Reptile learned Transformer or normal Transformer on Sensor-based Human Activity Recognition Problem

Zheng Qiwen [1]

[1]z5240149

August 2021

1 Abstract

Meta learning and transformer are two hot trends in deep learning area. In this research project, I combine simplified transformer with the reptile meta-learning method on Human Activity Recognition (HAR) problem, aiming to create a benchmark for four datasets and to show that positional encoding of the transformer is not always beneficial for test accuracy rate and reptile can generally boost the model performance, compared to normal back-propagation training schema.

2 Research Background

Recently, the vast increment of sensor devices have enabled the applications of sensor-based activity recognition, which make myriad applications such as smart homes, healthcare, and enhanced manufacturing become possible. Activity recognition is essential to humanity since it records people's behaviors with data that allows computing systems to monitor, analyze, and assist their daily life.

Considering the privacy issues of installing cameras in our personal space, sensor-based systems have dominated the applications of monitoring our daily activities, compared with video-based systems.

This field still faces many technical challenges, such as feature extraction, annotation scarcity and class imbalance. Many machine learning methods have been employed in human activity recognition. Recent years deep learning has embraced great prosperity in modeling high-level abstractions from intricate data in many area, such as computer vision and natural language processing. So it is inevitable development of deep learning in human activity recognition. Though deep learning is still confronted with reluctant acceptance by researchers owing to its abrupt success, bustling innovation, and lack of theoretical support, many deep learning models can works well and can achieve state-of-the-art in many datasets. Also, advanced computing resources like GPUs provide deep models with a powerful ability to learn features from more complex data and train the model faster. So in the project I choose to use a recently widely adopted deep learning model - transformer to see if it can achieve good performance on sensor-based HAR problem.

The dataset is always important for machine learning problems, for the research of HAR, many facilities and volunteers have contributed to create excellent open-source datasets. In this section, I would like to briefly mention some of them for the potential use of the whole research process, as well as how the datasets are used in the experiment.

MHEALTH[1]: The collected dataset comprises body motion and vital signs recordings for ten volunteers of diverse profile while performing 12 physical activities. Shimmer2 [BUR10] wearable sensors were used for the recordings. The sensors were respectively placed on the subject's chest, right wrist and left ankle and attached by using elastic straps. All sensing modalities are recorded at a sampling rate of 50 Hz, which is considered sufficient for capturing human activity.Each time the once data contains 24 column, corresponding to acceleration , electrocardiogram, gyro, magnetomete of different body part sensors with different axises(X, Y, Z). The last column is the label of the activity, ranging from 0-12 with 0 as the null class.

The data is cropped in the experiment with 10 people in total, each person has 18432 data records and each data record has 23 features that correspond to normalized data collected by sensors. Also the data is evenly distributed of activities across users.

PAMAP2[2, 3]: The dataset is collected by 3 Colibri wireless IMUs (inertial measurement units), with 100Hz smiling frequency. The position of three sensors is 1 over the wrist on the dominant arm, 1 on the chest, 1 on the dominant side's ankle. 9 subjects participated in the data collection, each of them had to follow a protocol, containing 18 different activities. Each of the data-files contains 54 column per row, with timestamp, activityID, heart rate, IMU hand, IMU chest, IMU ankle.

A cropped data set of PMP2 has been used that has 8 people with different number of data per person, from 111787 to 171277. Further each person performs irregular different distributed activities.

MARS: I cannot find information on Internet about this dataset, but after I analyse the data, it has 8 people in total and each person has done 5 activities with equally 50000 records per person and 19 features (except the user label and activity type) per data. Plus, the people perform equally distributed five activities.

