Faculty of
Technology

PhD Thesis

Detecting and Modelling Stress Levels in E-Learning Environment Users

Yee Mei Lim
March 2017

First Supervisor: Dr Aladdin Ayesh

Dissertation submitted to the De Montfort University in partial fulfilment of the requirements for the degree of Doctor of Philosophy

Abstract

A modern Intelligent Tutoring System (ITS) should be sentient of a learner's cognitive and affective states, as a learner's performance could be affected by motivational and emotional factors. It is important to design a method that supports low-cost, task-independent and unobtrusive sensing of a learner's cognitive and affective states, to improve a learner's experience in e-learning, as well as to enable personalized learning. Although tremendous related affective computing research were done in this area, there is a lack of empirical research that can automatically measure a learner's stress using objective methods. This research is set to examine how an objective stress measurement model can be developed, to compute a learner's cognitive and emotional stress automatically using mouse and keystroke dynamics. To ensure the measurement is not affected even if the user switches between tasks, three preliminary research experiments were carried out based on three common tasks during e-learning − search, assessment and typing. A stress measurement model was then built using the datasets collected from the experiments. Three stress classifiers were tested, namely certainty factors, feedforward back-propagation neural network and adaptive neuro-fuzzy inference system. The best classifier was then integrated into the ITS stress inference engine, which is designed to decide necessary adaptation, and to provide analytical information of learners' performances, which include stress levels and learners' behaviours when answering questions.

Acknowledgement

First of all, I would like to express my utmost gratitude to my first supervisor, Dr Aladdin Ayesh, for his invaluable non-stop support of my PhD study and related research. I would like to thank his continuous patience, motivation, encouragement and understanding throughout my entire study. His professional skills, knowledge and guidance helped me in all the time of research and writing of this thesis. Personally, he is my best advisor and mentor for my research life. I could not have imagined having this thesis submitted without his effort in pushing me constantly for this great achievement.

Besides Dr Ayesh, I would also like to dedicate my sincere appreciations to my second supervisor Dr Martin Stacey, not only for his insightful comments and encouragement of my research from time to time, but also for spending his valuable time to validate my research ideas despite his busy schedules. His critical questions that were raised during discussions have driven me to widen my research from various perspectives, other than gaining knowledge to become a better researcher.

My sincere thanks also go to Dr. Khor Siak Wang, my third supervisor for his motivation and encouragement. His guidance was important especially during the initial phase of my PhD study, which helped me to build my confidence in continuing the rest of my study independently.

Besides the supervisory team, massive gratitude is dedicated to my employer, Tunku Abdul Rahman University College, who funded my entire PhD study and conference fees. They also granted access for me to use their laboratory and facilities for experiments setup, as well as allowed their students to take part in the research. Special thanks to my superiors from the university college, including but not limited to Y Bhg Datuk Dr Tan Chik Heok, Assoc. Prof. Dr Ng Swee Chin, Assoc. Prof. Dr Loke Chui Fung, Ms Lim Mei Shyan and Ms Chok Len Mooi. Without their precious support, it would not be possible to complete this research.

Last but not least, it is a pleasure to thank those who made this dissertation possible, including my husband, my children, my beloved mother, brothers, colleagues and friends, for giving me the moral support I required. My regards and blessings are offered to all of those who supported me in any respect during the completion of the study. Finally, a special tribute is given to my late father, who encouraged me to continue my PhD study before he passed away.

Contents

List of Figures

List of Tables

Abbreviations

A	Attitude / Attention
ANFIS	Adaptive neuro-fuzzy inference system
Assessment	A task that instructs the students to perform mental arithmetic. See Chapter 5
B	Behaviour
B(M)	Mouse behaviour
B(K)	Keystroke behaviour
B(M, K)	The unification of both mouse and keystroke behaviours
CF	Certainty factor
D	Decision
FAR	False acceptance rate
FFBP	Feedforward back-propagation neural network
FRR	False rejection rate
EER	Equal Error Rate
Err	Error rate
ITS	Intelligent Tutoring System
KS	Average keystroke speed (number of keystrokes per second)
KL	Keystroke latency (down-down key latency)
KE	Total delete key and backspace key pressed
LMS	Learning management system
M	Motivation
M_r	Rational motivation
MS	Mouse speed
MID	Mouse idle duration
MIO	Mouse idle occurrences
MCL	Left mouse click
MCR	Right mouse click
Question	The question or instruction of a specific task for a participant to carry out during the experiment
ROC	Receiver Operating Characteristic curves
$S_{B(M)}$	Stress measured based on mouse behaviour difference between 2 tasks
$S_{B(K)}$	Stress measured based on keystroke behaviour difference between 2 tasks
$S_{B(M, K)}$	Stress measured based on both mouse and keystroke behaviours between 2 tasks
$S_{B(sensor)}$	This refers to either $S_{B(M)}$, $S_{B(K)}$ or $S_{B(M, K)}$
SP	Stress perception
S_{TD}	Stress measured based on time difference between 2 consecutive tasks. See Equation 7.3
Sensor	This includes *B(M)*, *B(K)* and *B(M, K)*
Search	A task that instructs the students to search a learning material. See Chapter 4
TD	Time duration
Timing	Time constraint
Typing	A task that instructs the participants to type a given text. See Chapter 6

Publications

Journal Article

[1] Y. M. Lim, A. Ayesh, and K. N. Chee, "Socio-demographic Differences in the Perceptions of Learning Management System (LMS) Design," *Int. J. Software. Eng. Appl.*, vol. 4, no. 5, pp. 15–35, 2013.

[2] Y. M. Lim, A. Ayesh and M. Stacey. The Motivation/Attitude-Driven Behaviour (MADB) Model in E-Learning and the Effects on Mouse Dynamics. *International Journal of Cognitive Informatics and Natural Intelligence (IJCINI*), vol.10, no. 3, pp. 38-53, 2016.

Book Chapter

[1] Y. M. Lim, A. Ayesh, and M. Stacey, "Using Mouse and Keyboard Dynamics to Detect Cognitive Stress During Mental Arithmetic," *in Intelligent Systems in Science and Information 2014*, vol. 591, K. Arai, S. Kapoor, and R. Bhatia, Eds. Switzerland: Springer, 2015, pp. 335–350.

[2] Y. M. Lim, A. Ayesh, and M. Stacey, "The Effects of Typing Demand on Emotional Stress, Mouse and Keystroke Behaviours," *in Intelligent Systems in Science and Information 2014*, vol. 591, K. Arai, S. Kapoor, and R. Bhatia, Eds. Switzerland: Springer, 2015, pp. 209–225.

Conference Proceeding

[1] Y. M. Lim, A. Ayesh, M. Stacey, and K. N. Chee, "Designing Learning Management System to Encourage E- Learning Sustainability," *in Innovation and Transformation in Learning and Teaching*, 2013, pp. 76–83.

[2] Y. M. Lim, A. Ayesh, and M. Stacey, "The Effects of Menu Design on Users' Emotions, Search Performance and Mouse Behaviour," *in IEEE 13th Int'l Conf. on Cognitive Informatics & Cognitive Computing (ICCI*CC'14)*, 2014, pp. 541–549.

[3] Y. M. Lim, A. Ayesh, and M. Stacey, "Detecting Cognitive Stress from Keyboard and Mouse Dynamics during Mental Arithmetic," *in Science and Information Conference 2014*, 2014, pp. 146–152.

[4] Y. M. Lim, A. Ayesh, and M. Stacey, "Detecting Emotional Stress during Typing Task with Time Pressure*," in Science and Information Conference 2014*, 2014, pp. 329–338.

[5] Y. M. Lim, A. Ayesh and M. Stacey, "Exploring direct learning instruction and external stimuli effects on learner's states and mouse/keystroke behaviours," *in 2016 4th International Conference on User Science and Engineering (i-USEr)*, 2016, pp. 161-166.

[6] Y. M. Lim, A. Ayesh, M. Stacey and L. P. Tan. "The Effects of Task Demand and External Stimuli on Learner's Stress Perception and Performance," i*n 4th TARC International Conference on Learning & Teaching2016(TIC 2016)*, 2016, pp. 115-119.

This page is intentionally left blank

CHAPTER 1: INTRODUCTION

It is important to develop an effective construct to measure users' cognitive performance and emotion state so that automated adaptation can be done to improve user experience and personalized learning. This research is set to examine how such a construct can be developed to measure a learner's cognitive load and emotion state, i.e. stress, automatically and objectively, using a low cost, less-invasive, computational feasible, and fully automated solution. If such a construct can be built, it can then be applied to an affective learning system to enable personalized adaptation, as well as to provide analytical information for teachers to review task demand based on learners' performances and their states. The following sections outline the problems and limitations of the related studies that justify the motivation of this research, the problem statement, the research objectives, the research questions, contributions, the methodology design, the theoretical framework adopted in the studies, and the scope of the research. The last section describes the structure of the thesis.

1.1 BACKGROUND OF THE PROBLEM

Affective computing and adaptive learning take important roles in the new pedagogies that might transform education [1]. Affective computing, as part of human-computer interaction research, takes into account a user's emotional states in order to produce a more usable system, or to influence a user's emotion such as increase motivation [2]. Adaptive learning uses data about a learner’s previous and current learning to provide highly personalized learning sessions through tailoring learning materials or contents, according to the learners' style, profile, interest, previous knowledge level, goal, and pedagogical aspect [3], [4]. The existing e-learning systems such as Blackboard [5] and Moodle [6] solely rely on learners' scores and time spent on a task. However, this is not enough to help teachers to identify a learner's emotion and engagement, which could be affected by the content or the demand of the task.

Existing research relates to affective learning have mostly adopted emotions defined by psychological research, such as the four quadrants of learning emotions as proposed by Kort et al [7], the Positive and Negative Affect Schedule (PANAS) scale by Watson et al [8], or Russell's Circumplex Model of Affect [9]. Although stress is found associated with learning performance [10], [11], there is a lack of empirical research that examines the relationships between learner's stress, cognitive behaviour, learning performance and their intrinsic behavioural characteristics such as mouse and keystroke dynamics. Stress, as according to Selye [12], can be classified into eustress, understress, overstress and distress, which could positively or negatively affect job performance besides health. Among these four types of stress, distress involves unresolved negative feelings of fear, anxiety and frustration, which builds psychological barriers to further learning. Stress is suggested by Lazarus & Folkman [13] as "a feeling experienced when a person perceives that demands exceed the personal and social resources the individual is able to mobilize", and is defined as "a state of mental or emotional strain or tension resulting from adverse or demanding circumstances" by the Oxford dictionaries [14]. From the American Institute of Stress (A.I.S) [15], stress apparently is viewed by most people nowadays as some unpleasant threat, a negative emotion, and as a synonym to distress as defined by Selye. Negative emotion may bring down the learning performance, which may be caused by the task demands itself, or other external factors that are related to the task [16]. If the factor that generates negative emotion can be determined, e-learning developers can redesign the learning process, including adapting the instructions and improving the learning environment, to enhance student's attitude in learning. Therefore, it is important to study how stress can be affected by certain factors, such as task demand and external psycho-physiological stimuli, how it affects learning performance, and how to enable stress to be computed automatically to enable adaptive learning.

There is a challenge in measuring learner's cognitive states and stress. Stress is considered as emotion that is subjective to human perception. To measure stress, objective measures can consist of task demand, available resources such as time duration, and influence of stress stimuli. Yet these objective measures cannot have the relevance and power of direct reporting of feelings about stress, hence it is particularly difficult to find objective criteria against which to validate self-report measures of stress [17]. For instance, given the same task demand and time constraint, two individuals could have different stress perceptions, dependent on how much the individual can tolerate the stress. Therefore, self-report survey is an important tool for the preliminary stage that requires large amount of samples in order for us to study the relationship between stress, job performance and learner behaviours when using mouse and keyboard, which help to build a valuable dataset for the analysis in the later stage. However, self-report survey is not appropriate when it comes to the measurement of a learner’s cognitive state. Cognitive load usually involves processes working with short-term and long-term memory, attention, motivation, behaviour [18]–[22], which is complicated compared to measuring stress alone.

To assess or measure cognitive load, the common approaches are subjective methods, physiological tests and task performance-based measurement [20]. Subjective methods such as surveys require users to perform self-assessment on their mental effort. This is simple but they are often prone to inaccurate and unreliable results. Physiological measurements may provide higher accuracy in measuring mental activities or emotions by collecting biological data, such as heart beat rate and eye activity [23], but they are considered invasive, the equipment are usually expensive and need special setup, hence cannot be implemented as part of normal software system. Task performance-based methods are objective and standardized measure of individual's task performance, cognitive ability, aptitude, and emotional functioning [24]. In a task-specific environment, user cognitive or emotional stress levels can be changed according to demand and control [25]. Misfit between job demands and individual capabilities intensifies the stress effect [26]. Task-performance-based methods are commonly used for socio-psychological research, but they are usually done using social science approach, which is lacking automation in cognitive computation and emotion detection. Other emotion detection methods include facial expressions recognition [27], [28]. Although promising accuracy can be produced, nevertheless special setup is needed, and they can be computationally expensive and intensive, which may be difficult to be implemented online.

To produce a construct that is able to quantify cognitive load and emotion, using a low cost, non-invasive, computational feasible and fully automated solution, some research examines the potential of using mouse or keystroke dynamics. Mouse and keystroke dynamics were initially studied as potential biometric authentication methods but they also demonstrated great feasibility in emotion detection over the past decade [29]–[32]. As standard input devices for a computer, keyboard and mouse enable a completely unobtrusive way of data collection as no special hardware or setup is needed, and can be captured easily during user's usual computer activities. Besides, small amount of features to be extracted also means that they can be easily processed online in order to sense learner’s states in real time, without greatly affecting the server or computer performances. Although both keyboard and mouse dynamics have been shown to differ according to emotion, most previous work has considered them in isolation. There is very little research done that unifies keystroke dynamics and mouse dynamics in emotion detection. The unification of both techniques is important as there is a risk of collecting misleading information from only one channel. For instance, keystroke dynamic analysis could be affected by long stops and irregular restarts [33], e.g. because the task requires the use of a mouse instead of a keyboard. Moreover, in a real application, users may use either the mouse or the keyboard or a combination of both for different tasks.

1.2 PROBLEM STATEMENT

Arising from the problems discussed above, the exact gaps in the knowledge are identified. It is crucial for teachers to understand that a learner's emotion and engagement could be affected by the content or the demand of the task. In order to help the teachers to identify the factors that cause negative emotion and poor learning behaviour, it is not enough to merely provide them number facts, such as duration spent and scores achieved for an assessment, which are done by most of the existing e-learning systems. If the factor that generates negative emotion, such as stress, can be determined automatically, an effective personalized adaptive learning system can be developed to help enhance student's engagement in learning, as well as assisting the teachers to redesign the necessary learning process and materials. To achieve the afore-mentioned, four challenges that must be overcome. First, the existing affective computing approaches, such as physiological measures and audio-visual computing, are either obtrusive, expensive or need special setup. It is not feasible to implement these as part of a normal online system. A cheap, ubiquitous and less invasive means of estimating users' emotion must be sought. Second, existing affective learning research considers emotion from multi-dimensions. It may be important to have a better understanding of the granularity of emotion of the learner. However, enabling measurement of rich granularity of emotion is extremely challenging. Third, numerous existing psychological research reported the effects of stress on job performance and behaviour, but there is a lack of empirical affective learning research that examines the relationships between learner's stress, cognitive behaviour and learning performance, although many other emotions have been studied. It is important to study the effects of task demand and external psycho-physiological stimuli on learner's stress and learning performance, since stress could result in negative feelings of fear, anxiety and frustration, which build psychological barriers to further learning. Therefore, this would be interesting and useful if stress can be measured automatically, as stress could be related to both cognitive stress and emotional stress. Fourth, some research over the past decade has started to examine the potential of using mouse or keystroke dynamics but most of them consider these methods in isolation. The unification of both techniques is important as there is a risk of collecting misleading information from only one channel, since not all tasks require the use of a single device. Furthermore, there is only a little research examining the correlations of a learner's emotions to his/her mouse and keystroke dynamics, although most of them found significant impacts of emotions on learners' mouse/keystroke behaviours. However, there is almost no research that studies the correlations of learner's stress to the learners' behaviours when using these devices to carry out some tasks in an e-learning environment.

1.3 RESEARCH OBJECTIVES

This research has two main objectives. First, it aims to produce the groundwork necessary to produce a cheap, task-independent, ubiquitous and less invasive means of estimating users' cognitive or emotional stress using mouse and keystroke dynamics. Second, it aims to outline possible extensions in affective and adaptive computing research, to build a model of intelligent tutoring system (ITS) that can track individual learner's stress and behaviour. It is believed that the proposed model of ITS would have many valuable application areas, such as providing motivation when necessary, adapting assessment materials according to individual, and providing analytical feedback to an examiner to adjust any possible mismatched expectation.

To achieve the two primary objectives, preliminary research will be carried out to study the relationships between task demand, external psycho-physiological stimuli, stress, cognitive states and mouse/keystroke behaviours, based on e-learning users with a case study at Tunku Abdul Rahman University College, a higher learning institution in Malaysia. Further, an empirical research on stress detection and modelling is conducted from the preliminary research, using artificial intelligence methods such as artificial neural network and fuzzy logic. Two applications of the stress measurement model built on mouse and keystroke dynamics focus on adaptive assessment and analytical feedback to examiner.

1.4 RESEARCH QUESTION AND HYPOTHESIS

The study sought to answer two research questions for achieving the desired solution:

Research Question 1: How can an effective construct that measures learner's cognitive states and stress level be developed by using mouse and keystroke dynamics?

Research Question 2: How can the construct that measures users' cognitive states and stress level using mouse and keystroke dynamics be applied in an intelligent tutoring system?

The challenge to answer the research questions above is there are lack of related research and existing dataset that are useful to construct the stress measurement model using mouse and keystroke dynamics. Therefore, before the first research question can be solved, preliminary research must be conducted to identify the feasibility of the construction of learner's cognitive states and stress measurement by using mouse and keystroke dynamics. The collected dataset will also be useful for us to test the stress measurement model in order to answer Question 1. Three hypotheses given in the preliminary research, which are designed based on the Motivation/Attitude-Driven Behavioural (MADB) model [22], are as follows:

Hypothesis 1: Direct instruction (such as assessment and typing demand), indirect instruction (such as search requirement) and external stimuli (such as menu design, time pressure, clock and/or countdown timer displays) affect stress perception and motivation.

Hypothesis 2: The correlations between instruction, external stimuli, stress perception, motivation, rational motivation, attitude, decision, and behaviour are significant.

Hypothesis 3: Behaviour affects mouse behaviour and keystroke behaviour.

We assume that if the hypotheses above are accepted, then mouse or keystroke dynamics could be considered as sensors that can sense the changes of learner's cognitive and stress level when task demand is changed significantly or when the stimuli is induced.

1.5 CONTRIBUTION

Generally, the research findings will aid in the understanding and application of affective and adaptive computing in educational technology areas, where a cheap, less invasive and task-independent solution that can be easily implemented is required as part of the online learning system. There are two major contributions made by this research as follows.

Adding new theoretical and empirical knowledge of stress measurement model in an e-learning environment using mouse and keystroke dynamics.

This research designs and constructs a stress measurement model that is useful for affective computing practitioners and researchers. The proposed stress measurement model has a few advantages: (1) to enable online monitoring of a learner's affective states by collecting only a few features of mouse and keystroke dynamics; (2) to allow computational measurement of a learner's stress, which is cost effective, less intrusive and no special setup of hardware is needed, and therefore can be implemented as part of a normal system; (3) the solution is task-independent and hence can be applied to any task involving searching, typing or assessment; (4) besides the proposed adaptive assessment and analytical feedback systems, the stress measurement model can be used in many other areas, such as to enhance user-centred design and improve user experience by enabling adaptive interface, for building an affective learning system to detect emotional or cognitive stress of learners.

Three different, preliminary, experimental research that study the effects of stress on learners' mouse/keystroke behaviours, which are reported in Chapters 4 to 6, are conducted as groundwork to build the stress measurement model. Besides helping us to develop the framework for the stress inference engine in the proposed adaptive learning system, the datasets generated from these experiments will also be useful for further related research in the future.

Adding a theoretical framework of a stress inference engine to the affective learning system developers.

The proposed framework of inference engine consists of three components. First is the neural network that estimates the stress level of learners based on mouse and keystroke dynamics. The accuracy of the measurement is validated against an objective measurement of stress based on time duration. Second is the fuzzy classification that classifies the stress level, whether it is increased significantly, decreased significantly or remained normal. Lastly the third component comprises the decision tree that decides when an adaptation of interface and learning content is needed. It identifies the poor learning behaviour that is anomalous during learning. It determines whether or not an assessment is considered significantly demanding, or much easier than expected. This inference engine is useful for the e-learning system developers in many ways. For example, to aid an adaptive system that can reengage a learner for the next learning task, and to generate useful analytical information to the examiners to review the performance of the learners based on their stress levels and behaviours, on top of their scores and duration spent on the task. A prototype is built according to the proposed framework based on the existing dataset, as a proof of concept to demonstrate its feasibility.

1.6 RESEARCH DESIGN

The research is first approached by reviewing the existing literature related to the development of affective learning and adaptive computing. Related psychological and technical literature will also be reviewed to identify the behavioural patterns in relation to stress and user's mouse and keystroke dynamics. Affective computing methods, particularly in detecting stress, will be critically reviewed and evaluated to identify the gaps. Various techniques, particularly dealing with mouse and keystroke dynamics, will be explored in designing the stress detection and modelling model.

The first research objective has sought a task-independent solution. To ensure the same solution works on different contexts even though the user might switch jobs in between, three experiments are set up based on three different tasks that are commonly done in an e-learning environment, i.e. searching for a learning material, assessment and typing. The three tasks are on different job areas, and they require different cognitive load resulting from various interactive elements in the tasks as suggested by Plass et al [34]. They argued that cognitive load is affected by two factors: the number of elements to be simultaneously processed in working memory; and the prior knowledge of the learner. For the first instance, solving mental arithmetic problems involves dealing with higher element interactivity than typing the pre-defined text. For the second instance, searching for appropriate learning materials on a page that is packed with texts may involve higher element interactivity than solving one mental arithmetic problem. These tasks that require higher element interactivity may also require prior knowledge about the element to be solved. Accordingly, the search task is set to study the effect of usability design on learner's stress and mouse behaviour. The assessment task studies the effects of task demand and external stimuli on learner's cognitive stress and mouse/keystroke behaviour. The typing task is set to study the effects of task length and familiarity on learner's emotional stress and mouse/keystroke behaviour. To simulate those tasks in the e-learning environment and to avoid the results to be affected by unfamiliarity with the interface when they begin the tasks, a mock-up application is built based on the learning management system (LMS) that was used by the university students, i.e. Blackboard™ Academic Suite[1]. The targeted experiment subjects are the undergraduate

[1] The institution has upgraded the LMS to *Blackboard Learn* ™- Enterprise License (9.1.100401.0) since 2012 after the experiments were conducted

students from Bachelor Degree in Computer Science, Bachelor Degree in Information Systems, and Bachelor Degree in Information Technology, who are aged between 18 to 24 years old. Participants from narrow specializations and ages are selected under the constraint to control the effect of socio-demographic difference on stress perception [35], when reacting to the interfaces in the search task. Additionally, the items to search are also IT subject-related, about which prior knowledge is needed when searching a desired learning material.

To model how the cognitive process and affective state such as stress drive human attention, decision and behaviours, the MADB model proposed by Wang [22] is adopted, with some slight modification to suit an e-learning environment, and added with mouse and keystroke behaviours to relate a learner's behaviour. The MADB model is explained in Section 1.7 and further discussed in Chapter 2. Models of behaviours, which include mouse behaviour, keystroke behaviour, job performance and learner profile are transformed from raw data automatically each time a learner finished a job. The collected raw data are transformed using the log_{10} function. Due to huge temporal variations of keystroke and mouse dynamics of a user, and also high behavioural differences between individuals, calibrated mouse and keystroke behaviours are collected before the system started the analyses, i.e. during login. Although the learner might have stress even before using the system, which is caused by external factor. However, the calibration is useful to provide a baseline for the system to determine the internal factor that may raise additional stress to the learner, such as the demand of a task, or the design of a learning material. Therefore, we consider the calibrated behaviours as the baseline condition, i.e. normal stress level, which are needed for the comparison with the learner's condition when the first learning activity is carried out. The subsequent behaviours with the previous condition are compared and analysed to determine whether the learner's stress has increased or decreased significantly, or remained stable (normal). To achieve that, the stress measurement model will be tested using three different stress classifiers, namely certainty factor (CF), feedforward back-propagation (FFBP) neural network, and adaptive neuro-fuzzy inference system (ANFIS). These classifiers are considered in the research as they can be useful in managing uncertainties and easily implemented in an online environment. Uncertainties emerge from stress perception variations between individuals even though they are given the same challenge and resources. These methods could also allow stress to be measured continuously over an online environment as they are less complicated in terms of architecture, so that the processing time of stress measurement could be done almost instantly without causing delay to both sides of client and server. The best classifier that produces the best accuracy in stress estimation will be adopted in the construction of the proposed intelligent tutoring system, which enables adaptive assessment and analytic feedback to the examiner.

1.7 THEORETICAL FRAMEWORK OF COGNITIVE STATES ASSESSMENT

Before an adaptive learning system can be built, it is crucial to study how formal cognitive processes during learning can be modelled and measured objectively and automatically. Cognitive load theory (CLT) explains psychological or behavioural phenomena resulting from instruction, and how human cognitive architecture, instructional design and learning are related to each other [34]. It emphasizes devising effective instructional procedures to enhance learning based on the understanding of human cognitive process working with long-term and short-term memory [20]. It also studies how cognitive processes relate to attention, attitude, engagement, motivation [18], [19], [34], and can be affected by emotional factors [22]. Unpleasant or negative emotions could inhibit the necessary resources being recruited for further cognitive process, which prevents optimal skill execution [36]. While motivation and attitude can drive individual's cognitive behaviour and triggers the transformation from thought into action, motivation has considerable impact on behaviour and influences the way a person thinks and feels [37], and whether they are mentally and physically ready to accept and execute learning tasks. Although most of the psychological research related to CLT explained the reasons why emotional and motivational factors should be considered when developing instructional procedures in a learning environment, there exists a lack of standards that devise how cognitive load could be measured objectively, or can be translated into technological solutions. Sensing human behavioural signals may include facial expressions, body gestures, non-linguistic vocalizations, and vocal intonations [38], but these data may be infeasible to be observed all together in real-time without the use of powerful tools, which could be expensive. Fortunately, Wang [22] proposed a model that rigorously and formally treated complicated human emotional and perceptual phenomena based on cognitive informatics theories and denotational mathematics, which was known as Motivation/Attitude-Driven Behaviour (MADB) model. MADB model was based on the Layered Reference Model of the Brain (LRMB) [39] and the Object-Attribute-Relation (OAR) model [40], to describe formally and quantitatively the relationship between emotion, motivation, attitude, and behaviour, and driven in a task-specific environment. Therefore, the MADB model can be easily adopted to measure how motivation processes drive human behaviours and actions, and how the attitude and decision-making process help to regulate and determine the action to be taken. Wang tested his MADB model in a software engineering organization, but we strongly believe that the model also suits e-learning environments. Therefore, it is important to carry out some preliminary research to examine how formal cognitive processes during e-learning can be modelled based on his work.

1.8 PROJECT SCOPE

The research area of this study is considered novel and there is lack of literature to support the especially complex state of human psychology and cognitive states. Accordingly, a few assumptions have to be made so that the research can be carried out. Firstly, we assume that stress perception can be quantified by the learners during the experiment survey. As discussed, it is difficult to find objective criteria against which to validate self-report measures of stress [17], since it is very much dependent on how individual perceives his/her feeling of a task demand and available resource. Therefore, it is assumed that the participants will answer truthfully and accurately to the survey when reporting their personal stress perception. Secondly, if the perceived stress is correlated to a learner's behaviour, and the learner's behaviour impacts mouse and keystroke behaviours significantly, then mouse or keystroke dynamics could be considered as sensors that can sense the changes of a learner's cognitive stress or emotional stress in a task-specific environment. Thirdly, the experiment's subjects are narrowed based on their specialization and age to avoid possible socio-demographic differences that affect the results. It is assumed that the learners have similar abilities in terms of prior knowledge needed for searching a desired material, mental arithmetic skills and typing skills.

There are several limitations of the research design. First, the research is set to only detect stress, which may not be good enough for affective learning. Affective learning usually requires better understanding of granularity of emotion, which is not limited to stress. However, we believe that it is still useful to be able to determine the stressor that causes student's troubled learning behaviour automatically by the e-learning system, which is important for teachers to enhance their learning materials, as well as the development of affective learning. Secondly, sample with narrow specialization and ages also mean the findings cannot be generalized. Thirdly, the limited capabilities of the keystroke and mouse loggers needed to capture the keystroke and mouse data, which are built by ourselves rather than employing a professional, might generate inaccurate data for the subsequent analysis. Fourth, the research only focuses on three common tasks that are carried out during e-learning, i.e. searching for a material, assessment and typing. This does not include other tasks such as reading, watching a video or listening to an audio clip. Lastly, only the design of the ITS architecture will be presented based on the groundwork carried out by this study, but no further empirical research will be carried out to validate the effectiveness of the ITS.

1.9 SUMMARY AND THESIS OUTLINE

The existing problems of the studies related to affective learning and adaptive computing were outlined. The problem statement was given to justify the motivation of the research. Two primary research objectives are presented, i.e. (1) to present the groundwork necessary to produce a cheap, task independent, ubiquitous and less invasive means of estimating users' cognitive or emotional stress using mouse and keystroke dynamics, and (2) to build a prototype of ITS that can track learner's stress states, and produce necessary adaptation to learners and analytic information to teachers. Two main contributions of the study have been identified. The outline of research design was presented, the theoretical framework adopted in the studies was briefly discussed. Lastly, the scope of the research was defined.

There are three main phases in the research. The first phase is important for data collection, as well as to examine the feasibility of using mouse and keystroke dynamics in stress measurement. Experimental studies will be carried out with some e-learning users of a higher learning institution in Malaysia. Three experiments are designed based on three different common tasks in e-learning environment, i.e. search, assessment, and typing. The results and analyses of the three experiments will be reported in Chapters 4, 5 and 6 respectively. The second phase focuses on constructing the stress measurement model and determining the best stress classifiers, namely certainty factors (CF), feedforward back-propagation (FFBP) neural network, and adaptive euro-fuzzy inference system (ANFIS). The detailed setup for the stress classifiers constructions will be covered in Chapter 7. The last phase will focus on designing two possible applications of the proposed stress measurement model for an ITS, i.e. adaptive assessment and analytical feedback to examiner. The detailed architectural design of the ITS, the processes involved in the stress inference engine, the design of adaptive assessment and the analytical feedback system that provides examiners information related to learners' behaviours, will be presented in Chapter 8.

Chapter 2 presents the background of the related studies, by introducing the development of affective learning, the importance of affective learning, and adaptive learning system in Section 2.1. Section 2.2 defines stress, which the study intends to measure. Stress is defined based on existing psychological literature, and cognitive state is measured based on the MADB model presented by Wang [22]. Section 2.3 aims to discuss the problems of the existing affective computing methods, and the emerging affect detection research using keystroke and mouse dynamics over the past decade. The chapter also intends to examine how emotion can be objectively measured by using task-performance-based technique. In order to identify how keystroke and mouse dynamics can be used in the stress measurement model, extensive studies on the current research on keystroke and mouse dynamics, which are useful for emotion detection, will be discussed in detail in Section 2.4 and Section 2.5. Lastly, the background work related to the experiments that involve search task, assessment task, and typing will be presented. This helps to justify the design of the experiments for each of these three tasks, so that the first research question can be answered. This chapter ended with a summary.

Chapter 3 will mainly focus on the design and experimental procedures of the preliminary research. The preliminary research experiments are important to examine the feasibility of using mouse and keystroke dynamics in measuring stress. The data collection during these experiments is vital for us to devise the stress measurement model. Section 3.1 would first provide the definition of stress in this research context. Section 3.2 explains the adoption and modifications of an existing theoretical framework proposed by Wang [22], namely MADB model. This model is used to compute learner's cognitive states using some objective measurements. Section 3.3 explains the design of stress stimuli and stress perception collection method. Section 3.4 briefs the sampling of participants. Section 3.5 then discusses the experiment procedures in the preliminary research. Section 3.6 illustrates the construction of the apparatus needed for the data collection, i.e. key logger and mouse logger, and the mock-up of existing e-learning system for the three different tasks, i.e. search, assessment and typing. Section 3.7 illustrates the features to be extracted for user behaviour modelling. Section 3.8 briefs the analysis methods. Finally, Section 3.9 concludes the chapter.

Chapter 4 mainly focuses on presenting the results and the statistical analyses of the experiments involving search task. Similarly, Chapter 5 discusses the assessment task, while Chapter 6 explains the results of the typing task. Each Chapter 4, 5 and 6 explains the sample collection and the results of the experiments, followed by the discussion and ended by a conclusion.

Chapter 7 presents the second phase of the research study. It first introduces the motivation of building the stress measurement model. Second section presents the validation methods of the performance produced by the three stress classifiers, i.e. CF, FFBP neural net and ANFIS. Section 7.3 then explains the stages of stress measurement and classifier's construction in detail. Section 7.4 presents the results and the statistical analysis of the results. Section 7.5 provides in-depth discussions on the analysis, and followed by a conclusion section.

Chapter 8 mainly focuses on the last phase of the research study. The detailed design of the ITS that applies the proposed stress measurement model will be provided. There are two main objectives in this chapter: (1) to design an adaptive learning system that provides adaptation of learning material when necessary, e.g. when anomalous learning behaviour is detected; and (2) to design a collective feedback reporting system that provides examiner the insights on students' performance and their behaviours when answering the questions. Section 8.1 explains the architecture of the ITS, with the details of each component in the architecture, which include the processes that involve examiner and learners, the inference engine that produces stress classification, the adaptive interface that motivates the identified disengaged learner, and the collective feedback to the examiners. The design of the inference engine will also be discussed in detail on how it produces stress classification and decision for adaptation. The chapter is ended by conclusion.

Finally, Chapter 9 reemphasizes the motivation of the research in Section 9.1. Section 9.2evaluates the limitations of the research and the experiment designs. Section 9.3 discusses the contributions to the e-learning practitioners, researchers and developers. Section 9.4presents potential future work for improvements. Lastly, the thesis is completed by the last concluding section.

This page is intentionally left blank

CHAPTER 2: A REVIEW OF AFFECTIVE COMPUTING IN E-LEARNING ENVIRONMENT

This chapter aims to provide background information related to the research, and to investigate the problems and limitations of existing methods used in constructing an affective learning systems. It also aims to review background information needed for the experimental designs of the three different tasks that are commonly done during e-learning, i.e. searching for learning material, assessment and typing. The chapter will first present the development of affective learning and adaptive systems in Section 2.1. Section 2.2 defines stress related to learning based on existing psychological literature. Stress, either emotional stress or cognitive stress in e-learning environment, is the affect that the study intends to measure computationally and automatically. The measurement of the cognitive state that is related to stress, which is mainly adapted from the Motivational Attitude-driven Behaviour model by Wang [22], will be explained further. Section 2.3 then investigates the existing objective measurements of affects. The problems of the existing affective computing methods will be reviewed, that justify the emerging methods of using keystroke and mouse dynamics. Extensive studies on the current research of keystroke and mouse dynamics, which are useful for emotion detection, will be discussed in detail in Section 2.4 and Section 2.5. Lastly, Section 2.6 presents the background work related to the experiments that involve searching, assessment and typing tasks. The chapter ends with a conclusion of the research background review.

2.1 AFFECTIVE LEARNING

In general, the term ‘affective’ refers to the generation of an affect or emotional response [2]. In recent decades, research in psychology and education has taken affects into account to enhance personalized learning, because of their influence in perception, reasoning, motivation, decision-making and learning [1], [7], [41]–[46]. Emotions, a.k.a. affects, guide social interactions, influence decisions and judgments, affect basic understanding, and can even control physical actions [47]. O’Regan [48] identified the emotions that were critical during online learning. His research positioned emotion as central and essential to the teaching/learning process. Eccles and Wigfield [49] studied the theoretical relations between motivation, beliefs, values and goals, and how these factors affect their achievement behaviours, such as why individuals choose to engage or disengage in different activities. Baker et al [50] found that the factors that cause learning problems and problematic behaviour could be due to boredom and confusion, and the factor for better learning is engaged concentration. These factors are determined by different interface qualities, pedagogical principles, and different materials. O'Neil and Spielberger [41] argued that serious stress and strain, degrade reception and inefficient learning, could be caused by learner's limited memory, attention span or decision-making capabilities despite having strongest motivation. Besides, LePine et al [10] found that stress associated with challenges had a positive relationship with learning performance, and that stress associated with hindrances had a negative relationship with learning performance in a learning environment. They also suggested that these stress-learning performance relationships were partially mediated by exhaustion and motivation to learn. Hence, fluctuation in motivation, losing concentration and unbearable stress that a learner has, are some of the issues that both learner and teacher must deal with.

However, in an online environment, even with the presence of a teacher synchronously, it is hard for the teacher to notice or address any affect-related problems of every learner, hence making him or her fails to recognize those unproductive emotional states like boredom and frustration. It is definitely not enough for the teacher to assess the performance of the learners by tracking only number facts, such as frequency of activities, number of posts, and marks obtained. If the teacher is unaware of the motivational problems of the learners and the factors that cause the students to behave as they do, then the teacher may not be able to foster the learner's concentration, or to improve his or her future performance. Therefore, the detection of emotions using advanced artificial intelligence approaches, which is known as affective computing, could be introduced in the e-learning system to automatically sense how learners experience feelings, engagement, and attention while learning. Automated affect measurement could help teachers to identify those stress factors that cause learner's poor learning behaviour. By discovering the factor that endangers learning, teachers or the affective learning system could adapt the content to reengage the learner's concentration in the subsequent learning experience.

This research will mainly focus on the development of automated affect measurement using objective measurement, using a low cost and unobtrusive solution. However, the research does not include the treatment of pedagogy to see how such system can improve students' learning attitudes. The learning theory [271] that investigates the effectiveness of learning, and studies the students' behaviour during learning will not be under our consideration too. The next sub-sections discuss the development of affective computing and adaptive computing in e-learning systems.

2.1.1 AFFECTIVE COMPUTING IN E-LEARNING

Affective computing, as part of human-computer interaction research, takes into account a user's emotional states in order to produce a more usable system, or to influence his/her emotion by increasing motivation. It can also be defined as methods and techniques that are related to the computer's capability to recognize, model, respond, and express emotions in order to interact effectively with users [51]. Affective computing can be applied in the areas of software engineering, development process improvement, education and e-learning, enhanced website customization, video games, and many other useful applications [32]. When applied in educational technology area or e-learning, there are a few advantages as discussed in the following sub-sections.

2.1.1.1 AUTOMATED COMPUTATION OF COGNITIVE STATES RELATED TO EMOTION, ATTITUDE, MOTIVATION, BEHAVIOUR AND LEARNING PERFORMANCE

Affective computing is important in cognitive computing to build an effective construct to measure users' cognitive performance and emotions, so that automated adaptation can be done to improve user's experience when using an e-learning system. Cognitive load theory (CLT) emphasizes devising effective instructional procedures to enhance learning based on the understanding of human’s cognitive process working with long-term and short-term memory [21], [52]. However, learner’s performance could also be affected by motivational and emotional factors as suggested by Beilock and Ramirez [36]. Therefore, emotion and motivational factors should be considered when developing instructional procedures in a learning environment, in order to ensure that the students are always ready to accept and execute demanding learning tasks.

