An Adaptive Recommender Engine for m-Health Solutions to Chronic Disease

Hao Ding
Department of Electrical and Electronic Engineering
Imperial College London
Email: hd1812@ic.ac.uk
Supervisor: Dr. Yiannis Demiris

***Abstract*—Chronic disease is the major cause of mortality and has been leading to larger burden globally. The aim of this project is developing a recommender engine that collaborates with mobile applications and other components to assist the healthcare of chronic diseases and advance the self-management of patients. The emphasis of this project is implementing the recommender system architecture, which uses a hierarchical multi-model structure that provides personalised action suggestions and improves the suggestion quality through learning algorithms. The proactive action suggestion mechanism and the multi-model architecture of this recommender system is likely to deliver better outcome than the traditional mechanisms used in mobile health apps. Currently the skeleton of the system is accomplished and a simple demonstration is available to show the operation of the recommender system. Models and learning algorithms that target specific aspects of chronic disease are awaiting to be implemented. This interim report shows the research, plan and future work of the project.**

I. Introduction

Chronic diseases, often referred as non-communicable disease, are defined by the World Health Organisation as "the diseases of long duration and generally slow progression". Examples of chronic diseases include diabetes, cancer, cardiovascular diseases and mental illness. In U.S, 75% of the healthcare spending is on chronic diseases[1]. In 2010, the median state medical cost is $1.8 billion for diabetes and $410 million for asthma[2]. The major causes of death among the U.S elderly population are cardiovascular disease and cancer, accounting for 41% of all deaths together[3]. In Europe, chronic diseases are responsible for more than 80% of all deaths[4]. Chronic diseases are the most common, costly and preventable of all health problems. It has been brought increasing challenges for government healthcare system and also resulted in the prolonged burden of individual patients.

This project aims to tackle the challenges in chronic disease healthcare by delivering the long-term support to patients. A distributed adaptive multi-model architecture will be used for action selection and recommendation. This project is likely to show satisfying performance in the long-term healthcare tasks owing to its adaptivity and proactivity. This report introduces the project context, the current progress and the future plans.

A. Background and Context

M-health, abbreviation of mobile health, can be defined as 'the emerging mobile communications and network technologies for healthcare systems' [5]. The m-health field is referred as a sub-segment of eHealth, the usage of computers, smartphones and other digital devices for healthcare service [6]. In recent years, mobile phones have becoming increasingly popular in promoting healthy lifestyle and encouraging physical activities[7]. In an NHS Guide for Developing Mobile Healthcare Applications published in 2014, NHS has stated that demand of healthcare apps has been increasing and so is the number of clinicians that use smartphones and tablets. A study predicted that 83% of doctors would use smartphones in work and the ratio is expected to increase over time[8]. Owing to the popularisation of mobile devices among various populations, m-Health has become increasingly suitable to deliver personalised and long-term healthcare service.

Regarding the treatment for chronic diseases, an international cooperation project in Jamaica has shown some interesting results. Local patients with chronic lifestyle disease were provided with an opportunity to receive lifestyle guidance and health education. Residents have begun to adapt their lifestyle patterns after obtaining the knowledge and instructions from public health professionals. It has emphasised in the article that, to change the behaviour pattern of patients, knowledge and support are necessary to improve their awareness in chronic diseases[9]. This report has provided the motivation of my project, which delivers healthcare support using the recommender system. The proactive feature of a recommender engine is likely to encourage patient engagement and benefit the prolonged healthcare process.

In my project, the recommender system means to behave as the back-end of health mobile applications. Unlike traditional health applications which only provide the passive functions of data recording and tracking, this project provides recommendations proactively based on the user status, which can be collected using wearable devices or smartphones. The proactive nature of this project is likely to lead to a more satisfying personal healthcare service since the guidance and assistance was considered to be significant in chronic disease healthcare, according to the project in JamaicaJamaica. This mechanism is likely to encourage the engagement of patients and most importantly, improve the awareness of self-management and lead to a healthy lifestyle that is beneficial in the long-run.

B. Scope and objectives

Note that the core architecture design, HAMMER[10] that is introduced in the next section, is a general architecture for action recognition and execution. It has a range of applications such as motor control and robot imitation learning. The idea behind this project is using the HAMMER architecture in long-term healthcare of chronic diseases. Throughout the project, there is likely to be the collaboration with other related projects that involves mobile devices and wearable devices, to realise higher goals in m-Health. Therefore, the work on interface design is also expected and ideally the cooperation between projects will form a complete system for chronic disease healthcare.

This project aims to implement the underlying architecture that provides patients with personalised suggestions on their diseases. The final deliverable is likely to operate on servers. It receives patient status information and targeted status and generates the optimal action suggestions based on the simulation results of the adaptive and personalised models.