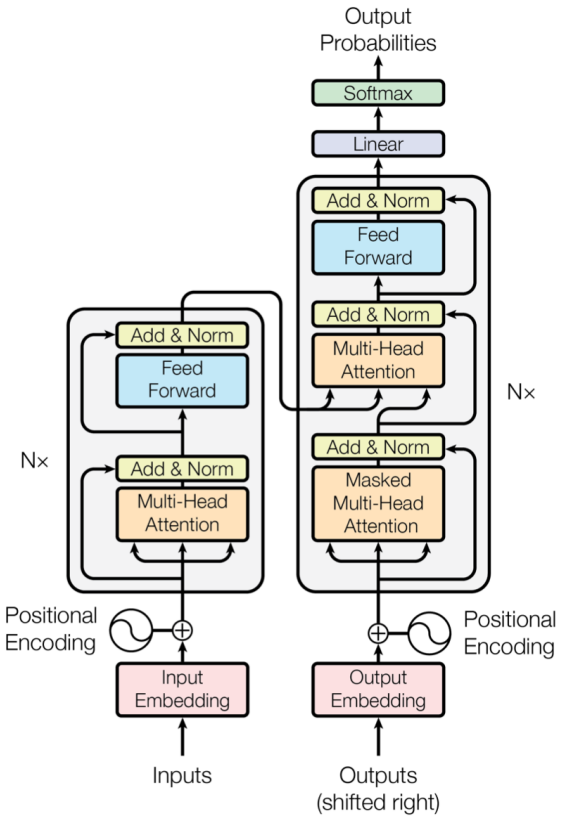
UCI HAR[4]: The experiments have been carried out with a group of 30 volunteers within an age bracket of 19-48 years. Each person performed six activities (WALKING, WALKING_UPSTAIRS, WALKING_DOWNSTAIRS, SITTING, STANDING, LAYING) wearing a smartphone (Samsung Galaxy S II) on the waist. Using its embedded accelerometer and gyroscope, we captured 3-axial linear acceleration and 3-axial angular velocity at a constant rate of 50Hz. The experiments have been video-recorded to label the data manually.

In the project, I used a cropped data that still has 30 users in total, 20 of which be the training set and 10 left be the test set, but there are only 9 features per data.

In this project, I refer the previous work from [5], which propose a multi-agent spatial-temporal attention model that uses CNN and LSTM to achieve high test rate on four datasets. I try another trending deep learning model - Transformer to compete with it, as well as creating a benchmark for further research.

3 Transformer Review

First proposed in [6], transformer is a neural network model that has proven to be especially effective for common natural language processing tasks. It uses the self attention-mechanism that looks at an input sequence and decides at each step which other parts of the sequence are important. Like LSTM, Transformer is an architecture for transforming one sequence into another one with the help of two parts (Encoder and Decoder), but it is different from the models because it does not contain any Recurrent Networks (GRU, LSTM, etc.). The general classical architecture of Transformer is shown as below:



Each encoder block is mainly composed of a multi-head self-attention module and a position-wise feed-forward network (FFN). Compared to the encoder blocks, decoder blocks additionally insert cross-attention modules between the multi-head self-attention modules and the position-wise FFNs.

Positional Encoding is a sinusoidal (sin/cos) function of the index with pre-defined frequency to leverage the order of the input sequence. The necessity of positional encoding on HAR problem is a main target in the project.

As transformer is designed to tackle sequence to sequence problem, it is applicable to HAR area as human activity is also a sequential data stream that can be viewed as parallel, what is different is that HAR data collected by sensor is usually continuous value of multi-dimension, so we don't need to do embedding to project it to a vector of multi-dimension. The input data would be passed through a double linear layers and then goes to either positional encoding or not.

For simplicity and effectiveness, I only use the encoding part, which is the left side of the transformer to deal with the HAR classification problem. Also in this experiment I only use two or three stacks of Nx, because more stacks can cause severe over-fitting and large amount of time to train. After that the output passes through linear layers and finally softmax layer to get the probabilities of the activities prediction.

4 Meta Learning Review

As traditional deep learning progressed in the last decades, it can achieve great successes when presented with large data sets and sufficient computational resources across various domains, including image recognition, speech recognition, even game playing. But the restrictions of large amount of data constrain the ability of deep neural networks to learn new concepts quickly, which is one of the defining aspects of human intelligence[7].

The field of Deep Meta-Learning advances at great speed, as one strategy to overcome this challenge. The normal way of the deep supervised learning aims to find a set of optimal parameters that minimize the a loss function on training set, whereas supervised meta-learning tends to find the best parameter set, such that the base-learner can learn new tasks (data sets) as quickly as possible.