Cognitive load theory also studies how cognitive processes relate to attention. Wang et al [19] define attention as a perceptive process of the brain, which individual selectively concentrates or focuses the mind and proper responses on external stimuli, internal motivations, and/or threads of thought. According to them, attention is triggered by all five primary sensory receptors, i.e. vision, hearing, smelling, taste and touch, but it is dominantly manipulated by the vision sensory receptor. Attention can also be triggered by derived internal senses of position, time, and motion at the sensation layer. Cognitive performance could also be affected by emotional, motivational and attitude factors. Wang [22] defines emotions as a set of states or results of human perception that interprets the feelings on external stimuli into either pleasant or unpleasant categories. Unpleasant or negative emotions could inhibit necessary resources being recruited for further cognitive process, which prevent optimal skill execution [36]. While motivation and attitude can drive individual's cognitive behaviour, and triggers the transformation from thought into action. Therefore motivation has considerable impact on behaviour and influences the ways a person thinks and feels [37]. Due to these reasons, emotional and motivational factors should be considered when developing instructional procedures in a learning environment, to ensure that the learners are always ready to accept and execute demanding learning tasks.

2.1.1.2 EVALUATION OF LEARNING CONTENT

Landowska [32] suggested that affective computing can be applied in e-learning especially those prepared for self-learning. It is crucial to track fluctuation of motivation and attention of the learning in distance or virtual environment. Failure of doing so could cause the learning processes to be paused or even abandoned. Research in affective computing has been investigating the methods to detect resources that are considered boring [53], [54] frustrating [50] or even stressful [55]–[58]. Information of the student's interaction with resources or interface is needed in order to monitor his/her emotional state, and to identify parts of resources that cause disengagement or weak learning performance, so that the overall learning quality could be enhanced. This application is not only useful for institutions that offer distance learning environments, but also extend the functionality of existing LMS such as Blackboard and Moodle.

Most of the existing LMS offer test analysis functions that provide statistics on overall test performance and individual test questions. Blackboard [5] and Moodle [6], for instance, they offer one key feature named item analysis, that provides discriminative information that helps examiners to recognize questions that might be poor discriminators of learner performance. With this information, the examiners shall be able to improve questions for future test administrations or to adjust credit on current attempts. This feature is certainly good to help the examiners to identify which question is considered good, fair or poor (or easy, medium or hard in terms of difficulty). Questions that are considered good and fair are better at differentiating between students with higher and lower levels of knowledge, while poor questions, which are easy or hard, are recommended for review. However, their analyses rely heavily on the learners' scores of the given test. This is certainly not enough for the examiners to comprehend the mistake made by a student whether is due to the high demand of question, or the student simply gave up or did not pay attention. It is important to note that emotions, attention and engagement are key drivers for learning [59]. If analytics of learner states such as emotions are introduced, the examiners will be able to track which learning students are following, and whether they are distracted, simply guessing answers to quiz tests, or really engaged in learning [1].

2.1.1.3 IMPROVING USER EXPERIENCE IN AN E-LEARNING SYSTEM

According to Kalbach [60], user experience is all the behaviour, thoughts and feelings a person has when encountering a product over time. A good user experience balances elements such as usefulness, usability and desirability. Kay & Loverock [61] predicted the changes in emotions would be correlated to changes in use of computers. Increased happiness and decreased negative emotions should translate into more frequent use of computers. Therefore, they suggested the importance of developing strategies to reduce negative emotions or to promote excitement with respect to promoting use of computers. Besides, Tidwell [62] argued that user interface design affects users' task completion and navigation experience. A study by Lazar et al [63] showed that between a third and a half of the time a user spent on computer is wasted on frustrating experiences. Amongst all the reasons, web navigation appears to be the largest cause of users' frustrations. It also shows that novice users suffer even more frustration than experienced user, as they do not have a lot of computer experience, and therefore can easily get frustrated. Besides, a website that is packed with many features is not necessary usable and effective [64]. When the users find a website unfriendly, confusing, overloaded with too much information, or they are unable to find the information they need, they will leave that site with frustration [65]. On the other side, Bee and Madrigal [66] and Hülsheger et al [67] suggested that satisfaction is positively related to user's enjoyment of the overall experience. In other words, if the overall experience using the system is positive, then the user's emotion toward the system is also positive. Therefore, it is important to take into consideration the emotional state of the users in e-learning environment in order to enhance learning performance and e-learning sustainability.

Affective computing is important in interactive software development and it would be good to have effective metrics to measure users' emotions, so that automated adaptation can be done to improve user's experience. To ensure the success of the e-learning system, it is critical to create a system that supports rather than frustrates users. According to Penna et al [68], the common step to start designing a successful e-learning system is to design usable user interfaces. Designing a usable interface is very important because it has a negative impact on user performance if it is not done correctly [69]. However, Cohen [70] suggested that appraisal of the environmental demands can be affected by many factors such as personality, cognitive styles and current mood states. Therefore, individual person may have different level of appraisal. Even though an interface is designed based on a good standard guideline, not everyone will perceive its usability in similar ways, and not all of them would have the same level of satisfaction of the same system. A survey by Lim et al [35] that studied the users' perceptions of 7 factors, i.e. whether a web page contains (1) confusing features, (2) too many features, (3) inconsistent layout, (4) unrecognisable hyperlinks, (5) no information of user's current location, (6) no explanation of features, and (7) ambiguous terms, found to be consistent with what was stated by Cohen. Their results show that when given the same LMS, users with different socio-demographic background, such as age, gender, experience and role, have varied perceptions and satisfaction with the system designs. Therefore, the one-size-fits-all approach in system design would probably not able to fit in all users' expectations. To improve personalized experience, research in affective computing and adaptive computing has been investigating various methods to detect user's emotion when using the system by measuring certain metrics, such as facial cue [71]–[73], speech and linguistic analysis [74], psycho-physiological state [75], etc., and to provide appropriate adaptation accordingly to engender positive feeling in users.

2.1.1.4 IMPROVING LEARNER-CENTRED DESIGN

The increasing heterogeneity of the users’ population, the diversification of learners’ learning needs and tasks, and the decreasing tolerance of users’ frustration motivate the application of the user-centred model in e-learning design [76]. Besides, due to the great variations in performance between individuals independently of age, interfaces should be tailored for each user or be adaptable [77]. User-centred design is one of the significant criteria to improve the usability of a system as it integrates requirements and user interface designs based on users' needs. By focusing on the end users, we ensure they are satisfied with a more efficient and user-friendly navigation experience, hence their loyalty and return visits will increase as the system supports rather than frustrates them. This will indirectly promote users' active participation and involvement in using the system to help the learners to learn the content more effectively.

Dhar & Yammiyavar [78] argued that a learner-centred design should be adopted over user-centred design when designing an e-learning platform. Learner-centred design (LCD) requires the design to be done by creating a characterization for each learner's profile based on individual personality, learning preferences, learning behaviours or styles, motivation background knowledge, experience with the course content and the system, location and culture, inter alia [79]–[84]. The theory of LCD was raised by Soloway et al [85] in 1994, in which they differentiate between user-centred model (UCM) and learner-centred model (LCM). They claim that UCM focuses on tasks, tools and interfaces (TTI), whilst LCM focuses on tools, interfaces, learner's needs and tasks (TILT). The TILT model suggests some scaffolding strategies for the special needs of the learner. For instance, coaching is needed to help students to acquire knowledge and practices of a task domain, tools must be adaptable to support a learner's growing expertise, and interface must allow learners to communicate and express themselves by the use of different media and mode. LCD in e-learning can be implemented through personalization or adaptive systems. In this approach, an intelligent system is built to personalize and adapt e-learning content, pedagogical models, and interactions between participants in the virtual learning environment to meet the individual needs and preferences. The learner model is an essential component in an adaptive e-learning system since it is used to modify the interaction between system and learners to suit the needs of individual learner [86]. Semantic analysis and intelligent agents appear to be the main technologies to implement personalization for e-learning systems [87]–[90]. Adaptive system makes the content changes automatically to fulfil the requirements of the individual learner.

2.1.2 ADAPTIVE E-LEARNING SYSTEM

The concept of adaptive computing emerged from the capability of a computer systems, such as embedded systems and distributed systems. It adapts one or more of its properties during run-time to improve its performance and design [91]. This provides a means to automatically map an application to specific hardware, and the hardware may be configured to a specific application in order to allow optimal performance. Hence in engineering field, adaptive computing is also known as reconfigurable computing. Unfortunately, this also imposes serious complications for the application developer when developing software for adaptive hardware, due to lack of hardware knowledge [92]. To bring the adaptive systems within reach of applications programmers, the development environment has to handle any hardware issues. As such, visual programming environments allow users to construct complex applications via the connection of basic operations. This level of programming removes any lower abstraction layers, including machine level programming, allowing the application designers to focus on the specific application [92]. Due to this, a further field of application for adaptive computing in computer science enables the researchers to utilize artificial intelligence techniques to allow the content and presentation of a computer program to be adjusted automatically, based on diverse properties such as user's action, user's profile, user's preference, etc.

A newly emerged paradigm of adaptive computing in modern learning approaches is known as adaptive e-learning. Unlike the traditional e-learning system which focuses on the quantity of information, adaptive e-learning must comprise a component called an adaptation system. An adaptation system is the central component of any e-learning system and is "responsible for tailoring learning materials or contents according to the learners' style, profile, interest, previous knowledge level, goal, pedagogical method, etc., to provide highly personalized learning sessions" [3]. Past research of adaptive e-learning proposed many strategies that enable the most appropriate content and presentation to be fitted to each individual user, based on the correct and continuous identification of the user learning styles [4], [93]–[95], personality [84], [96], emotions [96], [97], knowledge/capability [98]–[100] and others, such as web-browsing behaviour [101].

Zafar & Ahmad [3] categorized the e-learning activities as follows:

- Content (knowledge) representation, storage and management;
- Content distribution to variety of users including in-situ and mobile learners;
- Content presentation matching the learners interaction device ranging from desktop, laptop to handheld mobile devices (device adaptation);
- Personalization (adaption) and handling uncertainties related to user and knowledge modelling;
- Assessment of users knowledge or learning.

They argued that the technologies related to the last two are still evolving and have not been standardized yet, and their arguments still remain valid over the last decade. There are still many challenges and difficulties in the sense of technologies that need to be solved. For instance, presently used browsers and devices are having different technological capabilities, such as different support for markup languages, different media types that are supported, different size of the display, colour or sound capabilities, etc. Therefore, an adaptation of the content and the presentation is needed before it can be presented to the user. The adaptation can be done on the server, on a proxy or on the client [3]. Besides, in the distributed environment, it is essential to find ways to improve the collaboration between two software modules, as well as between software and hardware. This can be done through the development of multi-agent systems (MAS) and blackboard system [95], [97], [102]. Artificial intelligence (AI) methods can be incorporated in the design of the adaptive systems. For instance, some imparted psychological tests to categorize student's learning style or personality to personalize the learning process [93], [96]. Alternatively, intelligent agents can be developed to enhance interaction in the e-learning systems in intelligent way [95], [102]–[105]. Other AI research used ontology and the semantic web to represent information with a well-defined meaning, which can explicitly represent the knowledge about users to perform further classification [96], [106].

2.2 EMOTION AND STRESS

Existing research related to affective learning adopted emotion defined by psychological research, e.g. the four quadrants of learning emotions as proposed by Kort et al [7], the Positive and Negative Affect Schedule (PANAS) scale by Watson et al [8], or Russell's Circumplex Model of Affect [9]. It may be important to have better understanding of granularity of emotions that could impact learning performance. However, enabling automated detection of rich granularity of emotions is extremely challenging. Picard et al [44] argued that the affective state is hard to measure and cannot be directly measured [107]. There is a lack of clear theories that define emotions, which are constructs or conceptual quantities with fuzzy boundaries, and with substantial individual difference variations in expression and experience. The biggest challenge is to bring together research of theorists and practitioners from different fields, including psychology, neuroscience, physiology and social science, in order to refine the terminology with respect to affect and learning. Although there is research attempting to give a clear dimension on emotion flourishes in many disciplines and specialties, yet experts cannot agree on its definition [108].

Given that measuring emotions in large scale is difficult, this study aims to measure only stress instead of other emotions. Stress can degrade reception and cause inefficient learning [10], [41]. If possible to be detected automatically, it could be useful for affective computing developers to build effective e-learning that helps to identify the stressors that cause unproductive learning. The stressors may include mismatched demand by the teachers, frustrating resources, or bad usability design, which brings negative effect to learning.

The term "stress" was first coined by Selye [12], [109] in his earlier endocrinological research. Although his original work was unrelated to psychological or educational research, his classification of stress as follows is meaningful as each can positively or negatively affect learning.

Eustress	It is a kind of good stress. This is needed so that the human-being will thrive on some degree of stress in their lives. It is often seen as a motivating factor that stimulates everyone to greater achievements
Understress	It is also known as 'rustout', or under-stimulation. It has a very negative effect, often resulting in boredom, fatigue and dissatisfaction, which often causes a person losing interest in learning
Overstress	This occurs when one pushes himself or herself beyond his/her limits, which leads to the state of fight or flight
Distress	It involves unresolved feelings of fear, anxiety and frustration, which build psychological barrier to further learning

Although Selye argued that stress can be good or bad, most people viewed stress as some unpleasant threat, and was generally considered as being synonymous with distress (http://www.stress.org/what-is-stress). Oxford dictionaries [14] defined stress as "a state of mental or emotional strain or tension resulting from adverse or demanding circumstances". Lazarus & Folkman [13] defined stress as "a feeling experienced that a person perceives that demands exceed the personal and social resources the individual is able to mobilize", which concerned primarily on human emotion and feeling of stress. On the other side, cognitive psychologists identifying stress analytically from the fundamental components of mental life, such as attention and its allocation, memory systems, problem solving, decision making [110]. Therefore, we divided stress into two types: emotional stress and cognitive stress.

2.2.1 THE OBJECTIVE MEASUREMENT OF EMOTIONAL STRESS

The challenge for us to measure stress is to determine solid constructs that can objectively quantify the strength of stress. Objective measures can consist of the task demand strength, available resources such as time duration, and influence of external stress stimuli such as unpleasant environment [17]. Karasek [25] found that in a task-specific environment, user stress levels can be varied according to two factors: *demand* and *control*. Excessive demand such as meeting a deadline, and lack of control over workplace processes could significantly affect work performances. Johnson and Hall [111] proposed the Job Demand-Control-Support (JDCS) model to measure work stress and suggest that an iso-strain job (high demands-low control + low social support/isolation) could bring the most negative impact to the workers. They also believe that social support can moderate the negative impacts of high strain on well-being. Hence, by deliberatively changing the workload, control of tasks and added social support, the user’s stress level can be changed. Liao et al. [57] compare the inferred stress level against job demands through visual features, physiological, behavioural and performance evidences. Their experiments show that the inferred user stress level by their system is consistent with that predicted by Karasek.

Unfortunately these objective measures cannot have the relevance and power of direct reporting of feelings about stress, hence it is particularly difficult to find objective criteria against which to validate self-report measures of stress [17]. For instance, two individuals could have different stress appraisal even they are given the same task and resources, dependent on how they can cope with the stress. If stress is considered as a kind of emotion that is subjective to human perception toward a task demand [112], then self-report survey is an important tool for the preliminary research. Self-report survey is useful when large amount of samples must be collected for us to study the relationship between stress, job performance and learners' behaviours when using mouse and keyboard. It would help us to build a valuable dataset for the analysis in the later stage. However, a self-report survey may not be appropriate when comes to the measurement of a learner's cognitive states. Cognitive load usually involves processes working with short-term and long-term memory, attention, motivation, and behaviour [18]–[22], which is more complicated than measuring emotional stress alone.

2.2.2 THE OBJECTIVE MEASUREMENT OF COGNITIVE STRESS

Wang et al [19], [22] demonstrated work to show how the complicated human emotional and perceptual phenomena can be rigorously modelled and formally treated based on cognitive informatics theories and denotational mathematics, which is known as the MADB model. This provides a good base to examine the effects of demand and external stimuli on stress perceptions, cognitive states and behaviour of students during the search tasks. They argued that cognitive performance is related to attention, and could be affected by emotional, motivational and attitude factors. Attention is a perceptive process of the brain, which individual selectively concentrates the mind and focuses proper responses on external stimuli, internal motivations, and/or threads of thought. It is triggered by all five primary sensory receptors, namely vision, hearing, smelling, taste and touch, but dominantly manipulated by the vision sensory receptor. Besides, attention can also be triggered by derived internal senses of position, time, and motion at the sensation layer. Emotion is a set of states or results of human perception that interprets the feelings on external stimuli into either pleasant or unpleasant category. Unpleasant or negative emotion can prevent optimal skill execution and is bad for learning, as it could hinder necessary resources being employed for further cognitive process by human mental [36]. On the other side, the motivation and attitude of an individual can drive cognitive behaviour, trigger the transformation from a thought into an action, and have considerable impact on human behaviour, as well as influencing the ways a person thinks and feels [22], [37]. Due to these reasons, emotional and motivational factors should be considered when developing instructional procedures in a learning environment, to ensure that the learners are always ready to accept and execute demanding learning tasks.

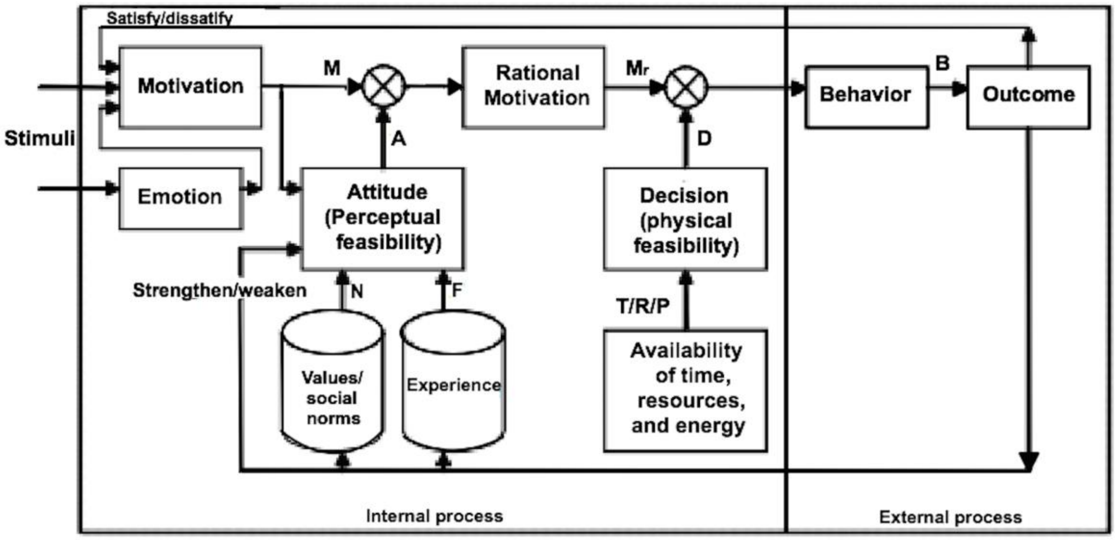


Figure 2.1. The MADB model proposed by Wang [22]

Figure 2.1. illustrates the MADB model proposed by Wang [22], which is based on the Layered Reference Model of the Brain (LRMB) [39] and the Object-Attribute-Relation (OAR) model [40]. The model can be used to formally describe how the motivation process drives human behaviours and actions, and how the attitude and decision-making process help to regulate and determine the action to be taken. Based on the MADB model, the strength of a motivation, M, is proportional to both the strength of emotion $|E_m|$, and the difference between the expectancy of desire E, and the current status S of a person. The mode of an attitude, A, is determined by both an objective judgment of its conformance to the social norm, N, and a subjective judgment of its empirical feasibility, F. A rational motivation M_r is defined as a motivation regulated by an attitude A (with a positive or negative judgment). A decision D is raised based on the basis of the availability of time T, resources R, and energy P. Lastly, behaviour B is driven by a rational motivation M_r and supported by a positive decision D toward the action. His research demonstrated how the MADB model was applied in a software engineering organization, but we envisage the model can also be fit into the e-learning environment. We would like to examine how formal cognitive processes during e-learning can be modelled by considering student's motivation, attitude and behaviour.

In the next section, some affective computing methods that are useful in the automation and computation of stress measurement in online learning environment will be discussed.

2.3 AFFECTIVE COMPUTING METHODS FOR E-LEARNING SYSTEM

If the emotion, mental load, cognitive efficiency, or a learner's instruction condition can be measured or diagnosed, it is believed that the effectiveness of learning can be enhanced, as adaptive learning materials and customized assessment can be given based on individual needs and performance. However, designing such measurement is challenging especially as there are a lack of data that can accurately define human emotions or cognition. Besides, we have four main concerns for building an affect monitoring system in a web environment: (1) the monitoring process should be continuous, (2) the method should be non-invasive, (3) the method should be cost-effective, and (4) the measurement of stress should be reliable [46], [113]. Firstly, the affect monitoring system must be able to collect the inputs from the learner's computer continuously once he has logged into the system. Then it should respond accordingly once the learner's behaviour is detected anomalous. Secondly, the affect measurement system should be unobtrusive so that it can capture the responses from the learners without creating additional stress. Ideally the users should not even be aware that they are being observed, and they can carry out their tasks as usual. Thirdly, considering in a web environment, the data are transferred using hypertext-transfer protocol (HTTP), therefore it is necessary to limit the amount of data transferred to and from the server to reduce the computers' load. The processing time should be done almost instantly without causing delay to both sides of the computer and server. Although the hardware and software are getting more advanced nowadays, but one can imagine the amount of data created from the users would be huge if the process has to be done continuously. Lastly, having a reliable measurement of emotional stress is most important. In 2004, Picard and his team [44] identified five main gaps in the methods for affective learning, one of them is to seek reliable measurements of emotional states symptoms, which is still remained as a challenge nowadays. Besides, the measurement should be context-independent, so that it can be applied regardless the type of task carried out by the user. In other words, the accuracy of the stress classification should not be affected even the student swaps between tasks, or he is already stressed out even before using the system.

The following sub-sections identify the existing affective computing methods and their limitations.

2.3.1 THE EXISTING AFFECTIVE COMPUTING METHODS

It is really a challenging task for computer engineers or scientists to compute a user's emotional state objectively as there is lack of solid definition and ground-truth to define emotion. To "quantify" these emotions, some research in computer vision utilizes the six "basic" emotions suggested by Ekman [114] that are readily manifested in facial expressions: sadness, happiness, anger, fear, disgust and surprise. Besides, video or audio inputs that enable face expressions, speech or body movements recognitions [72], [115]–[117] are possible to be automated, considered non-intrusive, and enable objective measurement of various types of emotions. Nevertheless, the assessments require the user to turn the input device on, which means it could also be switched off by users easily. Besides, the measurement of affect states using visual processing is often computationally intensive due to the large amount of data extracted. Therefore, this could overload the computer if the process has to be done continuously for long period.

Physiological tests are widely used to detect changes in cognitive functioning [20] and emotion [32] that are reflected in measurable physiological measurements, such as heart rate or eye activity [43], [45], [56], [118], [119]. Liao et al. [57] compared the inferred stress level against job demands through different modalities, which include visual features, physiological, behavioural and performance evidence. Their research showed that the inferred user stress level by their system is consistent with that predicted by psychological theories. Heiden et al [120], Zhai et al [119], Bennett et al [121], Setz et al [122] and many more also found physiological evidences that job demands or cognitive load affects human psycho-physiology. Although using physiological method is effective, it cannot be easily implemented without special equipment, and the additional equipment requires extra costs and labour maintenance, hence it is less flexible, i.e. cannot be easily integrated into normal system. Furthermore, physiological tests could be invasive to the users as some sensors need to be attached to human bodies, hence the users may not feel comfortable in carrying out the task normally.

To enable a more feasible objective measurement, some consider text analysis that heavily research the sentiment features from the text typed by the user [55], [123]. Other tools in text analysis also include web-logging, web proxies and activity logging to identify the activities undertaken in the platform by the users [124]. These methods do not require additional equipment and massive data can be obtained easily for analysis, but they still have their own constraints and may be difficult to construct. For instance, not all activities in e-learning require text inputs. Besides, tools that collect data on user activities have potential privacy implications [125], which data that others would find sensitive, damaging, or private could be gathered unintentionally. Furthermore, if the analysis is not carefully designed, noise or unwanted data could have affected the results.

2.3.2 THE EMERGING AFFECTIVE COMPUTING WITH KEYBOARD AND MOUSE

The introduction of the methods using non-visual peripherals and non-intrusive equipment, i.e. mouse dynamic and keyboard dynamic analyses, sheds light to a more flexible and inexpensive affective computing research. Maehr [126], Schuller et al [127] and Tsoulouhas et al [54] classify user's emotional states such as boredom, surprise, joy, anger, fear, disgust, sadness, happy and neutral using mouse dynamic analyses, while others utilize keystroke dynamic analyses to detect various emotional states such as stress, boredom, etc. [53], [55], [128]–[133]. Although most of these studies considered these methods in isolation, they produced promising results which are comparable to those using physiological or visual processing methods.

Standard input devices, such as a keyboard and mouse, enable a completely unobtrusive way of data collection as no special hardware is needed, and can be captured easily during user's usual computer activities. Features extracted from keystrokes may be divided into timing and frequency parameters. Mouse characteristics include both clicking and cursor movement measurements [134]. The amount of features to be extracted is relatively small compared to visual processing, which is suitable for continuous monitoring process. Besides, similar to physiological inputs, user's mouse dynamics and keystroke behaviour are considered intrinsic or behavioural characteristics, which could reflect the changes in cognition function and be captured into measurable responses such as typing pressure, speed and mouse movements [135]–[138]. Some research showed that keystroke dynamics and mouse dynamics are associated with boredom [53], [54], physical and cognitive stress [55], [139], emotional stress [113] and many other emotions [126]–[130], [140], [141] (see Table A1.1 in Appendix I). Therefore, mouse and keystroke dynamics analyses have the potential to be used for stress measurement in a web-based system, considering the collection and process of data can be automated, cost-effective, non-invasive and possible to provide more reliable measurement than subjective method.

However, by using mouse and keystroke dynamics analyses alone may not be sufficient. First, only small amount of information can be retrieved. The information produced by these devices is unstructured and differed from each other. Furthermore, different tasks require different devices to be used, and one would be idle for long time when another is in use. For instance, if we only analyse keystrokes, the results may be affected by long stops and irregular restarts [33]. A long stop could be due to the user’s attention being diverted to another activity, or the mouse device is used to perform an action rather than the keyboard. Moreover, in a real application, users may use either the mouse or the keyboard or a combination of both for different tasks. To enable better classification rates, mouse and keystroke dynamics analysis should be combined with other technique, such as task-performance-based analysis, to increase the reliability of the results.

The next section introduces and reviews the existing research related to keystroke dynamics and mouse dynamics.

2.4 KEYSTROKE DYNAMICS-BASED ANALYSES

Keystroke dynamic more easily understood as typing rhythms of a user using a computer keyboard. It is the detailed timing information that is normally represented by two basic measures: the time a key is depressed and the time the key is released. Such information is used to compute the duration of a keystroke and the latency between two consecutive keystrokes. From two consecutive keystrokes we may extract the elapsed time between two key events [142], [143], which are:

keystroke latency	The release of the first key and the depression of the next, also called flight time, or up-down time;
keystroke duration	The amount of time each key is held down, also called dwell time, key pressed time, or down-up time;
diagraph	The elapsed time between the depression of the first and of the second key, or down-down time;
trigraph	The elapsed time between the depression of the first and of the third key. As an example, suppose that a user is asked to type the text: surprise. The outcome of the sampling, when using trigraphs, could be the following: *sur 277; urp 255; rpr 297; pri 326; ise 235;* (where next to each trigraph is its duration in milliseconds).

Other key features that have been used are key press event (left click, right click), pause occurrences (how many times the user pauses typing), key codes (e.g. SHIFT key), key rate (how often a key is pressed), the elapsed time between 2 subsequent keys are released (up-up time), correction key use rate (backspace and delete key), punctuation key use rate, and a set of keys pressed (e.g. alphabets) [144].

The recorded keystroke timing data is then processed through specialized algorithm, which determines a primary pattern for future comparison. Hence, it can be used in both identification and authentication tasks, which is generally known as behavioural biometrics [145], or also be considered as a soft biometric [146]. Although it is commonly held belief that behavioural biometrics are not as reliable as physical biometrics that are used for authentication, the performance of keystroke dynamics was shown falls with respect to physical biometrics, such as fingerprints [147].

There are many potential advantages of using keystroke dynamics in data analysis [139], [140], [145], [147], [148]:

- It is believed that some characteristics of keystroke information, which form a unique biometric template of the user’s typing pattern, are as individual as a signature.

- It is neither obtrusive nor intrusive, as user will be using computer keyboard anyway. Furthermore keystroke dynamics, by design, has a non-invasive user interface. It can be implemented to quietly capture user typing during normal operation, thus making user unaware of the process.
- It allows data to be captured continuously over a length of time (unlike interview or psychological test where data can only be gathered periodically).
- We can further leverage behaviours in which the individual is already engaged.
- It is relatively inexpensive to implement, since it is already an essential hardware of the computer, hence it requires no extra equipment such as scanner or camera.
- Since no special equipment is needed, no extra human resource is needed for client-side installations or upgrades of the hardware.
- Almost every workstation has a keyboard; thus, the process of recognition is all done based on software only. With a software-only solution, users are not limited to individual or specific workstations.
- The technology can be embedded in any in-house software application.
- This technology does not require changes in existing network access policies.

However, as keyboard device only allows small amount of information to be retrieved, insufficient data collection could lead to less accurate data. For instance, a research by Kang and Cho [149] shows that equal error rate (EER) would increase when text size or number of keystroke characteristics decreases. Therefore, most of the research choose to use fixed text analysis to ensure a minimum number of keystrokes per sample in order to produce good results in their building models. Some also tend to improve the accuracy of the results by analysing the actual content typed by the users. The next sub-section discusses the differences between fixed text analysis and free text analysis.

2.4.1 KEYSTROKE DYNAMICS WITH TEXT ANALYSIS

Keystroke features (such as timing and key code) are usually combined with the analysis of the content typed by the user. Clues that show unique behavioural patterns are believed could be found from the text that user has typed. For instance, "the", which is a very common English word, along with common endings, such as “ing” may be entered far faster than the same letters in reverse order ("gni" or “eht”) to a degree that varies consistently by individual; the choice of words such as “behaviour” and “behaviour” for a nation who learns both American and United Kingdom English; and the common grammatical errors that are done by individual could also be varied by individual. Therefore, keystroke dynamics with text analysis could be used for user identification, after the process of authentication. Generally there are two types of analyses are done: fixed-text (static) analysis and free-text (dynamic) analysis [148].

2.4.1.1 FIXED-TEXT ANALYSIS

Fixed-text analysis means essentially that the analysis is performed on typing samples produced using the same predetermined and static text for all the individuals under observation. Most keystroke authentication methods fall within this category, e.g. [33], [128], [133], [145], [150]–[155]. The analysis of the fixed-text is often much easier than the free-text, where the words and languages used by the user will usually not to be included in the analysis. Besides, most research limit the sample data to be produced from structured and predefined text, such as password and static text, from only a few words to a few hundred words. Although the attained level of accuracy is far from being acceptable, or good performance could be achieved under very special conditions, it is hard to maintain in real applications [137], as most of the time users do not type predefined text in their work.

2.4.1.2 FREE-TEXT ANALYSIS

While fixed-text analysis limits the text used to perform analysis on keystroke dynamics, free-text analysis allows users to freely type whatever they want with any length. Nevertheless, Gunetti & Picardi suggest that a sample of around 800 characters should produce sufficient data for analysis. In a free-text analysis, classification is performed based on the available information entered by the user, therefore we could also refer the analysis of free text as dynamic analysis [148]. The advantages by extracting the n-graphs shared between two samples are to allow typing errors to be detected, and we could also compare samples made using different languages, provided the two languages share some legal n-graph [148]. The literature on keystroke analysis of free text is pretty limited, and the application of free-text analysis is mainly focusing on authentication [33], [146], [148], [149], [156], [157]. Generally the classification results from free-text analysis are considered less promising than fixed-text analysis, but some outcomes of identity recognition could reach a false rejection rate (FRR) of 4.83% and false acceptance rate (FAR) of 0.00489% [148].

The motivation of using free-text analysis to detect emotion is raised by some researchers in the keystroke dynamics-based authentication (KDA), which they argue that the recognition of user identity may be expected to "vary greatly under operational conditions in which the user may be absorbed in a task or involved in an *emotionally* charged situation" [156]. Bergadano et al. [137] also had similar viewpoint, where they argued that absolute values, such as keystroke duration and latency, may "greatly vary with the psychological and physiological state of the person providing the sample, but it is reasonable to expect the changes being homogeneous, affecting all of the typing characteristics in a similar way". Epp [128] also argued that the performances of keystroke dynamics are lower than other biometric modalities, and could be affected by the intra-class variability of the users behaviour that are pertaining to different typing behaviour when they are nervous, or angry, or even sad. Vizer et al. [55] explored the relationship between exposure to stress and changes in keystroke and linguistic features, by testing three different features when analysing the text, i.e. timing features, key features and text features. Timing features include time per keystroke, duration of pause and pause rate.; key features are the key codes such as deletion keys (e.g. delete, backspace), navigation keys (e.g. left arrow key), punctuation keys and other keys (e.g. SHIFT key); text features include language diversity (e.g. lexical diversity such as unique words), language complexity (e.g. noun, verb), cognitive operations (e.g. "think" and "reasoning"), language expressivity (e.g. modifier such as quickly, happily), affect words (e.g. "hate" and "like"), perceptual information (i.e. sensory information such as "hear", "feel"), and other non-immediacy words (such as "we", "everyone", "can", "not", "is"). As the results, the users demonstrate different typing behaviour and choice of text features under cognitive stress and physical stress.

2.4.2 DEALING WITH KEYSTROKE DYNAMICS-BASED ANALYSIS

Table A1.2 in Appendix I shows the summary of some research papers related to keystroke dynamics-based analyses. In general, most of them were done using fixed-text analyses, and the results of fixed-text analyses are more desirable and promising than the free-text analyses. Although the research by Monrose and Rubin [156] did not demonstrate successful recognition based on free-text, the research by Gunetti & Picardi [148] and Vizer et al. [55] show the contrast. One interesting observation from the summary table is most papers are using supervised learning. Supervised learning is useful to discriminate between users in a closed setting in which data can be collected for all the users, i.e. all users' data must be collected before classification can proceed. However, in real-life operational environments, this technique may be hindered by the non-uniform class problem as the number of classes may increase. In the case where public access to hosts is not restricted, unsupervised learning may be more suitable [158].

The experiments on keystroke dynamics may also face a risk of collecting bad samples. For instance, user's typing may vary substantially after a period of time, and their typing behaviour may change not due to stress, but caused by other factors such as sleepiness, intoxification, change of computer hardware and software, use of virtual keyboard, hardware defective, injury of hand or finger, or the user simply using a voice-to-text converter (instead of typing). Different keyboard designs may also affect the typing behaviour too, and different keyboard types may also affect the classification algorithm designs [149]. The variation of keyboard design parameters such as distance between keys, size of keys and requirement of pressure, are built to accommodate users with different physical abilities. Hence people with especially large or small hands may have difficulty in using standard keyboards [159].

Due to the above variations, there will be error rates to the system, and a valid solution that uses keystroke dynamics must take these elements into account. Although we may not be able to solve the external problems that we couldn't control, such as the physical wellness and life style of the users, we could take into considerations detecting a learner's states when he is doing e-learning using objective measurement. This will help his teacher or the adaptive system to analyse the effectiveness of the learning, to identify potential stress created by the design of the system or the learning materials. The objective measurement of stress could include keystroke dynamics-based analyses with other techniques. For instance, we could use machine learning to allow the system to learn the user's typing behaviour over time. We also propose a solution that combines both keystroke dynamics and text analysis, with mouse movement analysis since mouse is also an essential part of computer hardware, which is also non-obtrusive and not costly. Most importantly, some researchers use this technique to model user behaviour, and it is proven as good as keystroke dynamics [160]–[162]. More details of mouse dynamics recognition will be discussed in the next section.

For the text analysis, there are also some issues to consider, especially the length of the text samples. If it is too lengthy, then users will consider it a nuisance; if it is too short, then it will reduce the accuracy of the classifier. In addition, Monrose & Rubin [163] argue the limitation of using free-text recognition, of which the input is unconstrained, i.e. the user may be un-cooperative and environmental parameters are uncontrolled. Revette et al. [145] also observed that the experiment participants claim that "periodic checking of their typing style is obtrusive and considered as an unacceptable invasion of their privacy". Therefore, full consideration must be taken on the length of the text inputs, and participants must be briefed clearly on the expectations before they voluntarily take the roles.

2.5 MOUSE DYNAMICS-BASED ANALYSES

Mouse dynamics refer to the characteristics of user's actions received from a mouse input device while interacting with the graphical user interface. Mouse input devices may include similar devices that control the cursor movements, such as track ball, touch pad, touch screen. A mouse action can be classified into one of the following categories[2]:

Mouse pressed:	A mouse button is pressed
Mouse released:	A mouse button is released

[2] Java SE Documentation, 2014. Class MouseEvents. Oracle. Available at: http://docs.oracle.com/javase/7/docs/api/java/awt/event/MouseEvent.html

Mouse clicked:	A mouse button is clicked (pressed and released)
Mouse entered:	The mouse cursor enters the un-obscured part of component's geometry
Mouse exited (or out):	The mouse cursor exits the un-obscured part of component's geometry
Mouse motion (or movement):	The mouse is moved and the position of the cursor is changed
Mouse dragged:	The action starts with mouse button down, movement, and then mouse button up
Mouse wheel scrolled:	Mouse wheel events include scroll type (wheel unit scroll or wheel block scroll), scroll amount (number of units to be scrolled) and wheel rotation (rotated up or down)
Silence:	The elapsed time of no movement.

Therefore, there are many secondary data can be derived from the mouse actions, such as:

Mouse click data:	Pressed duration (the elapsed time between a mouse button is pressed and released), click occurrences (number of times a click event (left, right or double click) occurred)
Section hovering rate:	The number of times the mouse cursor entered and hovered a particular section on a page
Mouse movements data:	As mouse movement speed or velocity, acceleration, movement direction or angle, and travelled distance/curvature,
Mouse dragged data:	Mouse drag rate, drag duration (the elapsed time between a mouse button is pressed, moved and released)
Mouse wheel data:	Scroll speed, scroll duration, scroll rate, scroll occurrences

Mouse dynamics share the same advantages as keystroke dynamics, where it is not obtrusive nor intrusive, less costly, does not require additional special equipment and special setup, allows continuous monitoring, and can be applied on a keyboard-less application, such as touch-screen kiosk, touch-screen based ATM and point-of-sale systems.

Although the literature of mouse dynamics-based analyses is pretty limited, existing research papers show that the classification results are quite promising, mainly in the area of authentication or identity recognition. This is achieved by building a profile of each authorized user based on his/her mouse dynamics during enrolment, and the current behaviour can then be classified against the user profile into either a genuine or intrusive behaviour [127], [135], [136], [158], [160], [161], [164]–[169]. Some research even shows that mouse dynamics-based authentication is better than some well-established biometrics such as voice and face recognition systems [136],

[160], [161], [164], [165], which the outcomes of an identity recognition could reach the best false rejection rate (FRR) of 0.36% and false acceptance rate (FAR) of 0.0% [158].

Besides being used for authentication during a login process, mouse dynamics-based analysis is believed to be useful for building a personalized system. Chudá & Krátky [170] propose a user modelling process specialized for user identification in browsing the web, and a system that is able to track mouse dynamics characteristics and user's personality profile. Their results show a distinctiveness of individual characteristics among visitors, and the performance of their proposed identification method reaches an accuracy rate of 87.5%.Chudáand Krátky [138] also conducted a preliminary research to detect student’s cheating behaviour and learning style based on mouse dynamics gathered in an e-learning system. In some of the mouse dynamics-based emotion detection research [54], [126], [127], [138], and some other research that combines mouse dynamics with keystroke dynamics [113], [140], [144], they have achieved accurate recognition rates of above 80%, and a few of them also tested significant correlations between some mouse dynamics features and affect arousals.

Table A1.3 in Appendix I shows the summary of the past mouse dynamics-based research.

