time without interacting with the main chip of the microcontroller. This means that the main chip of the microcontroller could be in sleep mode for the predefined period. When the accelerometer's buffer is full, the accelerometer interrupts the main chip and transfers the stored acceleration data to it. We use the accelerometer's y-axis and z-axis to detect moments when the individual is moving the hand towards the face. Namely, we calculate the mean value for both axes, and if both of the values are above a predefined threshold value, the machine learning procedure for eating detection is activated. We used two axes for the trigger to reduce the possible situations in which our trigger is falsely activated. However, one can work only with one axis, which will result in more activated triggers. We could say that having more activated triggers is not desirable. However, if the eating detection method is not good enough to

detect eating after a trigger is activated during a meal, then the constraints of the trigger should be reduced.

The next step is the definition of stopping criteria for the machine learning model. The idea here is to stop the machine learning procedure after a specific number of windows if there is no eating detected. Each time our trigger is activated, the machine learning procedure is turned on for the next three buffers of data. The machine learning procedure is stopped if there is no positive prediction in any of the three windows. However, if there is at least one positive prediction, the machine learning procedure continues to work for another three new buffers. Also, the number of windows for which the machine learning procedure is active was experimentally obtained.

3.2 Machine-Learning Procedure

A detailed description of the used method can be seen in [9]. The method is based on machine learning and consists of the following steps: filtering the accelerometer and gyroscope data coming from the wristband, segmentation of the filtered data, feature extraction, feature selection, two stages of model training and predictions smoothing.

In the first step, the raw data were filtered with a 5th order median filter to reduce noise. Furthermore, the median filtered data was additionally filtered with low-pass and band-pass filters. Hence, we ended up with three different streams of data, median, low-pass and band-pass filtered data.

The accelerometer and gyroscope data were segmented using a sliding window of 15 seconds with a 3-second overlap between consecutive windows. This means that once we have 15 seconds of data, the buffer is adjusted to only store 3 seconds of new data. After that, each time the buffer is full, we add the new 3 seconds of data to the previous 15 seconds window and we drop the oldest 3 seconds from it. The reason for the length of the window is that it needs to contain an entire food intake gesture [10].

After the segmentation step, we extracted three different groups of features. Also, we included a feature selection step to improve the computational efficiency of the method, to remove the features that did not contribute to the accuracy and to reduce the odds of overfitting.

The training procedure for the method used in this study consists of three stages. The first two aim at training an eating-detection models on an appropriate amount of representative eating and non-eating data. The third step smooths the predictions of the model.

4 DATASET AND EXPERIMENTAL SETUP

For this study, we recorded data from 10 subjects (8 male and 2 female), ranging in age from 20 to 41 years. The data were recorded using a commercial smartwatch Mobvoi TicWatch S running WearOS, providing 3–axis accelerometer and 3–axis gyroscope data sampled at 100 Hz. The technical description of the sensors from the smartwatch shows that the recorded data is compatible with our target wristband for which we are developing our eating detection method. Additionally, the use of a commercially available smartwatch was an easier option for recording data. The collected dataset contains recordings

from usual daily activities performed by the subjects, including eating. The subjects were wearing the smartwatch on their dominant hand while recording. The smartwatch had an application installed on it, which enabled them to label the beginning and the end of each meal. There were no limitations about the type of meals the subjects could have while recording, which resulted in having 70 different meals included in the dataset. Furthermore, the subjects were also asked to act naturally while having their meals, meaning talking, gesticulating, using the smartphone, etc. The total data duration is 161 hours and 18 minutes, out of which 8 hours and 19 minutes correspond to eating activities.

For evaluation, the LOSO cross-validation technique was used. In other words, the models were trained on the whole dataset except for one subject on which we later tested the performance. The same procedure was repeated for each subject in the dataset. The results obtained using this evaluation technique are more reliable compared to approaches where the same subject's data is used for both training and testing, which show excessively optimistic results.

As mentioned before, smartwatches offer limited resources, one of which is the size of the RAM memory. Therefore, we analyzed models with different sizes to see whether the bigger and more complex models provide higher accuracy. We tested the performance of four different machine learning algorithms, Random Forest [11], Decision Tree [12], Logistic Regression [13] and LinearSVC [14].