C. Achievements

To date, the skeleton of the project architecture has been constructed and wrapped into a local server. Model templates and the overall structure have been established and the implementation of actual models as well as algorithms will commence in near future. To begin with, several aspects of lifestyle for patients with diabetes will be investigated, including the amount of the daily exercise as well as the stress level.

A simple demonstration of its operation has been finished, which shows how the hierarchical models operate sequentially or parallel, and how the actions are suggested according to the sequence of their confidence levels. Imitation learning is used to adjust model coefficients iteratively by observing the user states assuming that the user is under good healthcare. Furthermore, the communication through the local server has been tested by submitting data and commands to the server and observing the returned results.

II. State-of-the-art

A. Introduction

Mobile healthcare is not a new idea. Using mobile applications to monitor and assess patients with chronic diseases is likely to reduce the cost and burden of daily healthcare. Remote report and medical suggestions may enable more reliable, efficient and responsive service[11]. Research has been done on the feasibility of utilising texting in delivering diabetes information to the parents of children with type 1 diabetes. Positive user feedbacks were received from some parents and the report has concluded that the text message system has the potential in delivering diabetes information[12].

Currently, there are a wide range of commercialised products following different designs and techniques. A report of the United Nation Foundation and Vodaphone Foundation has separated mHealth applications into six categories: Education and Awareness, Remote Data Collection, Remote Monitoring, Communication and Training for Healthcare Workers, Disease and Epidemic Outbreak Tracking, and Diagnostic and Treatment Support [13]. This section introduces the commercialised products with similar applications, introduces the HAMMER architecture, which my project is based on, and explains the learning and modelling techniques which will be implemented as a part of the HAMMER architecture.

B. Current m-Health Applications

This section compares some commercialised m-health products following two different mechanisms, either passive or proactive. In the m-health market exist a wide range of products and a large number of them are focused on data collection. The Health app, a pre-installed app for iPhones, acts as a personal data hub. It can be connected to wearable devices and other third-party accessories to collect and analyse data. Another app, the AliveCor Mobile ECG, measures and records user heart rate. This type of product emphasises on data collection and analysis, which provides information for user to track and understand their health conditions. The product itself does not provide any suggestion, but requires user to take further steps after presenting them with data.

Some other products follow a different idea, that the app not only collects and analyses data, but also provides feedback to users and tells them the actions to take. For example, the HidrateSpark, tracks the amount of water consumed and reminds users to drink water in order to meet the customised daily water goal. The Pip, estimates the user stress level by measuring the Electrodermal Activiey (EDA). It uses gamification to encourage users to retain their stress levels. Such m-health applications provide proactive suggestions or solutions after obtaining user data.

My project follows the proactive design mechanism, actively providing suggestions to the user-end. After collecting and analysing user data, the recommender engine generates the optimal action to take in order the reach the target state. The details of the recommending mechanism are explained in the following section. In the past project in Jamaica, guidance and instructions are considered important in preventing chronic lifestyle diseases. Therefore, the proactive design of m-health system is likely to lead to higher healthcare efficiency and more engagement of patients in improving the lifestyle of patients with chronic diseases.

C. The HAMMER Architecture

1) Overview: The HAMMER architecture[10], a hierarchical multi-model structure developed in Imperial College London, will be used for action selection. Hammer is the abbreviation for hierarchical attentive multi models for execution and recognition [10]. It organises the motor control systems in a hierarchical way and performs the functions of imitating learning and competitive action suggesting. The attention mechanism allocates the limited resource depending on the significance level. This section explains the details of HAMMER architecture and how it will be utilised in long-term healthcare for chronic disease [10].

In motor control theory, inverse model is used to generate the motor commands in order to approach the target state[14], while forward model predicts the upcoming state following the inverse model command, which allows rapid error detection and system adjustment[14]. In the application of healthcare, an inverse model reads the current state of patient, for example, blood pressure, heart rate, etc, and the target state desired, for example, a reasonable glucose level. The inverse model then produces the action which is believed to lead the patient to the target state. A forward model reads the command, for example, taking pills, and forecasts the patient state by simulating the execution of the input command.

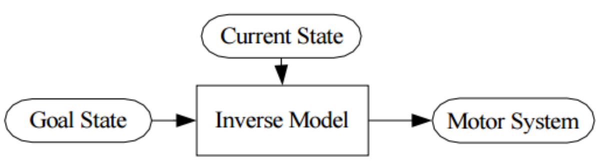

Fig. 1. the structure of an inverse model[15]

2) Inverse model and forward model: The basic unit in Hammer architecture is an inverse-forward model pair. The command generated from the inverse model is fed into the forward model directly, allowing the forward model to predict upcoming state of the command. In the learning phase, the predicted state is compared with the actual state and the error between them is used to adjust forward model. This allows the forward model to learn and imitate user reaction overtime and this adaptive feature has shown to be preferred in long-term chronic disease healthcare[16]. In the execution phase where multiple inverse models are present, the prediction mechanism of the forward model allows the outcome of various commands to be simulated and compared with each other and thus, the command which is most likely to lead to the desired outcome will be selected and delivered to the user[17].

