IJNRD.ORG ISSN : 2456-4184
INTERNATIONAL JOURNAL OF NOVEL RESEARCH AND DEVELOPMENT (IJNRD) | IJNRD.ORG
An International Open Access, Peer-reviewed, Refereed Journal

MEMORY AWARE ACTIVE LEARNING SENSOR FOR SYSTEM SERVER MONITORING

Gayathri S, Sangeetha P
Department of Computer Science and Engineering, JKK Munirajah College of Technology,
Erode, Tamilnadu - 638 506, India

Abstract: I propose a novel active learning framework for activity recognition and server monitoring. My work is unique in that it takes limitations of the oracle into account when selecting sensor data for annotation by the oracle. This capacity constraint is manifested not only in the number of queries that a person can respond to in a given time-frame but also in the time lag between the query issuance and the oracle response. I introduce the notion of mindful active learning and propose a computational framework, called EMMA, to maximize the active learning performance taking informativeness of system data, query budget, and human memory into account. I formulate this optimization problem, propose an approach to model memory retention, discuss the complexity of the problem, and propose a greedy heuristic to solve the optimization problem.

I design an approach to perform mindful active learning in batch where multiple system observations are selected simultaneously for querying the oracle. I demonstrate the effectiveness of our approach using three publicly available activity datasets and by simulating oracles with various memory strengths. I show that the activity recognition accuracy ranges from 21% to 97% depending on memory strength, query budget, and difficulty of the machine learning task. Moreover, I show that the performance of our approach is at most 20% less than the experimental upper-bound and up to 80% higher than the experimental lower-bound.

To evaluate the performance of EMMA for batch active learning, I design two instantiations of EMMA to perform active learning in batch mode. I show that these algorithms improve the algorithm training time at the cost of a reduced accuracy in performance. Another finding in our work is that integrating clustering into the process of selecting sensor observations for batch active learning improves the activity learning performance by 11.1% on average, mainly due to reducing the redundancy among the selected sensor observations. I observe that mindful active learning is most beneficial when the query budget is small and/or the oracle's memory is weak. This observation emphasizes advantages of utilizing mindful active learning strategies in mobile health settings that involve interaction with older adults and other populations with cognitive impairments.

Keywords: Active learning, wearable computing, machine learning, activity recognition, memory retention, cognitive factors, server monitoring.

1. INTRODUCTION

With the advent of the Internet-of-Things (IoT) paradigm, applications of sensor based systems have advanced significantly across many domains from health monitoring and autonomous vehicles to smart building and environmental monitoring. Mobile and wearable devices are being increasingly utilized, along with machine learning algorithms, to monitor physical and mental health, and to improve human well-being through clinical interventions.Most of these applications are human-centered in that they focus on monitoring human health and even interacting with humans to incorporate their feedback for improved performance of the system. The monitoring component often relies on computational algorithms that can detect important health events. For example, wearable sensors are extensively utilized to record human physiological data and then, computational algorithms such as machine learning models are applied for data analysis and to make predictions about events of interest. To train accurate machine learning models for different applications, such as activity recognition, an adequate number of labeled sensor data is required. However, data collections and related experiments are mainly done in laboratory settings where the experiments are highly controlled. Unfortunately, models that are trained based on sensor data collected in controlled environments and laboratory settings perform extremely poorly when utilized in uncontrolled environments and outside clinics. Therefore consideration of real-world and uncontrolled settings has become increasingly important.

Specifically, in human-centered applications, various limitations of human-beings, which can affect the performance of the trained models, need to be taken into account. For an activity recognition classifier to be accurate, one needs to collect and label sensor data in end-user settings. Therefore, active learning is a natural choice for labeling the data where the end-user acts as the oracle agent and we iteratively query the user for correct labels. Throughout this article, the terms 'end-user' and 'oracle' are interchangeably used. In such a human-centered monitor setting, it is critical to design active learning strategies that are mindful of the user's cognitive and compliance capabilities.

I recognize that human-beings are limited in their capacity to respond to, for example, prompts on their mobile devices. This capacity constraint is usually manifested in the number of queries that a person can, or will, respond to in a given time-frame and in the difference between the time that a query is made and when it has been responded to. This issue is critical in wearable-based continuous health monitoring where the amount of sampled data is orders of magnitude more than what the end-user can possibly annotate.

Server monitoring is crucial in ensuring system reliability and preventing downtime. I introduce the concept of memory aware active learning sensors for enhanced system monitoring.