The meta-learning is identified as three categories of Deep Meta-Learning approaches: i) metric-based, ii) model-based, and iii) optimization-based meta-learning techniques.[8] In the project, we mainly focus on optimization-based meta-learning.

Optimization-based techniques explicitly optimize for fast learning. It adopts bi-level method that the inner-level makes task-specific updates using some optimization strategy (such as gradient descent) and the outer-level optimizes the performance across tasks. Model-agnostic meta-learning (MAML)[9] uses a simple gradient-based inner optimization procedure (e.g. stochastic gradient descent) that can learn a good set of initialization parameters. MAML has obtained great attention within the field of Deep Meta-Learning due to its i) simplicity (only requires two hyperparameters), ii) general applicability, and iii) strong performance. But it can be quite expensive in terms of running time and memory to optimize a base-learner, because it computes higher-order derivatives from the optimization trajectories. To tackle the problem, Reptile[10] uses only first-order derivatives for the meta-learning updates that repeatedly samples a task, trains on the task, and moves the model weights towards the trained weights. It is an simple meta-learning technique that saves time and memory, also generate compatible models compared to MAML.

In the experiment, I use the reptile method to have a test on the effect on the HAR problem, aiming to show that reptile trained way can generally have a good performance when deal with high dimension feature data. Specifically, the task in the HAR problem refers to the data per user.

5 Experiment Setup

The experiment involves reptile trained and normal trained way of Transformer encoder part with and without Positional Encoding on four datasets (MHealth, PMP2, MARS and UCI). The models are implemented using pytorch with python 3.7 and the source code is on https://github.com/gakkistyle/Research-Project_o/Research-Project, where traditional

trained method is in tradition_lopo.py and reptile trained is in meta_reptile_lopo.py. All experiments are carried out on Tesla P100-PCIE-16GB in Google Colab. The specific parameters (optimizer, learning rates, epoch, etc) setup are included in the comment session of the code. Leave-one-person-out strategy is used that each time one user's data is chosen to be test data and others' be the training data. The exception is for UCI dataset that there are predefined training dataset (20 people) and test set (10 people), so we only train the models based on the training set, and test on the test set.

Also randomly pick 30 percent data in training users to be validation set, after each epoch accuracy test on whole validation set is recorded and a bunch of candidate models during the training steps are saved and would be chosen to do testing in the user-test phase. When test on unseen user's data, I record the highest accuracy rate among models and record the average highest accuracy and std among all the test users, which implies the upper bound of the model performance and its stability among different users. The test phase is different for the reptile trained model, the evaluation on test users is postponed, instead, we do a one-shot learning that the chosen models would do adaptation on the new data of the test users by randomly picking one data per class and letting the model do gradient decent to adapt to the mode of the new user. Then we can test on the whole test set and pick the best performance as the normal trained way.

6 Experiment Result

The results are compared with normal trained multi-agent spatial-temporal attention model[5], which was the state-of-the-art model in 2019 on the four datasets.

Acc	Tra-NPE	Tra-PE	Rep-NPE	Rep-PE	Mul-Agent
MH	91.23±2.16	91.45±1.62	92.78±3.35	93.06±2.81	**96.12±0.37**
PMP	88.32±6.06	89.84±8.61	93.06±2.85	**93.38±2.58**	90.33±0.62
MARS	82.31±3.47	79.4±3.19	85.17±2.45	83.79±6.36	**88.29±0.87**
UCI	81.27±11.54	72.98±13.2	70.42±0.362	50.48±0.28	**85.72±0.83**

The result shows that for most datasets, the multi-agent model still contributes the best performance, though the Reptile trained transformer can beat it on the PMP dataset. Also Reptile trained models can generate a better average performance on three datasets (PMP,PMP and MARS), compared with normal way trained transformer, with a higher average test accuracy, but it can not guarantee the reducement of the variance between users, as the results of the MH dataset imply. Plus, reptile models can consume much computation power and time, for example, the reptile with PE runs 53717 seconds whereas it only takes 8982 seconds for the normal trained PE model to get the results, nearly 6 times the gap. However, positional encoding part is not always suitable. For MHealth and PMP2 dataset, the model with PE generally outperforms the No-PE, while this is the opposite situation for MARS and UCI dataset, when without PE always gets higher test rates on unseen users' data. In terms of time consumption, however, PE part does not take much time (no PE: 8889 seconds, PE: 8982 seconds for the PMP dataset).