2.5.1 DEALING WITH MOUSE DYNAMICS-BASED ANALYSIS

Mouse dynamics recognition could offer promising and even high accuracy rates if the modelling design is done well. This approach also shares similar advantages as keystroke dynamics recognition where it does not require additional hardware or special setup, and it is unobtrusive and less computationally intensive, where it enables data to be captured continuously over a length of time. However, there are several limitations of utilizing mouse dynamics recognition alone. For instance, Shen et al. [160] define variability in mouse dynamics as variations of a user’s distinctive mouse operation patterns caused by the changes of the following factors:

a) Environment settings such as the height of the chair, the distance between mouse and body, usage of new mouse, etc.;
b) GUI settings such as screen resolution, pointer speed, etc. ;
c) Application scenario such as Internet browsing, etc.;
d) Skill level of a user, i.e. a user becomes more practiced in some operations or more accustomed to a change in the related settings;
e) Emotional states of a user: anger, despair, happiness, nervous, excitement, pressure etc.;
f) Physical conditions of users: tiredness, illness, etc..

The above factors bring uncertainties to mouse activities of a user and can have a serious impact on the accuracies of mouse dynamics for its application to personal identity recognition. Hence, Shen et al. [160] proposed a framework, which is called dimensionality reduction based approach, for how to tackle these problems that may result in the decrease of classification accuracy by these variations. However, our research interest is not focusing on the authentication but emotion recognition. Since mouse dynamics can be varied due to factor (e) above, we would like to examine how stress can be detected through the analysis on computer mouse activities.

Another obvious limitation is that this method seems hardly to be utilized alone. Many factors may affect the accuracy of the data collected. For instance, if the user does not use the mouse or rarely generate mouse events, this method will fail; any technical problems that occur on the mouse, e.g. malfunction, could also affect the results; laptop systems that are generally equipped with touch pads or touch screens may exhibit completely different user behaviour compared to using an external mouse. Therefore, the restriction should be applied to the participants, that they must use a normal, external, mouse device during the experiments. However, it is very difficult to control during the real-life operational environments. Therefore, the best solution is to combine this approach with other techniques.

2.6 THE BACKGROUND STUDY FOR EXPERIMENTS SELECTION

This section covers the background and literature review that are related to the research and experiment designs, to enable us to justify the selection of the approaches that can lead us to the solution. The main research consists of three different phases. The first phase focuses on preliminary research that examines the effects of three general tasks in an e-learning environment, namely search for a learning material, assessment and typing, on user's cognitive or emotional stress. The second phase focuses on designing and building a stress measurement system using various algorithms, i.e. certainty factors, artificial neural network and adaptive neuro-fuzzy inference system. Lastly the research aims to demonstrate how the stress detection system can be applied in an intelligent tutoring system. The following sub-sections present the related background of each component that needs to be done in this research.

2.6.1 AFFECT MEASUREMENT BASED ON TASK PERFORMANCE

Task-performance-based technique measures actual performance of the given tasks. This technique is also more reliable than the subjective method, as quantitative data such as success and failure rates of the task could be collected. Furthermore, some psychological theories find that in a task-specific environment, user stress levels can be varied according to two factors: demand (e.g. excessive demand on worker production, especially to meet a deadline) and control (e.g. lack of control over the process) [25]. Therefore, employing a task-specific environment in the experiments is also believed to be more relevant to a real-life e-learning environment, which user stress can be induced by deliberatively increasing the workload and reducing the control of task. Compared to most of the past research that induced users' emotions by visual (such as video, images) or audio (such as storytelling, jokes) effects, the latter are considered not applicable to the real-life application [128]–[130], [133]. It is important for an affective learning system to be capable of identifying the actual stressors that trigger negative emotions of users in order to provide the best appropriate action to tackle the issues.

Therefore, three different tasks that are commonly done in an e-learning environment, in which learner stress perception of the demand is measured, will be designed, i.e. searching for a learning material, assessment and typing. Since the three tasks are on different job areas, they may require different cognitive load resulting from various interactively elements in the tasks as according to Plass et al [34]. They argued that cognitive load is affected by two factors: the number of elements to be simultaneously processed in working memory, and the prior knowledge of the learner. For instance, solving mental arithmetic problems involves dealing with higher element interactively than typing the pre-defined text, and searching for appropriate learning materials on a page that packed with text may involves higher element interactivity than solving one mental arithmetic problem. The first search task is set to study the effect of usability design on learner's stress and mouse behaviour. The second assessment task studies the effects of task demand and external stimuli on learner's cognitive stress and mouse/keystroke behaviour. The last typing task is set to study the effects of task length and familiarity on learner's emotional stress and mouse/keystroke behaviour. The justifications of the experiment designs are given in the following Section 2.6.2. Section 2.6.3 discusses the stress classifier's learning and construction.

2.6.2 RESEARCH EXPERIMENTS

The following sub-sections provide some background required for the design of the three preliminary research experiments.

2.6.2.1 MENU SEARCH EFFECTS

To ensure the success of an e-learning system, it is critical to create a system that supports rather than frustrates users. The common step to start designing a successful e-learning system is to design usable user interfaces [68], because it has a negative impact on user performance if it is not done correctly [69]. User interface design does not only affect users' task completion and navigation experience [62], but also their satisfaction and enjoyment of the overall experience [66], [67]. Some research found that web design has significant impacts on users' navigation experience such as visual search and information retrieval performances. These impact factors include the appearance of hyperlinks, font size and type, colour, text length, line space, frame layout, background contrast, spatial layout, use of ambiguous terms that are difficult to understand and confusing, and poor organization and grouping of information [60], [63], [171]–[176]. However, existing research has not studied the effects on user's task performance when these factors are combined into a single web page. We envisage the factors may be significant when they are tested individually, but the interaction between these factors may also intensify the impact on users' navigation experience. Furthermore, it is unusual that web page design should only focus on one or two factors for real application development. The exploratory research hopes to identify the relations between these design factors and user stress perception. If there is a way to objectively measure the amount of the impact of user interface design on user’s task performance and navigation experience, then it will be much easier for most developers, whose include graphic designers, programmers, or content developers, to create a more usable e-learning system to ensure effective learning in a virtual environment. Accordingly, experiments will be developed to combine these factors into different menu designs for the participants to search for varied learning materials. Correlation tests will be conducted to analyse the relationships between the six design factors and the participants’ stress perceptions.

To determine the factors that could cause negative feelings (such as frustration, dislike, uncomfortable, etc.) to users in web environment, particularly in menu search task, we reviewed some related papers and filtered the factors as below.

- **Colour**: Certain colour combinations are found to reduce the accuracy, speed, legibility and visual search performance when users search text or icons on a web page. From the research findings by [173], [177]–[179], there is a common colour combination that is ranked inferior to legibility on a web page and decrease the speed in searching, which is green and red. Besides, Shieh and Chen [180] also found that red text on green background resulted in shortest view distance from the display device. On the other hand, the research by van Schaik and Ling [181] showed a significant effect of link colour both on accuracy and rated display quality, with blue links on a white background resulting in better outcomes than black links on a white background. However most of the above mainly tested the colour effect using luminance contrast with the cathode ray tube (CRT) display in their research, but the users could perform better with the liquid crystal display (LCD) [182], thin film transistor LCD (TFT-LCD) or light-emitting diode (LED) monitor.

- **Typography:** Ling & van Schaik [173], [174] and Shieh et al [177] found that typography affects performance significantly. Bernard et al [183] compared serif and sans serif fonts in 12- or 14-point size in a task where participants had to detect substituted words in text. They found that 14-point fonts were more legible, led to faster reading, and were preferred to the 12-point fonts. However, their participants preferred sans serif fonts although they performed tasks more quickly with serif fonts. Ling & van Schaik [174] also stated that their experiment subjects had a preference for Arial (sans serif) over Times (serif). Both research by Bernard et al and Ling & van Schaik found no differences in performance across a range of widely used fonts. However, as cited by Mills [184], the optimal size for characters on a computer display depends on the type of task being performed. Smaller characters produced faster reading speeds for reading task, but larger characters produced faster search times for menu search task.

- **Text Length:** Some research shows that users prefer shorter line lengths than the long one. According to Briem [185], shorter line lengths are easier to scan than longer ones. Ling and Schaik [174] suggested that although longer line lengths facilitate faster scanning, shorter lines may be better for accuracy. Users performed better with a line length of 70 characters in their experiments. However, there is lack of research examine the effect of hyperlink length on menu search task.

- **Menu Organization:** Earlier research shows that in general search tasks, alphabetical organization performs slightly better than random menu organization, but users' performance on search tasks improved directly when categorical menu organization was used [186], [187]. Mehlenbacher [188] suggested that a functionally categorized menu is more effective than an alphabetical menu, through which users make fewer errors with the categorization. On the other side, a research by MacGregor and Lee [189] found that menu search is consistent with systematic and sequential search. However the models by Hornof and Kieras [190] provided evidence that people do not decide on menu items individually but rather process many items in parallel, by using both systematic top-to-bottom and random visual search strategies. A plausible explanation they provided was people engage a low-level perceptual, cognitive and motor processing when selecting an item from a menu. Since categorized alphabetic menu is widely used in the web, and therefore more empirical research should be carried out to study its efficiency on navigation performance.

- **Terms Used:** A lot of frustrations occur during web navigation are caused by unpredictable interfaces [63]. A deceptive and confusing design can also be caused by inappropriate use of terminology, such as subject-specific and technical terminology that normal users would not understand, and it is worse when there is no explanation that provides cues to what the label should convey [60]. Confusion can also be induced when two different features are given similar names.

- **Scrolling:** Spool et al [191] argued that links should not be embedded in pages of text which requires the user to scroll down to find them. Readers' memory is supported by the fixed relationship between an item and its position on a given page. A scrolling facility is therefore liable to weaken these relationships and offers the reader only the relative position cues that an item has with its immediate neighbours [192]. Earlier research showed that when scrolling or dragging on the scroll bar did not move concurrently in response to the dragging, this can make the finding of the exact location extremely time-consuming [193]. Besides, if the user has to scroll more often to view the entire text, it interferes with concentration [194]. Most of the past research studied effects of scrolling on reading text on the screen, e.g. [194]–[197], but there is lack of research that examines the effects of scrolling a long menu on navigation experience, especially when the computer devices and browser capabilities are so advanced nowadays.

Lim et al [35] study the users' perceptions of 7 factors, i.e. whether a web page contains (1) confusing features, (2) too many features, (3) inconsistent layout, (4) unrecognisable hyperlinks, (5) no information of user's current location, (6) no explanation of features, and (7) ambiguous terms. They found that users with different socio-demographic background have varied perceptions and satisfaction with the system designs when they were given the same learning management system to use. Therefore, there is a need to improve personalized experience with affective computing and adaptive computing, so that user's emotion when using the system can be detected by measuring certain metrics (such as facial cue, duration spent to complete a task, psycho-physiological state, etc.).

Lim et al [198] reviewed the above six factors that could cause negative feeling (such as feeling uncomfortable or stressed) to users in a menu search task. Their results show that bad menu design, such as bad colour combination (blue link on red background), smaller font size (9 pt.), text without code (e.g. English for the IT Profession), abbreviated terms (e.g. MSDNAA), ambiguous terms (e.g. Bulletin Board vs. Bulletin Board for Staff), random display (not according to alphabetical order), and the need to scroll (the user needs to scroll the window down in order to locate an item) make users feel uncomfortable. Although colour combination does not affect job performance in general, it has an effect when it is combined with other factors. Bad setting of font (smaller font), text design (long text) and (ambiguous) term used may decrease job performance (such as increased attempt to revisit the questions, increased tendency to give up the task and increased task duration). Categorized display and no scrolling make users feel comfortable but they do not necessarily improve job performance. The description of a task to be searched is also believed to affect individual’s job performance, as the task may require the users to use more cognitive power to comprehend or to process the possible feature to be searched. In terms of the effects on mouse dynamics, their research also suggested that font size, text design, term used, feature organization and requirement to scroll significantly change mouse behaviour. The bad settings of 4 factors (namely colour, font size, text length, and term used) increase mouse idle duration and occurrences, and reduces mouse speed and mouse click. Lastly, their results also show that job performance gives impact on mouse behaviour. For instance, when the users make more errors or attempt to revisit the question (or to give up the task), they demonstrate longer mouse idle time (indicates that they do not move the mouse often), but fast mouse speed when they need to use the mouse. On the other side, when the users spend longer time in the search task, they demonstrate longer mouse idle time, but slower mouse speed. As the users agree that they feel stressed when they need to spend longer time in the search task, we can infer that if task duration increased, error count increased and attempt to revisit question (or to give up) also increased, and generally mouse speed is slower and mouse idle time is longer, then they are probably feeling stressed.

2.6.2.2 MENTAL ARITHMETIC AND COGNITIVE LOAD

Mental arithmetic problems under time pressure are widely used to induce cognitive stress [43], [122], [199]. A study by Imbo & Vandierendonck [200] suggested that larger numbers and borrow operations in arithmetic problems, which involve longer sequences of steps and require maintenance of more intermediate products, will place greater demands on human working memory. Once the demand has exceeded the working memory capacity and temporal limitations, then the task is deemed too challenging to be continued [52]. Although much research has investigated how attention, memory and computational processes support arithmetic calculations, less work has addressed how math performance can be influenced by emotional factors, such as stress. Beilock and Ramirez [36] suggested that stressful and emotion-inducing situations could lead to unwanted performance degradation even for relatively simple calculations in math performance, due to negative emotion could prevent or inhibit the recruitment of the appropriate cognitive resources necessary for optimal skill execution. However, Weinberg et al [201] argued that human attention to emotion stimuli may not be automatic nor obligatory. When the context of the emotion stimuli is not relevant to the task (such as seeing a picture of a crying face), humans may demonstrate little-to-no impact on the emotional modulated arithmetic task. In other words, the effects of the stimuli on cognitive process may depend on both of the attentional demands of the task and the salience of the stimuli [16]. The impact of negative emotion on performance decrement may be caused by the task demands itself (such as high requirements), or other factors that are related to the task (such as time pressure).

Research done by [202], [203] found that task demand is correlated to students' stress perceptions, job performance (duration spent, error rate and passive attempt), mouse behaviour (mouse speed, mouse click rate and mouse idle duration) and keyboard behaviour (keystroke speed, key latency). The correlation results are consistent with what was reported in [198]. However, when the task difficulty has increased, but the job performance, mouse and keystroke behaviours do not behave in a way that is expected, then anomalies become prominent. Anomalous behaviours indicate three possibilities: (1) there is a wrong assumption about the demand of the question; (2) qualitative difference in task demands, e.g. increment of the number of digits per number in the mental arithmetic, would require more working memory to process the task; or (3) the student is either understressed or overstressed, which is beyond their motivation limits. At this point, prediction of cognitive stress level would become invalid, as the students have already lost motivation to continue the task. Therefore, it is important to activate the adaptive content to motivate the students to continue the task. Their research also discovered that task demand is the main factor that influences student's stress perception, job performance, mouse and keystroke behaviours, but time pressure only provides a small significant effect.

2.6.2.3 TYPING TASK AND SUBJECT FAMILIARITY

There is a little research done to examine the effect of typing tasks on emotion since most of the tasks in an e-learning environment require text typing, e.g. post discussion. Besides, there is also a lack of research carried out to study the influence of subject familiarity on task performance and physiological behaviour. Tobias et al [204] suggested that lack of familiarity implies that the required cognitive resources or response needed for executing the task may not be available in the learner's repertory or memory. Therefore, it would require a more overt response for optimal learning from content with unfamiliar subjects. Hulme et al [205] also found that memory spans for unfamiliar words are lower than familiar words. Therefore, we would like to examine the effects of text length and language familiarity on user behaviour, such as typing rhythms, even though the effect could be small.

However, there are a few issues to consider in typing task demand. The main issue is there are high variations of individual typing skills such as typing speed, which are caused by individual expertise skills, experience, and environmental factors. According to Davidson et al [206], a typist's typing speed will increase if he or she is able to look far ahead. Far sight allows superior preparation and optimization of typing movement. Additionally, typing speed can be increased by 10-20% if full concentration is exerted, and habitual typing behaviour could be broken when individuals engage in activities that are deliberately prescribed to increase their typing speed, such as setting time pressure, and this often leads to mistakes. The second issue in typing task demand is regarding text length. Most research limits the experiments to produce samples from structured and predefined text in order to analyse keyboard dynamics. Many researchers strived to work with relatively short sample phrases, such as username and password, for example [132], [153]–[155], [207]–[209]. Others used free and long text in their studies, e.g.[148], [210]. However, most of their studies show that both fixed text and free text are equally useful for keyboard dynamics analyses, regardless the length of the text.

A research by Lim et al [211], [212] required 60 students to type 6 fixed texts (3 in familiar language and 3 in unfamiliar language, varied by lengths), which 30 of them were without any time constraint, while the rest in the experimental group were given 30 seconds time limit for each question. The results show that higher stress perception is associated with longer text length and lower familiarity of the language. High task demand generally results in longer task duration, higher error rate, slower mouse and keystroke speeds, longer mouse idle duration, and lower mouse idle occurrences and use of error key (such as delete key). They also found that time pressure does not necessarily affect how users perceive their stress levels but it affects task performance (shorter time completion but with higher error rate), mouse dynamics and keystroke dynamics. On the other side, language familiarity affects only task performance and keystroke behaviour, while text length changes mouse behaviour but not keystroke behaviour. This suggests that we should mainly look into task performance and mouse behaviour features if the typing tasks involve changes in length, and observe only task performance and keyboard behaviour to understand whether a person is familiar with the given material. Lastly, the measurement of user's emotional stress level will become invalid once he or she is overstressed or has lost motivation, which results in anomalous behaviours, such as unexpected job performance, along with abnormal mouse and keyboard dynamics.

2.6.3 STRESS CLASSIFIER'S LEARNING AND CONSTRUCTION

Stress is a kind of affective state that is hard to express and quantified clearly, is vague in some way, and lacking a fixed, precise definition. Furthermore, the mouse and keystroke features of a subject taken from different instances of the same level of stress could have wide variations. The stress perception variations between individuals when facing the same challenge is also one of the main sources of uncertainty in the stress measurement problem. The other concern we have is to find a cost-effective method to allow stress to be measured continuously over an online environment. Therefore, the classifier's learning algorithm should be less complicated so that the processing time of stress measurement could be done almost instantly without causing delay to both sides of client and server. If correlations between learners' stress, behaviour, mouse behaviour and keystroke behaviour are found significant in the preliminary research, then further step in designing the stress measurement model using keystroke and mouse dynamics, which is able to sense the changes in learner's emotion and cognition, could be carried out. Stress level is expected to be classified into one of the 3 outputs based on user mouse/keystroke behaviour, i.e. stress increased significantly ($SP = 1$), stress decreased significantly ($SP = -1$) or remains stable or normal ($SP = 0$). Three different approaches that can be useful in managing uncertainties and easily implemented in an online environment are certainty factors (CF), feedforward back-propagation neural network (FFBP) and adaptive euro-fuzzy inference system (ANFIS). Fuzzy logic will be considered for the stress inference engine development as part of the intelligent tutoring system.

The next sub-section explains the stages of constructing a stress classifier. The subsequent sub-sections explain CF model and the architectures of FFBP, ANFIS and Fuzzy Logic in detail.

2.6.3.1 STAGES OF CLASSIFIER CONSTRUCTION

The stages of classifier's construction of emotion measurement consist of data acquisition and feature extraction, creation of the training set containing labelled data and classifier's learning [32]. Data acquisition must be carried out automatically to collect samples that can objectively measure real world physical conditions, and the data could be converted into digital form for computer manipulation. However, not all data are necessarily useful for analysis and therefore feature extraction should take place before the data are processed. Feature extraction is mainly used to reduce the measurement and storage requirements, to minimize training and utilization times, so that the prediction performance can be improved. Therefore, one must carefully deliberate the necessary inputs to be captured from the users to ensure the measurement is reliable and effective. The common approaches used to acquire user's inputs for emotion recognition include subjective methods such as self-report, text extraction, physiological tests using physiological sensors, use of pressure-sensitive keyboard, video and/or audio recording and analysis, standard mouse and keyboard inputs, and task-performance based measurement [20], [32], [46]. This research mainly focuses on using standard mouse and keyboard inputs for data acquisition as they can be easily implemented as part of a normal system without special setup, hence cost-effective and unobtrusive.

2.6.3.2 CERTAINTY FACTORS

Certainty factors (CF) model was first introduced in MYCIN [213] as a way to represent uncertainty when a conclusion is made by a rule. Although this approach is questionable, many past and current expert systems do utilize certainty factors in several different forms. Heckerman and Shortliffe [214] argued that the CF model may be inadequate for the domains where appropriate recommendations of treatment are more sensitive to accurate diagnosis. However, considering stress measurement itself a highly subjective research, therefore using CF in stress measurement should be considered acceptable. The standard concept used in MYCIN requires each rule to be assigned a strength called certainty factor, usually by expert, lying in the interval [0, 1]. The premises of the rules are evaluated when a rule is fired, and each premise, E, is assigned a numeric value ranged from -1 to 1. Then the action part, H, of the rule is evaluated and conclusion is made with a certainty value, which $CF(H) = E \times CF(Rule)$. In particular, a CF between 0 and 1 means that the person's belief in H given E increases, whereas a CF between -1 and 0 means that the person's belief decreases. When we have more than one rule with the same hypothesis, then the certainty values of all relevant rules must be combined for a conclusion. The developers of the CF model did not intend a CF to represent a person's absolute degree of belief in H given E, $P(H|E)$, as does a probability theory [215], but they redefined CF to accommodate an infinite number of probabilistic interpretations [216]. Although CF violates certain restrictions in probability theory, e.g. a system can contain sets of mutually exclusive and exhaustive hypotheses with more than two elements, nevertheless it is still useful as it is easy to use and not critical to the system's performance.

2.6.3.3 FEEDFORWARD BACK-PROPAGATION NEURAL NETWORK

Feedforward back-propagation (FFBP) neural network, aka. multilayer feedforward neural network or back-propagation net, is a multilayer feedforward network trained by back-propagation (of errors) training method [217]. It is widely used in many areas such as classification and pattern recognition. A FFBP neural network consists of neurons, which are ordered into layers - an input layer, hidden layer(s) and an output layer. It operates in two modes: training and prediction mode. One dataset is needed for training and a test set is needed to predict. The training mode begins with randomly generated weights, and proceeds iteratively with back-propagation training algorithm. For a given training set, back-propagation learning is preferred to proceed in pattern mode over batch mode, as the former requires less local storage for each synaptic connection, and in online-process control, there are not all of training patterns available in the given time [218]. The crucial problem in the model selection is the number of hidden units and hidden layer to be used. According to Svozil et al [218], there is no way to determine a good network topology. It highly depends on the training cases, the amount of noise, and the complexity of the classification that you are trying to learn. It is strongly recommended to use one hidden layer as additional hidden layer makes the gradient more unstable and that training process would slow dramatically. Furthermore, tendency of FFBP neural network to 'memorise' data (the predictive ability) is substantially lowered if the number of neurons in hidden layer is increased.

2.6.3.4 ADAPTIVE NEURO-FUZZY INFERENCE SYSTEM

Fuzzy neural network is a hybrid system of fuzzy logic and neural network. Although there are several types of fuzzy neural network, the model that is relevant to rule-based system is the fuzzy rule-based system with learning ability, where fuzzy if-then rules are adjusted by iterative learning algorithms similar to neural network learning [219]. Therefore, unlike the static fuzzy inference system, fuzzy neural network is given the ability to learn and predict the outcome as neural network. One example of system that is classified under this type of fuzzy neural network is called adaptive neuro-fuzzy inference system (ANFIS) as proposed by Jang [220]. ANFIS gives fuzzy systems adaptive capability, by combining the fuzzy inference systems of Sugeno-type (FIS) and neural network, which is ideal for interpretation of nonlinear systems. Given an input-output dataset, the parameters of membership functions (in fuzzy variables of premises of the fuzzy rules) are tuned using back-propagation algorithm or in combination with a least squares type of method.

In the FIS part, a general fuzzy if-then rules given by [220] with 2 premises (inputs), 2 membership functions (for each premise) and a single output can be written as follows:

RULE *1:* If x is A_1 and y is B_1 then $f_1=p_1x + q_1 y + r_1$

RULE 2: If x is A_2 and y is B_2 then $f_2=p_2x + q_2 y + r_2$

RULE *3:* If x is A_1 and y is B_2 then $f_3=p_3x + q_3 y + r_3$

RULE *4:* If x is A_2 and y is B_1 then $f_4=p_4x + q_4 y + r_4$

where $x = [x_1, x_2]$ and $y = [y_1, y_2]$ as n-dimensional input vectors, f is an output variable, A and B are the fuzzy sets with 2 membership functions.

The number of rules is correspondent to the number of membership function of a premise by default. For instance, 2 inputs with 2 membership functions (for each fuzzy set) produce 4 rules (with different permutations). Two inputs with 3 membership functions (e.g. low, normal, high) produce 9 rules, and 4 inputs with 3 membership functions will produce 81 rules. Therefore, in terms of programming, the implementation of ANFIS in the inference engine of stress monitoring system could be more challenging than CF and FFBP neural networks. Nonetheless, ANFIS is believed to be good as it models the qualitative aspects of human knowledge and reasoning process without employing precise quantitative analyses. Furthermore it offers adaptive capability to fine tune the membership functions so as to minimize the output error measure and to maximize performance index [220].

2.6.3.5 FUZZY LOGIC

Fuzzy logic has been applied to many fields, from control theory to artificial intelligence. Fuzzy logic, introduced by Lofti Zadeh in 1965, is a form of multi-valued logic, in which the truth values can be in the range of continuous interval [0, 1] of real numbers, representing a degree of vagueness, rather than being only either 0 or 1 as in Boolean Logic. The truth-value in fuzzy logic is interpreted as fuzzy set [221]. The truth value of the member in a fuzzy set is determined by a membership function (MF). MF is a curve that defines how each point in the input space is mapped to a membership value (or degree of membership) between 0 and 1 [222]. When linguistic variables are used in a fuzzy inference system, these degrees may be managed by specific membership functions, which is used to reduce principles of reasoning to a code [223]. Mamdani's fuzzy inference method is the most commonly seen fuzzy methodology, due to its intuitiveness and it can be well suited to human input easily [224]. It was proposed by Ebrahim Mamdani [225] in 1975 for building control systems using fuzzy set theory. The method was based on Zadeh's work on fuzzy algorithms for complex systems and decision processes in 1973 [226]. The fuzzy inference process involves fuzzifying the inputs, applying the fuzzy operator, and expects the output membership functions to be fuzzy sets as well. After the aggregation process, there is a fuzzy set for each output variable that needs defuzzification [227]. Amongst the defuzzification methods, centroid is the most popular [228]. Centroid defuzzification method returns the centre of area under the curve, which is the point along the *x* axis about a shape would balance. Other defuzzification methods also include MOM, SOM, and LOM (stand for Mean, Smallest, and Largest of Maximum, respectively). These three methods key off the maximum value assumed by the aggregate membership function [229]. The MOM method computes the average of the fuzzy outputs that have the highest degree. It does not consider the entire shape of the output membership function but only select the points that have the highest degrees in that function [230]. There is no quick and fast rule to determine the best defuzzification method that is appropriate for this research. Although centroid is widely used, the method does not work when the output membership function has non-convex properties [231].

2.7 SUMMARY

Over the past decades, research in e-learning has begun to take emotions, a.k.a. affects or valences, into account, because their influence in perception, reasoning, decision-making and learning. Fluctuation in motivation, losing concentration and unbearable stress that a learner has, are some of the issues that both learner and teacher must deal with. By discovering the factors that endanger learning, the teacher or the adaptive e-learning system could adjust the content to reengage the learner's concentration in the subsequent challenging learning experience. Past research of adaptive e-learning proposed many ways on how the most appropriate content and presentation can be fitted to each individual user, based on the correct and continuous identification of the user learning styles or behaviours. However, there are still many challenges and difficulties in the sense of technologies that need to be solved. Despite these challenges, an affective learning system is believed to enable more effective learning. It allows automated computation of cognitive states, evaluation of learning content, improving user experience to enhance learning performance, and supports learner-centred design in the e-learning system to improve learning sustainability.

Keystroke and mouse dynamics have been adopted by a number of research that mainly study their effectiveness in authentication and identity recognition over the past two decades. Recent research in affective computing discovered the potential of keystroke dynamics and mouse dynamics in recognizing user’s emotions. web-based applications, including e-learning system, are controlled by the mouse and keyboard most of the time. As such, using these input devices in modelling and tracking user behaviour is considered non-obtrusive, user-friendly, cost-effective, enables continuous monitoring process and the measurement of user’s affects could be more reliable than subjective methods. However, to enable a reliable and objective measurement of stress by using mouse and keystroke dynamics analyses alone is not sufficient. These devices can only produce relatively small amount of information or references, which are unstructured and differed from each other. Furthermore, different tasks require different device to be used, and one would be idle for long time when another is in use. To enable a better classification rate, mouse and keystroke dynamics analyses should be complemented by other techniques, such as task-performance-based analysis, to increase the reliability of the results. Most of the past research induced users' emotions by visual or effects, which may not be relevant in the real-life e-learning environment. It is important to identify the actual stressors that trigger negative emotions of users in order to provide the best appropriate action to tackle the issues. Therefore, by using task-performance-based analysis, the level of stress arousal can be adjusted by deliberatively changing the workload and control of task given to the users, based on three different tasks, i.e. search, assessment and typing.

A modern ITS should be designed to be aware of the emotional state of a learner, and to intervene appropriately and only when a negative affective state of the learner is detected while he is stuck on a problem. If the stressor that generates the negative effect on learner's behaviour, e.g. high demand of question, can be determined automatically, then adaptation of learning materials could be made. Besides, a feedback related to the stressor could be channelled to the relevant teacher for fairer assessment, due to the possibility of mismatched expectation by the examiner (the teacher may think the question is reasonably fair, but it may be deemed too challenging by the students, and vice versa). Methods in producing the stress measurement have been studied, three different classifiers, namely certain factor, feedforward back-propagation neural network, and adaptive neuro-fuzzy inference system will be applied, and their efficiencies in stress measurement will be studied.

Chapter 3 will discuss the research methodology and the experiment designs in detail.

CHAPTER 3: RESERCH METHODOLOGY AND EXPERIMENT DESIGN

Figure 3.1 illustrates the three major phases of the research studies. The initial phase is set to test the feasibility of using mouse and keystroke dynamics for building an automated stress measurement model in a web-based learning environment. Experimental studies, which are described in this chapter, will be carried out to examine the relations of task demand and external psycho-physiological stimuli to stress, cognitive states and mouse/keystroke behaviours of some e-learning students from a higher learning institution in Malaysia. The results of the feasibility studies and the data analyses for three different e-learning tasks will be reported in Chapter 4 to Chapter 6 respectively. The second phase of the research carries out an empirical study to examine the best stress detection and modelling using mouse and keystroke dynamics, out of three artificial intelligence methods, namely certainty factors (CF), feedforward back-propagation (FFBP) neural network, and adaptive neuro-fuzzy inference system (ANFIS). The detailed setup for the three stress classifiers' constructions will be covered in Chapter 7. The last phase focuses on designing two possible applications of the identified stress measurement model to an ITS, which are adaptive assessment and analytical feedback to examiner. The detailed architectural design of the ITS, the processes involved in the stress inference engine, the design of adaptive assessment and the analytical feedback system that provides the examiner some information related to learners' behaviours, will be presented in Chapter 8.

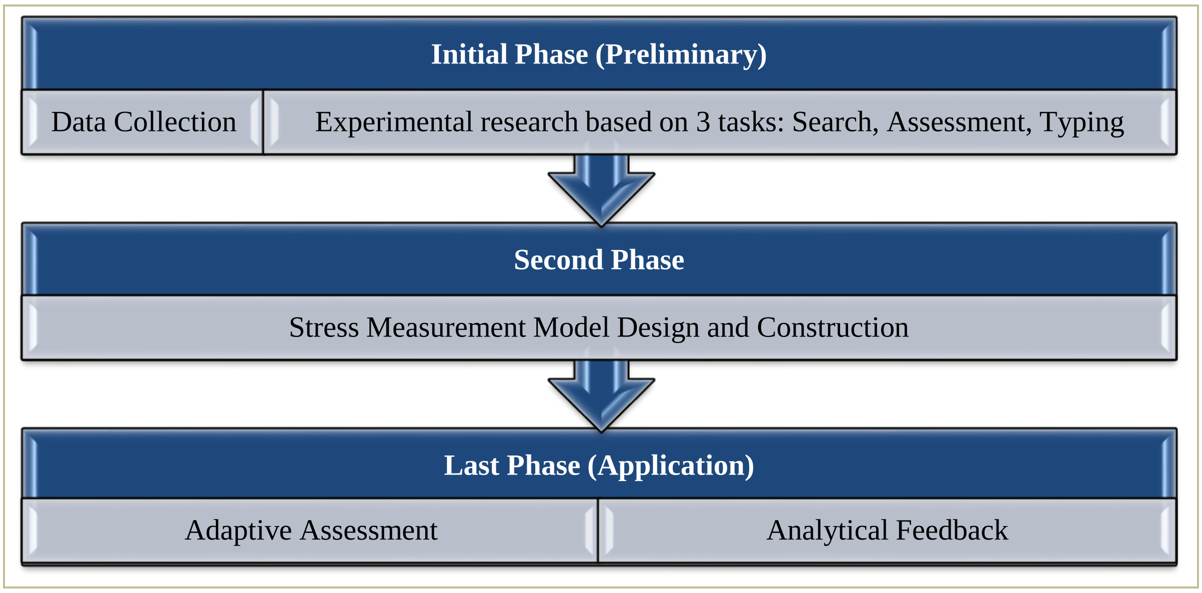


Figure 3.1. Research phase overview

Chapter 3 provides the detailed design and the procedures of the initial phase setup. Section 3.1 defines stress, which is the affective state examined in the research. Section 3.2 explains the adoption and adaptation of an existing theoretical framework proposed by Wang [22], namely MADB model. This model is useful for us to compute learner's cognitive states using objective measurements. Slight modifications are done on the MADB model so that it suits e-learning environment. As adaptations are made on the Wang's original MADB model, some tests must be carried out to validate the adjustment. Section 3.3 explains the creation of stress stimuli and stress perception collection method. Section 3.4 describes the sampling of participants. Section 3.5 describes the general experimental procedures. Section 3.6 illustrates the construction of the apparatus needed for the data collection, i.e. key logger and mouse logger, and the mock-up of an existing e-learning system for the three different tasks, i.e. search, assessment and typing. Section 3.7 describes the behaviour modelling. Section 3.8 explains the analysis methods. Finally, Section 3.9 concludes the chapter.

3.1 STRESS DEFINITION IN THE RESEARCH CONTEXT

Two kinds of stress are defined for this research: emotional stress and cognitive stress. Most people view stress as some unpleasant threat nowadays, and it is generally considered as being synonymous with distress as defined by Selye [12]. Emotional stress, stated as stress perception in our research, is therefore defined based on Selye, as *a perceived emotion that involves unresolved feelings of fear, anxiety and frustration,* which build psychological barrier to further learning. In terms of cognitive stress, as stated as cognitive states in our research context, it is *a human perception of stress in relation to various states of the fundamental components of mental load, such as motivation, attention allocation, memory resources, attitude, decision making and behaviour*, based on the MADB model as proposed by Wang [22]. These two kinds of stress are interrelated based on the MADB model, which human perception of stress, or emotional stress, would affect, or in relation to the states in human cognition, such as motivation and behaviour. On the flip side, the outcome of the behaviour could affect stress perception. Therefore, stress may comprise both kinds at the same time in our research context. The next section explains the MADB model that is adjusted to suit e-learning environment. The validation of the proposed MADB model will be carried out based on the three different tasks given to the participants during e-learning.

3.2 MOTIVATION/ATTITUDE-DRIVEN BEHAVIOUR (MADB) WITH MOUSE AND KEYSTROKE BEHAVIOUR

Wang et al [19], [22] suggested that cognitive performance is related to attention, and could be affected by emotional, motivational and attitudinal factors. Wang demonstrated how the complicated human emotional and perceptual phenomena can be rigorously modelled and formally treated based on existing cognitive informatics theories and denotational mathematics, which he named as MADB model. His previous work applied the MADB model in a software engineering organization, but we envisage the model can also be fit into the e-learning environment. It is interesting to examine how formal cognitive processes during e-learning can be modelled by considering student's motivation, attitude and behaviour. Since the environment of e-learning is different from a software engineering organization where Wang conducted case studies to formalize the MADB model, hence we have done some adaptation of the MADB model as follows.

1. The stress stimuli refer to the direct instruction, such as assessment and typing task, and indirect tasks, such as search, which need to be done to achieve a goal. External stimuli may also be raised by the environmental factors, such as the design of the user interface, display of countdown timer, and setting of time constraint. For example, if the system design does not fulfil usability standards, it may cause fatigue and unnecessary stress to the users; a presence of countdown timer may cause the students to feel nervous too.
2. Motivation can be weaken by unpleasant experience with the system, or poor job performance/outcome.
3. Attitude includes user's confidence with the task based on experience, the estimated effort to complete the task, or the amount of attention can be spent on a task. The combination of motivation and attitude gives impact on the rational motivation. Rational motivation enables a person to continue doing the task, if it is still within their acceptable effort to invest.
4. Decision is affected by time, resources and energy according to Wang. Therefore, time constraint and projected long completion time may reduce user’s estimated probability of success.
5. The combination of rational motivation and decision will affect the behaviour and job outcome.
6. The job outcome affects student's motivation and stress perception for carrying out the next task.
7. As we are interested in examining the feasibility of using mouse-dynamics and keystroke-dynamics-based analysis in detecting human emotion, motivation and attitude, we added the mouse behaviour and keystroke behaviour as the external behaviour in the MADB model. We assume that the mouse behaviour and keystroke behaviour are related to human behaviour, which is affected by motivation and decision.

The proposed MADB model in e-learning context is illustrated in Figure 3.2.

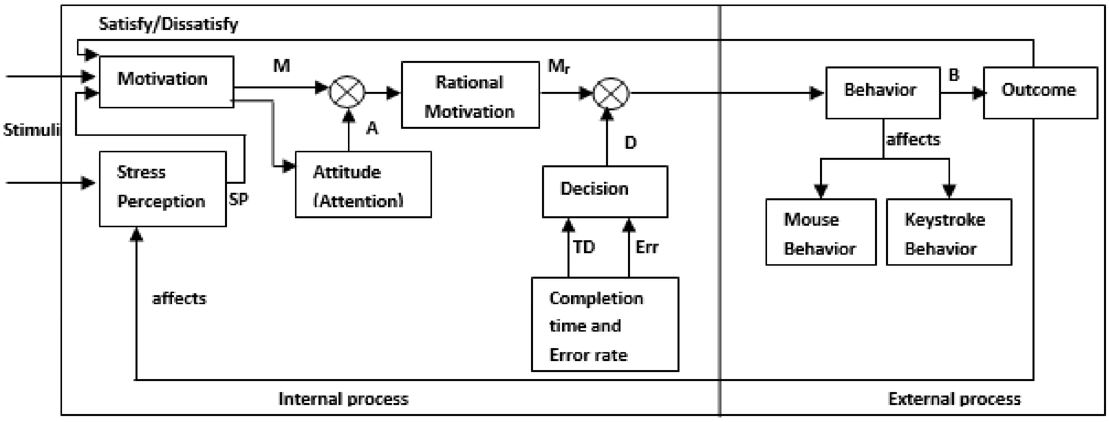


Figure 3.2. The application of MADB model in e-learning context with mouse and keystroke behaviours, adapted from [22]

To validate the application of Wang's MADB model in our research context, the following illustrate the seven assumptions that we make in the mathematic denotations.