2. LITERATURE SURVEY

Internet of things (IoT): A vision, architectural elements, and future directions

Ubiquitous sensing enabled by Wireless Sensor Network (WSN) technologies cuts across many areas of modern day living. This offers the ability to measure, infer and understand environmental indicators, from delicate ecologies and natural resources to urban environments. The proliferation of these devices in a communicating–actuating network creates the Internet of Things (IoT), wherein sensors and actuators blend seamlessly with theenvironment around us, and the information is shared across platforms in order to develop a common operating picture (COP). Fueled by the recent adaptation of a variety of enabling wireless technologies such as RFID tags and embedded sensor and actuator nodes, the IoT has stepped out of its infancy and is the next revolutionary technology in transforming the Internet into a fully integrated Future Internet. As i move from www (staticpages web) to web2 (social networking web) to web3 (ubiquitous computing web), the need for data-on-demand using sophisticated intuitive queries increases significantly.

Thispaper presents a Cloud centric vision for worldwide implementation of Internet of Things. The key enabling technologies and application domains that are likely to drive IoT research in the near future are discussed. A Cloud implementation using *Aneka*, which is based on interaction of private and public Clouds is presented. I conclude my IoT vision by expanding on the need for convergence of WSN, the Internet and distributed computing directed at technological research community.

A survey on the edge computing for the internet of things

The Internet of Things (IoT) now permeates our daily lives, providing important measurement and collection tools to inform our every decision. Millions of sensors and devices are continuously producing data and exchanging important messages via complex networks supporting machine-to-machine communications and monitoring and controlling critical smart-world infrastructures.

As a strategy to mitigate the escalation in resource congestion, edge computing has emerged as a new paradigm to solve IoT and localized computing needs. Compared with the well-known cloud computing, edge computing will migrate data computation or storage to the network "edge", near the end users. Thus, a number of computation nodes distributed across the network can offload the computational stress away from the centralized datacenter, and can significantly reduce the latency in message exchange. In addition, the distributed structure can balance network traffic and avoid the traffic peaks in IoT networks, reducing the transmission latency between edge/cloudlet servers and end users, as well as reducing response times for real-time IoT applications in comparison with traditional cloud services. Furthermore, by transferring computation and communication overhead from nodes with limited battery supply to nodes with significant power resources, the system can extend the lifetime of the individual nodes. In this project, i conduct a comprehensive survey, analyzing how edge computing improves the performance of IoT networks. I categorize edge computing into different groups based on architecture, and study their performance by comparing network latency, bandwidth occupation, energy consumption, and overhead. In addition, i consider security issues in edge computing, evaluating the availability, integrity, and the confidentiality of security strategies of each group, and propose a framework for security evaluation of IoT networks with edge computing. Finally, i compare the performance of various IoT applications (smart city, smart grid, smart transportation, and so on) in edge computing and traditional cloud computing architectures.

Approaching a human-centred internet of things

This project surveys recent Internet of Things (IoT) related HCI literature, and examines it in light of a comprehensive framework by Atzori et al. (2010). Mapping

HCI literature to this framework helped us understand the extent and the focus of IoT related HCIefforts, including a lack of HCI engagement with deeper human-centred perspectives of the IoT. It also revealed HCI considerations for the IoT which I added to the framework. This extended the framework to a tool for an HCI audience that can be used for 'thinking through' the design of IoT technologies. I close the project by demonstrating how this tool has been found useful in an IoT research project and at the same time illustrating my approach in how to engage more deeply with human centred concerns.

Neural network approach to forecast the state of the internet of things elements

The project presents the method to forecast the states of elements of the Internet of Things based on using an artificial neural network. The offered architecture of the neural network is a combination of a multilayered perceptron and a probabilistic neural network. For this reason, it provides high efficiency of decision-making. Results of an experimental assessment of the offered neural network on the accuracy of forecasting the states of elements of the Internet of Things are discussed.

2. PROPOSED SYSTEM

Memory awareness allows our sensors to adapt and improve over time.

Active learning sensors offer a new way to monitor servers in real-time.

The sensors have the ability to learn from past experiences and use this knowledge to better detect anomalies, reducing false alarms axx`nd improving overall performance.