3) Multi-model nature and Hierarchy: The inverse-forward pairs can be implemented into hierarchical structure. Multiple model pairs are able to operate parallel. The target state and

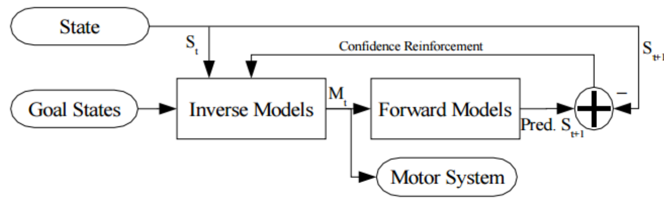

Fig. 2. the basic unit of the HAMMER architecture[15]

the user state are sent to all inverse models simultaneously, allowing inverse models to produce a number of actions. Each of the actions is fed into forward model, which predicts the corresponding user state for each command. The difference between the user state and the actual state has effect on the confidence levels of inverse models. The action suggested by the inverse model with the highest confidence level is selected after the learning phase.

Apart from the parallel feature, HAMMER also allows the hierarchical structure by arranging model pairs in various levels. Nodes in higher level represent the higher abstraction of low-level nodes, for example, the goal state[10]. Low-level models can also be combined to form higher level models[15]. Experiments have demonstrated that the abstraction mechanism contributes to the better performance in behaviour recognition[15].

D. User Modelling and Learning Algorithms

In order to maintain the long-term engagement of patient in chronic disease healthcare, the system has to be adaptive and able to automatically adjust to the changes happened on users. For example, the system can learn the changes in users preference, physical and mental condition, personality and interests. To provide prolonged personalised service, the system has to change its behaviour based on the changes of user behaviour. Research has shown that the adaptive mechanism is preferred in human-robot interactionadapt. Machine learning is becoming increasingly popular in user modelling field[18].

User preference is one of the most commonly used information in learning about new users. There are a range of machine learning techniques that are beneficial to user modelling. The collaborative filtering recommender system is good at selecting items that the new user is likely to have an opinion about [19], [20].especially in the case that the information of the new user is limited. This is likely to be the situation of chronic disease treatment for young children. Bayesian network are commonly used in distinguishing user abilities and learning styles[21], [22], [23], [24]. Feedback based on learning style has been shown to contribute to better learning outcomes[24]. Support vector machine has been used to recommend learning resource to user[25]. Regression models has been used to learn the user preference on performance feedback[26].

In the application of machine learning for user modelling,

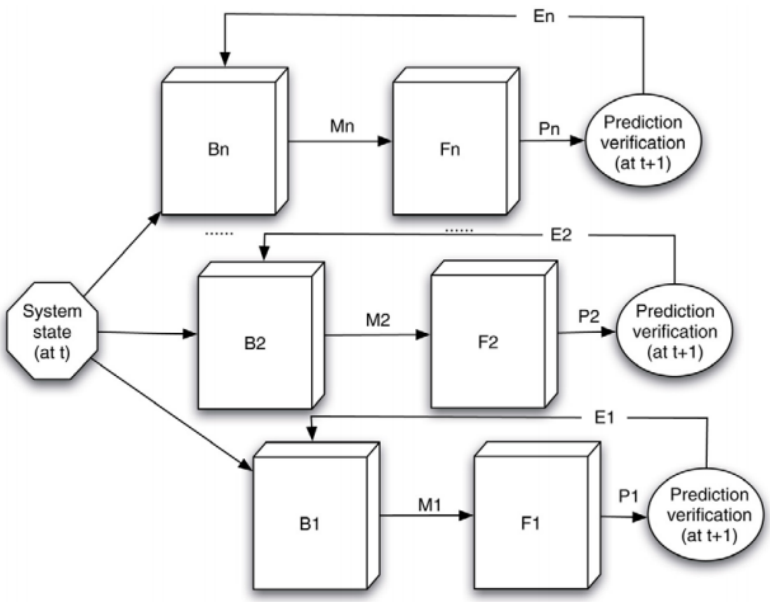

Fig. 3. the basic parallel structure of the HAMMER architecture. Inverse models (B1-Bn) generates various commands (M1-Mn)that are received by forward models (F1-Fn). Error signals (E1-En)are sent back to inverse models[10]

large and labelled data set might be required and the computational complexity might limit the performance of machine learning algorithms[18]. Therefore, the careful selection of machine learning techniques based on the challenges in different situations is vital to the system performance. Furthermore, machine learning techniques are also likely to be combined to provide personalised learning experience[27].