1. Stress stimuli comprise tasks varied by different job natures, i.e. search, mental arithmetic and typing, and with added external stimuli such as menu design, time constraint, clock display and countdown timer display. The stimuli are strengthen by the increased level of difficulty in each type of tasks. The stress stimuli are explained in Section 3.3. We measure the stress perception (*SP*) of the task on a 7-Likert scale [232] (1 indicates strongly disagree that he/she is stressed, and 7 indicates strongly agree), as follows:

$$1 \leq SP \leq 7 \tag{3.1}$$

2. Motivation can be weaken by high stress perception *SP*. The strength of motivation *M* is reduced by higher *SP* and the expectancy of desire *E* and the current status *S*. Rational motivation can be affected by motivation (*M*), emotion (*SP*) and attitude (*A*). Desire *E* is defined as how strong the person is willing to continue the task, that is:

$$E = \begin{cases} 0 \; if attempt\ to\ continue\ the\ current\ task \\ 1 \quad if\ attempt\ to\ give\ up\ the\ current\ task \end{cases} \tag{3.2}$$

   And the current status S is defined as follows.

$$S = the\ total\ number\ of attempts\ that\ a\ person\ gave\ up\ the\ previous\ tasks \tag{3.3}$$

3. Since the expectancy of desire *E* and the current status *S* of a student can be absolutely none, *M* is defined proportional to the strength of stress perception *SP*, the expectancy of desire *E*, and the current status *S* of a person Therefore the motivation *M* is computed as follows:

$$M = 100 - \left(\frac{SP + E + S}{C}\right) \tag{3.4}$$

where C = the constant to accomplish the expected motivation, which is averaged by the number of tasks given. C is included to normalize the value of M in the scope of [0..100]. For instance, if the maximum value of SP =7, maximum E = 1 and maximum S = 9, then C = 17/100 = 0.17.Lower value of Mindicates low motivation, and higher M means stronger motivation.

4. Rational motivation M_r is defined as a motivation regulated by an attitude A [22]. To compute M_r, we define A as the amount of attention to be spent on a task. We designed two distinguished ways to measure the strength of A (in the scope of [0...5]). For the indirect instruction (i.e. searching for a desired material), we measure A as the attempt to revisit the task instruction is observed, which is:

$$A = 5 - number\ of\ attempt\ of\ an\ individual\ to\ revisit\ the\ task\ instruction \quad (3.5.1)$$

For the direct instruction with time pressure, i.e. assessment and typing tasks, A is measured based on the passive attempt to wait for the time is up instead of submitting the answer earlier (passive attempt = 1 if true, else 0), which is:

$$A = 5 - passive\ attempt \quad (3.5.2)$$

We assume that A is low if there is a need to revisit the given instruction. Rational motivation M_r is then defined as:

$$M_r = \frac{M \times A}{500}, \text{ so that } M_r \text{ is in the scope of [0...1]} \quad (3.6)$$

5. Behaviour is affected by the rational motivation M_r and decision D, and the changes of behaviour can be observed from mouse/keystroke dynamics. According to Wang, decision is a binary choice on the basis of availability of time T, resources R and energy P. However, as we are looking for an objective measurement that can compute decision D automatically, we assume that a decision to continue a task D, is reduced by the increment of total task duration TD or error rate of the task, Err:

$$D = 1 - max(TD, Err) \quad (3.7)$$

where $0 \leq (TD, Err) \leq 1$. TD is the increment or decrement rate of current task duration$T$(in milliseconds) compared to the accumulated average duration from the previous tasks, T_{ac}. However huge variations of time duration can be sensitive to generate significant difference even small departures from homogeneity and the assumption of normality, hence the collected data are transformed using thelog_{10} function.

$$TD = \frac{T - T_{ac}}{T} \quad (3.8)$$

Err refers to the accumulated average error rate of the executed tasks, as follows:

$$Err = \frac{\sum_i^n X}{N} \quad (3.9)$$

where x = the accumulated number of errors from the previous task, and N = total number of tasks

6. Behaviour B determines the action to continue a task. We assume that if the external stimuli, such as menu design, task demand and time constraint, are perceived unpleasant, then the chance that B determines the action to continue the task is low, as the motivation M is reduced and stress perception SP is increased, and the attitude A is also reduced, which further decrease rational motivation M_r. Bad external factor is also believed to increase task duration T and error rates Err [198], which will reduce the decision D to continue doing the task. Since the behaviour B is driven by the rational motivation M_r, and decision D, B is defined as

$$B = min(M_r, D) \tag{3.10}$$

7. Past research found that user's mouse behaviour and keystroke behaviour can be affected by task demand and stress perception [55], [113], [198]. Therefore, we envisage the correlations between behaviour B, mouse behaviour $B(M)$ and keystroke behaviour $B(K)$ can be significant. Detailed models of keystroke behaviour and mouse behaviour are illustrated in Section 3.7.3 and Section 3.7.4 respectively.

3.3 STRESS STIMULI AND STRESS PERCEPTION COLLECTION

The stimuli used in the experiments to induce stress are varied according to task, as follows.

3.3.1 TASK A: SEARCH FOR A LEARNING MATERIAL (MENU SEARCH)

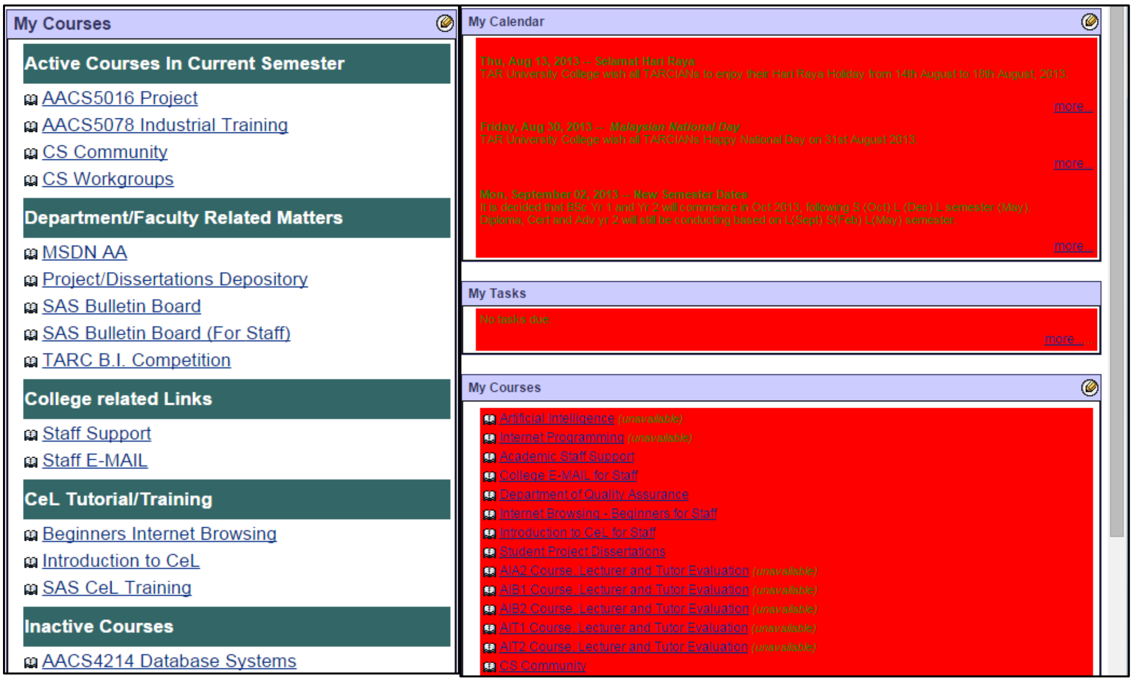
Preliminary research [198] identified six factors that could cause negative emotion such as frustration, dislike, and uncomfortable feelings to users during a menu search task. The six factors are (1) colour, (2) font size, (3) text length, (4) menu organization, (5) term used, and (6) the need to scroll the menu. We limited the research to two levels of each factor, to prevent overly huge number of combinations, which result in 64 different combinations of menu design. The detailed combinations according to questions are shown in Table A2.1 in Appendix II Part B. For each combination of the six factors, a single web page for each menu design is built, and therefore 64 different web pages for the experiments are produced. Table 3.1 shows the good and bad settings of each factor. Figure 3.3 shows sample web pages with different settings. To avoid carry-over effects [233] that might affect the later performance when a participant attempts the same instruction more than once, 64 different questions are introduced, therefore there are 64 different items to be searched in the entire search task. Detailed arrangements of the design factor combinations for each question, and the actual instructions given to the participants are provided in Appendix II Part C. All the web pages are designed with the same layout according to an existing e-learning system, except that the main menu is changed according to different factor level. To avoid any potential biases or judgments, completely randomized design is utilized, which the order of the questions assigned to the participants are done at completely random manner. Besides, the randomized trials are automatically generated by the system itself, so that no participant should follow the task in the same sequence as others.

A survey form that consists of the following questions was designed to evaluate the impacts of menu design on the participants' emotion stress. The survey form will be displayed at the end of Task A.

1. You feel stressed if you need to take longer time to search for a feature in the website (select (1) for strongly disagree to (7) for strongly agree)
2. Rate your feeling when searching for a feature if the page is designed with the following setting. Select (1) for strongly uncomfortable to (7) for very comfortable
    a. if the text colour is blue on white background (goodColour).
    b. if the text colour is blue on red background (badColour).
    c. if the font size is bigger (bigFont).
    d. if the font size is smaller (smallFont).
    e. if the label is provided WITH course code, e.g. "AACS5078 Industrial Training" (text with code).
    f. if the label is displayed WITHOUT course code, e.g. "Industrial Training" (text without code).
    g. if the text is shorten, e.g. DQA (abbreviated term).
    h. if the text is lengthen, e.g. Department of Quality Assurance (longText).
    i. if the term used clearly represents the feature you are looking for (clear).
    j. if the term used to represent a feature is ambiguous and confusing (ambiguous).
    k. if the features are organized (categorized).
    l. if the features are not organized and displayed randomly (random).
    m. if you can view all features without scrolling down the page (noScroll).
    n. if you have to scroll down the page to view the feature (scroll).

Table 3.1: The Setting of Colour, Font, Text, Term, Organization and Scroll.

No.	Factor, x	Good setting (x = 0)	Bad setting (x = 1)
1	Colour	*goodColour*: a blue link on a white background.	*badColour* = a blue link on a red background.
2	Font	*bigFont*: font size of 13 points (pt.), Arial.	*smallFont*: to the font size of 9pt, Arial.
3	Text	*shortText*: the link consists of not more than 3 words and with module code (e.g. *AACS4134 Internet Programming*) or with abbreviation (e.g. *MSDNAA*). When abbreviation is used, a drop-down tooltip will be shown to explain the full term when the user hovering the link.	*longText*: the link consists of at least 3 words and without any code (e.g. *English for the IT Profession*).
4	Term	*clear* term: the term used to describe a link is clear and direct, which the users should be able to recognize the link without much cognitive processing power.	*ambiguous* term: there is another link with similar term or function exists on the same page, which can cause confusion (e.g. *Bulletin Board* and *Bulletin Board for Staff*). This type of term requires more cognitive processing power so that the users would need to comprehend the actual link to be searched.
5	Organization	*categorized* organization: the links are functionally categorized and sorted alphabetically.	*random* organization: there is no categorization and the links are displayed randomly.
6	Scroll	*noScroll*: the links are displayed on top on the page and no scrolling is required in order to hit the required link.	*Scroll*: the links are displayed on the bottom right corner, so that the users need to scroll down the page to reach the required link.



(A) (B)

Figure 3.3. Menu design with different settings of (A) goodColour, bigFont, shortText, clearTerm, categorized organization and noScroll, and (B) badColour, smallFont, longText, ambiguous term, random organization and need to scroll.

3.3.2 TASK B: ASSESSMENT (MENTAL ARITHMETIC)

This research is to analyse how keyboard and mouse behavioural patterns change according to task demands and external psycho-physiological stimuli during mental arithmetic. Ten different mental arithmetic problems with different levels of complexity are set. Each question is displayed on an individual web page. The participants must answer the questions on the mock-up online-assessment website by doing mental arithmetic, i.e. no calculator and no calculation on paper. Ten different mental arithmetic problems with diverse complexity, as shown in Table 3.2, were given to the students. The task demand is elevated from Question 1 to Question 10, with respect to the increment of amount of digits per number, and the amount of numbers in the question, as well as the use of summation, deduction and multiplication operation. The participants must type the answer into a designated textbox on the page. To force the student to use the mouse, the "Enter" key is disabled, and he or she must click the "Submit" button in order to submit the answer. Group 000 is not given any time constraint; hence the members must click the *Save* button to proceed to the next question. Only the question is displayed on the screen but there is no information about the time, i.e. no clock nor timer display. On the other side, a time limit of 30 seconds for each question is introduced to all the experimental groups. Group 100 is having the same interface as Group 000 except that the members are informed that they must complete the answer within 30 seconds, otherwise the page will be submitted automatically. Group 110 is given a clock display that is updated every second, Group 101 is given a count-down timer that flashes every second in yellow background, and Group 111 would see both clock and timer displays on the screen. Figure 3.4 shows sample clock and countdown timer display. Figure 3.5 displays the sample interface shown to the participants of Group 111, who are given both clock and timer.

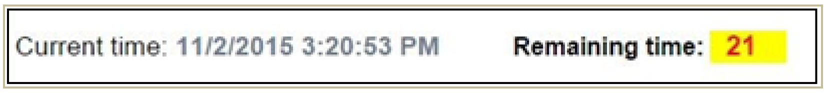


Figure 3.4. Clock display and countdown timer display

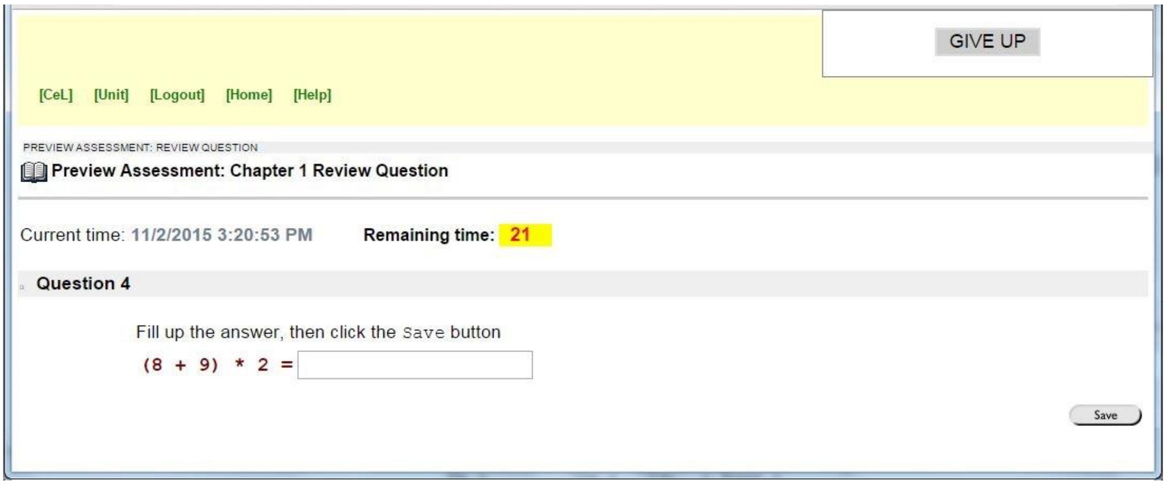


Figure 3.5. The sample web page for Group 111 with a clock display and a countdown timer that flashes every second

Table 3.2. Mental Arithmetic Problem and Demand

Task	Max digits in number	Amount of numbers	Arithmetic problem
1	1	2	6+2
2	1	2	9*4
3	1	3	6*5-1
4	1	3	(8+9)*2
5	2	3	7-8*10
6	2	4	58+20*(8-6)
7	2	4	67-2*(4+2)
8	3	5	(880+12+50-520)*2
9	3	5	105+83*5-3*60
10	3	5	561-81*5+3*610

3.3.3 TASK C: TYPING AND SUBJECT FAMILIARITY (TEXT TYPING)

This task enables the examination of the typing task demand and language familiarity effects on emotional stress, student's task performance, and mouse and keystroke behaviours. Six different typing tasks are set based on different text length and language familiarity. Three fixed texts are set in English as familiar language, and three in German as unfamiliar language. The requirements of the typing tasks are shown in Table 3.3. To determine the time limit to be given to the participants, a pilot test with 13 samples is conducted. The average duration to complete Question 3 is 26,730ms (or 26.73 seconds), Question 4 is 30,602 ms, Question 5 is 30,247 ms (100% of them made more than 40 typing errors), and Question 6 is 24,952 ms (76.92% of them made more than 40 typing errors). Therefore, a 30-second time limit is set for all the experimental groups to complete each task. The pilot results show that there is little possibility to complete 63 words within 30 seconds without any error. The reason to set much longer text but insufficient time for Question 5 and Question 6 is to push the participant's performance beyond limit, especially when they are under time pressure. Longer text is also believed to lead to boredom, tiredness and fatigue [54]. Figure 3.6 shows the distribution of typing errors for the pilot test. Similar to the assessment task, Group 000 is allowed to complete the questions without any time limit. Group 100 is given 30 seconds constraint but without clock nor timer display. Group 110 is given a clock display that is updated every second. Group 101 is given a count-down timer that flashes every second in yellow background, and Group 111 is given both clock and timer displays.

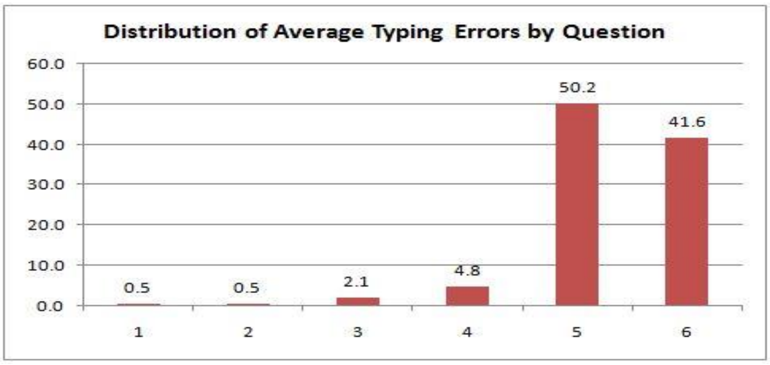


Figure 3.6. Distribution of Typing Errors by Question (sample size=13)

Table 3.3: Typing Task Demand.

Q	Text	Characteristics		Length	
		length	familiarity	Words	letters
1	Time flies like an arrow.	short	familiar	5	21
2	IchbringeSiezumFlughafen.	short	unfamiliar	5	25
3	Study by Lazar (2003) has shown that about one third of the time on computer is spent on frustrating experiences.	medium	familiar	20	94
4	Was denkenSiedaruber? Ichfahre morgen nach Dresden. Wannisst du zuMittag? Das schmeckt! Schonen Tag noch, Tschau.	medium	unfamiliar	20	99
5	Vizer stated that cognitive-stress tasks such as mental-multiplication and number-recall are widely used to induce cognitive-stress. Their results show that those keystroke-features that can be changed by cognitive-stress include keystroke-pause-length, keystroke-time, deletion-keys, navigation-keys and other keys (such as letter-keys and number-keys). However, we are more interested to examine the user-interface factors that may cause cognitive-stress in the e-learning environment, which include navigation designs.	long	familiar	63	459
6	Jeder hat das Recht auf Bildung. Die Bildungistunentgeltlich, zummindesten der Grundschulunterricht und die grundlegendeBildung. Der Grundschulunterrichtistobligatorisch. Fach- und Berufsschulunterrichtmussenallgemeinverfugbargemach twerden, und der HochschulunterrichtmuBallengleichermaBenentspreche ndihrenFahigkeitenoffenstehen. Die BildungmuB auf die volleEntfaltung der menschlichenPersonlichkeit und auf die Starkung der Achtungvor den Menschenrechten und Grundfreiheitengerichtet sein.	long	unfamiliar	63	451

To enable typing using conventional US keyboard, those umlauted vowels (e.g. ä, ö, and ü) in German language are replaced with basic alphabets (e.g. a, o, u)

3.4 SAMPLING OF PARTICIPANTS

The experimental and quantitative studies will be carried out with the convenience sampling method [234]. Convenience sampling is the most commonly used sampling method in behavioural science studies, where researchers simply get participants who are available and willing to respond. However, the sample must be students who have an e-learning system in their institution. In terms of sample size, we accept the margin of error (E) to be 10%, with a 90% confidence level (α=0.10). The recommended size is 67 for each experiment, based on the following [235] :

$$n = 0.25\left(Z_{\propto/2}/E\right)^2 \tag{3.11}$$

where$Z_{0.05}$=1.64 and E=0.1.

A total of 190 second-year undergraduate students who studied Bachelor Degree in Computer Science, Bachelor Degree in Information Systems, and Bachelor Degree in Information

Technology from Tunku Abdul Rahman University College, Malaysia, aged between 20 to 29 years old, were approached for their participations. Participants from narrow specializations and ages were selected under the constraint to control the effect of socio-demographic difference, such as age, on their stress perception when reacting to the interfaces during the experiments [35]. In addition, the searching task involves items that are IT subject-related, which prior knowledge is needed when searching a desired learning material. Although other socio-demographic factors, such as disability and gender, could affect the aims of the project, they are not being considered as control variables. This is because most of the students studying the above-mentioned programmes are male, and they do not have any disclosed disability. Since convenience sampling is utilized, the experiments are conducted during their classes with the consent by their teachers, hence all students are invited to participate.

The participants were randomly assigned to different design treatments and a control group in the preliminary research study. However, there was no control group for the laboratory experiments in the search task, where all students would run the same experiments with the same sets of search instructions. As for the assessment and typing tasks, the students were randomly assigned into 5 different groups, i.e. they were given either with/without time constraint or timing, with/without clock display and with/without countdown timer display. The groups were named following the code system below:

```
Timing (0 or 1) + Clock (0 or 1) + Timer (0 or 1)
```

where 0 means not available and 1 means available.

Group 000: It is the control group. The members are required to complete all 10 questions without any time constraint. They are required to click the Save button in order to proceed to the next question. There is no clock display nor countdown timer. There are 30 and 32 students who take part in the assessment and typing tasks respectively.

Group 100: This group is not given any display of clock nor countdown timer, but the members are given 30 seconds time constraint to complete each question in a task. There are 34 and 32 sudents who take part in the assessment and typing tasks respectively.

Group 101: This group is given a countdown timer that flashes every second with yellow background on the computer screen. The members are given 30 seconds time constraint. There are 31 and 32 sudents who take part in the assessment and typing tasks respectively.

Group 110: This group is given a digital clock displayed on the computer screen that shows the current date and time, which is updated every second.There is no countdown timer display. The members are given 30 seconds time constraint. There are 35 and 36 sudents who take part in the assessment and typing tasks respectively.

Group 111: This group is able to see both clock display that is updated every second, and a countdown timer that flashes continuously in yellow background on the computer screen. The members are given 30 seconds time constraint. For each assessment and typing tasks, there are 30 sudents who take part in the experiments.

For all the experimental groups, all questions will be submitted automatically once the time is up, if the participant did not submit the answer manually.

Fourteen sessions of experiments were conducted within 2 weeks. As the participants were given an option to withdraw from the experiments at any time, not all of them completed all the tasks. For those who provided valid data for the subsequent analyses, there were 151 participants for search task, 160 participants for assessment task, and 162 participants for typing task. Amongst these 190 students, 88.8% of them were male and almost all of them (99.4%) had at least one year of experience using the Blackboard e-learning system.

3.5 EXPERIMENTAL PROCEDURES

To ensure the same solution works even the e-learning learner switches to different task in between, three different experiments are setup to examine the effects of the tasks on learner's stress states. The three tasks that are commonly done in e-learning environment are (1) searching for a learning material, (2) assessment and (3) typing, which are already explained in Section 3.3.

The apparatus: To simulate those tasks in the e-learning environment and to avoid the results to be affected by unfamiliarity with the interface when they begin the tasks, a mock-up application is built based on the LMS that was used by the university students, i.e. Blackboard™ Academic Suite[3]. To collect the primary data from mouse and keyboard, two programs are written in Java and VB.NET separately to acquire mouse raw data and the virtual-key codes generated by the Windows platform. The collection of mouse raw data is recorded every 10 milliseconds, and their respective event time in milliseconds. The collection of keyboard raw data includes hit key code and its respective event time in milliseconds. To avoid huge variations of data that can be sensitive to detect significant difference even small departures from homogeneity and the assumption of normality, hence the collected raw data are transformed using the log_{10} function.

[3]The institution has upgraded the LMS to Blackboard Learn ™- Enterprise License (9.1.100401.0) since 2012 after the experiments were conducted

To protect user's privacy, the virtual-key codes are transformed into special codes automatically by the program. For instance, a number key or a letter key is recorded as 'k', delete key as '?' and backspace key as '*'. The construction of key and mouse loggers used for raw data collection will be presented in Section 3.6. All the experiments are conducted in a computer laboratory that is equipped with computers that run on Windows 7 with 17” monitor (resolution of 1024x768 pixels). In order to reduce invariabilities of mouse movements and typing behaviours that would affect the results, the students must use normal, external and common mouse and keyboard devices during the experiments. Every computer is equipped with 3.10 GHz CPU, 4GB RAM, an external standard QWERTY HID (acronym for Human Interface Device) keyboard and an external HID-compliant mouse. The web pages that show instructions and questions would run on Google Chrome web browser by default.

The consent: Before the experiments, the students are required to give consent to carry out the experimental tasks based on voluntarily basis. The details in the consent form are given in Appendix II Part A.

The calibration: Once they have agreed and proceeded to next page, they are required to perform calibration of their keystroke behaviours through a mock-up login page. Besides typing the usual username and password that the students are already familiar with, they are given a short sentence, i.e. “The quick brown fox jumps over the lazy dog” to reduce the practice effect [233] that may affect the calibration result. To calibrate their mouse behaviours, they are required to click 5 different hyperlinks that are distributed across the 4 corners and the centre of the screen as shown in Figure 3.7.

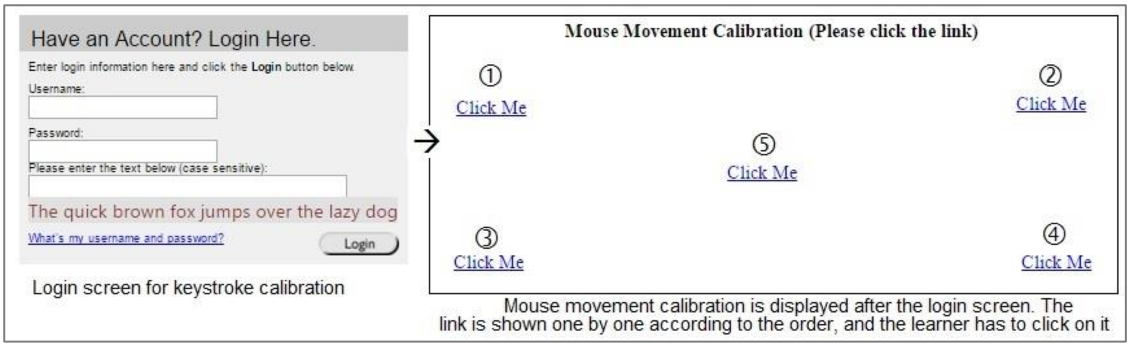


Figure 3.7. Keystroke and mouse movement calibrations required when login

The instructions: After the calibration process, the participants are given an instruction page to understand what activities they must do next each time before they start a new task. When they are ready, they need to click the *Start* button, and then an additional instruction regarding the first question of the task will be shown. When the first question of the task is displayed, the start time (in milliseconds) of the question will be recorded. For each question, if they wish to give up and skip to the next question, they could click the *Give Up* button placed on the top right corner of the screen. Once the *Give Up* button is hit, desire E and current status S (as defined in Equation 3.2 and Equation 3.3) are collected by the system automatically. Once a question is submitted or skipped, the end time (in milliseconds) is recorded. The entire experiments involving three tasks should take about 30 to 40 minutes for each participant. If any of the participants does not wish to complete the entire experiments, s/he can withdraw from the experiments at any time.

The next sub-sections present the detailed procedures for each task, i.e. search, assessment and typing.

3.5.1 TASK A: SEARCH FOR A LEARNING MATERIAL (MENU SEARCH)

Before the participants start the actual search task, a general instruction is displayed to guide them the area of search, as shown in Figure 3.8. For each of the 64 search instructions given to the participants, it requires the participants to read the instruction or cue of what module to search prior to the search action. When the participants are ready, they should click the *Start* button as shown in Figure 3.9. The start time in milliseconds would then be recorded. They should find the desired module on the dedicated menu as shown in Figure 3.8. They are required to click the correct hyperlink based on the given cue. If they are unable to locate the correct hyperlink, and wish to skip to the next question, they may click the *Give Up* button on the top right corner. For every mistake that a student makes, the number of attempt of the same task will be increased by one, and the accumulated average error rate *Err* (as defined in Equation 3.9) is computed. Any participant who wishes to revisit the question after losing focus, he or she could click the *Restart* button to recollect the instruction of the search task. Once the *Restart* button is hit, attention A (as defined in Equation 3.5.1) is collected by the system automatically. Upon completion of every question, or when the *Give Up* button is pressed, the end time (in millisecond) is recorded, and task duration is computed. The next question is randomly assigned to the participants until they complete all 64 questions.

Finally, a learner's self-report stress perception form that is explained in Section 3.3.1 is displayed, so that the impacts of menu design on user emotional stress can be evaluated.

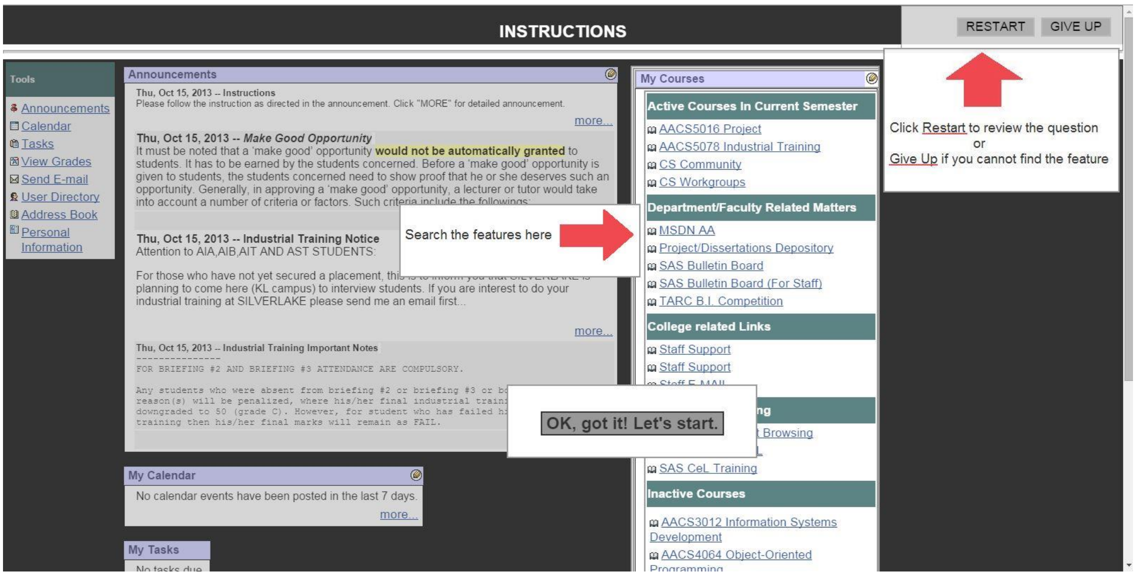


Figure 3.8. A guide given to the participants about the search task before start

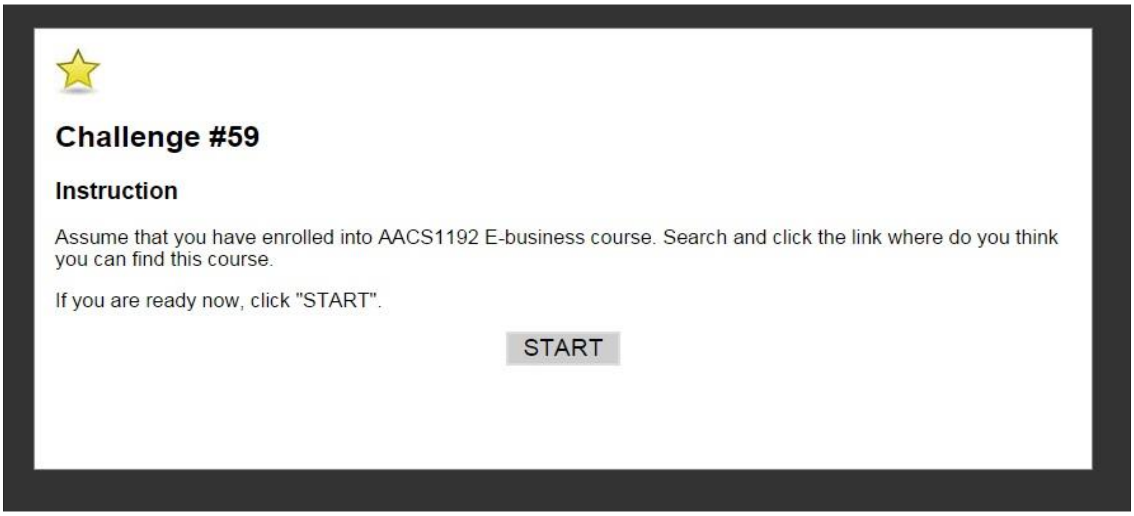


Figure 3.9. Sample instruction given to the participants prior to the search task

3.5.2 TASK B: ASSESSMENT (MENTAL ARITHMETIC)

In the assessment task, the participants are required to answer 10 arithmetic questions using mental ability, i.e. no calculator to be used, and working the solution on a paper is prohibited. Before the participants start solving the actual mental arithmetic problem, a general instruction is displayed to inform what they should do during the task. The instruction is shown in Figure 3.10. For each of the 10 arithmetic problems given, the participants are required to type the answer into a textbox. A sample interface of a mental arithmetic question is shown in Figure 3.11. To force the use of mouse so that mouse dynamics could be collected, the *Enter* key is disabled so that the participants must use a mouse to click on the *Save* button to submit the page. For the experimental groups who are given a time constraint, if they do not click the Save button before the time is up, the page will be submitted automatically when the time limit is reached. If the page is submitted automatically by the system, then attention *A* (as defined in Equation 3.5.2) will be computed. Anyone who wishes to skip to the next question, they may click the *Give Up* button on the top right corner. If the answer submitted by a participant is wrong, or if the student gives up the question, the error of the question will be set as one, and the accumulated average error rate *Err* (as defined in Equation 3.9) is computed. Upon completion of every question, or when the *Give Up* button is pressed, the end time in milliseconds is recorded, and task duration *TD* is computed. Then the next question is displayed according to the pre-determined order as shown in Table 3.2.

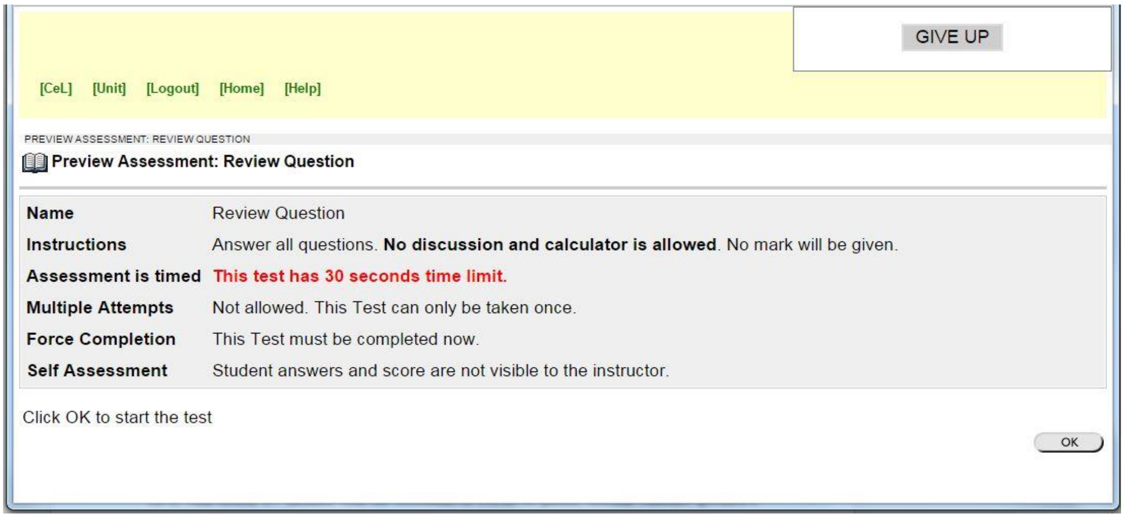


Figure 3.10. The sample instruction page prior to the first mental arithmetic question

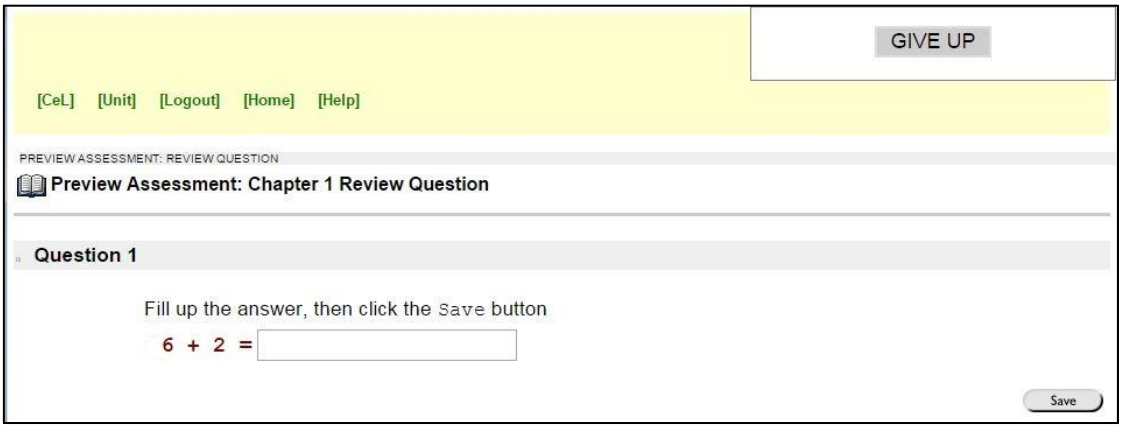


Figure 3.11. The sample web page for Group 000 and Group 100. The students can click the Give Up button on the top right corner, or the Save button on the bottom right corner to submit the answer

3.5.3 TASK C: TYPING AND SUBJECT FAMILIARITY (TEXT TYPING)

In the typing task, the participants are required to type the pre-determined text into a textbox. There are 6 questions with various text lengths, with 3 questions in English and 3 in German. Before the participants start the actual typing task, a general instruction is displayed to inform what they should do in the task. The instruction is shown in Figure 3.12. A sample interface of a typing question is shown in Figure 3.13. To force the use of the mouse so that mouse dynamics could be collected, the *Enter* key is disabled so that the participants must use a mouse to click on the *Save* button to submit the page. For the experimental groups who are given a time constraint, if they do not click the *Save* button before the time is up, the page will be submitted automatically when the time limit is reached. If the page is submitted automatically by the system, then attention A (as defined in Equation 3.5.2) will be computed. Anyone who wishes to skip to the next question may click the *Give Up* button on the top right corner. The amount of typographical mistakes made by a participant in a given text upon submission or giving up is counted and scaled using the log_{10} function. The accumulated average error rate Err (as defined in Equation 3.9) is then computed. Upon completion of every question, or when the *Give Up* button is pressed, the end time (in millisecond) is recorded, and task duration is computed. Then the next question will be displayed according to the pre-determined order as shown in Table 3.3.