My contributions in this project can be summarized as follows:

(i) I introduce mindful active learning as a general approach for budget-aware and delay-tolerant active learning in human-in-the-loop mobile systems. Mindful active learning takes into account the possibility that the oracle may forget the past events, particularly when there is a large time difference between the query and the activity being performed, and at the same time being constrained on the maximum number of queries that can be made;

(ii) To account to human memory in the active learning process, i propose an approach to model memory retention based on the Ebbinghaus forgetting curve;

(iii) I formulate mindful active learning as an optimization problem and propose an approach to solve this problem; and

I evaluate the performance of my algorithms for activity recognition using severaldatasets involving wearable and mobile sensors.

4. SYSTEM DESIGN

4.1 System Interfaces

Example systems interface requirements:

A.System1-to-System2 Interface

The <external party> will create and send a fixed length text file as an email attachment to be imported into the System2 system for payroll calculation. This file must be received on EDIT dayby 4:00 PM in order to be processed in the EDIT night run. The requirements below document the file specifications, data transfer process, and specific schedule. This file is referred to as "FileName" in this document.

File Structure and Format

A1. The FileName file is a fixed length text file.

A2. The FileName file is an unformatted ASCII file (text-only).

A3. The FileName file contains a batch totals record and several detail records.

4.2. Random Forest Classifier

Random forests are a type of machine learning algorithm that is used for classification and regression tasks. A classifier model takes data input and assigns it to oneof several categories. For example, given a set of images consisting of dogs and catsimages, a classifier could be used to predict whether each image is of a dog or a cat. In anutshell, a random forest algorithm works by creating multiple decision trees, each ofwhich is based on a random subset of the data. Decision trees are a type of algorithm thatmakes predictions by looking at the data inputs and determining which category theybelong to. Random forests take this one step further by creating multiple decision trees and then averaging their results.

This helps to reduce the chance of overfitting, which is when the algorithm only works well on the training data and not on new data. Random forestsare a powerful tool for machine learning and can be used for a variety of tasks such as facial recognition, fraud detection, predicting consumer behavior.

Stock market predictions. It can be considered as an ensemble of several decision trees.The idea is to aggregate the prediction outcome of multiple decision trees and create a final outcome based on the averaging mechanism (majority voting). It helps the model trained using the random forest to generalize better with the larger population. In addition, the model becomes less susceptible to overfitting / high variance. Here are the key steps of random forest algorithm:

(i) Take a random sample of size n (randomly choose n examples with replacement bootstrap)

(ii) Grow the decision tree from the above sample based on the following:

a) Select m features in a random manner out of all the features

b) Create the tree by splitting the data using m features based on the objective function (maximizing the information gain)

c) Repeat the above steps for k number of trees as specified.

d) Aggregate the prediction outcome of different trees and come up with a final prediction based on majority voting or averaging.

The diagram below represents the above mentioned steps:

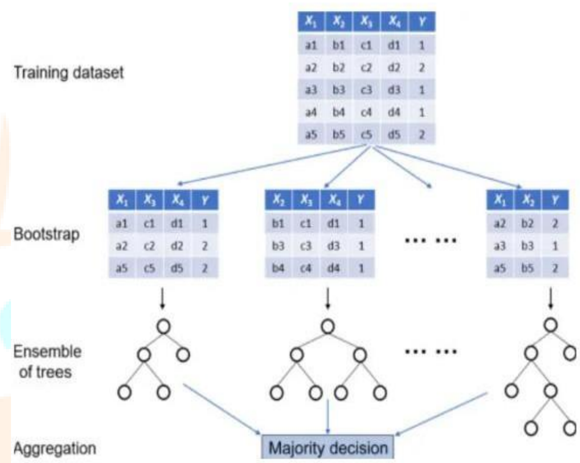


Fig. Key steps of random forest algorithm

4.3. Ensemble Classifier

The word ensemble is a Latin-derived word which means ‘union of parts’. The regular classifiers that are used often are prone to make errors. As much as these errors are inevitable they can be reduced with the proper construction of a learning classifier. Ensemble learning is a way of generating various base classifiers from which a new classifier is derived which performs better than any constituent classifier. These base classifiers may differ in the algorithm used, hyper parameters, representation or the training set.