III. Implementation Plan

A. Tasks

The implementation of my project is mainly consist of three parts:

- Architecture Design
- Interface Design
- Model Implementation

The main architecture is based on the HAMMER architecture described in the previous section. Though not all features of HAMMER are likely to be utilised in this project due to the time limit, the architecture design will still include the multi-model structure as well as the attention mechanism. Details about the implementation so far have been included in the next two section.

The interface between the recommender engine and external software depends on the environment of the application. In the test stage, the HAMMER architecture is developed independently, and then wrapped into a local server, which exchanges information with the browser through passing around JSON object. Currently, there are two potential projects that might include HAMMER in near future: Firstly, a coursework project on machine learning and mobile healthcare. In this project, mobile device will be connected to the HAMMER architecture and deliver a demonstration of using the adaptive feature of HAMMER in mobile healthcare. This might requires the local host to be exposed online and to allow other device to access the local IP address. Secondly, it may be used as a component of the Personal Assistant for healthy Lifestyle (PAL) project, which tackles the challenges of self-management among young patients. Joining the PAL project needs the communication framework used for PAL participants.

Model implementation is another important module. Low-level models might need to be hard coded to deal with different aspects while higher level models might be generated by combining low-level models [15]. As a starting point, simple low-level inverse models will be implemented using basic machine learning techniques, for example, the decision tree which provides meaningful information on output commands. The first model will be tested in the coursework project mentioned previously. The relation between the glucose level and daily exercise is likely to be investigated in this project first.

B. Achievements

So far, the skeleton of the HAMMER architecture is nearly accomplished while some advanced features such as high-level abstraction requires more work to finalise. Demonstrations on the parallel and sequential operation are also available. Fig 4 has shown the simplified version of the UML diagram of HAMMER implementation structure. Please find the detailed UML diagram in Appendix A for an overview of my actual implementation.

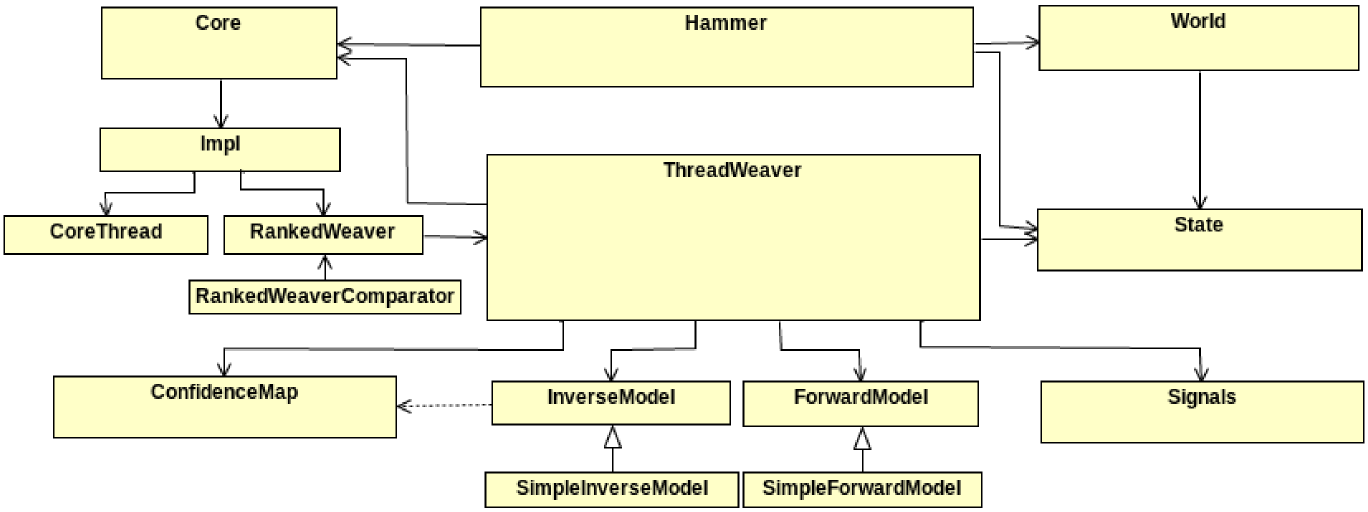

Fig. 4. the UML diagram showing the relation between building blocks in the HAMMER architecture. Detailed explanations are available on the achievements section

The 'Hammer' is the main interface for programmers and it provides many functions including specifying the 'Core', updating state information and specifying model implementation. The 'Core' is the control hub of the operation, providing access to various threads, models and functions. The 'Core' may contains a number of threads, each consists of an inverse-forward model pair. The relation between threads is implemented as a property named 'dependencies', which affects the execution sequence of threads. The 'World' class simulates the user state change for early-stage testing purpose. Note that to compromise various data types, general types are used in the current implementation, allowing more freedom in real world application.