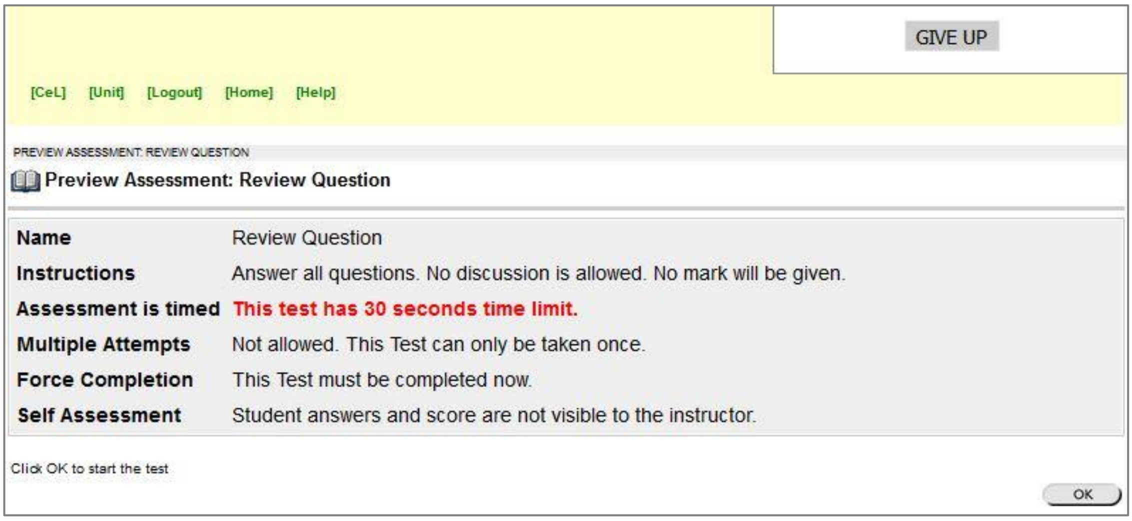


Figure 3.12. The sample instruction page prior to the first typing question

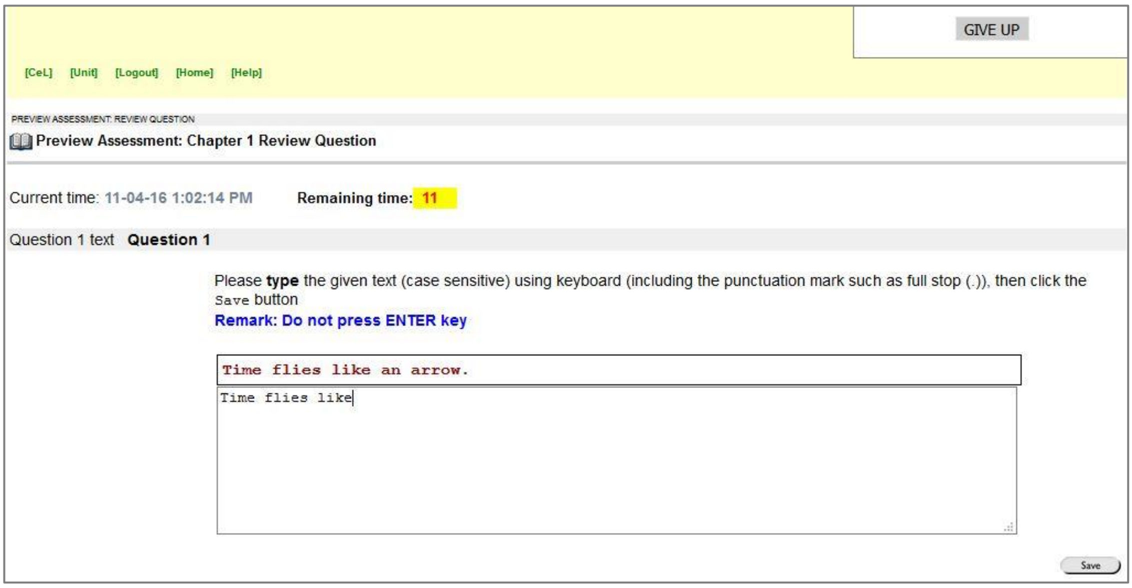


Figure 3.13. The sample web page for Group 111 with a clock display and a countdown timer that flashes every second

3.5.4 ETHICS

Evaluations should be performed in a professional and ethical manner. Participants' rights must be protected. All participants must sign a consent form and receive information documents, explaining the purpose and those findings are used for stated purposes only. Anonymity is assured as no private data that reveal the individual identity are collected. Participation is voluntary, and a participant could withdraw at any time. They receive no reward for participating. Questionnaires are pilot tested to avoid any potential misunderstanding. These evaluations are conducted in an ethical and socially responsible manner.

In the search task, some of the pages are purposely designed with low usability, such as inappropriate combination of text colour and background colour, smaller font size, etc., which may cause eye fatigue or eye strain. The participants are given this information at the very beginning before they agree to continue the experiments. They are provided the option to give up on the task by clicking the *Give Up* button, if they feel uncomfortable with the page and could not proceed. They are allowed to withdraw from the experiments at any time if they do not wish to continue.

Since a key logger is used to record user's keystrokes, privacy must be embedded into the design and architecture of the system. We must be offering measures as strong privacy defaults, appropriate notice and empowering user-friendly option [125]. Therefore, the users should be given an option for choosing not to be observed by the real-life system. The actual data of the keys used, which reflect the original content of the text (such as username and password) must not be stored. If have to be stored, these data must be encoded for the use of the analysis purpose only, for instance a number key or a letter key is recorded as 'k', delete key as '?' and backspace key as '*'. The actual hit key-codes will not be stored. We need to ensure that at the end of the process, all collected data are kept confidential, secure and safe. User's profile is identified through randomly generated keys and no data that reveals the participant's identity will be kept in the database. Upon the completion of the research, all data shall be securely destroyed in a timely fashion.

3.6 DATA COLLECTION AND APPARATUS DESIGN

At the heart of the stress inference engine is the hooking module designed to monitor keystroke and mouse dynamics as background process. To enable keystrokes and mouse dynamics to be collected, a custom function must be added into the Windows for the relevant I/O event types. The hooking process must be global, i.e. it must be able to monitor user's keystroke and mouse behaviours outside the context of the host application. This is important as the user may switch between tasks or windows. This section presents the designs of a key logger and a mouse logger that collect keystroke and mouse raw data before the data are recomputed and modelled into user behaviour for the continuous stress monitoring process.

3.6.1 THE CONSTRUCTION OF KEY LOGGER

The difficulty of constructing a key logger is to enable an effective and safe software to users. People often relate key logger as a surveillance tool or spyware, which can be embedded in user's computer that allows information to be intercepted or transmitted to an unknown third party [236]. There is a high possibility that only few users would agree to risk their privacy and security if a key logger has to be used to monitor their emotion. Besides, there are some technical challenges that must be solved in order to ensure a robust and reliable system. The requirements of the key logger are as follows.

- The key logger must be carefully designed with additional protection to ensure the sensitive input, such as username and password, are filtered and excluded from being stored in any form of the database.

- The data collection must be efficient and speedy. The code inside the core hook function should not only be reliable but it must be able to record the data without delaying the performance of the entire system. As such, the hook function should only focus on gathering data on the raw key data using suitable data structure, and transfer the data from buffer into proper storage that should not delay the system including the inference engine itself.

- The data storage must be effective. As for each keystroke produces repeated key code, i.e. once on the key down event and once on the key up, this means that we should be careful on the storage of the raw data especially for a task that requires a user to type a lot of text. The raw data should be manipulated and processed quickly into desired information in a timely manner and the old storage should then be wiped out for new incoming data.

- The keyboards that are used may be varied by users. The quality of the keyboard used might affect the user's mood too. Therefore, it is important to enable a calibration to be done before the actual real-time data to be collected. The keystroke behaviour collected during this process is considered a 'normal' keystroke behaviour, as the task should not induce additional stress to the user. For instance, a calibration can be done during the login process, to ensure readings from the subsequent 'normal' keystroke dynamics are consistent with the keystroke during calibration. Measurements are traceable when the subsequent keystroke can be related to the calibrated keystroke data through statistical comparisons.

The sub-sections below explain the technical implementation of the key logger module.

3.6.1.1 KEY LOGGER DESIGN AND DEVELOPMENT

There were a few technical problems encountered when building the key logger at the initial stage. The key logger was first built using Java so that it is platform independent. Accordingly, the key logger can be installed and run on any operating system. Unfortunately pure Java does not support global key hook due to Java Virtual Machine (JVM) security issues [237]. Therefore, keyboard listeners in Java only work if the registered component has the focus on the window. If any window loses its focus, e.g. is minimized, then it is not possible to track any keyboard events anymore. This is unusable especially for a web-based e-learning system which activities should be focused on the web browser but not the Java window. To enable global keyboard and mouse listeners for Java, JNativeHook in the Java Native Interface (JNI) library can be used to enable listening for global shortcuts. To accomplish this task, JNativeHook leverages platform-dependent native code, such as C or C++, through Java's native interface to create low-level, system-wide hooks and deliver those events to the application (https://code.google.com/p/jnativehook/). However, this method is inflexible as the programmer needs to write the code in both Java and a native code, and the outcome is platform dependent. If used in Mac OS and Linux, they could not work as a different platform that provides its own virtual key codes. Using JavaScript is easy and but the code must be tied to the web pages, and so it does not provide great flexibility in detecting stress in any page or any website. Lastly, we considered detecting keystrokes using VB.NET as it does not only provide full library of keyboard events, but it enables a key logger to be built without using hooks. Although the system is platform dependent, but this allows speedy process to detect the pressed keys by simply using the `GetAsyncKeyState()` and `GetKeyState()` built-in functions [238], [239].

3.6.1.2 `GETASYNCKEYSTATE()` AND `GETKEYSTATE()` IN VB.NET

Both `GetAsyncKeyState()` and `GetKeyState()` can be used to determine whether a key is up or down at the time the function is called. However, there is a difference between the two. While processing a keyboard input, we may need to determine the status of another key besides the one that generated the current message, for instance, a user may press `SHIFT+A` to type a capital letter of '`A`'. The key logger must check the status of the `SHIFT` key whenever it receives a keystroke message from the '`A`' key. The key logger can use the `GetKeyState()` function to determine the status of a virtual key at the time the current message was generated; and it can use the `GetAsyncKeyState()` function to retrieve the current status of a virtual key [240]. The differences between `GetAsyncKeyState()` and `GetKeyState()` given by the [238], [239] are shown in Table 3.4. Table 3.5 illustrates the requirements of the environment to develop the key logger using `GetAsyncKeyState()` and `GetKeyState()` functions.

Table 3.4: Differences between `GetAsyncKeyState()` and `GetKeyState()` Functions [238], [239]

`GetAsyncKeyState`	**`GetKeyState`**
To determine whether a key is up or down at the time the function is called, and whether the key was pressed after a previous call to `GetAsyncKeyState`.	Retrieves the status of the specified virtual key. The status specifies whether the key is up, down, or toggled (on, off—alternating each time the key is pressed). An application calls `GetKeyState` in response to a keyboard-input message. This function retrieves the state of the key when the input message was generated.
It should be used to retrieve the current state for an individual key regardless of whether the corresponding keyboard message has been retrieved from the message queue	It should be used to retrieve status information for an individual key.

Table 3.5: Requirements of `GetAsyncKeyState()` and `GetKeyState()` Functions [238]

Minimum supported client	Windows 2000 Professional [desktop apps only]
Minimum supported server	Windows 2000 Server [desktop apps only]
Header	Winuser.h (include Windows.h)
Library	User32.lib
DLL	User32.dll

3.6.1.3 KEYSTROKE DATA DESIGN AND STORAGE

It is important to strike a balance between gathering user input for statistical analysis and the level of trust required by the end users. Therefore, we dispense the actual data and encode them into a less meaningful representation that is sufficed for further statistical inference. As a result, the actual virtual key code is filtered and encoded accordingly as shown in Table 3.6. By encoding the actual virtual key code, no one should be able to capture or steal the actual private or sensitive data such as username and password from the storage. Figure 3.14 shows the user interface of the key logger with encoded virtual key on the dialog box for testing purpose.

Table 3.6: Encoded Virtual Key for Data Storage and Privacy Control

Virtual Key Code	**encoded key**	**meaning**
8	*****	backspace
13	**l**	newline
16	**#**	shift
1-31	**-**	other system key, except 8, 13 and 16
32	**s**	space bar
46	**?**	delete
48 – 57	**n**	number
64 – 122	**C**	alphabet
128 - 255	**+**	other special character

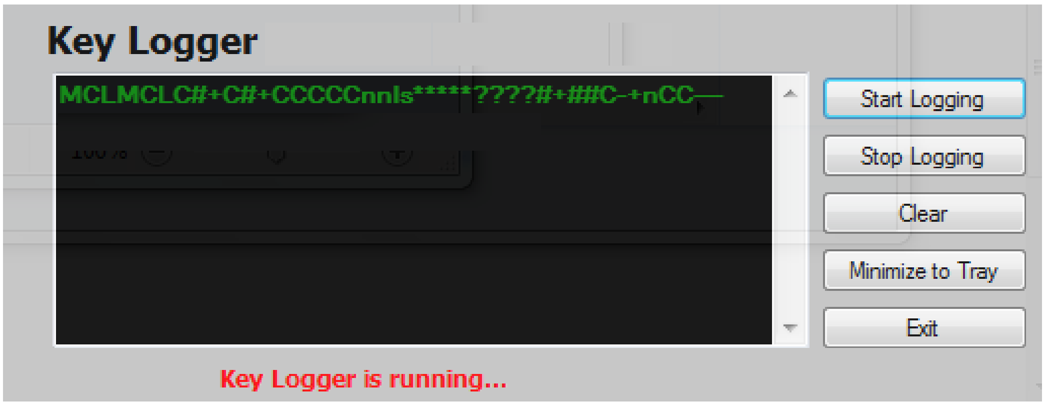


Figure 3.14. Interface of key logger

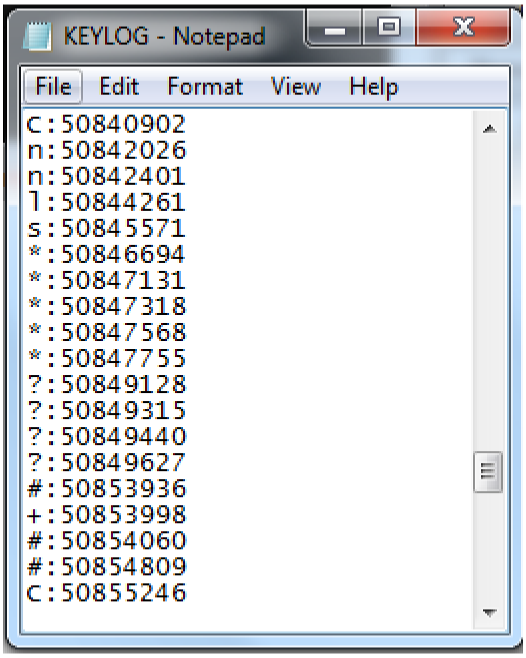


Figure 3.15. Sample data stored in the text file "`keylog.txt`" with the format `Encoded code:Time Stamp`

Every keystroke data is stored into the local hard drive as a text file. The reasons of using text file are simply because it is very simple to create and it is considered an efficient storage of binary document, and it is very commonly used in the read/write process [241]. Since only text is stored for processing, a relational database is not under consideration as inserting or retrieving data into/from the database server in a frequent timely manner could lead to delay in the performance of the system, while appending data into text file can be done instantly. Furthermore, we do not need to run complicated queries in order to retrieve the data from the database. Figure 3.15 shows how the actual encoded virtual key stored in a text file.

3.6.1.4 KEYSTROKE DATA PRE-PROCESSING

Before the user's keystroke dynamics are modelled into keyboard behaviour, some pre-processing is needed to compute the raw data into the forms that are useful for statistical inference. Algorithm 3.1 to Algorithm 3.3 illustrate the procedures to compute the error key rates

(KE), typing speed (KS), and keystroke latency (KL), which are used for the keystroke behaviour modelling (see Section 3.7.3).

ALGORITHM 3.1: TO DETECT ERROR KEY RATE

Keystroke Error (KE):

```
        if(keyMsg.Equals("?"))
                deleteCount++
        if(keyMsg.Equals("*"))
                backspaceCount++
        KE = ∑ deleteCount + ∑ backspaceCount
```

We are interested to find out the number of typing errors produced by the user when he or she is typing a given text. This can be done by detecting the frequency of the delete key or backspace key used.

ALGORITHM 3.2: TO DETERMINE TYPING SPEED

Keystroke Speed (KS):

```
        if(keyMsg.Equals("C") || keyMsg.Equals("n") || keyMsg.Equals("+"))
                keystroke++
       KS = ∑ keystroke / ∑ duration* 1000 //number of keystroke per second
```

As the given text only consists of alphabets (C), numbers (n) and special characters (+), we are only interested to determine the speed a user uses to type the text. Therefore, we do not put other keys that are used (e.g. system key) into the computation.

ALGORITHM 3.3: TO GET KEYSTROKE DOWN-DOWN LATENCY

Keystroke Latency (KL)

```
       if        ∑ keyPress = ∑ keyMsg
                   t1 = getTimeStamp(previousMsg)
                   t2 = getTimeStamp(nextMsg)
                   keyPressDuration = t2 – t1
       then
                 KL = ∑ keyPressDuration / ∑ keyPress
```

Lastly, to reduce computation load to the system, we only consider the Down-Down key latency, which is the elapsed time between 2 subsequent keypresses. We determine the average duration of a single keypress rather than the total keypress time over the total duration.

3.6.2 THE CONSTRUCTION OF MOUSE LOGGER

Building an application that captures only keyboard input is insufficient, as not all applications require input from a keyboard. It is important to complement the application with mouse input, which it receives mouse input in the form of messages that are sent to its windows. The difficulty of constructing a mouse logger is that the application must be able to deal with large amount of generated data in speedy and effective manner. The requirements of the mouse logger are similar to key logger, which are as follows:

- The data collection must be efficient and speedy. The code inside the mouse hook function must be reliable, and is able to record the data without delaying the performance of the entire system. As such, the hook function should only focus on gathering data on the raw mouse data using suitable data structure, and transfer the data from buffer into proper storage that should not delay the system.

- The data storage must be effective. Every mouse click will generate repeated virtual key code, i.e. once on the mouse button down event and another on button up. Besides, mouse input data such as mouse position and time stamp of the mouse event are collected with the interval of 10 milliseconds (ms). The data collection is huge as there are at least 600 mouse data to be recorded every minute. Therefore, the raw data should be manipulated and processed quickly into desired information in a timely manner, and the old storage should then be wiped out for new incoming data.

- There may be different models of mouse for different computer users. The quality of the mouse might affect the user's emotion too. Therefore, it is important to enable a calibration to be done during the login process before the actual real-time data to be collected. The mouse behaviour collected during this process is considered as 'normal' mouse behaviour, which the task shall not induce unnecessary stress to the user. This is to ensure the readings from the subsequent 'normal' mouse dynamics are consistent with the mouse activities during calibration. Measurements are traceable when the subsequent mouse dynamics can be related to the calibrated mouse data through statistical comparisons.

The sub-sections below explain the technical implementation of the mouse logger module.

3.6.2.1 MOUSE LOGGER DESIGN AND CONSTRUCTION

The mouse logger should capture the user's mouse movements continuously and constantly. When the user moves the mouse, the system moves a bitmap on the screen called "mouse cursor". The mouse cursor consists of a single-pixel point called "hot spot", which points the position of the cursor that contains horizontal (x) and vertical (y) coordinates. When a mouse event occurs, the window that contains the hot spot typically receives the mouse message resulting from the event [242]. There were a few technical problems encountered when building the mouse logger at the initial stage. The mouse logger was initially planned to be integrated together with the key logger that was built earlier using VB.NET. Unfortunately, although the application does not require the window to be active or have the keyboard focus in order to receive a mouse message, only the foreground window can capture mouse input. When a background window attempts to capture mouse input, it receives messages only for mouse events that occur when the cursor hot spot is within the visible portion of the window [242]. Therefore, we moved the development to Java using the `processing.core.PApplet` class, which is an easier solution that allows the mouse logger to capture the cursor hot spot (*x* and *y* coordinates), and mouse events such as mouse pressed, mouse released, mouse moved, mouse dragged, mouse button and mouse wheel events. We also planned to include mouse wheel events in the analysis. However, a problem with mouse wheel data collection was encountered later, in which the mouse wheel rotations are not played back or recorded properly [243]. Besides, the Visual Basic 6.0 IDE does not have built-in support for scrolling by using the mouse wheel, so the IDE ignores the WM_MOUSEWHEEL message [244]. A special driver, i.e. VB6 Mouse Wheel.exe, needs to be included to send out messages that can be caught by the application, aside from standard events such as mouse moved. Unfortunately, although this problem was solved later by using the `PApplet` class, the captured mouse wheel data were either incomplete or experienced slight delay in data recording. This made the data collection became very much unreliable. As such, we decided to drop mouse wheel input from the data collection later.

Table 3.7: Java Methods Used to Capture Mouse Events and Mouse Position [245]

`draw()`	There can only be one `draw()` function for each sketch and `draw()` must exist if the code needs to run continuously or to process events such as `mousePressed()`. This is needed as we still need to capture the cursor hot spot position although there is not mouse activity at all. To capture the hot spot position, we could insert the following code inside the `draw()` method:
	`mousePosition = MouseInfo.getPointerInfo().getLocation();`
	`int x = mousePosition.x;`
	`int y = mousePosition.y;`

`mouseMoved()`	It is called every time the mouse moves and a mouse button is not pressed. This is used to capture the mouse speed.
`mousePressed()`	It is called once after every time a mouse button is pressed.
`mouseReleased()`	It is called every time a mouse button is released

`mouseClicked()`	It is called once after a mouse button has been pressed then released. This is used to determine which mouse button is clicked, e.g.:
	`button = e.getButton();`
	`if(button == 1) //left button`
	`msg = "MCL ";`
	`else if(button == 3) //right button`
	`msg = "MCR = ";`

3.6.2.2 THE PROCESSING.CORE.PAPPLET CLASS

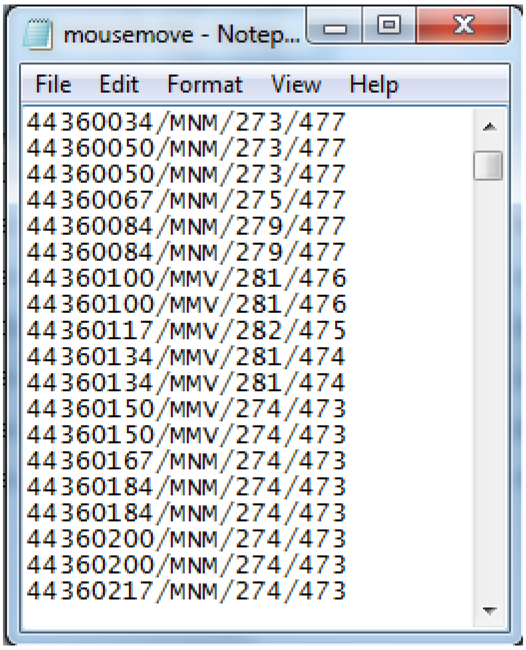
`PApplet` is the base class for all sketches that use `processing.core`. `Processing` uses active mode rendering [245]. The methods used to capture the mouse events are shown in Table 3.7.

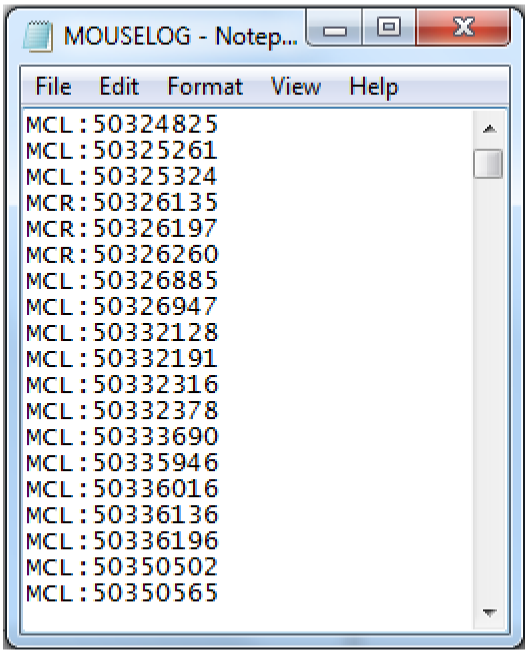
3.6.2.3 MOUSE DYNAMICS DATA DESIGN AND STORAGE

To ease data retrieval and pre-processing for the preparation of the mouse behaviour modelling, the captured mouse raw data are encoded as shown in Table 3.8. Similar to key logger, all mouse data are stored into the local hard drive as text files, as text file is an efficient storage especially when complicated query is not needed, and large amount of data need to be stored and processed in a timely manner. Figure 3.16 shows how the actual encoded mouse data are stored in a text file. Figure 3.17 displays the user interface of the mouse logger with encoded data on the dialog box for testing purpose. Figure 3.18 shows the mouse motion tracker window that draws the mouse motion of a user on the monitor regardless which window is active.

Table 3.8: Encoded Mouse Events for Data Storage

Encoded key	meaning
MP	Mouse Pressed
MR	Mouse Released
MNM	Mouse Not Moved (Idle)
MMV	Mouse Moved
MCL	Mouse Click (Left)
MCR	Mouse Click (Right)





(A) (B)

Figure 3.16. Sample data stored in the text file (A) "mousemove.txt" and (B) "mouselog.txt" with the respective format Time Stamp/Encoded code/x position/y position and Encoded code:Time Stamp

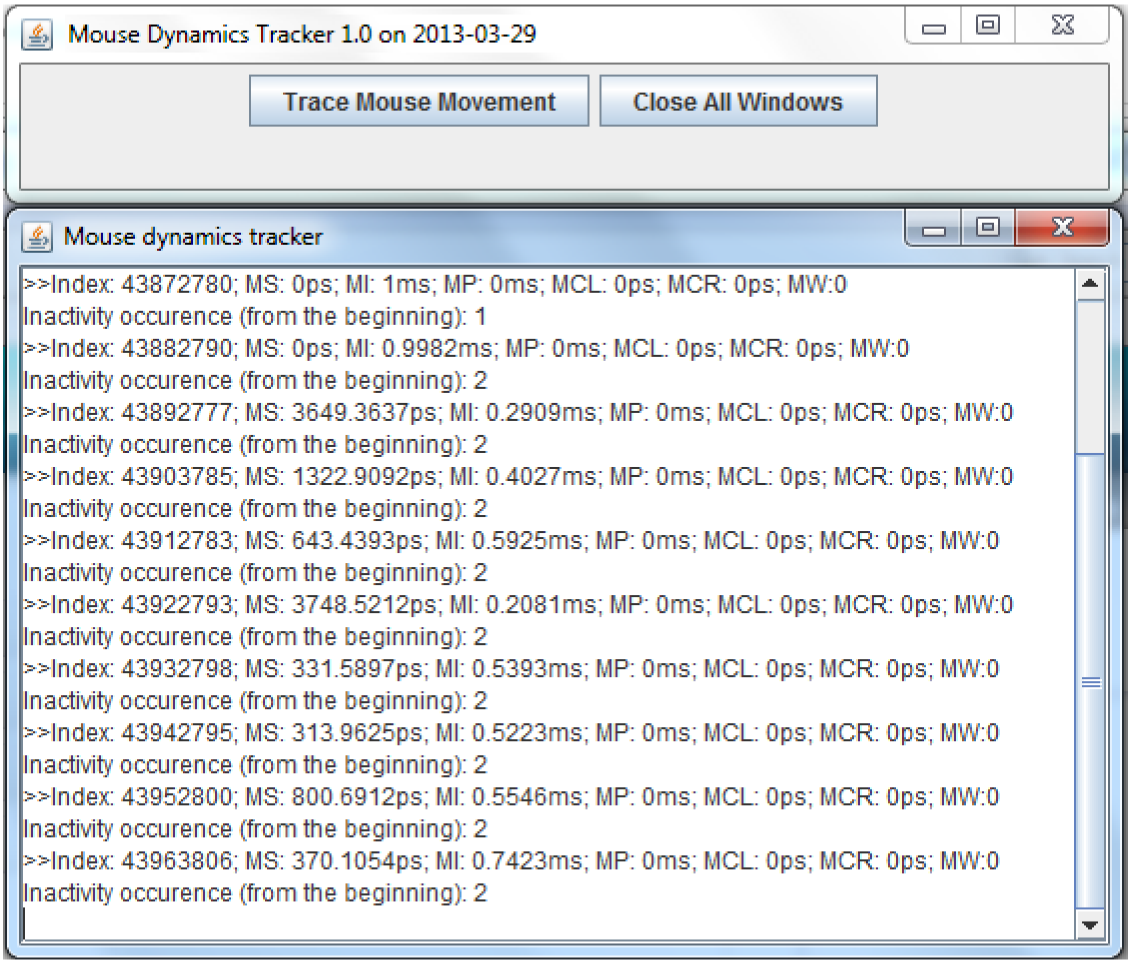


Figure 3.17. The user interface of the mouse logger

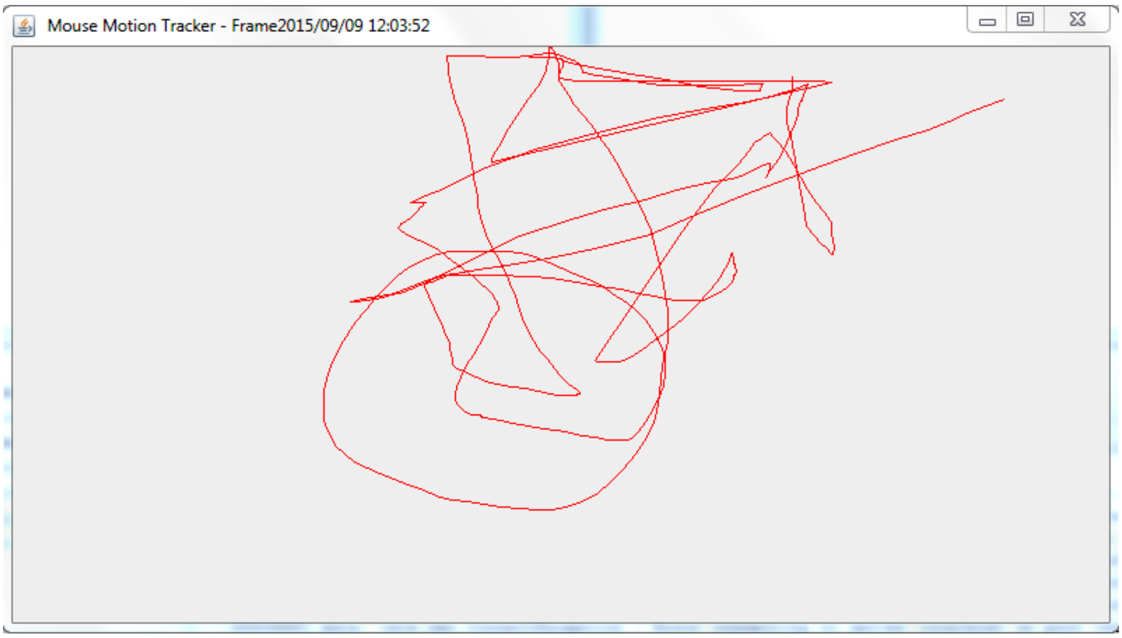


Figure 3.18. Mouse motion tracker window that draws the mouse motion of the user

3.6.2.4 MOUSE DATA PRE-PROCESSING

Before the user's mouse dynamics are modelled into mouse behaviour, some pre-processing is needed to compute the raw data into the forms that are useful for statistical inference later.

Algorithm 3.4 to Algorithm 3.8 illustrate the procedures to compute the mouse speed (MS), mouse idle occurrences (MIO), mouse idle duration (MID), mouse left press duration (MPL), mouse right press duration (MPR), mouse left click rate (MCL) and mouse right click rate (MCR), which are used for the mouse behaviour modelling (see Section 3.7.4).

ALGORITHM 3.4: TO DETECT MOUSE SPEED

```
Mouse Speed (MS):
    if      t1 = getTimeStamp(msgPrevious)
              t2 = getTimeStamp(msgNext)
              moveDuration = t2 – t1
              distance = Math.sqrt(Math.pow(x2-x1,2) + Math.pow(y2-y1,2));
   then
              MS = ∑ distance/ ∑ moveDuration*1000; // pixel per second
```

Mouse speed is determined when only the mouse is moved, therefore the speed is computed against the total moving duration, but not the total task duration.

ALGORITHM 3.5: TO DETECT MOUSE IDLE OCCURENCES

```
Mouse Idle Occurences (MIO):
    if (msgPrevious.Equals("MNM") && !msgNext.Equals("MNM"))
   MIO ++
```

"`MNM`" (the abbreviation of "mouse not moving") is recorded when there is a mouse inactivity detected, and this record will be stored based on 10 milliseconds (ms) interval, until the mouse is active again. The idle occurrences, MIO, will only be updated once when `MNM` is detected until another mouse event occurs. For instance, consider a mouse that is idle for 60 ms until it is moved again, MIO will only be increased by 1 as there is only 1 occurrence of inactivity.

ALGORITHM 3.6: TO GET TOTAL MOUSE IDLE DURATION

```
Mouse Idle Duration (MID):
   if (msgPrevious.Equals("MNM") && msgNext.Equals("MNM"))
          t1 = getTimeStamp(msgPrevious)
          t2 = getTimeStamp(msgNext)
   idleDuration = t2 – t1
   MID = ∑ idleDuration
```

Different from MIO, the Algorithm 3.6 is to record the total mouse idle duration, i.e. the total elapsed time between two "`MNM`" messages that are captured. For instance, consider a mouse that is idle for 60 ms until it is moved again, then MID = 60.

ALGORITHM 3.7 TO GET MOUSE LEFT PRESS DURATION

```
Mouse Left Press Duration (MPL):
   if(msgPrevious.Equals("MCL") && msgNext.Equals("MCL")
              t1 = getTimeStamp(msgPrevious)
              t2 = getTimeStamp(msgNext)
              leftPressDuration = t2 – t1
    MCL = ∑ leftPressDuration / ∑ duration
```

```
ALGORITHM 3.8 TO GET MOUSE RIGHT PRESS DURATION
Mouse Right Press Duration (MPR):
        if(msgPrevious.Equals("MCR") && msgNext.Equals("MCR")
                t1 = getTimeStamp(msgPrevious)
                t2 = getTimeStamp(msgNext)
                leftPressDuration = t2 – t1
         then
                MCL = ∑ rightPressDuration / ∑ duration
```

Both of the Algorithm 3.7 and Algorithm 3.8 are similar except that one is determining the duration of left mouse button press and another is for right mouse button press. This is to determine the duration that a user takes to press a mouse button before the button is released.

```
ALGORITHM 3.9 TO GET MOUSE LEFT CLICK RATE
Mouse Left ClickRate (MCL):
        if(msgPrevious.Equals("MCL") && msgNext.Equals("MCL")
                leftClick ++
        MCL = ∑ leftClick
```

```
ALGORITHM 3.10 TO GET MOUSE RIGHT CLICK RATE
Mouse Right ClickRate (MCR):
        if(msgPrevious.Equals("MCR") && msgNext.Equals("MCR")
                rightClick ++
        MCR = ∑ rightClick
```

For Algorithm 3.9 and Algorithm 3.10, we are interested to determine the frequency of mouse button clicks. As one click event will generate a repeated message, i.e. once when the button is down and another is generated when the button is up, therefore 2 consequential MCL (or MCR) message will increase MCL (or MCR) by one.

3.7 BEHAVIOUR MODELLING

Not all data collected are necessarily useful for analysis and therefore feature extraction should take place before the data are analysed. Feature extraction is mainly used to reduce the measurement and storage requirements, and to minimize training and utilization times, so that the prediction performance can be improved. To model the user behaviour efficiently, that are three key features to be included: keyboard typing rhythms, mouse activities, and task performance such as errors made. Therefore, ***User behaviour*** is defined as a dataset that describes the user's keystroke dynamics, mouse dynamics and list of task behaviours. We assume that the identified key features could be affected by emotional factors, particularly stress. The keystroke behaviour and mouse behaviour are computed and constructed after each question that the user has performed. Alternatively, to enable continuous stress monitoring without the information whether an instruction is completed, we may allow the keyboard activities and mouse movement to be computed every 10 seconds, as this is the best interval recorded by Tsoulouhas et al. [54]. Several datasets are built to model user behaviour, the tasks that he or she has performed and the correspondent keystroke and mouse dynamics. The sub-sections below illustrate that models that we build for the stress inference process.

3.7.1 USER BEHAVIOUR MODEL

Table 3.9 shows the user behaviour models that are recorded in the individual *user behaviour* dataset, *B(U)*. Table 3.10 shows the user default behaviour dataset, $B(U_0)$.

Table 3.9: User Behaviour, *B(U)*

Property	Description	Remark
UserID	User ID	Each user is given a randomly generated number (maximum 5 digits). The ID is generated before the calibration process started.
$B(U_0)$	User default behaviour	This records the default keystroke behaviour, mouse behaviour, task performance behaviour and stress perception that are captured during the calibration process.
List<*Task*>	List of tasks	Task is a dataset that records the correspondent task ID, keystroke behaviour, mouse behaviour, task performance behaviour, user stress perception regarding the task, and stress level classification based on the correspondent keystroke and mouse behaviours (see Table 3.11).

User default behaviour is set during the calibration process, which is used to determine whether the user stress is stable (normal), increased or decreased.

The mathematical representation or formulation of the user behaviour dataset is therefore defined as follows:

$$B(U) = \langle UserID, B(U_0), List\langle Task\rangle\rangle \qquad (3.12)$$

where

$$B(U_0) = \langle B(T_0), B(K_0), B(M_0), SP_0\rangle \qquad (3.13)$$

3.7.2 TASK AND TASK PERFORMANCE BEHAVIOUR MODEL

Task is a dataset that measures activities related to the tasks that a user has completed. We will classify the stress level produced by the task based on the correspondent mouse and keystroke behaviours. Table 3.11 describes the detailed features of the *Task* dataset.

Task performance B*(T)*, describes the performance of a given subtask, such as the completion time, number of errors made and passive attempt. Table 3.12 shows the details of the key features in the task performance behaviour.

The mathematical formulation of the task dataset is therefore defined as below:

$$Task = \langle TaskID, B(T), B(K), B(M), SP, S_{B(K)}, S_{B(M)}, S_{B(M,K)}\rangle \qquad (3.14)$$

where

$$B(T) = \langle TD, Err, PA\rangle \qquad (3.15)$$

Table 3.10: User Default Behaviour, $B(U_0)$

Property	Description	Remark
$B(T_0)$	Default Task Performance	$B(T)$ includes the task duration, error rate (e.g. wrong answer), use of error key (e.g. backspace or delete), and passive attempt (i.e. give up attempt or wait until the time is up (see Table 3.12). $B(T_0)$ stores the default values of task performance variables, which is 0.
$B(K_0)$	Default Keystroke Behaviour	$B(K)$ is a dataset that includes keystroke latency, typing speed and error key rate (see Table 3.13). $B(K_0)$ stores the default dataset of keystroke dynamics collected during the calibration process.
$B(M_0)$	Default Mouse Behaviour	$B(M)$ includes the movement speed, elapsed time, mouse idle occurrences, mouse press, and mouse click rate (see Table 3.14). $B(M_0)$ stores the default dataset of mouse dynamics collected during the calibration process.
SP_0	Stress Perception	SP is collected through a survey that enables the participants to (subjectively) assess their stress level when performing a task. Each time after the students completed a task, a self-report survey with 7-point Likert scale will be displayed – "You felt stressed when answering the previous question", where 1 for strongly disagree and 7 for strongly agree. SP_0 is collected during the calibration process.

Table 3.11: Task

Property	Description	Remark
TaskID	Task ID	This records the question number of a particular subtask, e.g. there are 10 questions for the mental arithmetic task. The task ID for Question 1 is `taskB-1`.
$B(T)$	Task performance behaviour	$B(T)$ includes the task duration, error rate (e.g. wrong answer), use of error key (e.g. backspace or delete), and passive attempt (i.e. give up attempt or wait until the time is up) (see Table 3.12).
$B(K)$	Keystroke behaviour	This is the keystroke behaviour correspondent to a particular subtask. This is null for menu search task as it does not require any keyboard input (see Table 3.13).
$B(M)$	Mouse behaviour	This is the mouse behaviour correspondent to a particular subtask (see Table 3.14)
SP	Stress perception	SP is collected through a survey that enables the participants to (subjectively) assess their stress level when performing a task. Each time after the students completed a task, a self-report survey with 7-point Likert scale will be displayed – "You felt stressed when answering the previous question", where 1 for strongly disagree and 7 for strongly agree. SP is unavailable for menu search task.
$S_{B(K)}$	Stress measurement based on $B(K)$	This is the projected value by the stress inference engine that represents the stress level of the user correspondent to a subtask. $S_{B(K)}$ is generated based on the correspondent keystroke behaviour. $S_{B(K)}$ is null for the menu search task as it does not require keyboard input.
$S_{B(M)}$	Stress measurement based on $B(M)$	This is the projected value by the stress inference engine that represents the stress level of the user correspondent to a subtask. $S_{B(M)}$is generated based on the correspondent mouse behaviour.
$S_{B(M,K)}$	Stress measurement based on $B(M)$and $B(K)$	This is the projected value by the stress inference engine that represents the stress level of the user correspondent to a subtask. $S_{B(M, K)}$ is generated based on the unified mouse and keystroke behaviours. $S_{B(M,K)}$ is null for the menu search task as it does not require keyboard input.