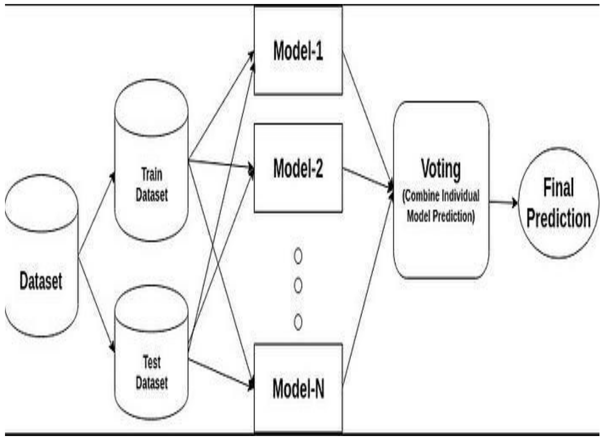


Fig. Basic Techniques for Ensemble Classifer

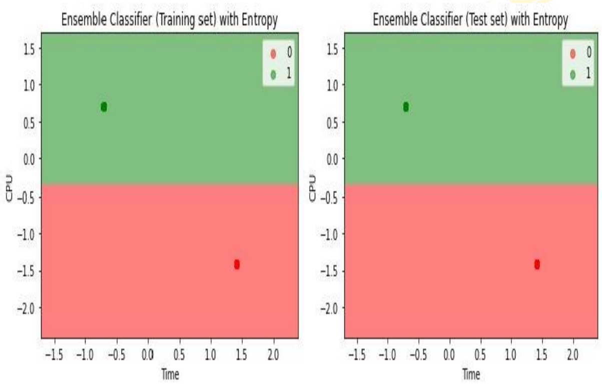


Fig. Ensemble Classifier Training and Test set

4.4. GRU(Gated Recurrent Unit)

Gated Recurrent Units (GRU) and Long Short-Term Memory (LSTM) have been introduced to tackle the issue of vanishing / exploding gradients in the standard Recurrent Neural Networks (RNNs).
In this article, I will give you an overview of GRU architecture and provide youwith a detailed Python example that you can use to build your own GRU models.
GRU's place within the Machine Learning universe

A complete Python example of building GRU neural networks with Keras and Tensorflow libraries.

4.4.1.How does GRU work?

GRU is similar to LSTM, but it has fewer gates. Also, it relies solely on a hidden state for memory transfer between recurrent units, so there is no separate cell state. Let's analyze this simplified GRU diagram in detail (weights and biases not shown)

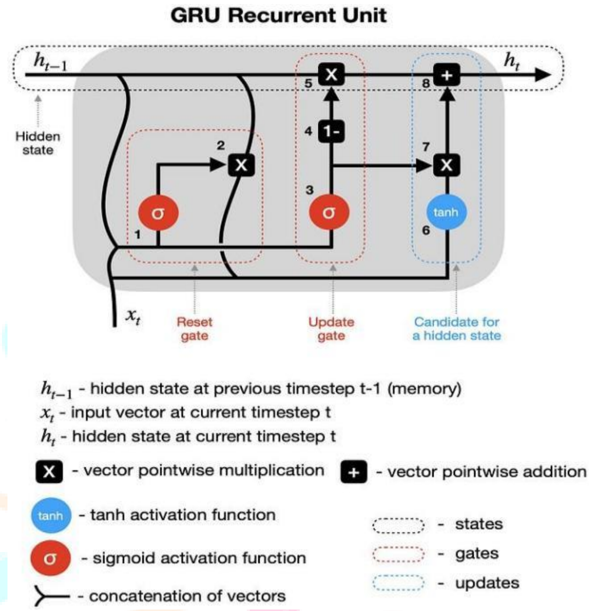


Fig. GRU Works

4.4.2.Training and evaluating GRU model

Here are a few things to highlight before we start.

*I will use sequences of 18 months to predict the average temperatures for the next 18 months. You can adjust that to your liking but beware that there will not be enough data for sequences beyond 23 months in length.

*I will split the data into two separate dataframes — one for training and the other for validation (out of time validation).

*Since we are creating a many-to-many prediction model, we need to use a slightly more complex encoder-decoder configuration. Both encoder and decoder are hidden GRU layers, with information passed from one to another via a repeat vector layer.

*A repeat vector is necessary when i want to have sequences of different lengths, e.g., a sequence of 18 months to predict the next 12 months. It ensures that we provide the right shape for a decoder layer. However, if your input and output sequences are of the same length as in my example, then you can also choose to set return_sequences=True in the encoder layer and remove the repeat vector.

*Note that i added a Bidirectional wrapper to GRU layers. It allows us to train the model in both directions, which sometimes produces better results. However, its use is optional.