We analyzed the following evaluation metrics: recall, precision and F1 score. These evaluation metrics are the most commonly used metrics for classification tasks like ours and give a realistic estimate of the efficacy of the algorithm. Also, the final results were obtained from the whole recordings by each subject. The reason for this is mainly to give a real picture of how good the developed method is in real-life settings.

5 RESULTS

The primary use of the trigger is to reduce the activity of the machine learning procedure. However, for the efficiency of the trigger, a very important requirement is when and how often the trigger is activated during a meal. In order to achieve accurate predictions, we want the trigger to be activated as soon as the meal is started. Additionally, the percentage of activated triggers during a meal should be bigger compared to noneating segments. For this purpose, we explored which window size works best with our trigger. Table 1 shows the results achieved in the conducted experiments. We tested two different window sizes with two slide values for each window, resulting in a total of four combinations.

Table 1: Different window size for the trigger procedure.

Window and slide size	Trigger activation time	% of activated triggers	Meals detected
3 - 1	36 s	34.2	68/70
3 - 3	41 s	32.6	68/70
15 - 3	48 s	42.0	55/70
15 - 5	41 s	42.0	54/70

Table 2: Results of eating detection procedure achieved with different algorithms and their model size.

Algorithm	Precision	Recall	F1 score	Model size
Random Forest	0.84	0.75	0.79	36339 KB
Logistic Regression	0.70	0.71	0.70	1.25 KB
LinearSVC	0.69	0.71	0.70	1.8 KB
Decision Tree	0.59	0.65	0.62	175 KB

The used combinations for the window and slide size are shown in the first column of the table. The second column shows the average time needed for the trigger to be activated for the first time after a meal is started. The third column shows the average percentage of triggered windows during a meal. These two columns were used as a metric for selecting the optimal size of a window and slide between the windows. The last column shows the number of meals when the trigger was activated. The values for the second and third columns were obtained only from the meals for which the trigger was activated. Row-wise comparison between these two columns shows the results obtained with each different combination of a window and slide. We can see that the most optimal combination regarding the average time needed for a trigger to be activated after a meal is started is a window size of 3 seconds with a slide of 1 second between two windows. Therefore, in our further analysis, we used this combination. The optimal window size of 3 seconds is expected if we have in mind that the usual intake gesture lasts around 2 seconds. Longer windows fail to detect the gesture while having a meal because usually we have two or three intakes in 15 seconds and the mean value over the whole window is low.

Table 2 shows the final results obtained using the whole method described in Section 2. Row-wise comparison between the used evaluation metrics shows the results obtained using the different algorithms shown in the first column. Additionally, the last column of the table represents the final model size. We can clearly see that the results achieved with Random Forest are better than the remaining algorithms. However, if we compare the model size of the best performing algorithm with the remaining algorithms we can say that the results achieved using Logistic Regression and LinearSVC are acceptable. Additionally, the precision value of 0.84 shows that the combination of trigger and machine learning procedure can differentiate between eating and noneating segments. However, the recall value of 0.75 suggests that a more accurate method regarding the eating periods is needed.

We also analyzed how much time each of the previously described algorithms was active during the noneating period. The results from this experiment are shown in Table 3. Additionally, in this table we can see the false positive rate during the noneating period. The best results are achieved using a Random Forest classifier, which is active only 20% of the whole noneating period. This means that our trigger-based procedure reduces the usage of the machine-learning procedure for 80%. However, this number also depends on the detection method because once it is activated, the eating predictions extend the active time of the method.

Table 3: Comparison of active time and false positive rate of the machine learning algorithms during noneating period.

Algorithm	Active time during noneating period	False positive rate
Random Forest	20%	1.36%
Logistic Regression	22%	2.18%
LinearSVC	22%	2.34%
Decision Tree	23%	3.93%

6 CONCLUSION AND FUTURE WORK

In this paper, we presented a method that can accurately detect eating moments using a 3-axis accelerometer and gyroscope sensor data. Our method consists of an energy-efficient trigger and a machine-learning procedure, which is started only after the trigger is activated. We evaluated this method using a dataset of 70 meals from 10 subjects. The results from the LOSO evaluation showed that we are able to recognize eating with a precision of 0.84 and recall of 0.75.