Table 3.12: Task Performance Behaviour, *B(T)*

Property	Description	Remark
TD	Task duration	This is the total duration of a user to complete a subtask (milliseconds)
Err	Error of task	To check whether the task consists of error(s) (Err = 0 if no error; Err > 0 if the answer is wrong)
PA	Passive attempt	To determine whether the user has any attempt to give up the task (Give Up button is pressed) or wait until the time is up. PA = 999 if attempt to give up; PA = 1 if attempt to wait until the timer ends

3.7.3 KEYSTROKE BEHAVIOUR MODEL

Table 3.13 shows the key features of keyboard dynamics. It should be noted that they are crucial features, as they will greatly vary from person to person.

Table 3.13: Keystroke Behaviour, *B(K)*

Property	Description	Remark
KErr	Keystroke error rate {delete key rate, backspace key rate}	We assume that the users will use delete or backspace key to correct their mistakes. We want to determine the use frequency of these 2 keys in the duration of a task (see Algorithm 3.1).
KS	Keystroke (typing) speed	Number of keystrokes (see Algorithm 3.2). We assume that the key typing rhythms could be unusual when a user emotion is shifted. For instance, the user could probably pound on the keyboard out of frustration.
KL	Keystroke latency	The average elapsed time between two keypress (see Algorithm 3.3)

The mathematical formulation of the keystroke behaviour dataset is therefore defined as below:

$$\mathrm{B(K)} = < \mathrm{KE, KS, KL} > \tag{3.16}$$

3.7.4 MOUSE BEHAVIOUR MODEL

Table 3.14 shows the key features of mouse dynamics. Similar to keystroke dynamics, they will greatly vary from person to person.

Table 3.14: Mouse Behaviour, *B(M)*

Property	Description	Remark
MS	Movement speed	The average speed, i.e. the distance in pixels per millisecond (see Algorithm 3.4).
MIO	Inactivity/silence occurrences	Number of occurrences of inactivity between 2 events (see Algorithm 3.5).
MID	Inactivity/silence duration	The average elapsed time between 2 events, i.e. no mouse activity (see Algorithm 3.6).
MPL	Press duration (left button)	The average hold duration of a mouse button is pressed before it is released (see Algorithm 3.7 and Algorithm 3.8). We assume that the user may press the mouse button longer when deliberating a task.
MPR	Press duration (right button)	
MCL	Mouse click rate (left button)	The number of clicks of left / right mouse button (see Algorithm 3.9 and Algorithm 3.10). We assume that user may click the mouse button repeatedly out of frustration.
MCR	Mouse click rate (right button)	

The mathematical formulation of the mouse behaviour dataset is therefore defined as below:

$$B(M) = < MS, MIO, MID, MPL, MPR, MCL, MCR> \quad (3.17)$$

The features of *MPL*, *MPR*, and *MCR* are removed later due to either no or insufficient data are collected by the mouse logger.

3.8 ANALYSIS METHOD

Statistical analyses are carried out to accomplish a few aims. Firstly, they are used to explore the important factors that affect learners' stress perception and motivation on a given task. Secondly, they are important to examine the relations between the stress stimuli, stress perception, cognitive states, mouse behaviour and keystroke behaviour. Lastly, the statistical tests allow us to validate the proposed MADB that was modified to suit e-learning environment. To test the significant effects of the stimuli on learners' states, univariate analysis of variance (ANOVA) [246], multivariate analysis of variance (MANOVA) [247], [248] and linear regression [249]are used. Spearman Correlation and Pearson Correlation Tests are conducted to test the relations of different variables in the experiments. According to Gravetter and Wallnau [250], the Pearson correlation test is useful to evaluate the linear relationship between two continuous variables, i.e. ratio or interval scale of measurement. While on the other side, Spearman correlation is often used to evaluate relationships involving ordinal variables, such as the setting of task demand and external stimuli.

3.9 CONCLUSION

In an affective e-learning environment, it is important to develop a construct that can help measuring perceived mental state, such as motivation, emotional stress and cognitive load, to further adapt instruction to improve self-learning performance. The construct must be able to be quantified, computerized and automated to measure perceived mental effort. As discussed in Section 2.3, there are four concerns in building such system in the web environment: (1) the monitoring process should be continuous, (2) the method should be non-obtrusive, (3) the method should be cost-effective, and (4) the measurement of stress should be reliable, which the measurement should be context-independent, so that it can be applied regardless the type of task carried out by the user. In other words, the accuracy of the stress classification should not be affected even the student swaps between tasks, or s/he is already stressed out even before using the system. The existing research using keystroke and/or mouse dynamics-based analysis provides us a good perspective in overcoming the four concerns rose above, and this type of analysis is also believed to be more reliable than the subjective method.

CHAPTER 4: MENU SEARCH EFFECTS ON MOTIVATION / ATTITUDE-DRIVEN BEHAVIOUR (MADB) AND MOUSE DYNAMICS

Three general hypotheses to be achieved in this research were presented in Section 1.4. However, to answer these hypotheses, three different experimental studies have to be carried out based on three different tasks, namely searching, mental arithmetic and typing. The respective results of these experiments are presented in this chapter, as well as in Chapters 5 and 6. Accordingly, each chapter would discuss the specific hypotheses to be achieved in the respective task. This chapter examines the effects of menu design on learners' stress, cognitive states and mouse behaviour during the search task. Six factors of web design that could cause stress to users during search task, i.e. (1) colour, (2) font size, (3) text length, (4) menu organization, (5) term used, and (6) the need to scroll the menu, are incorporated into the menu design. The research limits each factor to two levels to prevent overly huge number of combinations, hence resulting 64 different combinations of menu design. Learners' stress perceptions of the task demands are gathered using a user self-report survey with 7-point Likert Scale. Cognitive states are measured based on the MADB model proposed by Wang [22], which formally and quantitatively defines the relationship between emotion stress, motivation, attitude, and behaviour. The adaptation of the MADB model was discussed in Section 3.2.

There are three specific hypotheses to be answered in this chapter, which are derived from Section 1.4, to validate the proposed MADB model applied in e-learning environment:

Hypothesis 1: Indirect instruction, i.e. search requirement, and external stimuli, i.e. menu design, affects learner's stress perception and motivation.

Hypothesis 2: The correlations between indirect instruction, external stimuli, stress perception, motivation, rational motivation, attitude, decision, and behaviour are significant.

Hypothesis 3: Behaviour affects mouse behaviour *B(M)*

This experimental study only focuses on examining the effect of indirect instruction, i.e. the search requirement, but no direct instruction is given to the participants. Besides, the search task does not involve keystroke data collection, since they are expected to search the desired learning materials from a designated menu using only mouse device.

Section 4.1 presents the results of the hypotheses testing. Section 4.2 provides the discussion of the results, and lastly conclusion is given to conclude the hypotheses.

4.1 RESULTS

4.1.1 SAMPLES

Initially there were 190 participants who voluntarily participated in the experiments, which require them to search 64 different materials on a designated menu with different designs. Unfortunately,39 of them did not complete the experiments since they were given option to give up at any time. Therefore 9,664 valid responses from151 participants were achieved at last for the following statistical analyses. Each session of the search task took about 30 minutes for a participant. Majority of the 151 participants were male (90.07%), aged 20-29 years old (95.36%), had more than 2 years of experience in the Blackboard LMS (84.11%). In term of frequency of use, 99.34% used the LMS at least once in a year. The detailed demographic distributions are shown in Table 4.1.

Table 4.1: Demographic Background

Factor	Value	Frequency	Percentage
Gender	Female	15	9.93%
	Male	136	90.07%
Age	below 20	7	4.64%
	20-29	144	95.36%
Experience	never	1	0.66%
	below 1 year	7	4.64%
	1-2 years	16	10.60%
	above 2 years	127	84.11%
Frequency	never	1	0.66%
	1 or 2 times in a year	9	5.96%
	less than 4 times in one semester	36	23.84%
	at least 1 time each week in a semester	40	26.49%
	more than 10 times in one semester	65	43.05%

4.1.2 THE EFFECTS OF INDIRECT INSTRUCTION AND MENU DESIGN FACTORS ON USER'S STRESS PERCEPTION (*SP*) AND MOTIVATION (*M*)

The 64 search instructions given to the participants provide no significant impact to both students' stress perception *SP* and motivation *M*. The means of the participants' responses of their *SP* on the design factors are shown in Table 4.2. Based on the survey, which was explained in Section 3.3.1, generally the students agree that they feel uncomfortable if they need to take a longer time duration to search for a feature in the website ($\mu = 5.90$). They feel comfortable with the menu design if it is equipped with good colour, big font size, text with code, longer text length, clear term, categorized organization, and without the need to scroll down the menu. On the flip side, they feel uncomfortable if the menu design contains bad colour, smaller font size, text without code, abbreviated term, ambiguous term, random display of features and the need to scroll down the menu. Interestingly, when we test the effects of the six factors, i.e. Colour, Font, Text, Organization, Term and Scroll on their *SP*, the individual factor does not provide main effect on *SP* unless it interacts with other factors (see Table 4.3). From the analysis, only the interactions between (1) Term and Scroll, (2) Colour, Text, Term and Organization, and (3) Colour, Font, Text, Term and Organization, are significant to provide effects on *SP*. This suggests that the design factors are not significant when they are tested individually, but the interactions between these factors intensify the impact on users' emotion. When we test the effects of the six factors on motivation *M*, only Organization and Scroll appear to be the main effects that affect *M*, and the interactions between (1) Term and Scroll, and (2) Colour, Text, Term and Organization, are also significant.

Table 4.2: The Means of the Learners' Perceptions of Menu Design

Question	Mean, μ	Question	Mean, μ
1: feel stressed if need to take longer time to search	5.90	2h: long text	4.97
2a: good colour	5.71	2i: clear term	5.11
2b: bad colour	**1.67**	2j: ambiguous term	**3.06**
2c: big font	5.29	2k: categorized display	5.52
2d: small font	**3.25**	2l: random display	**2.34**
2e: text with code	5.26	2m: no scrolling is needed	5.09
2f: text without code	**3.09**	2n: scrolling is needed	**3.43**
2g: abbreviated term/short text	**3.09**		

The scale is 1 (strongly disagree or uncomfortable) to 7 (strongly agree or comfortable)

Table 4.3: The Effects of Instruction and Menu Design on *SP* and *M*

Factor		*p(SP)*	*p(M)*
Instruction	Question	.4743	.4317
External Stimuli	Colour	.3842	.8981
	Font	.6759	.7316
	Text	.4227	.3083
	Term	.9508	.3925
	Org.	.4307	**.0138**
	Scroll	.8748	**.0345**
Interaction	Term * Scroll	**.0031**	**.0051**
	Colour * Text * Term * Org.	**.0053**	**.0049**
	Colour * Font * Text * Term * Org	**.0299**	.1136

Effect is significant at p<0.05 (2-tailed) level. Other interactions between factors are not significant.

4.1.3 CORRELATIONS BETWEEN INSTRUCTION, MENU DESIGN AND COGNITIVE STATES

Spearman Correlation tests are done to test the correlations of instruction, i.e. search requirement, and external stimuli, i.e. menu design, to stress perception *SP* and other cognitive states. Pearson Correlation tests are then used to test the relationship between *SP* and cognitive states. The detailed results are given in Table 4.4. Although the instructions do not give significant impact on students' stress perception *SP* and motivation *M*, it has significant relations to *M*, Decision *D* and Behaviour *B* according to Spearman correlation tests. In terms of the external stimuli, i.e. menu design, all individual factors do not correlate to *SP*. Only the Organization factor has a negative correlation to *M*, i.e. when the menu is designed with randomized organization, the motivation to continue the task becomes lower. It is also interesting to note that when the factors are turned to bad setting ($x = 1$), the students' behaviour becomes significantly lower. Therefore, poor menu design may affect students' actions to continue next task.

Table 4.4: Correlations of Question and Menu Design to Stress Perception and Cognitive States

Factor			*SP*	*M*	*A*	*Mr*	*D*	*B*
Instruction	**Question**	*r*	.0028	-.0326	.0424	.0266	-.1127	-.0839
		p	*.7816*	**.0013**	**3x10^-5^**	**.0089**	**1x10^-28^**	**1x10^-18^**
External Stimuli	**Colour**	*r*	-.0101	.0029	-.0263	-.0136	-.0172	-.0216
		p	*.3208*	*.7734*	**.0097**	*.1826*	*.0903*	**.0339**
	Font	*r*	.0057	-.0048	-.0351	-.0206	-.0154	-.0295
		p	*.5768*	*.6394*	**.0006**	**.0427**	*.1313*	**.0038**
	Text	*r*	.0123	-.0127	-.0558	-.0414	-.0300	-.0375
		p	*.2277*	*.2105*	**4x10^-8^**	**5x10^-5^**	**.0032**	**.0002**
	Term	*r*	-.0012	-.0080	-.0402	-.0258	-.0619	-.0647
		p	*.9085*	*.4293*	**8x10^-5^**	**.0112**	**1x10^-9^**	**2x10^-10^**
	Org.	*r*	.0124	-.0207	.0454	.0055	-.0625	-.0435
		p	*.2221*	**.0423**	**8x10^-6^**	*.5880*	**8x10^-10^**	**2x10^-5^**
	Scroll	*r*	-.0051	-.0075	.0455	.0188	-.0939	-.0654
		p	*.6135*	*.4620*	**8x10^-6^**	*.0646*	**2x10^-20^**	**2x10^-10^**
Affect	**SP**	*r*	-	-.7316	-.0091	-.2941	-.0019	-.0355
		p	-	**0**	*.3737*	**4x10^-192^**	*.8541*	**.0005**
Cognitive States	**M**	*r*	-.7316	-	-.0173	.3770	.1944	.2104
		p	**0**	-	*.0897*	**0**	**7x10^-83^**	**4x10^-97^**
	A	*r*	-.0091	-.0173	-	.9193	.0105	.2681
		p	*.3737*	*.0897*	-	**0**	*.3022*	**1x10^-158^**
	Mr	*r*	-.2941	.3770	.9193	-	.0864	.3313
		p	**4x10^-192^**	**0**	**0**	-	**2x10^-17^**	**3x10^-246^**
	D	*r*	-.0019	.1944	.0105	.0864	-	.9430
		p	*.8541*	**7x10^-83^**	*.3022*	**2x10^-17^**	**-**	**0**
	B	*r*	-.0355	.2104	.2681	.3313	.9430	-
		p	**.0005**	**4x10^-97^**	**1x10^-158^**	**3x10^-246^**	**0**	-

Significant correlation exists between two features at p < 0.05 (2-tailed) level if it is bolded.
Highlighted cell indicates negative correlation coefficient, r.

Pearson correlation coefficient tests show that *M* is significantly and inversely related to *SP*. When *SP* increased, *M* decreased. *M* has no correlation with *A*, indicating that *M* may not affect *A*. A linear regression is conducted to verify the effect of *M* on *A*. The regression test shows no significant impact of *M* on *A* ($p = 0.0897$). This indicates that the motivation in search task does not affect the attention, i.e. the need to revisit the same instruction. Both *M* and *A* has significant correlations to *Mr*. This is congruous with the fourth assumption made in Section 3.2. Behaviour *B* is correlated to *SP* ($r = -0.036$, $p=0.0005$), *M* ($r= 0.210$, $p = 0.04e^{-95}$), *A* ($r=0.268$, $p= 0.01e^{-156}$), M_r ($r=0.331$, $p=0.03e^{-244}$) and *D* ($r=0.943$, $p=0$). To confirm the effect of M_r and *D* on *B*, we ran linear regression tests and the results show significant impacts of both M_r and *D* on *B* ($p = 0.03e^{-244}$ and $p = 0$ respectively). This result is congruous with the fifth assumption made for the MADB model as stated in Section 3.2. Interestingly we also found significant correlations between M_r and D. The relations between M_r and D may indicate that rational motivation may have an effect on decision as well. A regression test conducted later has validated the effect of M_r on D ($p = 0.02e^{-15}$). To verify the effects of B on SP and M in the proposed MADB model, the regression tests also have validated the effects ($p=0.0005$ for SP and $p = 0.04e^{-95}$for M). To validate the last assumption made for the MADB, i.e. the correlations of B to mouse behaviour $B(M)$, the next section discusses the results.

4.1.4 CORRELATIONS BETWEEN BEHAVIOUR AND MOUSE BEHAVIOUR

We would like to examine whether the changes of behaviour in cognition function, B, would affect the user's mouse behaviour, $B(M)$. To understand how B affects $B(M)$, a multivariate analysis of variance (MANOVA) test is conducted. The result shows that the effect of B on $B(M)$ is significant (see Table 4.5). Wilks' lambda (λ) test is then used to consider differences over all the characteristic roots. The smaller the value of Wilks' lambda, the greater the implied significance [249], while high values indicate that the effects are very small and could be ignored [251]. The result in Table 4.5 indicates that the effect size of 0.6066 is considered significant, and should not be ignored since the value is moderate.

Table 4.5: The Multivariate Tests of Behaviour on Mouse Behaviour *B(M)*

Effect	Dependent Variable	Sig. *p*-value	Wilks' Lambda value, *λ*
B(M)	MS	**.0003**	0.6066
	MID	**$2{\times}10^{-22}$**	
	MIO	**.0051**	
	MCL	**.0001**	

Effect is significant at p<0.05 (2-tailed) level.

Pearson Correlation tests are then conducted to examine the relation of B to $B(M)$. The detailed correlation coefficient test result is shown in Table 4.6. Significant relations between B and all the mouse features, except *MIO*, can be observed. When B increases, *MS* would decrease ($p=0.05e^{-22}$), *MID* increases ($p=0.03e^{-21}$) and *MCL* is reduced ($p=0.01e^{-6}$). This indicates that when the behaviour is improved in a search task, the mouse action will become slower in general.

Table 4.6: Correlation Coefficients among MADB, Stress Perception and Mouse Behaviour

		B	MS	MID	MIO	MCL
B	*p*		✓	✓		✓
MS	*r*	-.1025				
	p	**$5\text{x}10^{-24}$**			✓	✓
MID	*r*	.1009	.0024			
	p	**$3\text{x}10^{-23}$**	*.8144*		✓	
MIO	*r*	-.0088	-.1573	.5497		
	p	*.3865*	**$2\text{x}10^{-54}$**	0		✓
MCL	*r*	-.0580	.2063	-.0033	-.1035	
	p	**$1\text{x}10^{-8}$**	**$2\text{x}10^{-93}$**	*.7452*	**$2\text{x}10^{-24}$**	

Significant correlation exists between two features at p < 0.05 (2-tailed) level if it is ticked (✓).
Highlighted cell indicates negative correlation coefficient, r.

4.2 DISCUSSION

Experiments and statistical analyses were conducted to answer the hypotheses, namely (1) indirect instruction and external stimuli have significant effects on stress perception *SP* and motivation *M*, (2) indirect instruction and external stimuli are correlated to *SP* and cognitive states (*M*, attitude *A*, rational motivation *Mr*, decision *D*, and behaviour *B*), and (3) behaviour *B* are correlated to mouse behaviour *B(M)*. The results are critically discussed in the following sub-sections. The outcome of the experiments also validates the revised MADB model. The following subsections also provide more detailed discussions on the three hypotheses.

4.2.1 THE EFFECTS OF INDIRECT INSTRUCTION AND MENU DESIGN FACTORS TO USER'S STRESS PERCEPTION AND MOTIVATION

The results in this paper suggest that menu design can be a stimulus that gives impact to users' stress perception and motivation, but not the search instruction. The survey respondents agreed that longer task completion time could increase their stress perception. According to the participants, pleasant experience (feeling comfortable) is caused by good menu design, if it is equipped with good colour, big font size, code, longer text length, clear term and categorization, and without the need to scroll down the menu. They feel uncomfortable if the menu design contains bad colour, smaller font size, text without code, abbreviated term, ambiguous term, random display of features and the need to scroll down the menu. Interestingly, when we test the effects of the six factors, i.e. colour, font size, text length, feature organization, term used and the need to scroll, on their stress perceptions of the search tasks, the individual factor does not provide significant effect, but the effect is raised when the design factors interact with each other. When we test the effects of the six factors on motivation, only feature organization and the need to scroll appear to be the main effects. The effects on motivation are also significant when some factors interact with each other. There is no effect of instruction on stress perception and motivation. The results could be affected by the length of the experiments as there are considerable amount of questions to be answered. The participants may feel bored and tired toward the end of the experiments, and hence the stress perceived by them could be affected by this uncontrolled factor rather than the search instruction.

4.2.2 THE CORRELATIONS OF INDIRECT INSTRUCTION AND EXTERNAL STIMULI TO LEARNER'S STRESS PERCEPTION AND COGNITIVE STATES

Instruction does not correlate to learners' stress perception *SP*, but it is significantly related to motivation *M*. All the design factors have no significant relationship with *SP*. Only organization is correlated to *M*. Although no correlations of instruction and menu design are found, nevertheless significant effects of stimuli on both *SP* and *M* are sufficient to validate the first and second assumption in the proposed MADB model (see Section 3.2), which stimuli would affect *M* and *SP*. Significant negative correlation between *SP* and *M* indicates that motivation can be weaken by high *SP*. Motivation *M* and Attitude (or attention) *A* are found significant correlated to rational motivation M_r. These signification correlations validate the third assumption in the MADB model. Decision *D*, which is affected by time duration and errors of the task, has significant correlation to Behaviour *B*, indicating that when a decision to continue the task is made, user's behaviour will improve. This validates the fourth assumption of the MADB model. The fifth assumption is that the combination of *Mr* and *D* will affect *B*. Significant correlations and effects of M_r, *D* to *B* are congruous with the fifth assumption. The sixth assumption states that the outcome of *B* affects *M* and *SP* for carrying out next task. Significant correlations and regression tests of *B* to *M* and *SP* show consistent results to validate the sixth assumption. As behaviour produces the outcome (action) of the task, this verifies that the outcome affects the motivation and stress perception in the model. It is also interesting to discover that rational motivation M_r does not only correlate to decision *D*, but it also significantly affects *D*. Besides, *M* does not affect *A* in the search task, therefore the motivational state of the student may not affect the attention he or she pays on the task, particularly the need to revisit the question during a search task.

4.2.3 THE EFFECT AND CORRELATION OF BEHAVIOUR *B* TO MOUSE BEHAVIOUR *B(M)*

We examine the relationship between behaviour and user's mouse behaviour, to identify the potential of recruiting mouse dynamics analysis in the development of an automated stress measurement model in the future. It is observed that behaviour is significantly correlated to mouse dynamics, such as mouse speed, mouse idle duration and mouse left click rate, but not the mouse idle occurrences. Greater behaviour value leads to slower mouse movements, such as low mouse speed, longer mouse idle duration and lesser mouse click. The effects of behaviour on mouse behaviour are significant and could not be ignored.

4.2.4 THE VALIDATION OF MOTIVATION/ATTITUDE-DRIVEN BEHAVIOUR (MADB) MODEL

In literature, the MADB model was tested in a software engineering organization by Wang [22], but we adapted his model to suit the context of e-learning. Accordingly, validations must be carried out by adequate experiments. The effects of the behaviour on mouse dynamics are examined. A case study of search task effects has been carried out with the assistance of 151 students from a higher education institution in Malaysia. The statistical tests suggest a few important discoveries to validate the MADB model. From the empirical analyses, we found that the results are consistent with the MADB model formalized by Wang [22]. The seven assumptions made in Section 3.2 are confirmed as follows.

1. Menu design can be considered as an external stimulus, which significantly affects students' stress perception and motivation.
2. Motivation is significantly affected by stress perception. The strength of motivation is weaken by higher stress perception and the desire to give up the task.
3. Attitude is determined by the attention that a student can spend on one task. Attitude is low when there is a need to revisit the given instruction. Although Wang suggested that motivation can affect attitude, we found no congruent correlation between motivation and attitude from this study, particularly in search task.
4. Decisions are affected by time constraints and error rates. Estimated longer completion time and higher error rate may reduce their perceived probability of success. The combination of rational motivation and decision would affect the behaviour, which determines the action to be carried out. Rational motivation is also found significantly correlated to decision. This suggests that the motivational state may affect a learner’s decision to continue the task.
5. Correlations between the rational motivation, decision and behaviour are significant. High rational motivation and decision result in higher behaviour value. Higher behaviour value indicates stronger decision to continue to task. Therefore, the combination of rational motivation and decision provide impact to the learner’s behaviour or the outcome of his/her behaviour.
6. Behaviour or the outcome of behaviour has significant correlation with motivation and stress perception. Better behaviour leads to lower stress perception and higher motivation. Thus, we conclude that the behaviour or task outcome affects learner’s motivation and his/her stress perception.
7. Behaviour is significantly correlated to mouse dynamics such as mouse speed, mouse idle duration and mouse left click rate. Stronger behaviour strength results in slower mouse movements and lesser mouse clicks in general.

Based on the above results and discussion, the MADB model in e-learning, particularly during search task, is revised and shown in Figure 4.1.

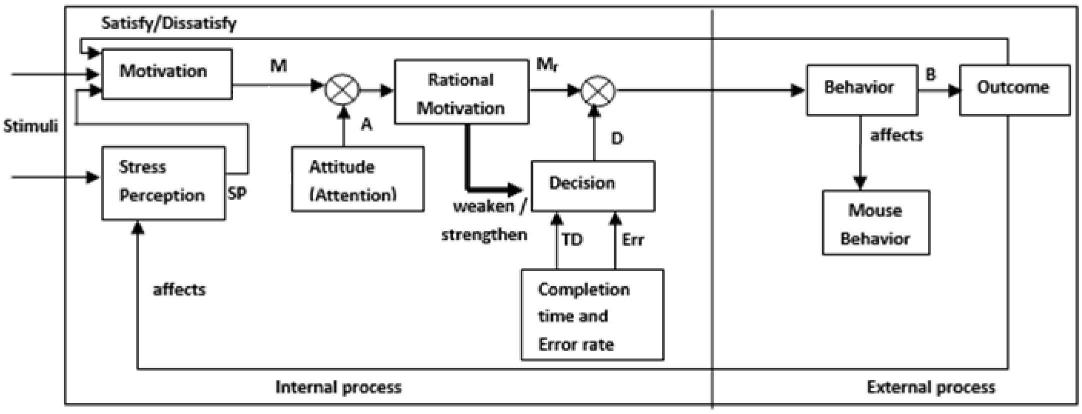


Figure 4.1. The revised MADB Model in the e-learning context with mouse behaviour during the search task

4.3 CONCLUSION

A revised version of the MADB model that was adapted based on e-learning context is proposed. In this preliminary study, the validation is done based on search task. Since the impact of student's behaviour on mouse dynamics is significant, there is a high potential and feasibility to enable automated computation of student’s stress and cognitive processes by simply observing the learner’s mouse behaviour. The next chapter will discuss the preliminary research that explores the effects of direct instruction and external stimuli such as time constraint, clock display and timer display, on learners' stress perception and cognitive states during mental arithmetic tasks.

This page is intentionally left blank.

CHAPTER 5: DIRECT INSTRUCTION AND EXTERNAL STIMULI EFFECTS ON MOTIVATION/ATTITUDE-DRIVEN BEHAVIOUR (MADB), KEYSTROKE DYNAMICS AND MOUSE DYNAMICS

Previously in Chapter 4, the effects of indirect instruction, i.e. search requirement, and external stimuli, i.e. menu design, on learners' stress perception, cognitive states and mouse dynamics were examined. Overall, congruent results with what was proposed by Wang [22] were found. This chapter continues to study the effects of direct learning instruction and external stimuli on learner's stress perception and cognitive states during an online assessment. Experiments are set to explore how formal cognitive processes are affected by mental arithmetic tasks in an e-learning system. Direct instruction refers to 10 mental arithmetic problems that the students must solve using their mental skills. External stimuli are invoked by imposing time constraint, and/or a display of clock that is updated every second, and/or a display of countdown timer that flashes every second in yellow background. Cognitive states are measured based on the MADB model adapted from what was proposed by Wang [22]. Learners' stress perceptions on the tasks are gathered using a user self-report with 7-point Likert scale. The participants are assigned to 5 different groups randomly, i.e. Group 000, Group 100, Group 101, Group 110 and Group 111. Group 000 is not given time constraint (Timing = 0, Clock = 0, Timer = 0), and the rest are given 30 seconds for each of the 10 questions (Timing = 1). Group 101 has a countdown timer display (Timer = 1), Group 110 has a clock display (Clock = 1), while Group 111 has both displayed on the screen. The detailed settings of the experiments were presented from Section 3.2 to Section 3.5. There are three specific hypotheses to be achieved in this chapter as shown below, which are derived from the three hypotheses as discussed in Section 1.4, to validate the proposed MADB model applied in e-learning as stated in Section 3.2.

Hypothesis 1: Direct instruction (Question), i.e. mental arithmetic, and external stimuli such as time constraint (Timing), clock display (Clock) and countdown timer (Timer), have significant effects on learner's stress perception and motivation

Hypothesis 2: The correlations between direct instruction, external stimuli, stress perception, motivation, rational motivation, attitude, decision, and behaviour are significant.

Hypothesis 3: Behaviour significantly affects mouse behaviour *B(M)* and keystroke behaviour *B(K)*

The following sections present the results of the hypotheses testing, followed by the discussions. Lastly a conclusion of this chapter is given.

5.1 RESULTS
5.1.1 SAMPLES

Out of the 190 students who voluntarily participated in the research studies, there were a total of 160 participants who completed the assessment task. Each session of the assessment task took about 5 minutes for each participant. Among the 160 participants, majority were male (88.75%), aged 20-29 years old (95.63%), had have more than 2 years of experience in the Blackboard e-learning system (86.25%). There were 99.37% of them who had used the system for at least once a year. The detailed demographic distributions are shown in Table 5.1. In terms of the subject groups, there were 30 participants in Group 000, 34 participants in Group 100, 31 participants for Group 101, 35 participants for Group 110, and 30 participants for Group 111.Unfortunately, there were 8 students did not complete all questions although they did most of them, of whom 5 students were from Group 000. To enable us to obtain balanced numbers of participants in each group, the missing values were imputed with mean substitution by the average of each variable (e.g. replace the missing values of Question 10 with the average values of Question 10). Finally, we achieved 1600 sample data for statistical analyses.

Table 5.1: Demographic Background

Factor	Value	Frequency	Percentage
Gender	Female	18	11.25%
	Male	142	88.75%
Age	below 20	7	4.38%
	20-29	153	95.63%
Experience	Never	1	0.63%
	below 1 year	7	4.38%
	1-2 years	14	8.75%
	above 2 years	138	86.25%
Frequency	Never	1	0.63%
	1 or 2 times in a year	12	7.50%
	less than 4 times in one semester	43	26.88%
	at least 1 time each week in a semester	40	25.00%
	more than 10 times in one semester	64	40.00%

5.1.2 THE EFFECTS OF DIRECT INSTRUCTION AND EXTERNAL STIMULI ON USER'S STRESS PERCEPTION (SP) AND MOTIVATION (M)

Based on the results of the univariate analysis of variance (ANOVA) test, as shown in Table 5.2, the effects of Question, Timing, Clock, Timer, and the interactions between Question and Timing, Question and Clock, and Clock and Timer are significant. Figure 5.1 shows that stress perception *SP* increased and motivation *M* decreased significantly when the task demand is elevated from Question 1 to Question 10 (except Question 7, which is perceived less stressful than Question 6).

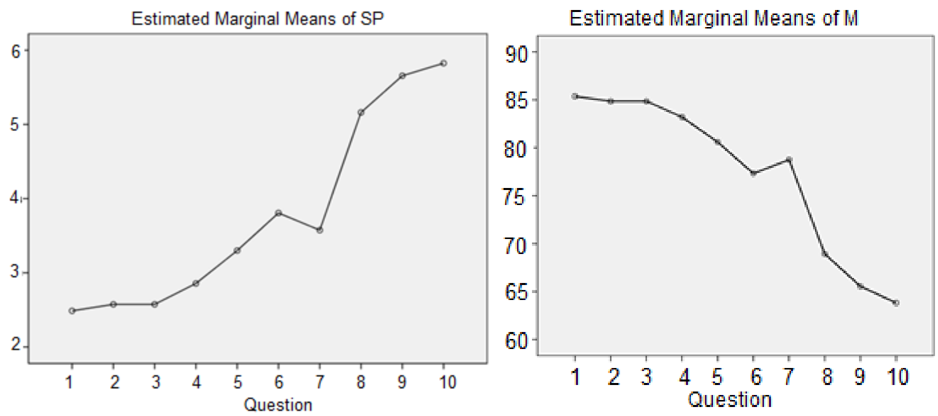
In terms of external stimuli, for the group of students who are given a time constraint (Timing = 1 for Group 100, Group 101, Group 110 and Group 111), their *SP* is generally lower, and *M* is significantly higher than those students who are not a given time constraint (cf. Figure 5.2). By comparing only those students who are given a time constraint, i.e. excluding Group 000, the students who are given a clock display generally have lower *SP* and higher *M* than those without a clock display (cf. Figure 5.3) or those with timer (cf. Figure 5.5 and Figure 5.6). For those who are given a countdown timer, their *SP* is significantly higher and *M* is lower than others (cf. Figure 5.4), and *SP* becomes worst when they are given only a countdown timer display (cf. Figure 5.5 and Figure 5.6).

Table 5.2: Test Between Question, Timing, Clock and Timer significant effects on *SP* and *M*

Factor		*Sample Size*	*N*	*p(SP)*	*p(M)*
Direct Instruction	**Question**	All	1600	**6×10^{-115}**	**2×10^{-115}**
External Stimuli	**Timing**	000,100	630	**.0042**	**.0015**
	Clock	Timing = 1	1300	**.0152**	**.0367**
	Timer	Timing = 1	1300	**1×10^{-5}**	**1×10^{-5}**
Interaction	**Question * Timing**	000,100	630	**.0360**	**.0081**
	Question * Clock	Timing = 1	1300	**.0055**	**.0037**
	Clock * Timer	Timing = 1	1300	**.0170**	**.0311**

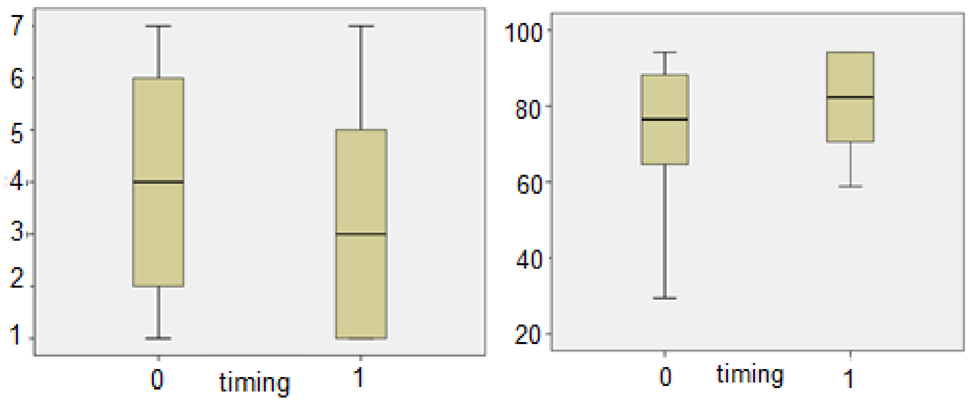
Effect is significant at p<0.05 (2-tailed) level. Other interactions between factors are not significant.

At the beginning, the students who are not given a time constraint (Group 000) have slightly lower *SP* and higher *M* than those with a time constraint (Group 100). However, from the third question onwards, their stress levels increased and motivation decreased beyond the other group (cf. Figure 5.7). Despite the effect of the interaction between Question and Timer not being significant, the students who are given a timer experience greater *SP* and lower *M* until Q7. After that, their *SP* and *M* are indifferent with those who have not given timer (cf. Figure 5.8). For the groups who are given a clock display, their *SP* is generally lower and *M* is higher than others, except when the Question demand becomes more challenging from Question 8 to Question 10 (cf. Figure 5.9).