*Also, i need to use a Time Distributed wrapper in the output layer to predict outputs for each timestep individually.

*Finally, I have used MinMaxScaling in this example because it has produced better results than the unscaled version.

*You can find both scaled and unscaled setups within Jupyter Notebooks in my GitHub repository (link available at the end of the article).

5. RESULT AND DISCUSSION

In this section, I present results on the performance of EMMA for active learning, comparative analysis, and the performance of batch-EMMA active learning.

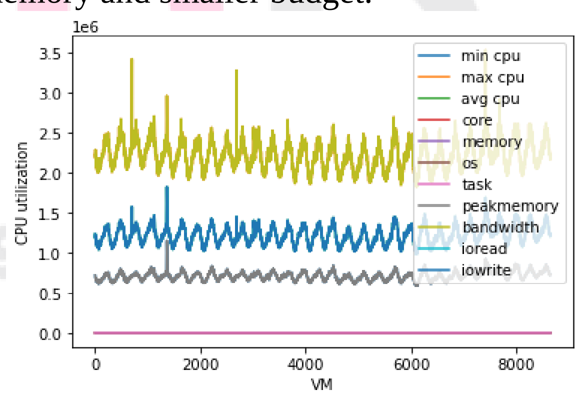
5.1. Performance of EMMA

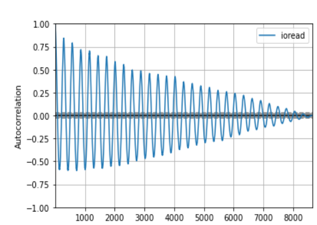
As a first analysis, i evaluated the performance of EMMA using different metrics (i.e., accuracy, precision, recall, F1-score) and examined how these metrics change as various parameters of the algorithm change. To this end, i con-ducted multiple experiments by changing the algorithm pa-rameters including query budget, B, and memory strengths. After EMMA constructed a labeled dataset, i trained an activity recognition classifier using the labeled dataset and utilized the trained model over the test set to measure performance metrics for each of the three datasets discussed previously. The details of the platform which is used for running our analysis is as follows: Intel(R) Core (TM) i5-3230M CPU @ 2.60GHz

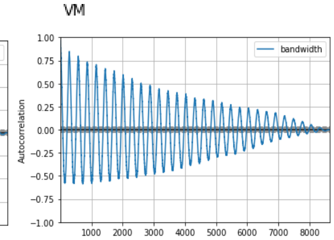
5.2. Comparative Analysis

I compared the performance of EMMA with that of several active learning approaches including EMA, MMA,upper-bound (UB), and lower- bound (LB). For a description of these alternative approaches. For brevity, i focus on accuracy as our main perfor mance measure here . Similar to my analysis in the previous section, i examined the performance of each algorithm while the budget value ranged from 0 to 200 and the memory strength was set such that different memory retention levels, R 1 (10 %– 99 %), R 2 (25 %– 99 %), R 3 (50 %– 99 %), and R 4 (70 %–99 %) are obtained. The accuracy numbers in these graphs re pre sent average values computed over all the users in each dataset.

The results show that the performance improvement due to using EMMA over EMA and MMA is most notable when the amount of budget is small and the memory is weak (i.e., memory retention level of R 1). This observation emphasizes the importance of considering both uncertainty and memory retention in health monitoring applications where data collection is extremely costly and end-users are likely to be cognitively impaired. It can be seen from the results that EMMA achieves an average accuracy that is 13 . 5 % higher than that of EMA and 14 % higher than that of MMA for cases of weaker memory and smaller budget.