The demonstration designed for now is a simple test on HAMMER operation. In the 'World' class simulates patient states, including age, glucose level, etc. In the learning phase, the glucose level starts from a random number and gradually approaches a reasonable range after receiving commands from the demonstrator. The confidence levels of inverse models are incremented or decremented depending on the errors between the corresponding predicted state from the forward model and the actual state. After entering the execution phase, the action suggested by the most confident model is selected. This demonstration shows a simple imitation learning process. Note that the inverse model is designed to be adaptive but the forward model has been assumed to a good user model, just for the simplicity of initial test. This demo also shows the parallel operating of models at the same level and the sequential operation of models belonging to different abstract level, though the high-level model here is only a dummy model, just to show the operation of the hierarchical structure.

The HAMMER has also been wrapped into a local Tomcat server using REpresentational State Transfer (REST) API, allowing the browser to communicate with HAMMER through sending JSON object. REST API is convenient to use in terms of resource management and software interface design. Figure 5 has shown the communication between the three major parties in RESTful service. The REST API manages server resource under various entry points and performs actions depending on user requests.

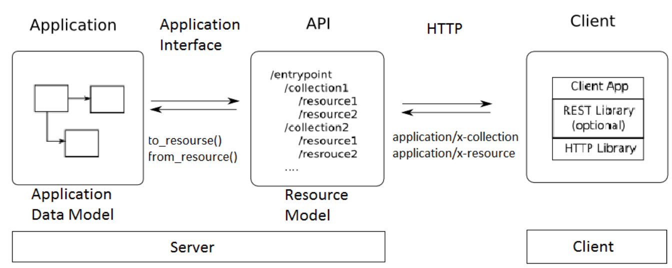

Fig. 5. The interaction between RESTful API components. A similar figure is available following the link: http://restful-api-design.readthedocs.org/en/latest/scope.html

A simple demonstration of moving robot using REST API and HAMMER has been implemented. After receiving a JSON object that specifies the robot position, the target position and the moving speed, HAMMER runs several models which guess the robot speed. HAMMER observes the robot position change and modifies the model confidence level based on the prediction errors. All the confidence levels are returned to the browser and the one with the highest confidence level actually guessed the correct number. This demonstration shows the multi-model operation of HAMMER with RESTful API and shows the potential of accessing HAMMER to external devices.

Overall, the achievement has shown the reasonable project progress and the potential feasibility of ideas. The skeleton of HAMMER has been constructed and the demonstration has shown the potential to connect HAMMER to other devices. More research and implementation are necessary to lead to a complete design.

C. Future Work

Considering the three main tasks throughout this project, the majority work of the architecture design has been finished and the remaining tasks will be accomplished in the following four months. At the same time, sufficient time left for enhancing the project and finalising the report.

Future work includes but not limited to:

- Architecture Design
  - Implement the hierarchical feature
  - Research and implement Stochastic Context-Free Grammar (If necessary)
- Interface Design
  - Research and design interface for mobile device, e.g. cloud server
  - Implement the HAMMER wrapper to fit into the framework in PAL (If necessary)
- Model Implementation
  - Background research on data source, hardware and background knowledge relates to chronic disease
  - Research suitable machine learning algorithms, e.g. Bayesian network
  - Implementation of inverse and forward models

Note that the above list is for reference only. It is possible that the cooperation with external projects might add to the work load. It largely depends on the progress of my project as well as other participants' work. A flexible timetable with contingency time has been included in the Appendix B.

D. Challenges

Many potential challenges have been foreseen so far.

First of all, a large amount of labelled data is necessary to train the model. Note that since labelled data set might violate the privacy policy, acquiring sufficient and reliable training data is a challenging task. In the worst case scenario, data will be simulated using computer programs, which is less reliable than the data collected from patients.

Secondly, HAMMER is mean to be a general structure that allows different data types while the data used by each model is specific. Such design adds to the complexity of the model implementation. For example, data collected from different medical device might be in different format. As a starting point, it is reasonable to focus on several basic data types first. Complex data type can be handled by implementing data converter modules or designing different models. The extra work caused by data format mismatch largely depends on the scenario where HAMMER is used and therefore, this problem should not be focused on in early-stage design.

Thirdly, the interface design in this project also depends on its application scenario. As a result, the HAMMER architecture itself should be implemented independently, for the convenience of later application in various environments.

Finally, the model implementation is considered as a challenging task. There are a range of machine learning techniques available and their performance is likely to vary as data types and coefficients. Although HAMEMR allows multiple models to operate parallel, implementing models with good performance can still be a challenging task.

E. Time Plan

The major task in this project is software development, which can be delivered following the Software Development Life Cycle (SDLC), including planning, design, implementation and evaluation. More than one iteration of the SDLC will be included in this project plan to meet the various deliverable deadlines and to adapt to the potential changes in project requirements. Based on the current project progress, a simplified time plan with important dates for deliverables is outlined in Table I. The detailed project plan is included in the Appendix B.