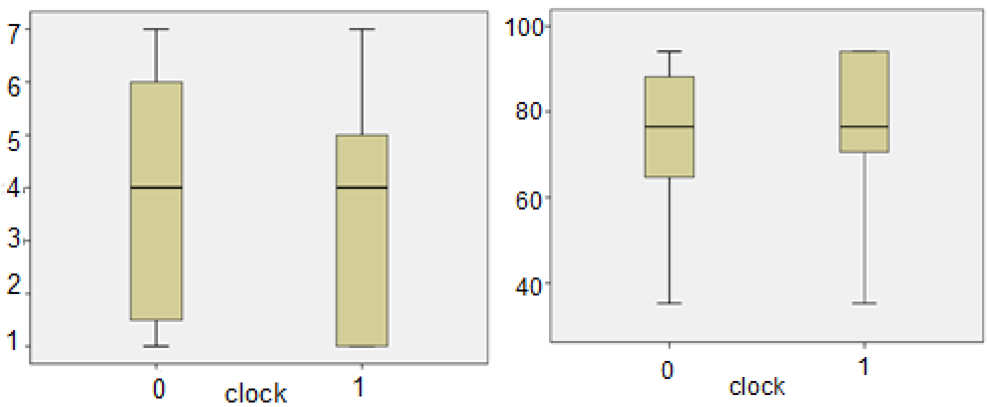
(A) (B)

Figure 5.1. Question effect on SP (A) and M (B) (sample size 1600)



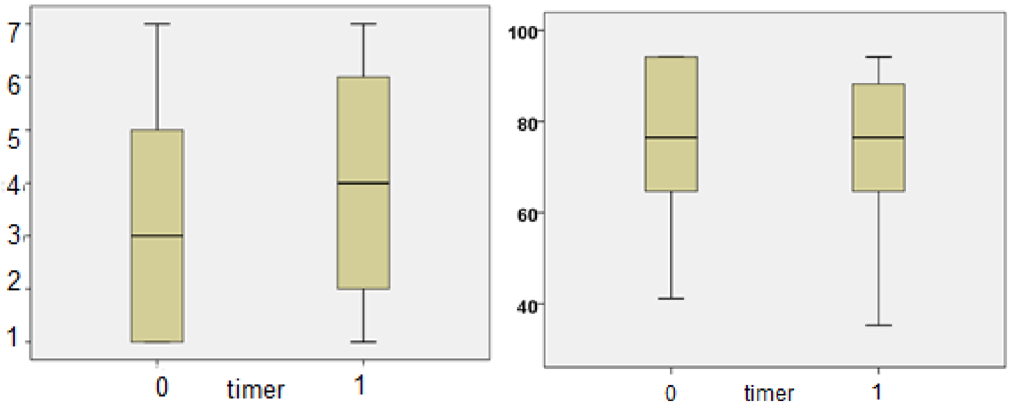
(A) (B)

Figure 5.2. Timing (time constraint) effect on SP (A) and M (B) (Group 000 vs. Group 100)



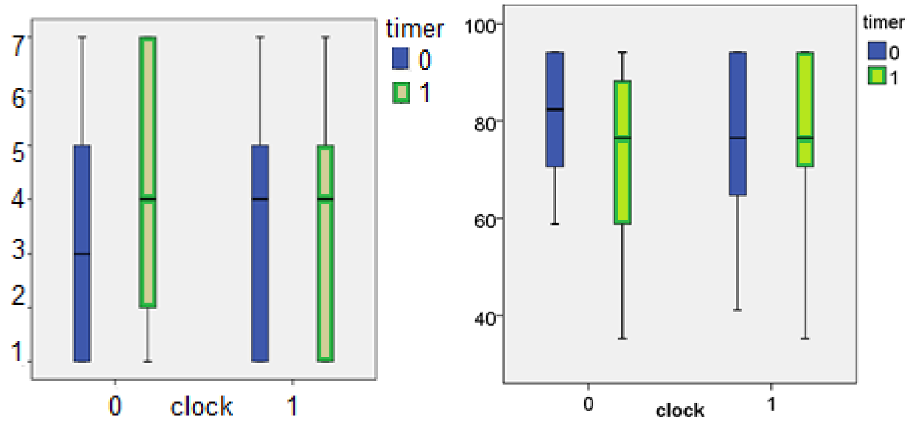
(A) (B)

Figure 5.3. Clock effect on SP (A) and M (B) (Timing = 1, sample size 1300)



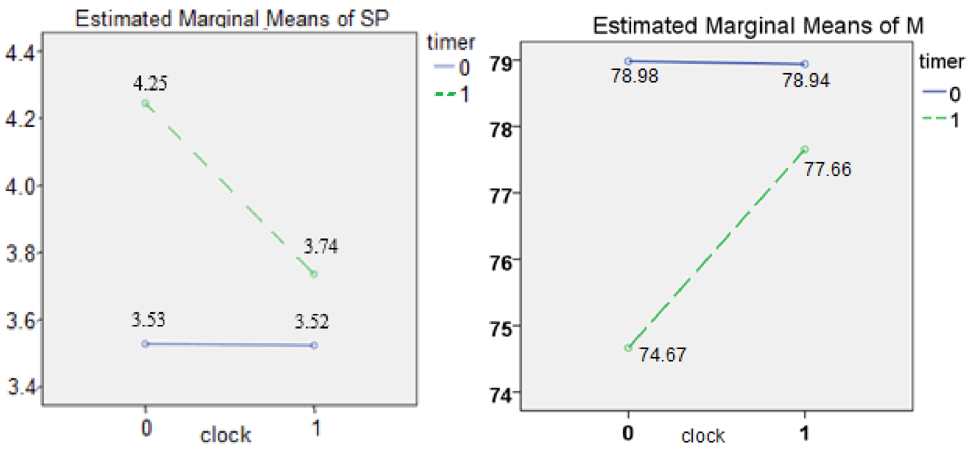
(A) (B)

Figure 5.4. Timer effect on SP (A) and M (B) (Timing = 1, sample size1300)



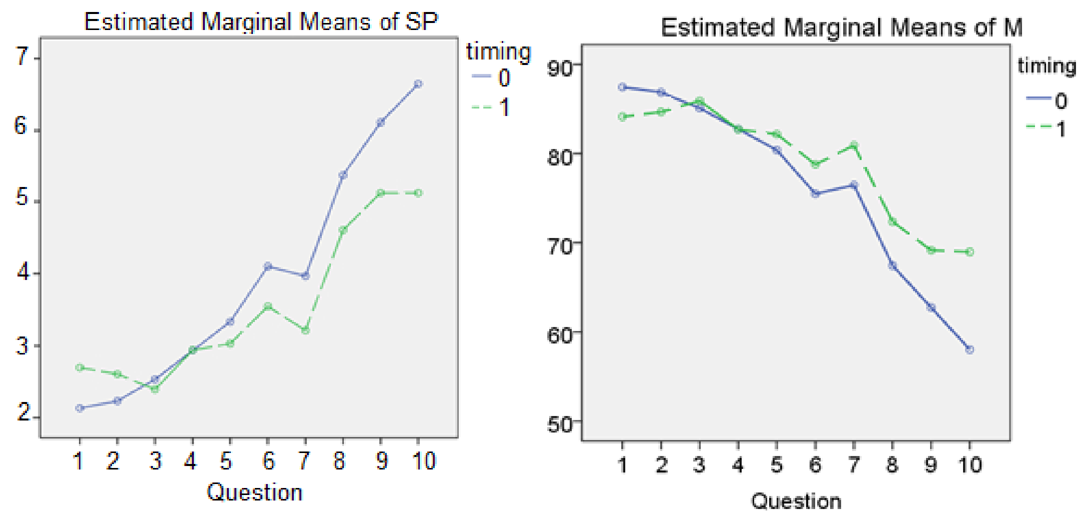
(A) (B)

Figure 5.5. Box plot of Clock and Timer effects on SP (A) and M (B) (Timing = 1, sample size 1300)



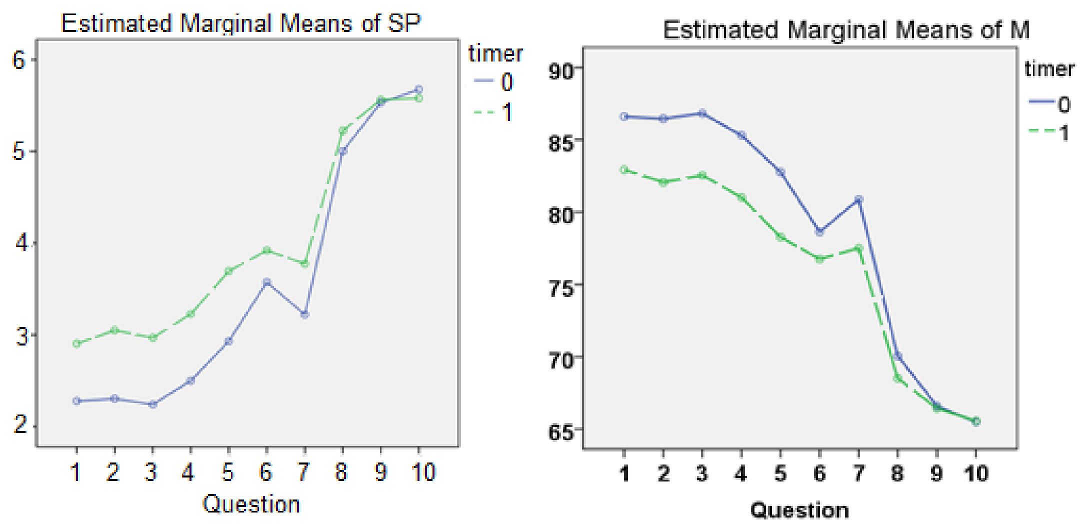
(A) (B)

Figure 5.6. Clock and Timer effects on SP (Timing = 1, sample size 1300)



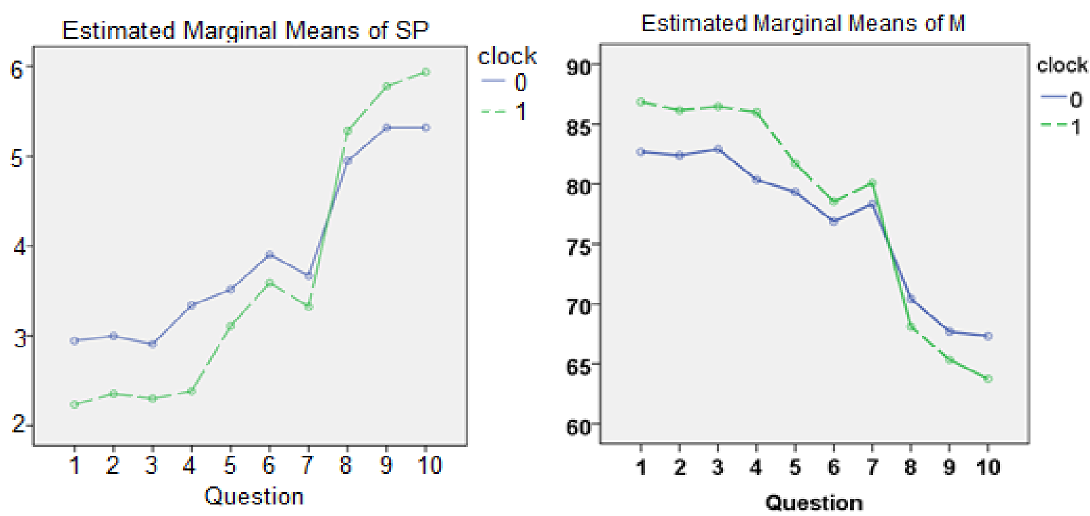
(A) (B)

Figure 5.7. Question and Timing effects on SP (A) and M (B) (Group 000 vs. Group 100)



(A) (B)

Figure 5.8. No significant interaction effects of Question and Timer on SP (A) and M (B) (Timing = 1, sample size=1300)



(A) (B)

Figure 5.9. Task Demand and Clock effect on SP (Timing = 1, sample size=1300)

5.1.3 THE CORRELATIONS BETWEEN DIRECT LEARNING, EXTERNAL STIMULI, STRESS AND COGNITIVE STATES

We performed the Spearman correlation tests to determine the correlations between direct instruction (Question), external stimuli (Timing, Clock and Timer), stress perception *SP* and motivation *M*. The significant correlations of direct instruction and external stimuli to stress perception *SP* and motivation *M* are found. As shown in Table 5.3, both direct instruction and external stimuli are correlated to motivation *M* and stress perception *SP*. When task demand increased or a countdown timer display is given, *SP* rose significantly. Interesting, when time constraint or clock display are given, *SP* becomes lower. *M* has an inverse correlation to *SP*. When *SP* increased, *M* decreased. *M* also correlates to attitude *A*. *A* is computed based on the passive attempt in the assessment task, in which the participant would wait until the time is up. The effect of *M* on *A* is significant based on a regression test ($p = 0$). Both *M* and *A* are correlated to rational motivation *Mr*. *Mr* and decision *D* are also significantly correlated to behaviour *B*. Both effects of M_r and *D* on *B* are significant according to regression tests ($p = 0.07e^{-11}$ and $p = 0.05e^{-32}$respectively). *B* significantly correlates to *M* and *SP*. The effects of *B* on *M* is significant from a regression test ($p = 0.03e^{-7}$). There is also a significant effect of *B* on *SP* ($p = 0.03e^{-7}$). When *B* improves, lower *SP* can be observed. Accordingly, *B* affects both *M* and *SP*.

Table 5.3: Correlations among Question, Timing, Clock, Timer, *SP* and *M*

Factor		Sample		SP	M	A	Mr	D	B
Instruction	Question	All	*r*	.5271	-.5386	-.1479	-.1850	-.7623	-.2221
			p	**$4x10^{-115}$**	**$5x10^{-121}$**	**$3x10^{-9}$**	**$9x10^{-14}$**	**$3x10^{-304}$**	**$3x10^{-19}$**
External Stimuli	timing	000,100	*r*	-.0934	.1017	-.1972	-.1882	.0044	-.1881
			p	**.0191**	**.0106**	**$6x10^{-7}$**	**$2x10^{-6}$**	*.9121*	**$2x10^{-6}$**
	clock	100, 101, 110, 111	*r*	-.0560	.0479	.1472	.1472	.0002	.1427
			p	**.0435**	*.0846*	**$1x10^{-7}$**	**$1x10^{-7}$**	*.9953*	**$2x10^{-7}$**
	timer	100. 101, 110, 111	*r*	.1032	-.1057	.1356	.1255	.0321	.1291
			p	**.0002**	**.0001**	**$9x10^{-7}$**	**$6x10^{-6}$**	*.2469*	**$3x10^{-6}$**
Affect	SP	All	*r*		-.9779	-.0921	-.1461	-.4881	-.1477
			p	-	**0**	**$2x10^{-4}$**	**$4x10^{-9}$**	**$2x10^{-96}$**	**$3x10^{-9}$**
Cognitive States	M	All	*r*	-.9779	-	.0880	.1446	.4809	.1443
			p	**0**	-	**$4x10^{-4}$**	**$6x10^{-9}$**	**$2x10^{-93}$**	**$7x10^{-9}$**
	A	All	*r*	-.0921	.0880	-	.9837	.1918	.9803
			p	**$2x10^{-4}$**	**$4x10^{-4}$**	-	**0**	**$1x10^{-14}$**	**0**
	Mr	All	*r*	-.1461	.1446	.9837	-	.2242	.9968
			p	**$4x10^{-9}$**	**$6x10^{-9}$**	**0**	-	**$1x10^{-19}$**	**0**
	D	All	*r*	-.4881	.4809	.1918	.2242	-	.2860
			p	**$2x10^{-96}$**	**$2x10^{-93}$**	**$1x10^{-14}$**	**$1x10^{-19}$**	-	**$2x10^{-31}$**
	B	All	*r*	-.1477	.1443	.9803	.9968	.2860	-
			p	**$3x10^{-9}$**	**$7x10^{-9}$**	**0**	**0**	**$2x10^{-31}$**	-

Significant correlation exists between two features at p < 0.05 (2-tailed) level if it is bolded. Highlighted cell indicates negative correlation coefficient, r.

5.1.4 EFFECTS AND CORRELATIONS OF BEHAVIOUR TO MOUSE BEHAVIOUR AND KEYSTROKE BEHAVIOUR

We envisage the changes in cognition function can be reflected and captured by mouse and keystroke dynamics. To understand how the changes of behaviour *B* affects keystroke behaviour *B(K)* and mouse behaviour *B(M)* as proposed in Section 3.2, the Pearson Correlation tests are conducted to observe the correlations between *B*, *B(M)* and *B(K)*. Although the error keys *KErr*, such as backspace and delete keys, were included in the experiments, the amount of the keys used by the participants was too small, which was not enough to be used in the tests. Therefore, *KErr* was excluded from the tests. The MANOVA tests show that the effects of Behaviour *B* on *B(M)* and *B(K)* are significant (see Table 5.4). Wilks' lambda (λ) considers differences over all the characteristic roots. The smaller the value of Wilks' lambda, the greater the implied significance [249]. From the results of the tests conducted based on mental arithmetic, the effects of *B* on both *B(M)* and *B(K)* are considered strong since the λ values are low.

Table 5.4: The Multivariate Tests of Behaviour on Mouse Behaviour *B(M)* and Keystroke Behaviour *B(K)*

Effect	Dependent Variable	Sig. *p*-value	Wilks' Lambda value, *λ*
Mouse Behaviour	MS	**$8x10^{-8}$**	.0136
	MID	**$2x10^{-56}$**	
	MIO	**$8x10^{-59}$**	
	MCL	**$7x10^{-8}$**	
Keystroke Behaviour	KS	**$8x10^{-05}$**	.1209
	KL	**.0047**	

Effect is significant at p<0.05 (2-tailed) level. Sample size = 1600

Table 5.5: Person Correlation Coefficients among MADB, Stress Perception, Mouse Behaviour and Keystroke Behaviour

		B	MS	MID	MIO	MCL	KS	KL
B	*p*		✓	✓	✓		✓	✓
MS	*r*	.2061						
	p	**$8x10^{-17}$**		✓	✓	✓	✓	✓
MID	*r*	-.2714	-.1795					
	p	**$2x10^{-28}$**	**$4x10^{-13}$**		✓	✓	✓	✓
MIO	*r*	.0631	-.3626	.1616				
	p	**.0116**	**$7x10^{-51}$**	**$8x10^{-11}$**		✓	✓	✓
MCL	*r*	.0333	.2885	-.2029	-.2155			
	p	*.1826*	**$5x10^{-32}$**	**$2x10^{-16}$**	**$3x10^{-18}$**		✓	✓
KS	*r*	.0514	.0926	-.2619	-.1628	.1201		
	p	**.0397**	**.0002**	**$2x10^{-26}$**	**$6x10^{-11}$**	**$1x10^{-6}$**		✓
KL	*r*	-.1234	-.0645	.2584	.0756	-.0843	-.8926	
	p	**$7x10^{-7}$**	**.0098**	**$8x10^{-26}$**	**.0025**	**.0007**	**0**	

Significant correlation exists between two features at p < 0.05 (2-tailed) level, except MCL.
Highlighted cell in grey indicates negative correlation coefficient, r. Sample size = 1600.

The results shown in Table 5.5 show that *B* is significantly correlated to *B(M)* (except *MCL*) and *B(K)*. When *B* improves, *MS* increased ($p=0.084e$-^{15}), *MIO* increased ($p=0.0116$), *KS* increased ($p=0.0397$), but *MID* decreased ($p=0.021e^{-26}$), and *KL* decreased ($p=0.074e^{-5}$), which indicate that the student's mouse and keystroke action become faster when his or her behaviour is improved, particularly during online assessment.

5.2 DISCUSSION

Experiments and statistical analyses were conducted to answer the three specific hypotheses. First, direct instruction (Question) and external stimuli (Timing, Clock and Timer) have significant effects on stress perception *SP* and motivation *M*. Second, direct instruction and external stimuli are correlated to *SP* and cognitive states, which include motivation *M*, attitude *A*, rational motivation *Mr*, decision *D*, and behaviour *B*. Third, behaviour *B* affects and are correlated to mouse behaviour *B(M)* and keystroke behaviour *B(K)*. Detailed discussions are provided in the following sections. The outcomes of the experiments also validate the consistency between the revised MADB model as proposed in the menu search task, as discussed in Chapter 4, and the online assessment task in this chapter.

5.2.1 THE EFFECTS OF DIRECT INSTRUCTION AND EXTERNAL STIMULI TO LEARNER'S STRESS PERCEPTION AND COGNITIVE STATES

Both direct instruction (Question) and external stimuli (Timing, Clock and Timer) give significant impacts on users' stress perception and motivation. As expected, demanding questions and a display of countdown timer increase stress perception and decrease motivation. However, interestingly the participants feel less stressful and more motivated when a time constraint is implemented, as well as when a clock is displayed on the screen, although they are given a time pressure. This could associate to the argument by Karasek [25], which user stress levels are varied according to two factors in a task-specific environment: demand and control. Excessive demand on production especially meeting a deadline and lack of control over the process usually generate a higher stress level. Although meeting deadline could be deemed as a high demand, we found that stress perception is correlated to the duration of task completion. Longer task completion time could increase stress perception [202], [211], [252], [253]. When people take a longer time to complete a task, they usually perceive the task as more stressful. Therefore, this could explain the reason why the students from Group 100 (Timing = 1) feel less stressed than the students from Group 000 (Timing = 0) as they spend a shorter time to complete a question. Besides, by being informed about the available resource, i.e. time constraint, as long as the students believe that they can complete the work before the deadline, the sense of control improves and hence perceived stress levels would be low.

As for the participants who are given a clock display, when compared to those without a clock display, a clock allows the user to have an ability to control his or her work. We looked at research on the influence of clocks and timers on human behaviour. Burle & Casini [254] studied how physiological arousal affects the rate of an internal pacemaker, and the way attention affects time estimation. A number of diverse observations indicate that arousal manipulations can change the rate of the pacemaker of an internal clock [255]. In short, increased attention to time, by showing users a clock or a timer, and an increase in physiological arousal, such as under time pressure, can lead to different time estimations. However, misestimate of duration in emotional situations can occur, and it is difficult to decide which mechanism, whether it is the attention raised or the induced physiological arousal, actually affects the sense and direction of time duration [256]. Compared to those who have no idea about the remaining time, the clock display may help the learners to estimate time and hence control their pace, which might help to lower their stress perception. However, for those who are given a countdown timer that flashes every second, it does not only increase the attention to time, but it might also create additional physiological arousal, i.e. stress, on top of the given time pressure.

5.2.2 THE CORRELATIONS OF DIRECT INSTRUCTION AND EXTERNAL STIMULI TO LEARNER'S STRESS PERCEPTION AND MOTIVATION

The Pearson correlation tests suggest a few important discoveries to validate the MADB model that we adapted from Wang [22]. We found some consistent results with those we have found in the menu search task. First, behaviour B is correlated to stress perception SP and motivation M. As behaviour produces the outcome or action of the task, this verifies that behaviour outcome affects motivation in the model. However, in the assessment task, there is no significant effect of B on SP, although Pearson correlations show that B is related to SP. Greater value of behaviour is correlated to lower stress perception but higher motivation. Stress perception is inversely correlated to motivation. When stress perception is higher, motivation becomes lower. Motivation and rational motivation are related to a decision, suggesting that the motivational state may affect the decision of a student to continue the task. We also found a consistent discovery, i.e. rational motivation M_r significantly affects decision D in both search task and assessment task.

Despite consistent results, we have also obtained differences between the menu search task and mental arithmetic task. Significant correlations between motivation and attitude are not found in the menu search task. However, the correlations between these two variables are found significant in the mental arithmetic task. Besides, motivation also significantly affects attitude. The difference between the two tasks is mainly due to two different methods that are used to compute the attention spent on the tasks. In the menu search task, attention was computed based on the attempt to revisit a question, while in the assessment task, the attention is computed based on the attempt to wait till the time is up. Therefore, we may assume that the motivational state of the student may affect the attention he or she pays during the assessment, i.e. attempt to wait till the time is up, rather than the attempt to revisit a question as tested in the menu search task.

5.2.3 THE EFFECTS AND CORRELATIONS OF BEHAVIOUR *B* TO MOUSE BEHAVIOUR AND KEYSTROKE BEHAVIOUR

Significant correlations between behaviour *B*, mouse behaviour *B(M)* and keystroke behaviour *B(K)* are found, except mouse click. This shows a great potential of recruiting mouse dynamics and keystroke dynamics analysis in developing an automated cognitive and affective states sensing in e-learning users. Although the correlations between *B* and *B(M)* also exist in the search task, the effect is different. Firstly, in the previous menu search task, a greater behaviour value would lead to a slower mouse movement, such as lower mouse speed, higher mouse idle duration and lesser idle occurrences. However, in the assessment task, the mouse movements become faster during mental arithmetic when the behaviour value is higher. This difference is due to two different computations being used in calculating the attitude *A*. Secondly, behaviour *B* is affected by either rational motivation *Mr* or decision *D*, and *Mr* is affected by motivation *M* and *A*. *A* is determined by the passive attempt to wait until the time is up in the assessment task, i.e. *A* is low if a passive attempt occurs. On the flip side, *A* is computed based on the attempt to revisit the question in the menu search task. Thirdly, for the assessment task, *B* improved if the students take proactive step to submit the question earlier. Improvement of *B* leads to faster mouse movements, as the students would like to submit the answer as fast as possible before the time is up. On the other hand, stronger behaviour strength results in slower mouse movements and lesser mouse clicks in the search task.

5.2.4 THE VALIDATION OF MADB MODEL

We tested the MADB model applied in the e-learning context adapted from what was proposed by Wang [22]. We found major consistency between menu search task and assessment task. The results corroborate the three specific hypotheses we made earlier, i.e. (1) direct instruction and external stimuli have significant effects on stress perception and motivation; (2) the correlations between direct instruction, external stimuli, stress perception and cognitive states are significant; and (3) the correlations between behaviour, keystroke behaviour and mouse behaviour are significant. Therefore, we confirm the seven assumptions made in Section 3.2 as follows:

1. Direct instruction (task demand) and external stimulus (time pressure, countdown timer and clock display) can significantly affect learners' stress perception and motivation.

2. Motivation correlates to stress perception. The strength of motivation *M* is reduced by higher stress perception and the desire to give up the task. Hence, motivation is weaken by higher stress perception *SP*.

3. Attitude includes user's confidence with the task based on experience, the estimated effort to complete the task, or the amount of attention can be spent on a task. In this study based on online assessment, attitude is determined by the attention that a student can spend on one task. Attitude is high when the student submits the task before the time is up. Motivation can affect a learner's attitude.

4. Decision is affected by time constraint and error rate. Estimated long completion time and high error rate may reduce their perceived probability of success. The combination of rational motivation and decision will affect the behaviour that determines the action to be carried out. We also found that rational motivation is significantly correlated to decision, which suggests that the motivational state of the student may affect his or her decision to continue the task.

5. Significant correlations are found between rational motivation, decision and behaviour. Greater rational motivation and decision result in higher behaviour value. Higher behaviour value shows stronger decision to continue to task. Therefore, the combination of rational motivation and decision affects behaviour or the outcome of behaviour.

6. Behaviour or the outcome of behaviour has significant correlation with motivation and stress perception. High behaviour leads to low stress perception and high motivation. The task outcome affects student's motivation and correlates to stress perception for carrying out next task.

7. Behaviour significantly affects mouse dynamics and keystroke dynamics. Strong behaviour strength results in higher mouse and keystroke movements in general, particularly in the assessment task.

Based on the results, the revised MADB model in e-learning context is found consistent with the proposed MADB model in Section 3.2. The proposed model for assessment task is shown in Figure 5.10.

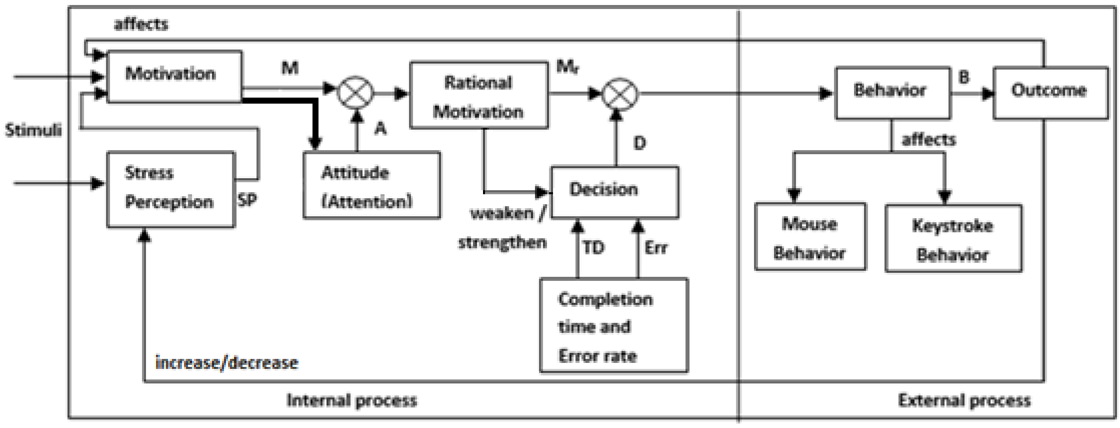


Figure 5.10. The revised MADB model in the e-learning context with mouse and keystroke behaviours during the assessment task

5.3 CONCLUSION

Based on what we have found from this research, the revised version of MADB model that is applied in the menu search task is also found consistent with the assessment task, although some minor discrepancies are found. Since the impacts of a student's behaviour on mouse dynamics and keystroke dynamics could be observed, we strongly believe that there is a potential to compute a student's cognitive processes with emotions, motivations and attitude, by observing the changes of mouse behaviour and keystroke behaviour in an online environment. The next chapter will discuss the preliminary research that explores the effects of direct instruction, text length, language familiarity, and external stimuli such as time constraint, clock display and timer display on learners' states during typing task.

This page is intentionally left blank

CHAPTER 6: TYPING DEMAND AND EXTERNAL STIMULI EFFECTS ON MOTIVATION/ATTITUDE-DRIVEN BEHAVIOUR (MADB), KEYSTROKE DYNAMICS AND MOUSE DYNAMICS

Previously, Chapter 4 and Chapter 5 studied the effects of menu search, mental arithmetic, and external stimuli such as menu design and timing factors on learners' stress perceptions and cognitive states. This chapter continues to study the effects of typing task demand and external stimuli on learner's states. Experiments are set to explore how formal cognitive processes are affected by the typing task in an e-learning system. The demand of the typing task is elevated by increasing the length of the pre-defined texts for the participants to type. To simulate the familiar task and unfamiliar task effects, English is introduced as a language that the learners are familiar with, and German language that they are totally unfamiliar with. There are a total of 6 questions (Question) with various text length (Length) and language familiarity (Familiarity) to be typed in the typing task. The detailed setting of the typing task demand was presented in Section 3.3.3. Similar to the assessment task in Chapter 5, external stimuli are invoked by imposing time constraint (Timing), and/or display of a clock (Clock) and/or a countdown timer that flashes every second (Timer). Cognitive states are measured based on the MADB model adapted from what was proposed by Wang [22].

Learners' stress perceptions on the tasks are gathered using a user self-report with 7-Likert scale. The participants are assigned to 5 different groups randomly, i.e. Group 000, Group 100, Group 101, Group 110 and Group 111. Group 000 is not given any time constraint, and the rest are given 30 seconds for each of the 6 questions. Group 101 has a countdown timer display, Group 110 has a clock display while Group 111 has both displayed on the screen. Three specific hypotheses for this chapter are given as follows, which are derived from the three hypotheses as discussed in Section 1.4, to validate the proposed MADB model applied in e-learning as stated in Section 3.2.

Hypothesis 1: Typing task demand that includes text length and language familiarity, and external stimuli, i.e. time constraint, clock display and countdown timer, have significant effects on learner's stress perception and motivation

Hypothesis 2: The correlations between typing task demand, external stimuli, stress perception, motivation, rational motivation, attitude, decision, and behaviour are significant.

Hypothesis 3: Behaviour significantly affects mouse behaviour *B(M)* and keystroke behaviour *B(K)*

The following sections present the results of the hypotheses testing, followed by the discussions. Lastly a conclusion of this chapter is given.

6.1 RESULTS

6.1.1 SAMPLES

Table 6.1: Demographic Background

Factor	Value	Frequency	Percentage
Gender	Female	17	10.49%
	Male	145	89.51%
Age	below 20	9	5.56%
	20-29	153	94.44%
Experience	Never	1	0.62%
	below 1 year	7	4.32%
	1-2 years	15	9.26%
	above 2 years	139	85.80%
Frequency	1 or 2 times in a year	13	8.02%
	less than 4 times in one semester	43	26.54%
	at least 1 time each week in a semester	39	24.07%
	more than 10 times in one semester	66	40.74%

One hundred and ninety students from Bachelor Degree in Computer Science, Bachelor Degree in Information Systems, and Bachelor Degree in Information Technology were recruited on a voluntarily basis, without any incentive. Only 162 of them completed the typing task. Among these 162 participants, the majority are male (89.51%), aged 20-29 years old (94.44%), have more than 2 years of experience in the Blackboard e-learning system (85.80%), and about 40% of them use the system for more than 10 times in one term (40.74%). The detailed demographic distributions are shown in Table 6.1. There are 32 of them from Group 000, 32 from Group 100 and 101 respectively, 36 from Group 110 and 30 from Group 111.All of them passed the English test in Malaysian Certificate of Education, but none of them know German language. Based on the 162 participants who completed the typing tasks, we achieved 972 sample data (N=972) for statistical analyses. Statistical tests are conducted to perform the analysis.

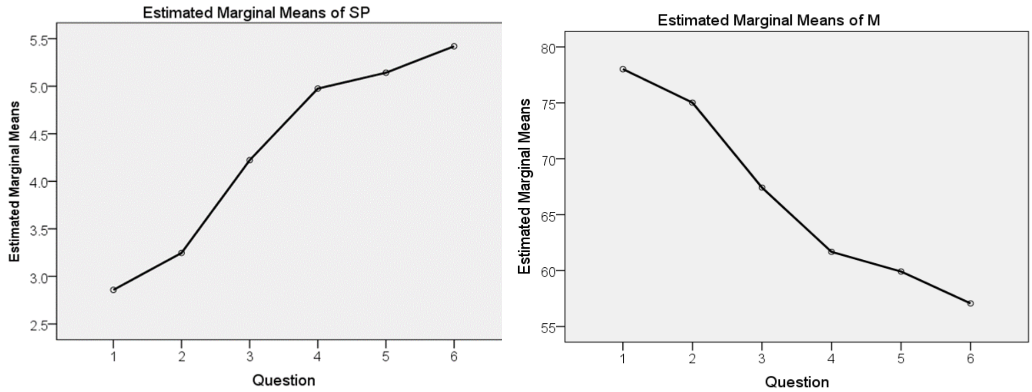
6.1.2 THE EFFECTS OF DIRECT INSTRUCTION AND EXTERNAL STIMULI ON USER'S STRESS PERCEPTION (SP) AND MOTIVATION (M)

Based on the univariate analysis of variance (ANOVA) test, the effects of Question, Length, Familiarity, and Timer are significant. From Table 6.2, there are no significant effects of time constraint and clock display on *SP* and *M* at all. The interactions between effects are also not significant. Figure 6.1 shows that *SP* increases and *M* decreases significantly when the task demand is elevated from Question 1 to Question 6. *SP* increases and *M* reduces significantly when the text length increases (see Figure 6.2). When familiar language is introduced, *SP* reduces and *M* increases significantly (see Figure 6.3). In terms of external stimuli, for those who are given a countdown timer, their *SP* is generally higher and *M* is lower than others (see Figure 6.4).

Table 6.2: Test Between Question, Timing, Clock and Timer significant effects on *SP* and *M*

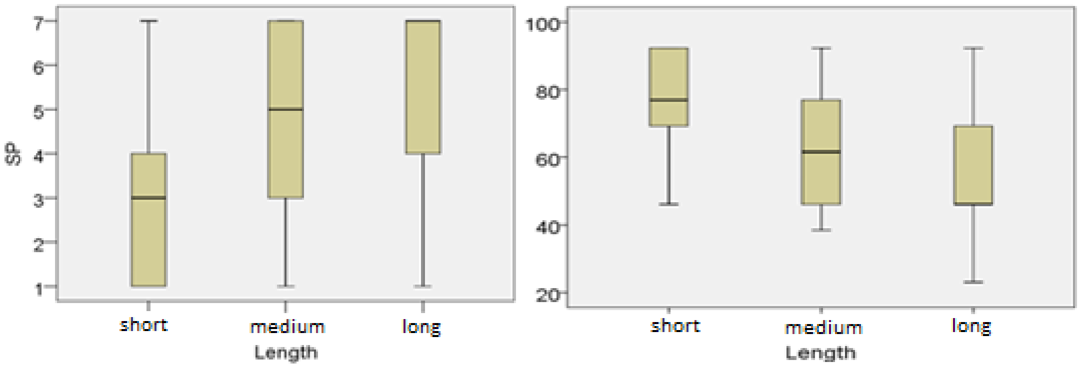
Factor		*Sample Size*	*N*	*p(SP)*	*p(M)*
Direct Instruction	Question	All	972	**$3\text{x}10^{-39}$**	**$8\text{x}10^{-41}$**
	Length	All	972	**$7\text{x}10^{-39}$**	**$2\text{x}10^{-40}$**
	Familiarity	All	972	**.0012**	**.0008**
External Stimuli	Timing	000,100	384	.6282	.9831
	Clock	Timing = 1	780	.2352	.3436
	Timer	Timing = 1	780	**.0084**	**.0070**

Correlation is significant at p<0.05 (2-tailed) level (highlight in bold). All interactions between factors are not significant.



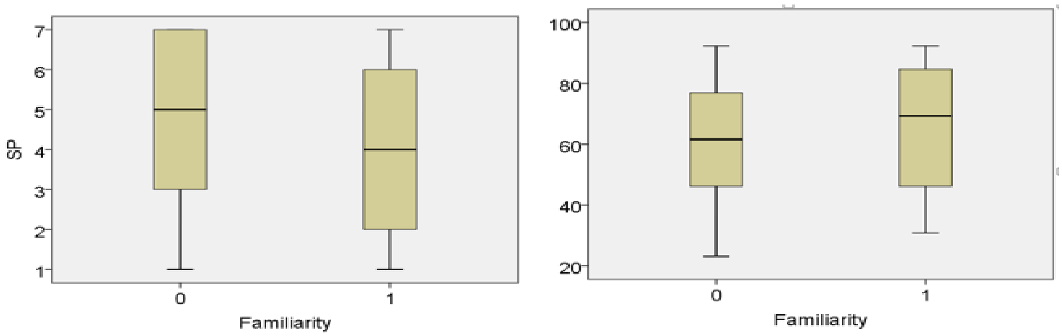
(A) (B)

Figure 6.1. Question effect on SP (A) and M (B) (sample size 972)



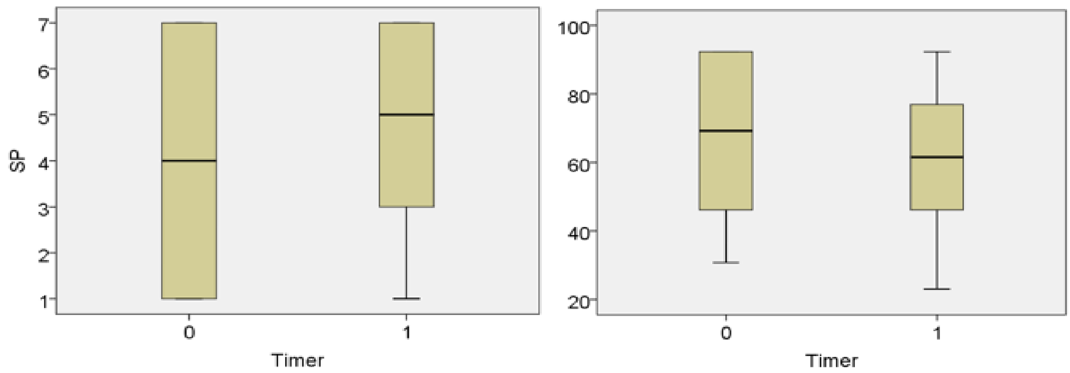
(A) (B)

Figure 6.2. Length effect on SP (A) and M (B) (sample size 972)



(A) (B)

Figure 6.3. Familiarity effect on SP (A) and M (B) (sample size 972)



(A) (B)

Figure 6.4. Timer effect on SP (A) and M (B) (Timing = 1, sample size 780)

6.1.3 THE CORRELATIONS BETWEEN TYPING DEMAND, EXTERNAL STIMULI, AND COGNITIVE STATES

The Spearman Correlation test is performed to determine the correlations between typing task demand, external stimuli, i.e. Timing, Clock and Timer, stress perception *SP* and motivation *M*. Significant correlations between the stress stimuli, *SP* and *M*, have been found. As shown in Table 6.3, when task demand (Question) or text length (Length) increases, or language familiarity (Familiarity) reduces, both *SP* and *M* decrease significantly. In terms of external stimuli, only Timer is found correlated to *SP* and *M*. When the timer is displayed, *SP* increases and *M* becomes significantly lower has an inverse correlation to *SP* ($p=0$). When *SP* increases, *M* would decrease. *M* also correlates to attitude *A*. *A* was computed based on passive attempt in the assessment task, i.e. the attempt that a participant would wait until the time is up. The effect of *M* on *A* is significant based on a regression test ($p = 0.01e^{-20}$). Both *M* and *A* are correlated to rational motivation *Mr*. *Mr* and decision *D* are also significantly correlated to behaviour *B*. Both effects of M_r and *D* on *B* are significant according to regression tests ($p = 0$ and $p = 0.02e^{-132}$ respectively). *B* significantly correlates to *M* and *SP*. The effects of *B* on *M* is also significant from a regression test ($p = 0.03e^{-293}$). There is also a significant effect of *B* on *SP* ($p = 0.09e^{-291}$), which was observed during the menu search task in Chapter 4, but not during the assessment task in Chapter 5. This indicates that *B* affects both *M* and *SP* in both menu search and typing task but not during the mental arithmetic. However, when *B* improves, lower *SP* and higher *M* can be observed in all menu search, mental arithmetic and typing tasks.

Table 6.3: Correlations among Direct Instruction, External Stimuli, Affect and Cognitive States

Factor		Sample		SP	M	A	Mr	D	B
Instruction	Question	All	*r*	.4163	-.4229	-.3142	-.4611	-.6145	-.5601
			p	**5x10^{-42}**	**2x10^{-43}**	**1x10^{-23}**	**3x10^{-52}**	**6x10^{-102}**	**2x10^{-81}**
	Length	All	*r*	.4036	-.4100	-.3105	-.4495	-.6044	-.5432
			p	**2x10^{-39}**	**1x10^{-40}**	**4x10^{-23}**	**2x10^{-49}**	**7x10^{-98}**	**1x10^{-75}**
	Familiar	All	*r*	-.1039	.1057	.0592	.1069	.1248	.1389
			p	**.0012**	**.0010**	*.0649*	**.0008**	**1x10^{-4}**	**1x10^{-5}**
External Stimuli	timing	000,100	*r*	.0275	-.0147	-.4346	-.1023	-.2266	-.1355
			p	*.5914*	*.7739*	**4 x10^{-19}**	**.0451**	**7 x10^{-6}**	**.0078**
	clock	100, 101, 110, 111	*r*	-.0499	.0414	.0132	.0377	-.0782	.0280
			p	*.1638*	*.2478*	*.7133*	*.2928*	**.0290**	*.4348*
	timer	100. 101, 110, 111	*r*	.0949	-.0966	.0600	-.0602	-.0360	-.0985
			p	**.0080**	**.0069**	*.0941*	*.0930*	*.3150*	**.0059**
Affect	SP	All	*r*	-	-.9959	-.3228	-.9565	-.3998	-.8889
			p	-	**0**	**5x10^{-25}**	**0**	**1x10^{-38}**	**0**
Cognitive States	M	All	*r*	-.9959	-	.3138	.9599	.3979	.8914
			p	**0**	-	**1x10^{-23}**	**0**	**3x10^{-38}**	**0**
	A	All	*r*	-.3228	.3138	-	.5552	.5545	.5565
			p	**5x10^{-25}**	**1x10^{-23}**	-	**1x10^{-79}**	**2x10^{-79}**	**4x10^{-80}**
	Mr	All	*r*	-.9565	.9599	.5552	-	.5016	.9405
			p	**0**	**0**	**1x10^{-79}**	-	**5x10^{-63}**	**0**
	D	All	*r*	-.3998	.3979	.5545	.5016	-	.6327
			p	**1x10^{-38}**	**3x10^{-38}**	**2x10^{-79}**	**5x10^{-63}**	-	**8x10^{-110}**
	B	All	*r*	-.8889	.8914	.5565	.9405	.6327	-
			p	**0**	**0**	**4x10^{-80}**	**0**	**8x10^{-110}**	-

Significant correlation exists between two features at p < 0.05 (2-tailed) level if it is bolded. Highlighted cell indicates negative correlation coefficient, r.