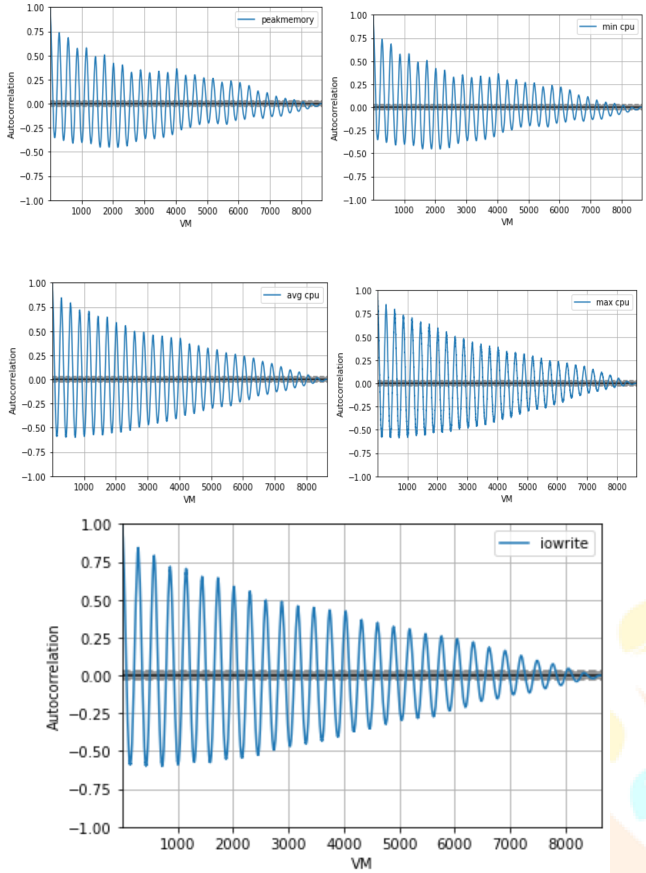


Fig. Comparative analysis data sets

5.3. Time Complexity Analysis

In this section we present our results on the run-time of the greedy approach for EMMA as well as that of th e brute-force approach. First, for the greedy algorithm, we measured the run-time for all three datasets when changing the budget and then averaged the results over all the users and number of iterations. The results it can be observed fro m these graphs, the time complexity of EMMA is linear in the the amount of query budget, which is consistent with our theoretical analysis.

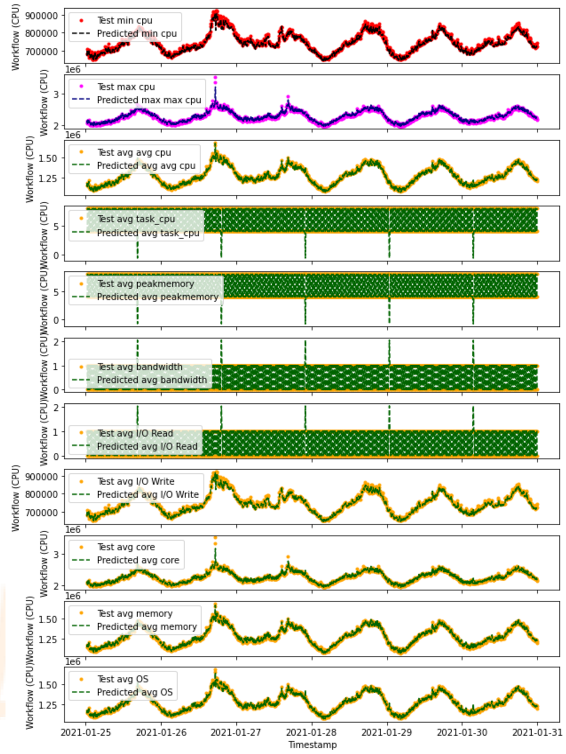


Fig. Workflow of time analysis

6. FUTURE WORK

In this paper, we focused on pool-based active learning. My ongoing work involves developing mindful active learning strategies that make query decisions on-the-fly as server monitoring data become available in real-time.

In my experiments i assumed that the time delay is equal to the difference between query time and sampling time. However, it is possible that the system may not respond to the query immediately. If the user prefers to proactively initiate the labeling process, the active learning process needs to recompute the queried observations by adding the time difference between query time and annotation time to reflect the time delay in our formulation. Furthermore , if there are multiple sessions in a day that the user intend to annotate the sensor data, this model can be used for each session separately. I also plan to investigate active learning solution s that take into account the possibility of delayed responses through context-sensitive active learning.

In this work, i simulated the memory strength of the end-user for validation purposes. My future work also focuses on conducting user studies that involve cognitive assessment of the user where i will assess the oracle's memory retention quantitatively.

7. CONCLUSION

Prior research on active learning takes in formativeness of data and query budget into account when selecting the data for query. In this project, i showed that cognitive constraints of the oracle are of significant importance that can greatly compromise active learning performance. I posed an optimization problem to combine data uncertainty with memory retention for use in ubiquitous and mobile computing applications. I derived a greedy approximation algorithm to solve the proposed mindful active learning problem. My extensive analyses on three publicly available datasets showed that EMMA achieves up to 97% accuracy for activity recognition using wearable sensors. I also showed that integrating memory retention improves the active learning performance by 16%.