Date	**Deliverable**
01.02.2016	Interim report
02.03.2016	An Integrated System for Demonstration
16.03.2016	1st Prototype
15.05.2016	2nd Prototype
06.06.2016	Abstract
15.06.2016	Final report
20.06.2016	Final Presentation

TABLE I
IMPORTANT DATES WITH DELIVERABLES

Notice that the 'Integrated system for Demonstration' is the first systematic design combining HAMMER, interface and model. It needs to show the feasibility of the whole system and prepares for the first prototype that works with the mobile device to deliver basic mobile healthcare service. The second prototype is a complete system with better model implementation covering various aspects of chronic diseases. It might has more than one interface, not only providing web service for mobile device, but only communicating with other components in the PAL project. Details of the second prototype are awaiting to be finalised.

IV. Evaluation Plan

The evaluation plan of the project includes evaluating the three major components separately and evaluating the project as a whole. Various aspects of the system will be assessed, such as operation speed, computational complexity, and model accuracy. A systematic way to evaluate software product is available online[28].

First of all, qualitative measure can be used to justify whether the system design is reasonable and it provides a rather intuitive answer to the quality of the work.

Module	Evaluation Criteria
Architecture Design	Whether the design is compatible with different data types. Whether all the important features of HAMMER are fully utilised.
Interface Design	What type of hardware or software are supported by the interface. How convenient the interface is for external hardware or software to access.
Model Design	Whether the model outputs match our expectation intuitively. Whether the machine learning algorithms selected are suitable. Whether the models are capable of simulating most behaviours of the demonstrator.
Overall	How well the user experience is. Whether the suggested action is logical. Whether the project is well commented and documented for future development.

Secondly, quantitative measure is able to provide accurate and comparable assessment on system performance. Some criteria are listed in the following.

Module	Evaluation Criteria
Architecture Design	The number of data types supported. The maximum number of threads, models and layers supported.
Interface Design	The transmission speed under various conditions. The maximum size of package supported. The number of users that can serve at the same time.
Model Design	Classical methods to evaluate the performance of machine learning algorithms, e.g. confusion matrix, cost curves and statistical tests. Performance evaluation under various conditions, e.g. sparse training data. The number of aspects covered by inverse models. The data package size that can be handled by each model.
Overall	Overall action selection accuracy. System response time and Computational complexity. System stress test.

Since there is a large chance to cooperate with external project, it will be interesting to see how different components collaborate together and form a larger system. Though the performance of the whole system is dependent on other components, it is still an opportunity to test my part under a more complicated and more practical environment.

V. Conclusion

This report has introduced the background and context of the project, explained the current progress and future implementation plan, and shown a number of methods to evaluate the performance of the project. The whole project aims to provide a solution to long-term chronic disease m-Health. The design is based on the HAMMER architecture and is able to actively deliver action suggestions to patients. It is likely to bring satisfying performance in long-term m-Health due to the adaptive and proactive features of the design. Other work such as interface design provides the opportunity to collaborate with other related projects, which is likely to lead to a complete system design and realise higher overall goals.

It is exciting to see the future development of this project, in particular, the collaboration with other projects to form a larger healthcare system for chronic diseases. It brings more uncertainty into the project and costs more effort on interfacing with other components. But this is a great chance to test the architecture in the real world scenario which is challenging and stimulating.