The presented results are important because both the training and the evaluation data were recorded in uncontrolled real-life conditions. We want to emphasize the real-life evaluation since it shows the robustness of the method while dealing with plenty of different activities that might be mistaken for eating as well as recognizing meals that were recorded in many different environments while using many different utensils. The proposed method can also deal with interruptions while having a meal, such as having a conversation, using the smartphone, etc. Additionally, we believe that the energy efficiency of the proposed method is very important. The proposed technique uses a trigger to activate the machine learning procedure and it is able to reduce the active time of the machine learning procedure for almost 80%. If we have in mind that the wristbands are devices with limited resources, we could say that even small reductions in resource usage can be significant for longer battery life.

The initial results achieved in this study are encouraging for further work in which we expect to improve the eating detection method. In the near future, we plan to optimize our machine learning procedure to detect eating periods more accurately once the trigger is activated. Furthermore, we want to overcome the problem with false positives predictions. For this problem, we believe that a more sophisticated method for selecting representative noneating data will help to recognize the problematic activities and directly include them in the training data. Also, we plan to investigate personalized threshold values. We believe that personalized values for the threshold will help

to activate the trigger during eating periods more easily. Additionally, this could reduce the activation of the machine-learning procedure during non-eating periods. Also, we plan to explore memory efficient methods for storing the models in memory.

REFERENCES

[1] World Health Organization. World Health Statistics 2015. Luxembourg, WHO, 2015

[2] Public Health England. Data Factsheet: Adult Obesity International Comparisons. London, 2016. http://webarchive.nationalarchives.gov.uk/20170110165728/http://www.noo.org.uk/NOO_pub/Key_data

[3] M. Mirtchouk, D. Lustig, A. Smith, I. Ching, M. Zheng, and S. Kleinberg. Recognizing eating from body-worn sensors: Combining freeliving and laboratory data. Proc. ACM Interact. Mob. Wearable Ubiquitous Technol., 1(3):85:1–85:20, Sept. 2017

[4] E. Thomaz, I. Essa, and G. D. Abowd. A practical approach for recognizing eating moments with wrist-mounted inertial sensing. In Proceedings of the 2015 ACM International Joint Conference on Pervasive and Ubiquitous Computing, UbiComp '15, pages 1029–1040, New York, NY, USA, 2015. ACM.

[5] Y. Dong, J. L. Scisco, M. Wilson, E. Muth, and A. W. Hoover. Detecting periods of eating during free-living by tracking wrist motion. IEEE Journal of Biomedical and Health Informatics, 18:1253–1260, 2014

[6] O. Amft, H. Junker, and G. Troster. Detection of eating and drinking arm gestures using inertial body-worn sensors. In Ninth IEEE International Symposium on Wearable Computers (ISWC'05), pages 160–163. IEEE, 2005.

[7] P. Navarathna, B. W. Bequette, and F. Cameron, "Wearable Device Based Activity Recognition and Prediction for Improved Feedforward Control," in Proceedings of the American Control Conference, 2018, doi: 10.23919/ACC.2018.8430775.

[8] K. Kyritsis, C. Diou, and A. Delopoulos, "Detecting Meals in the Wild Using the Inertial Data of a Typical Smartwatch," in Proceedings of the Annual International Conference of the IEEE Engineering in Medicine and Biology Society, EMBS, 2019, doi: 10.1109/EMBC.2019.8857275

[9] Stankoski S, Resçiç N, Mezic G, Lustrek M. Real-time Eating Detection Using a Smartwatch. InEWSN 2020 Feb 17 (pp. 247-252).

[10] Xu Ye, Guanling Chen, and Yu Cao. Automatic eating detection using head-mount and wrist-worn accelerometers. In 2015 17th International Conference on E-health Networking, Application Services (HealthCom), pages 578–581, Oct 2015.

[11] T. K. Ho. Random decision forests. In Proceedings of 3rd international conference on document analysis and recognition, volume 1, pages 278–282. IEEE, 1995.

[12] P. H. Swain and H. Hauska, "The decision tree classifier: Design and potential," in IEEE Transactions on Geoscience Electronics, vol. 15, no. 3, pp. 142-147, July 1977, doi: 10.1109/TGE.1977.6498972.

[13] Lee, Youngjo, John A. Nelder, and Yudi Pawitan. Generalized linear models with random effects: unified analysis via H-likelihood. Vol. 153. CRC Press, 2018.

[14] Chang, Chih-Chung, and Chih-Jen Lin. "LIBSVM: A library for support vector machines." ACM transactions on intelligent systems and technology (TIST) 2.3 (2011): 1-27.