In this work, i simulated the memory strength of the end-user for validation purposes. My future work also focuses on conducting user studies that involve cognitive assessment of the user where i will assess the oracle's memory retention quantitatively.

Moreover, by comparing the performance of EMMA with multiple competing algorithms as its variations, i showed that including both informativeness of samples and memory effects on noisy and incorrect labels, results in EMMA being less dependent on the datasets compared to other algorithms. This indicates that the results obtained by EMMA are more consistent across different datasets and machine learning tasks. Finally, the gap between the performance of EMMA and other algorithms is most notable with small budgets and weak memories.

ACKNOWLEDGEMENT

The author would like to express their gratitude to everyone who helped them write this article and provided data for this research.

REFERENCE

[1] Zhila Esna Ashari, Student Member, IEEE, Naomi S. Chaytor, Diane J. Cook, Fellow, "Memory Aware Active Learning in Mobile Sensing System" IEEE, and Hassan Ghasemzadeh, Senior Member, IEEE, 2022.

[2] M. Rofouei, M. Pedram, F. Fraternali, Z. Esna Ashari and H. Ghasemzadeh, "Resource-efficient computing in wearable systems", Proc. IEEE Int. Conf. Smart Comput., pp. 150-155, 2019.

[3] O. Sener and S. Savarese, "Active learning for convolutional neural networks: A core- set approach", Proc. 6th Int. Conf. Learn. Representations, 2018.

[4] P. Alinia, I. Mirzadeh and H. Ghasemzadeh, "ActiLabel: A combinatorial transferlearning framework for activity recognition", 2020.

[5] P. Bachman, A. Sordoni and A. Trischler, "Learning algorithms for active learning",Proc. 34th Int. Conf. Mach. Learn., pp. 301-310, 2017.

[6] J. Gubbia, R. Buyyab, S. Marusica, and M. Palaniswamia, "Internet of things (iot): A vision, architectural elements, and future directions," Future Generation Computer Systems, vol. 29, no. 7, Sep. 2013.

[7] W. Yu, F. Liang, X. He, W. G. Hatcher, C. Lu, J. Lin, and X. Yang, "A survey on the edge computing for the internet of things," IEEE Access, vol. 6, nov 2017.

[8] T. L. Koreshoff, T. W. Leong, and T. Robertson, "Approaching a human-centred internet of things," in Proceedings of the 25th Australian Computer-Human Interaction Conference: Augmentation, Application, Innovation, Collaboration, Adelaide, Australia, Nov. 2013.

[9] I. Kotenko, F. Skorik, and S. Bushuev, "Neural network approach to forecast the state of the internet of things elements," in Proceedings of the XVIII International Conference on Soft Computing and Measurements (SCM), St. Petersburg, Russia, Jun. 2015.

[10] P. Ni, C. Zhang, and Y. Ji, "A hybrid method for short-term sensor data forecasting in internet of things," in Proceedings of the 11th International Conference on Fuzzy Systems and Knowledge Discovery (FSKD), Xiamen, China, Jun. 2014.

[11] R. Fallahzadeh and H. Ghasemzadeh, "Personalization without user interruption: Boosting activity recognition in new subjects using unlabeled data," in 2017 ACM/IEEE 8th International Conference on Cyber-Physical Systems (ICCPS), April 2017, pp. 293–302.

[12] B. Settles, "Active learning,"

Synthesis Lectures on Artificial Intelligence and Machine Learning, vol. 6, no. 1, 2012.

[13] P. Bachman, A. Sordoni, and A. Trischler, “Learning algorithms for active learning,” in Proceedings of the 34th International Conference on Machine Learning, Sydney, Australia, Aug. 2017.

[14] S. Redmond, N. Lovell, G. Yang, A. Horsch, P. Lukowicz, L. Murrugarra, and M. Marschollek, “What does big data mean for wearable sensor systems?” Yearbook of medical informatics, vol. 23, no. 01, pp. 135–142, 2014.

[15] R. K. Sah and H. Ghasemzadeh, "Adversarial transferability in wearable sensorsystems", 2020.

[16] S. A. Rokni and H. Ghasemzadeh, "Share-n-Learn: A framework for sharing activity recognition models in wearable systems with context-varying sensors", ACM Trans. Des. Autom. Electron. Syst., vol. 24, 2019.