APPENDIX A
UML DIAGRAM OF CURRENT HAMMER IMPLEMENTATION

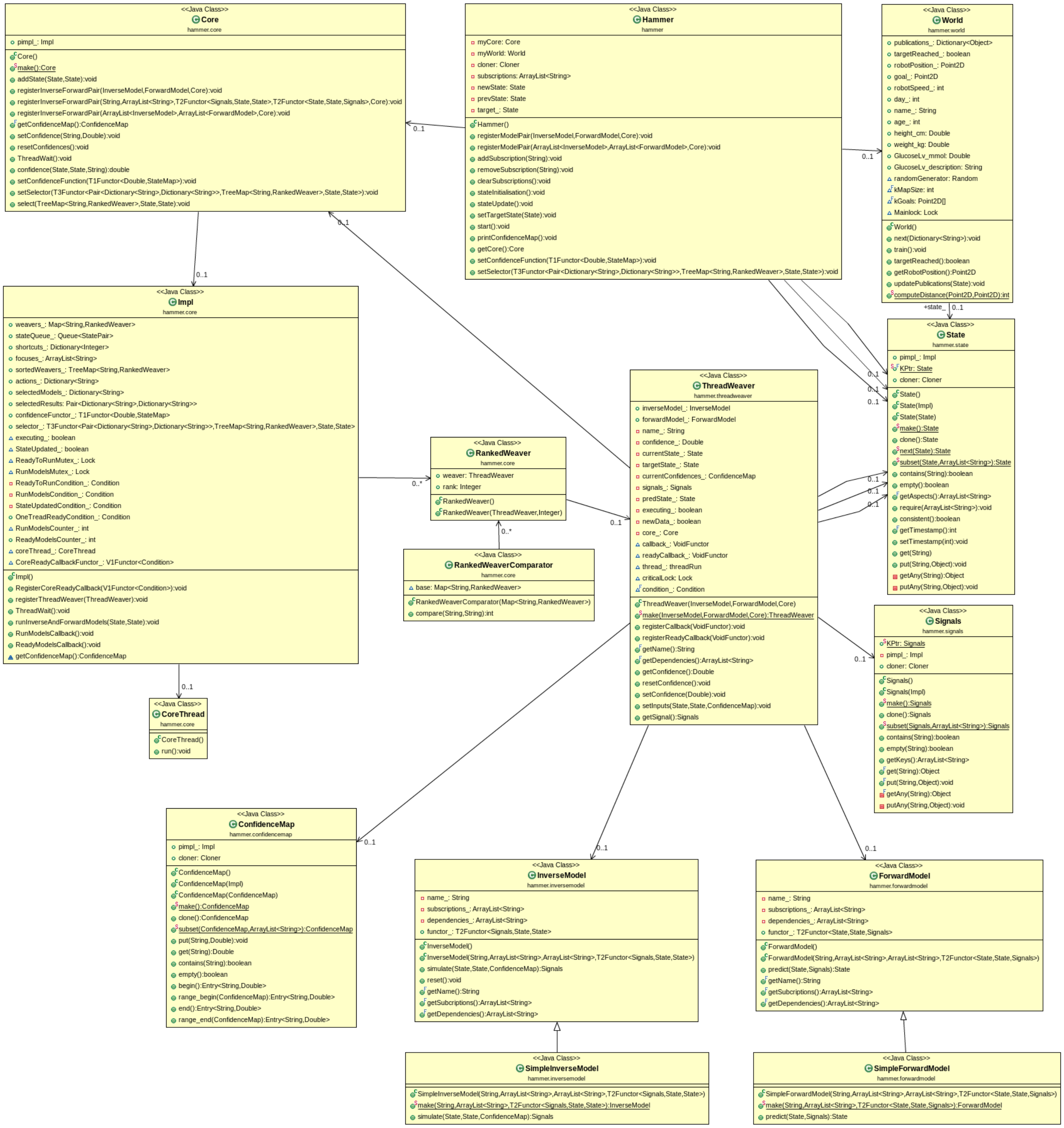

Fig. 6. the UML diagram showing the relation between building blocks in the HAMMER architecture. Detailed explanations are available in the achievements section

APPENDIX B
PROJECT TIMEPLAN

Date	**Task**	**Deliverable**	**Milestone**
20.01.2016	Background research		
27.01.2016	Initial Project Plan		Planning
01.02.2016	Initial Design	Interim report	Design
09.02.2016	Interface with iOS device		
21.02.2016	Model Implementation		Implementation
02.03.2016		A integrated system for demonstration	
19.03.2016	Design Finalisation		Test and Evaluation
16.03.2016		1st Deliverable	First prototype delivered
23.03.2016	Project Plan		Planning
30.03.2016	Overall Design		Design
14.04.2016	Model Implementation		
21.04.2016	Hierarchical Feature		
01.05.2016	Interface Design for PAL		Implementation
08.05.2016	Design Finalisation		Test and Evaluation
15.05.2016		2nd Deliverable	Second prototype finished
22.05.2016	Further enhancement		
06.06.2016		Abstract	
15.06.2016		Final report	
20.06.2016		Final Presentation	

TABLE II
THE DETAILED PROJECT TIME PLAN.

LIST OF FIGURES

LIST OF TABLES

REFERENCES

[1] H. Bush, "Tackling the high cost of chronic disease," *Hospitals & health networks / AHA*, vol. 86, no. 10, p. 34, 2012. ID: TN-medline23163172.

[2] J. G. Trogdon, L. B. Murphy, O. A. Khavjou, R. Li, C. M. Maylahn, F. K. Tangka, T. A. Nurmagambetov, D. U. Ekwueme, I. Nwaise, D. P. Chapman, and D. Orenstein, "Costs of chronic diseases at the state level: The chronic disease cost calculator," *Preventing chronic disease*, vol. 12, p. E140, 2015. ID: TN-medline26334712.

[3] C. for Disease Control (US), C. for Disease Control, P. (US), and H. P. National Center for Chronic Disease Prevention and, "Chronic disease notes and reports : Cdnr," *Chronic disease notes and reports : CDNR.; Chronic diseases notes and reports*, 1988. ID: 44IMP-ALMA-DS5152368760001591.

[4] N. Kamel, "The growing chronic disease burden in europe: a role for respiratory health professionals?," *European Respiratory Journal*, vol. 43, no. 5, pp. 1249–1251, 2014.