Reptile tends to over-fitting when features are less, as the case for the UCI with only 9 features per class, compared to other data sets, which have 23, 52, 19 features respectively.

So when the number of features is small (below 10), it is better to use normal trained no PE transformer, which can generate a much higher accuracy compared to other ways. In other words, when features are complex, as the case of MH and PMP suggest, reptile trained PE transformer can output a promising model. However, the distribution of different activities across users has less effect, as the results of the MARS (19 features, uniform distributed) and PMP (52 features, irregular distributed) show, PMP has much higher test accuracy, as well as not much higher variance (2.58 compared to 2.45). So the number of features collected by sensor can be essential part for HAR problem.

7 Conclusion

The project creates a benchmark for simplified transformer combined with normal or reptile way of training in four datasets. The work presents a experimental result to show the effectiveness of positional encoding and reptile trained schema. From the perspective of the test on four datasets, the positional encoding is redundant when feature number is few and can drop the test performance to some extent, so it is worthwhile to try without PE when use transformer. Reptile meta learning can boost the performance but require more computation power and features as well, otherwise it can produce even more over-fitting models.

The number of features can determine the final performance of the models, it is intuitive to get the conclusion since the more available sensor data from different position of the human body, the more convinced that what kind of activities the person is doing. But in reality sensor data can be difficult to obtain and it is uncomfortable for the human to carry sensors in numbers of position on body, so it is also important to design a light weight sensor to collect body data from more aspects.

Further research can be based on the simplified transformer to create other variants, try more complex connection between layers and leverage more advanced GPU to train the model. Also meta-learning is also a good trending in AI area, as there are much more collected data nowadays, the ability to learn from the data and try adaptation on a few new data is essential to make the model simpler as well as achieve high accuracy.

References

[1] Oresti Banos, Rafael Garcia, Juan A. Holgado-Terriza, Miguel Damas, Hector Pomares, Ignacio Rojas, Alejandro Saez, and Claudia Villalonga. mhealthdroid: A novel framework for agile development of mobile health applications. In Leandro Pecchia, Liming Luke Chen, Chris Nugent, and José Bravo, editors, *Ambient Assisted Living and Daily Activities*, pages 91–98, Cham, 2014. Springer International Publishing.

[2] Attila Reiss and Didier Stricker. Introducing a new benchmarked dataset for activity monitoring. 06 2012.

[3] Attila Reiss and Didier Stricker. Creating and benchmarking a new dataset for physical activity monitoring. 06 2012.

[4] Davide Anguita, Alessandro Ghio, Luca Oneto, Xavier Parra, and J Reyes-Ortiz. A public domain dataset for human activity recognition using smartphones. 01 2013.

[5] Kaixuan Chen, Lina Yao, Dalin Zhang, Bin Guo, and Zhiwen Yu. Multi-agent attentional activity recognition, 2019.

[6] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, and Illia Polosukhin. Attention is all you need, 2017.

[7] Brenden M. Lake, Tomer D. Ullman, Joshua B. Tenenbaum, and Samuel J. Gershman. Building machines that learn and think like people. *Behavioral and Brain Sciences*, 40:e253, 2017.

[8] Mike Huisman, Jan N. van Rijn, and Aske Plaat. A survey of deep meta-learning. *Artificial Intelligence Review*, 54(6):4483–4541, Apr 2021.

[9] Chelsea Finn, Pieter Abbeel, and Sergey Levine. Model-agnostic meta-learning for fast adaptation of deep networks. *CoRR*, abs/1703.03400, 2017.

[10] Alex Nichol, Joshua Achiam, and John Schulman. On first-order meta-learning algorithms, 2018.