[5] C. Doukas, T. Pliakas, and I. Maglogiannis, "Mobile healthcare information management utilizing cloud computing and android os," in *Engineering in Medicine and Biology Society (EMBC), 2010 Annual International Conference of the IEEE*, pp. 1037–1040, IEEE, 2010.

[6] V. Consulting, "mhealth for development: the opportunity of mobile technology for healthcare in the developing world. washington, dc and berkshire, uk: Un foundation-vodafone foundation partnership. 2009," 2015.

[7] P. Klasnja and W. Pratt, "Healthcare in the pocket: mapping the space of mobile-phone health interventions," *Journal of Biomedical Informatics*, vol. 45, no. 1, p. 184, 2012. ID: TN-medline21925288.

[8] B. DC, "Smartphones in clinical practice, medical education, and research," *Archives of Internal Medicine*, vol. 171, no. 14, pp. 1294–1296, 2011.

[9] Y. Tateda, T. Kawamura, T. Yoshida, and T. Yamanaka, "Health education as part of health promotion and prevention of chronic lifestyle diseases in an international cooperation project," *International Congress Series*, vol. 1267, pp. 51–58, 2004. ID: TN-sciversesciencedirect-elsevierS0531-5131(04)00072-X.

[10] Y. Demiris and B. Khadhouri, "Hierarchical attentive multiple models for execution and recognition of actions," *Robotics and autonomous systems*, vol. 54, no. 5, pp. 361–369, 2006.

[11] B. H. Dobkin and A. Dorsch, "The promise of mhealth daily activity monitoring and outcome assessments by wearable sensors," *Neurorehabilitation and neural repair*, vol. 25, no. 9, pp. 788–798, 2011.

[12] S. C. Wangberg, E. Arsand, and N. Andersson, "Diabetes education via mobile text messaging," *Journal of telemedicine and telecare*, vol. 12 Suppl 1, p. 55, 2006. ID: TN-medline16884582.

[13] V. W. Consulting, "mhealth for development: the opportunity of mobile technology for healthcare in the developing world," *Washington Dc and Berkshire, UK*, 2009.

[14] R. P. Cooper, "Forward and inverse models in motor control and cognitive control," 2010.

[15] M. Johnson and Y. Demiris, "Abstraction in recognition to solve the correspondence problem for robot imitation," *Proceedings of TAROS*, pp. 63–70, 2004.

[16] A. Sekmen and P. Challa, "Assessment of adaptive human–robot interactions," *Knowledge-Based Systems*, vol. 42, pp. 49–59, 2013.

[17] D. M. Wolpert and M. Kawato, "Multiple paired forward and inverse models for motor control," *Neural networks*, vol. 11, no. 7, pp. 1317–1329, 1998.

[18] G. I. Webb, M. J. Pazzani, and D. Billsus, "Machine learning for user modeling," *User modeling and user-adapted interaction*, vol. 11, no. 1-2, pp. 19–29, 2001.

[19] A. M. Rashid, I. Albert, D. Cosley, S. K. Lam, S. M. McNee, J. A. Konstan, and J. Riedl, "Getting to know you: learning new user preferences in recommender systems," in *Proceedings of the 7th international conference on Intelligent user interfaces*, pp. 127–134, ACM, 2002.

[20] K. A. Bruffee, *Collaborative learning: Higher education, interdependence, and the authority of knowledge*. ERIC, 1999.

[21] N. Li, W. Cohen, K. R. Koedinger, and N. Matsuda, "A machine learning approach for automatic student model discovery," in *Educational Data Mining 2011*, 2010.

[22] A. Bunt and C. Conati, "Probabilistic student modelling to improve exploratory behaviour," *User Modeling and User-Adapted Interaction*, vol. 13, no. 3, pp. 269–309, 2003.

[23] S. Schiaffino, P. Garcia, and A. Amandi, "eteacher: Providing personalized assistance to e-learning students," *Computers & Education*, vol. 51, no. 4, pp. 1744–1754, 2008.

[24] S. M. Parvez and G. D. Blank, "Individualizing tutoring with learning style based feedback," in *Intelligent tutoring systems*, pp. 291–301, Springer, 2008.

[25] A. Al-Hmouz, J. Shen, J. Yan, and R. Al-Hmouz, "Enhanced learner model for adaptive mobile learning," in *Proceedings of the 12th international conference on information integration and web-based applications & services*, pp. 783–786, ACM, 2010.

[26] M. SUAREZ, "Predicting students appraisal of feedback in an its using previous affective states and continuous affect labels from eeg data,"

[27] A. Al-Hmouz, J. Shen, J. Yan, and R. Al-Hmouz, "Enhanced learner model for adaptive mobile learning," in *Proceedings of the 12th international conference on information integration and web-based applications & services*, pp. 783–786, ACM, 2010.

[28] M. Azuma, "Software products evaluation system: quality models, metrics and processesinternational standards and japanese practice," *Information and Software Technology*, vol. 38, no. 3, pp. 145–154, 1996.