6.1.4 THE EFFECTS AND CORRELATIONS OF BEHAVIOUR TO MOUSE BEHAVIOUR AND KEYSTROKE BEHAVIOUR

To understand how the changes of behaviour *B* affects keystroke behaviour *B(K)* and mouse behaviour *B(M)*, the effects of *B* on *B(M)* and *B(K)* are examined using Multivariate Analysis of Variance test (MANOVA) [248]. Pearson Correlation test is then conducted to observe the correlations between *B*, *B(M)* and *B(K)*. We reduced the sample size and use only Question 1 to Question 4 in the tests, as Question 5 and Question 6 consist of high number of outliers for the mouse and keystroke data. The outliers are caused by the intentional insufficient time constraint given to the participants. Therefore, a sample size of 648 (N = 648) is used in this study.

The MANOVA tests in Table 6.4 show that the effects of Behaviour *B* on *B(M)* and *B(K)* are significant. Wilks' lambda (λ) considers differences over all the characteristic roots. The smaller the value of Wilks' lambda, the greater the implied significance [249]. Hence, the effect of *B* on *B(K)* is stronger than *B(M)* in the typing task. Since the causation effects of *B* on *B(M)* and *B(K)* are prominent, we study the correlations between *B* and the features of *B(M)* and *B(K)*. The result in Table 6.5 shows that *B* is significantly correlated to *B(M)* and *B(K)*. When *B* increases, *MS* also increases ($p=0.054e^{-5}$), *MIO* increases ($p=0.07e^{-20}$), *KS* increases ($p=0.0012$), but *MID*, *MCL*, *KL* and *KErr* decrease ($p=0.06e^{-19}$, $p=0.002$, $p=0.0063$, and $p=0.0061$respectively), which indicate that the student's mouse and keystroke action become faster when behaviour is improved.

Table 6.4: The Multivariate Tests of Behaviour on Mouse Behaviour *B(M)* and Keystroke Behaviour *B(K)*

Effect	Dependent Variable	Sig. *p*-value	Wilks' Lambda value, λ
Mouse Behaviour	MS	**$3x10^{-8}$**	.8221
	MID	**$7x10^{-40}$**	
	MIO	**$7x10^{-47}$**	
	MCL	**$1x10^{-5}$**	
Keystroke Behaviour	KS	**.0056**	.3474
	KL	**.0416**	
	KErr	**.0201**	

Table 6.5: Person Correlation Coefficients among MADB, Stress Perception, Mouse Behaviour and Keystroke Behaviour

Behaviour	Feature		B	MS	MID	MIO	MCL	KS	KL	KErr
	B	*p*								
Mouse Behaviour, B(M)	MS	*r*	.1955							
		p	**$5x10^{-7}$**		X	X	X	X	X	X
	MID	*r*	-.3572	-.0695						
		p	**$6x10^{-21}$**	.0770						
	MIO	*r*	.3652	.0237	-.4328					
		p	**$7x10^{-22}$**	.5463	**$6x10^{-31}$**					
	MCL	*r*	-.1205	.0188	-.1370	-.1648				
		p	**.0021**	.6332	**.0005**	**$2x10^{-5}$**			X	X
Keystroke Behaviour, B(K)	KS	*r*	.1267	.0336	-.2251	.1021	-.0840			
		p	**.0012**	.3935	**$7x10^{-9}$**	**.0093**	**.0326**			
	KL	*r*	-.1071	.0069	.3424	-.1580	.0238	-.8921		
		p	**.0063**	.8602	**$3x10^{-19}$**	**$6x10^{-5}$**	.5455	**$5x10^{-225}$**		X
	KErr	*r*	-.1077	-.0516	.1982	-.0975	-.0126	-.1352	.0319	
		p	**.0061**	.1895	**$4x10^{-7}$**	**.0130**	.7500	**.0006**	.4172	

Significant correlation exists between two features at p < 0.05 (2-tailed) level, except MS.
Highlighted cell in grey indicates negative correlation coefficient, r.

6.2 DISCUSSION

Experiments and statistical analyses were conducted to answer the hypotheses, namely (1) typing demand (Question, Length and Familiarity) and external stimuli (Timing, Clock and Timer) have significant effects on stress perception *SP* and motivation *M*; (2) typing demand and external stimuli are correlated to *SP* and cognitive states that include motivation *M*, attitude *A*, rational motivation *Mr*, decision *D*, and behaviour *B*; and (3) behaviour *B* are correlated to mouse behaviour *B(M)*and keystroke behaviour *B(K)*. The results are critically discussed in the following sections. The outcome of the experiments also validates the consistency between the revised MADB model as proposed in the menu search task in Chapter 4, the assessment task in Chapter 5, and the typing task in this chapter.

6.2.1 THE EFFECTS OF DIRECT INSTRUCTION AND EXTERNAL STIMULI ON USER’S STRESS PERCEPTION AND COGNITIVE STATES AND THEIR CORRELATIONS

Direct instruction (Question) gives significant impacts on users' stress perception and motivation. As expected, questions with a longer length and/or low (language) familiarity increase stress perception *SP* and decrease motivation *M*. Longer text length indicates that the time duration estimated to complete the task would be longer. Humans could be more stressed over the time taken to complete a task [198], [252]. The finding of familiarity effects on *SP* and *M* also corroborates the research by Tobias et al [204] and Hulme et al [205]. Tobias et al suggested that lack of familiarity implies that the required cognitive resources or response needed for executing the task may not be available in the learner's memory. A more overt response could be required for optimal learning from content with unfamiliar subjects. Hulme et al found that memory spans for unfamiliar words are lower than familiar words, which could significantly affect cognitive states. In terms of external stimuli, only timer display provides significant effects on *SP* and *M*, although time pressure (Timing) and clock display affected users’ *SP* and *M* significantly during the assessment task. This could be due to the same amount of time constraint being allocated to both assessment and typing tasks, however the estimation of time generated by individual might be different between the two tasks, due to different perception of the work amount. The participants in the assessment task may estimate a smaller amount of time to solve the mental arithmetic problems initially, but those who attempted the typing task may estimate a longer time to complete typing the sentence(s). Davidson et al [206] argued that typing speed will increase if the individual is able to allow preparation and optimization of typing movement by seeing the text far ahead. The habitual typing behaviour could be broken when stimuli such as time pressure are induced, which could increase their typing speed, but also often leads to mistakes. Davidson's claims could be observed from Table 6.3, as timing and clock are both correlated to decision *D*, which was computed based on the time duration and errors made. But this does not mean that time pressure and clock display could generate strong impact on learner's stress perception and motivation, as much as a timer can do.

6.2.2 THE CORRELATIONS OF DIRECT INSTRUCTION AND EXTERNAL STIMULI TO USER'S STRESS PERCEPTION AND MOTIVATION

Typing demand (Question) gives significant impacts on users' stress perception and motivation. As expected, questions with longer length and/or low (language) familiarity increase stress perception and decrease motivation. In terms of external stimuli, only timer display provides significant effects on *SP* and *M*, although time pressure (Timing) and clock display changed users’ *SP* and *M* during the assessment task in Chapter 5.

The Pearson correlation coefficient tests suggest a few important discoveries to confirm the MADB model. We found some consistent results with what we have found in the menu search task (Chapter 4) and assessment task (Chapter 5). First, behaviour *B* is correlated to stress perception *SP* and motivation *M*. As behaviour produces the outcome (action) of the task, this verifies that the outcome affects the motivation and stress perception in the model. A greater value of behaviour results in lower stress perception but higher motivation. Stress perception is negatively correlated to motivation. When stress perception is higher, motivation becomes lower. Motivation and rational motivation are related to decision, suggesting that the motivational state may affect the decision of a student to continue the task. We also observe significant effects of behaviour *B* on mouse behaviour *B(M)* and keystroke behaviour *B(K)* that could be caused by the motivation and decision of a student. The significance level of *B* affecting *B(K)* is greater than affecting *B(M)* in the typing task.

Despite consistent results being found, we have also obtained some discrepancies among the menu search task, assessment task and typing task. First, the correlation between motivation and attitude is not found in the menu search task, but we found significant effect of motivation on attitude in both assessment task and typing task. The reason is both assessment task and typing task consider the attempt to wait till the time is up in the computation of attitude *A*, but on the other side menu search task considers the attempt to revisit a question when calculating *A*. As a conclusion, the motivational state of the student is correlated to the attention he or she pays during the assessment or typing task, i.e. attempt to wait till the time is up, rather than the attempt to revisit a question as tested in the menu search task.

6.2.3 THE CORRELATIONS OF BEHAVIOUR *B* TO MOUSE BEHAVIOUR AND KEYSTROKE BEHAVIOUR

Behaviour *B* provides significant effects on both Mouse Behaviour *B(M)* and Keystroke Behaviour *B(K)*, but the strength of the effect is stronger on *B(K)* than *B(M)* in the typing task, which is expected as typing task involves lesser mouse activities. Significant correlations among behaviour *B*, mouse behaviour *B(M)* and keystroke behaviour *B(K)* are found, including mouse click (which is not found in the assessment task in Chapter 5). This shows a great potential for recruiting mouse dynamics and keystroke dynamics analyses in developing an automated cognitive and affective states measurement in e-learning users. Although the correlations of *B* to *B(M)* and *B(K)* also exist in the previous menu search task, the effect is different. For a greater behaviour value, instead of leading to slower mouse movements (such as lower mouse speed, higher mouse idle duration and lesser idle occurrences) as found in the menu search task, the mouse movements become faster in both assessment task and typing task. This difference is because the menu search task has a different approach in the experiment as compared to the assessment and typing tasks. There is no control or experimental groups in the menu search task as no time constraint is given to the participants. Therefore, in the menu search task, *A* is computed based on the attempt to revisit the question. Since there is no time constraint, the participants' behaviours are not affected by any timing factor.

On the other side, *A* is determined by the passive attempt to wait until the time is up (*A* is low if passive attempt occurs) in the assessment task and typing task. For both assessment and typing tasks, *B* improved if the students take proactive step to submit the question earlier. Improvement of *B* leads to faster mouse movements, as the students would like to submit the answer as fast as possible before the time is up. It is also interesting to observe that mouse speed does not play an important role in this typing task. It is not correlated to any other mouse or keystroke features (although it is correlated to *B*). We anticipated that this could happen as this task focuses on typing, but surprisingly correlations between other mouse and keystroke features could be observed. This again shows the importance of unifying both mouse and keystroke dynamics to collect user's states so that they complement each other.

6.2.4 THE VALIDATION OF MADB MODEL

We tested the MADB model applied in the e-learning context and we found major consistencies between menu search task, assessment task and typing task so far. The results corroborate the three hypotheses we made earlier, i.e. (1) typing demand and external stimuli have significant effects on stress perception and motivation; (2) the correlations between typing demand, external stimuli, stress perception and cognitive states are significant; and (3) the correlations of behaviour to keystroke behaviour and mouse behaviour are significant. Therefore, we confirm the seven assumptions made in Section 3.2 in Chapter 3:

1. Typing task demand and external stimulus (such as countdown timer) can significantly affect students' stress perception and motivation.
2. Motivation is affected by stress perception. The strength of motivation *M* is reduced by higher stress perception and the desire to give up the task. Hence, motivation is weaken by stress perception *SP*.
3. Attitude includes user's confidence with the task based on experience, the estimated effort to complete the task, or the amount of attention can be spent on a task. In our studies, attitude is determined by the attention that a student can spend on one task. Attitude is high when the student submits the task before the time is up. Motivation can affect attitude as suggested by Wang [22].
4. Decision is affected by time constraint and error rate. Projected long completion time and high error rate may reduce their estimated probability of success. The combination of rational motivation and decision will affect the behaviour that determines the action to be carried out.

We also found that rational motivation is significantly correlated to decision, which suggests that the motivational state of the student may affect his or her decision to continue the task.

5. We found significant correlations between the rational motivation, decision and behaviour. High rational motivation and decision result in higher behaviour value. High behaviour value shows a stronger decision to continue to task. Therefore, the combination of rational motivation and decision affects behaviour or the outcome of behaviour.
6. Behaviour or the outcome of behaviour has significant correlation with motivation and stress perception. High behaviour leads to low stress perception and high motivation. Thus, the task outcome affects student's motivation and stress perception for carrying out next task.
7. Behaviour is significantly correlated to mouse dynamics and keystroke dynamics. Strong behaviour strength results in higher mouse and keystroke movements in general.

Based on the results, the revised MADB model in e-learning context, particularly during typing task is found consistent with the proposed MADB model in Section 3.2. The proposed model for typing task is shown in Figure 6.5 below. The model is found generally consistent with the model proposed in search task and assessment task.

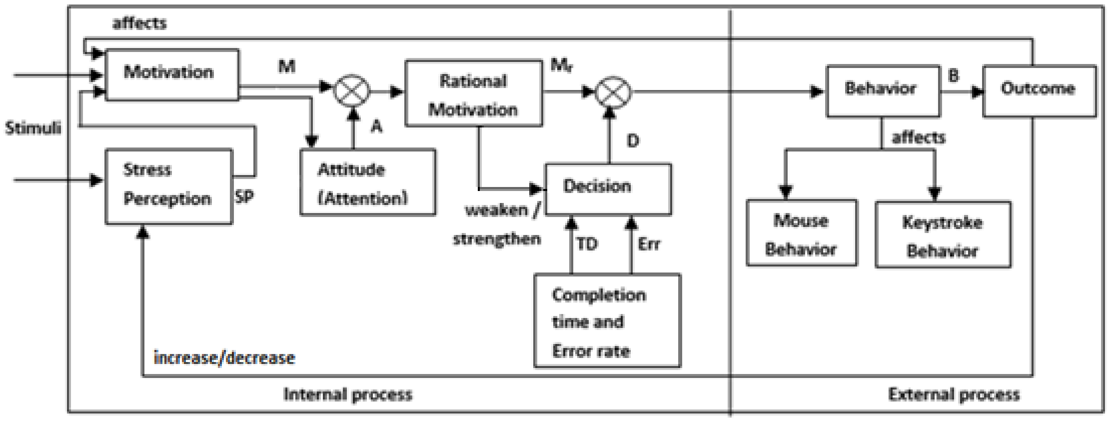


Figure 6.5. The revised MADB model in the e-learning context with mouse and keystroke behaviours during the typing task

6.3 CONCLUSION

Based on the findings from this research, the revised version of MADB model that is applied in the menu search task and assessment task is found generally consistent with the typing task, although some minor discrepancies are found. Since the impacts of student's behaviour on mouse dynamics and keystroke dynamics could be observed, we strongly believe that there is a potential to compute student's cognitive processes with emotions, motivations and attitude, by observing the changes of mouse behaviour and keystroke behaviour. Therefore, a stress measurement model based on mouse and keystroke dynamics can be built. The design and validation of the stress measurement model is explained in the next chapter.

CHAPTER 7: CONSTRUCTION OF THE STRESS CLASSIFIER

It would be desirable to have a means of assessing learner's stress levels in a task independent way through an e-learning system. It is especially important for an adaptive learning system, which is able to take into account user's cognitive and emotional states, to increase disengaged learner's motivation or to enhance personalized learning. The few signals produced by mouse dynamics and keystroke dynamics allow human-computer interaction researchers and developers to design and build a cost-effective and unobtrusive system, which can measure real world individual's affective or cognitive states. It is also important to ensure the measurement to be task-independent, so that it can be applied anywhere regardless the type of task carried out by the user. The accuracy of the stress measurement should not be affected even the student swaps between tasks, or he or she is already stressed even before using the system, which might be mishandled by the adaptive system.

The experiments reported in Chapters 4, 5 and 6 show that: (1) direct instruction, i.e. assessment and typing demand, and external stimuli, i.e. menu design, time pressure, clock and/or countdown timer displays, have significant impacts on stress perception S and motivation M; (2) stress perception S, motivation M, rational motivation M_r, attitude A, decision D and Behaviour B are significantly correlated; and (3) behaviour B significantly affects and correlates to mouse behaviour $B(M)$and keystroke behaviour $B(K)$.The findings (2) and (3) unfold the possibility to use both keystroke and mouse as sensors in gathering digital data that are useful in detecting the changes of user's cognitive, behavioural and emotional states. Hence, this gives a great motivation for us to continue designing and developing an effective, automated and objective method to measure learner's stress.

Accordingly, this chapter focuses on designing and building a stress measurement model, based on the datasets collected from the three preliminary research experiments that were reported in Chapter 4, 5 and 6. Section 7.1 below explains the motivation of this research. Section 7.2 presents the testing criteria that examine the best technique from the three selected classifiers used to measure stress, namely certainty factors (CF), feedforward back-propagation neural (FFBP) networks and adaptive neuro-fuzzy inference system (ANFIS). The justifications of the selected classifiers were given in Section 2.6.3. Section 7.3 explains the stages of emotion stress measurement and classifier's construction, which consist of data acquisition and feature extraction, creation of the training set and sample set containing labelled data, and the classifiers' architectures. Section 7.4 presents the results and analysis, followed by the discussion of the three classifiers' performances, in term of overall accuracy, false acceptance rate (FAR), false rejection rate (FRR) and equal error rate (ERR) of each model, in Section 7.5. Lastly Section 7.6 concludes the chapter.

7.1 INTRODUCTION

The main challenge of the implementation of mouse/keystroke-based analysis lies within the reliability of stress measurement. It is important to produce a reliable stress measurement that is generic or context-independent, which can monitor stress for any task in the same system. Three different activities in an e-learning environment were setup during the previous preliminary research experiments, such as searching for a desired learning material, assessment and typing, by introducing menu search (to typify search activity), mental arithmetic (to typify assessment) and typing pre-defined text (to typify typing activity). As search tasks only involve mouse input, keystroke dynamics analysis was excluded from the tests but it was included in the assessment and typing tasks.

To enable continuous stress monitoring in an online platform, we believe that measuring the stress state by computing the differences of task durations and mouse/keystroke behaviours between 2 tasks, or 2 time intervals, is useful. Besides, considering each user has individual differences in how they interact with interfaces using the devices when performing a task, we compare each individual's mouse and keystroke data against his/her time duration of completing a task, to get a sense of generally increasing, decreasing and stable (normal) stress levels. The measurement of stress is done based on either only mouse dynamics $S_{B(M)}$, keystroke dynamics $S_{B(K)}$ or the unification of both $S_{B(M, K)}$, which can be useful since not all the tasks require the use of both devices. For instance, while performing a typing task, data of mouse dynamics could be absent for a long time, then the measurement shall be solely based on keystroke dynamics. However, we must certain that the variability of tasks should not affect this computation, so that a universal method in measuring a learner's stress level can be created. Despite that, even if the task variability may significantly affect the computation based on mouse/keystroke behaviour, the effect would only last temporarily after the task is switched, if time interval-based computation is implemented.

To find an objective way to validate our proposed method, instead of relying on user self-report survey, or a physiological method that is usually hard to achieve a large number of participants, we compare the estimated $S_{B(M)}$, $S_{B(K)}$ and $S_{B(M, K)}$ against the stress level measured based on time duration S_{TD}. A few research reported the relationships between time pressure, stress, job performance, and decision making [257], [258], and humans are more stressed over time [198], [252]. Hence, a simple assumption is made in this research, i.e. when a task demand is elevated, the time spent on the task is expected to increase. If the increment rate of the time spent is within the anticipated range, then the behavioural outcome of the user is deemed stable (normal). However, if the task requires much more time than expected, then the task could be more challenging than what the examiner imagined. Vice versa, if the task takes significantly much shorter than expected, then the question might be either too easy, or the student may demonstrate anomalous behaviour, e.g. did not answer the question seriously.

7.2 TESTING CRITERIA

The Research Question 1, as specified in Section 1.4, is to find out how an effective construct that measures a learner's cognitive states and stress level can be developed by using mouse and keystroke dynamics. Accordingly, there are three criteria to be tested. These testing criteria are vital to find out the effectiveness of the proposed methods in order to produce the optimal stress measurement model. The testing criteria are as follows:

1. Can S_{TD} and $S_{B(Sensor)}$ be generally used for the 3 tasks, i.e. search, assessment and typing?
2. How close would the $S_{B(Sensor)}$ be with S_{TD}, using certainty factors (CF), feedforward back-propagation (FFBP) neural network and adaptive neuro-fuzzy inference system (ANFIS)?
3. How are CF, FFBP and ANFIS different in terms of stress measurement accuracy?

The first criterion is crucial as we need a stress measurement that is context-independent, so that it can be applied regardless the type of task carried out by the user. If the measurement is different from task to task, then it is probably not adequate to be used as a generalized measurement if the effect of task on stress measurement is high. To validate this method of measurement, we need to test the following hypotheses:

1.1 There is no difference in terms of S_{TD} between 3 tasks, i.e. search, assessment and typing.

1.2 There is no difference in terms of $S_{B(M)}$, $S_{B(K)}$, and $S_{B(M,K)}$ between the 3 tasks.

The second criterion is important as to allow the method to be implemented in an online environment. We may not know how long it would take a user to complete a task. If the measurement based on mouse/keystroke dynamics is close to the measurement based on the amount of time the user takes, then it is possible to enable continuous stress monitoring by merely observing mouse and keystroke dynamics. We examine the distance between $S_{B(Sensor)}$ and S_{TD} by using the following methods:

2.1 The probability that the $S_{B(Sensor)}$ of a single user will fall within the range of (S_{TD} - *0.5*, S_{TD} + *0.5*), considering the interval of S_{TD} is [-1, 1]. In other words, what is the chance if S_{TD} = 1 and $S_{B(Sensor)}$> 0.5?

2.2 The conditional probability, $P(normal(S_{TD})|normal(S_{B(Sensor)}))$, that $S_{B(Sensor)}$ of a single user falls within normal distribution of $S_{B(Sensor)}$, will also fall within the normal distribution of S_{TD}. In other words, if $S_{B(Sensor)}$ is "normal", then what is the chance that S_{TD} is also normal?

For the third criterion, the performance of the three models, i.e. CF, FFBP neural network and ANFIS, lies within the accuracy of the measurement. We measure the performance by checking the overall accuracy, false acceptance rate (FAR), false rejection rate (FRR) and equal error rate (ERR) of each model, which are defined as follows.

Accuracy	The measure of likelihood that the normal stress level (($Y_d(S_{TD})$ = 0) is measured to be normal ($Y(S_{B(Sensor)})$ = 0), and vice versa.
FAR	The measure of the likelihood that the normal stress level (($Y_d(S_{TD})$ = 0) is wrongly accepted as non-normal stress level (($Y(S_{B(Sensor)})$ = −1 or $Y(S_{B(Sensor)})$= 1)
FRR	The measure of the likelihood that the non-normal stress level ($Y(S_{TD})$ = −1 or $Y(S_{TD})$ = 1) is accepted to be normal ($Y(S_{B(Sensor)})$ = 0).
EER	A common way used in biometric research, to compare the accuracy of methods with different ROC (relative operating characteristic) curves. EER is the rate at which both FAR and FRR are equal. It is often used as an indicator to tell which method is better than others although it is not necessary that the classifier must operate based on EER. Usually the method with lowest EER is the best [259].

The desired output of S_{TD}, $Y(S_{TD})$, with the threshold of 1 standard deviation away (*stdev*) from the mean (mean(TD)) is activated by the following function

$$Y(S_{TD}) = \begin{cases} 1 \text{ if } S_{TD} > mean(S_{TD}) + stdev(S_{TD}) \text{, indicates stress increased} \\ -1 \text{ if } S_{TD} < mean(S_{TD}) - stdev(S_{TD}) \text{, indicates stress decreased} \\ 0 \text{ if otherwise, indicates stress is stable (normal)} \end{cases} \quad (7.1)$$

where $mean(S_{TD})$ = 0.0144 and $stdev(S_{TD})$ = 0.3813 based on a total of 12,144 records, which are collected during the previous preliminary research experiments.

To simplify the computation process, as shown in Figure 7.1, we assume that if the difference of the duration spent for the current question is at least one standard deviation from the mean, i.e. 68% are normal data, then the stress level has either increased or decreased, otherwise the stress level remains stable or normal.

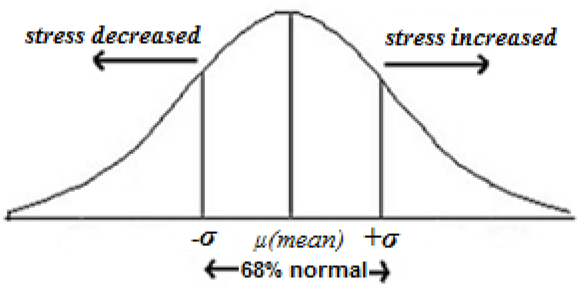


Figure 7.1. Standard deviation function of stress measurement S_{TD}

We use the threshold of one standard deviation away (*stdev*) from the mean ($mean(S_{B(Sensor)})$) to determine the actual output of $S_{B(Sensor)}$, $Y(S_{B(Sensor)})$, which is activated by the following crisp function.

$$Y\left(S_{B(Sensor)}\right) = \begin{cases} 1 \text{ if } S_{B(Sensor)} > mean(S_{B(Sensor)}) + stdev(S_{B(Sensor)}) \text{, indicates stress increases} \\ -1 \text{ if } S_{B(Sensor)} < mean(S_{B(Sensor)}) - stdev(S_{B(Sensor)}) \text{, indicates stress decreases} \\ 0 \quad \text{if otherwise, indicates stress is stable/normal} \end{cases} \quad (7.2)$$

where $mean(S_{B(M)}) = 0.0354$, $stdev(S_{B(M)}) = 0.1283$ based on a total of 12,144 records of all tasks; mean$(S_{B(K)}) = 0.0245$, $stdev(S_{B(K)}) = 0.0738$, $mean(S_{B(M, K)}) = 0.0245$, $stdev(S_{B(M, K)}) = 0.1820$ based on 2562 records of both assessment and typing tasks.

7.3 CONSTRUCTION OF THE STRESS CLASSIFIER

The following subsections explain the stages of the classifier's construction of an emotion measurement model, which consist of data acquisition and feature extraction, creation of the training and sample set containing labelled data, and lastly the construction of classifiers, namely certain factors (CF), feedforward back-propagation (FFBP) neural networks and adaptive neuro-fuzzy inference system (ANFIS).

7.3.1 DATA ACQUISITION AND FEATURE EXTRACTION

Data acquisition must be carried out automatically to collect digital samples that can objectively measure real world conditions. Feature extraction is mainly used to reduce the measurement and storage requirements, to minimize training and utilization times, so that the prediction performance can be improved. Primary data, including the raw data from mouse and keyboard and their event time, were collected by using a key logger and a mouse logger during the preliminary experiments based on the search, assessment and typing tasks (see Chapter 4 to Chapter 6). To construct the stress classifier, 2 types of input data are needed. First, time duration (*TD*) that the student spent on each question must be measured. Second, mouse/keystroke behaviours are used to measure the changes of stress when the task demand is altered. As the search task does not require keyboard input, the keystroke dynamics-based analysis is excluded from this task. Both mouse and keystroke are included for both assessment and typing tasks.

The mouse behaviour *B(M)*is defined as a dataset that captures the mouse features for each task, as follows:

B(M) = <MS, MID, MIO, MCL>, where
MS = Average mouse speed (pixels per second)
MID = Total mouse inactivity duration (ms)
MIO = Total mouse inactivity occurrences
MCL = Left click rate per ms

The keystroke behaviour *B(K)*is defined below:

B(K) = <KS, KL, KErr>, where
KS = Average keystroke Speed (number of keystrokes per second)

KL = Keystroke latency (down-down key latency)
KErr = Total delete key and backspace key pressed

Unfortunately, insufficient data of *Kerr* were collected during the assessment task, therefore *KErr* is excluded from the following experiments in this chapter. All the collected data are normalized using the log_{10} function.

7.3.2 CREATION OF THE TRAINING SET AND SAMPLE SET

As the variability of users' habits in using mouse, keyboard and the time they would spend on a question is high, therefore only the difference of a user's task duration and mouse/keyboard activities between the current question and the previous question will be considered. There are two benefits of doing this: first, it is able to reduce the variability between 2 persons; second, this also allows us to construct a personalized stress measurement, to compare whether the current task is deemed more challenging than the previous task, or whether the current stress level of the user has changed significantly compared to a moment ago. To enable stress measurement from time duration and mouse behaviour, the features are re-computed with correlation coefficient values obtained from the Pearson correlation test. Correlation coefficients are used to measure the presence of the relationship among time duration *TD*, user's stress perception of each question, and mouse behaviour and/or keystroke behaviour features, which we obtained from the experiments conducted from all three tasks with total samples of 12,144 data. These coefficients yields can be fixed as default parameters in order to build a stress measurement system. Although the parameters are fixed in this research, it is recommended for the future affective system to generate dynamic and adaptable parameters based on a personified set of rules relating stress of each person individuality, such as what has been suggested by Arevalillo-Herráez et al [260].

The stress measured based on time duration, S_{TD}, is defined as follows:

$$S_{TDk} = amp(r_{STD} * \frac{STD_k - STD_{k-1}}{STD_{k-1}}) \tag{7.3}$$

where the parameters, r_{TD} = 0.3710, k = the current question, k-1 = previous question (if k is the first question, then k-1 is the calibration), and *amp* is a function to amplify the output as the signal is too weak, so that the S_{TD} values would be in the range of [-1, 1]. The *amp* function is needed because after the data transformation of *TD* using the log_{10} function, the difference of *TD* between 2 questions is very small. Small difference of *TD* would result in a huge difference between SP_{TD} and $SP_{B(Sensor)}$, and hence affect the results. Accordingly, *amp* is set to10 in this case study.

The stress measurement values based on the changes of mouse and keystroke features, between 2 questions are as follows:

$$S_{featurek} = r_{feature} * \frac{feature_k - feature_{k-1}}{feature_{k-1}} \tag{7.4}$$

where *feature* consists of *MS*, *MID*, *MIO*, *MCL*, *KS* and *KL*. The parameters of each feature are $r_{MS} = -0.1503$; $r_{MID} = 0.3278$; $r_{MIO} = -0.0279$; $r_{MCL} = -0.0474$, $r_{KS} = -0.1111$; and $r_{KL} = 0.0919$ respectively. Similar to r_{TD}, these parameters are the correlation coefficients obtained from the Pearson correlation test against user self-evaluated stress perception. All the *S* values must be in the range of [-1, 1] to ease the classifier learning process later.

Table 7.1 shows the number of training sets and sample sets prepared for each task.

Table 7.1: Distribution of Training Sets and Sample Sets for the Three Tasks

<table>
<tr><th>TASK</th><th>Number of participants</th><th>Number of records</th><th colspan="2">Training set</th><th colspan="2">Sample set</th></tr>
<tr><td></td><td></td><td></td><td>Positive/ normal (0)</td><td>Negative/ anomalous (1 & -1)</td><td>Positive/ normal (0)</td><td>Negative/ anomalous (1 & -1)</td></tr>
<tr><td rowspan="2">SEARCH (64 questions)</td><td rowspan="2">151</td><td rowspan="2">9,582</td><td>4136</td><td>1764</td><td>2580</td><td>1102</td></tr>
<tr><td colspan="2">5,900</td><td colspan="2">3,682</td></tr>
<tr><td rowspan="2">ASSESSMENT (10 questions)</td><td rowspan="2">159</td><td rowspan="2">1,590</td><td>827</td><td>133</td><td>549</td><td>81</td></tr>
<tr><td colspan="2">960</td><td colspan="2">630</td></tr>
<tr><td rowspan="2">TYPING (6 questions)</td><td rowspan="2">162</td><td rowspan="2">972</td><td>520</td><td>80</td><td>318</td><td>54</td></tr>
<tr><td colspan="2">600</td><td colspan="2">372</td></tr>
<tr><td rowspan="2">TOTAL</td><td rowspan="2">171</td><td rowspan="2">12,144</td><td>5483</td><td>1977</td><td>3447</td><td>1237</td></tr>
<tr><td colspan="2">7,460</td><td colspan="2">4,684</td></tr>
</table>

7.3.3 THE CONSTRUCTION OF THE STRESS CLASSIFIER

Stress, is a kind of affective state that is hard to express and quantify clearly, which is vague, and lacking a fixed, precise definition. Furthermore, the mouse and keystroke features of a subject taken from different instances of the same level of stress could have wide variations. The stress perception variations between individuals when facing the same challenge is also one of the main sources of uncertainty in the stress measurement problem. The other concern we have is to find a cost-effective method to allow stress to be measured continuously over an online environment. Therefore, the classifier's learning algorithm should be less complicated so that the processing time of stress measurement could be done almost instantly without causing delay to both sides of client and server. Three different approaches that can be useful in managing uncertainties and easily implemented in an online environment are certainty factors (CF), feedforward back-propagation neural network (FFBP) and adaptive neuro-fuzzy inference system (ANFIS). The measured stress level is to be grouped into 3 classes based on mouse/keystroke behaviour: stress increased (*Stress* = 1), stress decreased (*Stress* = −1) or remains stable/normal (*Stress* = 0). The CF model and the architectures of FFBP and ANFIS for stress measurement are explained in the following sub-sections.

7.3.3.1 CERTAIN FACTORS

Each premise in the inference rule is correspondent to $S_{feature}$ (Equation 7.4), the output of the rule is a certainty factor (*CF*) in the range of -1 and 1, represents a measure of belief (stress increased if $CF > 0$) or disbelief (stress decreased if $CF < 0$). The computation of the measured stress level is similar to MYCIN [213], but we have made some slight adjustments. The certainty factors of each rule are obtained using the correlation coefficients between two variables.

Rule 1: If MS decreased, then S increased

$$\mathrm{CF}(S_{B(M)})_{\mathrm{k}} = \mathrm{r}_{\mathrm{MS}} * \frac{\mathrm{MS}_{\mathrm{k}} - \mathrm{MS}_{\mathrm{k-1}}}{\mathrm{MS}_{\mathrm{k-1}}} \tag{7.5}$$

Rule 2: If MID increased, then S increased

$$\mathrm{CF}(\mathrm{S}_{\mathrm{B(M)}})_{\mathrm{k}} = \mathrm{r}_{\mathrm{MID}} * \frac{\mathrm{MID}_{\mathrm{k}} - \mathrm{MID}_{\mathrm{k-1}}}{\mathrm{MID}_{\mathrm{k-1}}} \tag{7.6}$$

Rule 3: If MIO decreased, then S increased

$$\mathrm{CF}(\mathrm{S}_{\mathrm{B(M)}})_{\mathrm{k}} = \mathrm{r}_{\mathrm{MIO}} * \frac{\mathrm{MIO}_{\mathrm{k}} - \mathrm{MIO}_{\mathrm{k-1}}}{\mathrm{MIO}_{\mathrm{k-1}}} \tag{7.7}$$

Rule 4: If MCL decreased, then S increased

$$\mathrm{CF}(\mathrm{S}_{\mathrm{B(M)}})_{\mathrm{K}} = \mathrm{r}_{\mathrm{MCL}} * \frac{\mathrm{MCL}_{\mathrm{k}} - \mathrm{MCL}_{\mathrm{k-1}}}{\mathrm{MCL}_{\mathrm{k-1}}} \tag{7.8}$$

Rule 5: If KS decreased, then S increased

$$\mathrm{CF}(S_{B(K)})_{\mathrm{k}} = \mathrm{r}_{\mathrm{KS}} * \frac{\mathrm{KS}_{\mathrm{k}} - \mathrm{KS}_{\mathrm{k-1}}}{\mathrm{KS}_{\mathrm{k-1}}} \tag{7.9}$$

Rule 6: If KL increased, then S increased

$$\mathrm{CF}(\mathrm{S}_{\mathrm{B(K)}})_{\mathrm{k}} = \mathrm{r}_{\mathrm{KL}} * \frac{\mathrm{KL}_{\mathrm{k}} - \mathrm{KL}_{\mathrm{k-1}}}{\mathrm{KL}_{\mathrm{k-1}}} \tag{7.10}$$

The values of r_{MS}, r_{MID}, r_{MIO}, $r_{MCL,}$ r_{KS}, and r_{KL} are given in Equation 7.4.

The cumulative value of the certainty of the hypothesis, CF($S_{B(M)}$), in each rule is updated by the combination formula given in Equation 7.11 below.

$$\mathrm{CF}(\mathrm{R1},\mathrm{R2}) = \begin{cases} \mathrm{CF}(\mathrm{R1}) + \mathrm{CF}(\mathrm{R2}) - \mathrm{CF}(\mathrm{R1}) \times \mathrm{CF}(\mathrm{R2}) \text{ if } \mathrm{CF}(\mathrm{R1}) > 0 \; and CF(R2) > 0 \\ \mathrm{CF}(\mathrm{R1}) + \mathrm{CF}(\mathrm{R2}) + \mathrm{CF}(\mathrm{R1}) \times \mathrm{CF}(\mathrm{R2}) \text{ if } \mathrm{CF}(\mathrm{R1}) < 0 \; and CF(R2) < 0 \\ \frac{\mathrm{CF}(\mathrm{R1}) + \mathrm{CF}(\mathrm{R2})}{1 - \min(|\mathrm{CF}(\mathrm{R1})|, |\mathrm{CF}(\mathrm{R2})|)} \text{ if otherwise} \end{cases} \tag{7.11}$$

7.3.3.2 FEEDFORWARD BACK-PROPAGATION NEURAL NETWORK

Supervised learning is utilized to predict the outcomes of stress based on 3 different training sets, i.e. mouse features, keystroke features, and the combination of all features. Accordingly, three neural networks are formed using the back-propagation training. The first neural network is used to predict the stress based on the changes of mouse features $S_{B(M)}$, the second network is used to predict the stress based on the changes of keystroke behaviour $S_{B(K)}$, and the last network is to predict stress based on the changes of all features $S_{B(M, K)}$. The numbers of hidden neurons of the networks are correspondent to the numbers of inputs. The four inputs for the first neural network are S_{MS}, S_{MID}, S_{MIO} and S_{MCL}. The second network consists of only 2 inputs, i.e. S_{KS} and S_{KL}. The last network consists of all 6 inputs. All inputs are defined in Equation 7.4. There is only one hidden layer for each network. The distribution of training sets and sample sets are described in Table 7.1. The output target for both networks is the desired output of $Y(S_{TD})$ (-1, 0 or 1) as computed in Equation 7.1. Since the inputs and the measurement of stress are in the interval of [-1, 1], the *tansig* function is used as the transfer function from the input layer to output layer, which will also return an output, *Y*, in [-1, 1] (stress increased if $Y > 0$ or stress decreased if $Y < 0$). The algorithm of *tansig* function [261] is a follows:

$$\text{tansig}(n) = 2/(1+\exp(-2*n))-1 \tag{7.12}$$

After the training, to incorporate the classifier as the inference engine in the stress monitoring system, only the feedforward phase of the training algorithm need to be applied. The application procedure is as shown in Algorithm 7.1.

ALGORITHM 7.1. APPLICATION PROCEDURE OF FEEDFORWARD ANN [217]

Initialize trained weights, v_{ij} and w_{jk}

for each input vector, **x**, do

for i=1 till n: set activation of input unit x_i // x is the input

for j=1 till p

$z\_in_j = v_{0j} + \sum_{i=1}^{n} x_i v_{ij}$ // the net input to the hidden unit j (Z_j);

$z_j = \text{tansig}(z\_in_j)$ // the output signal of Z_j

for k = 1 till m

$y\_in_k = w_{0k} + \sum_{j=1}^{p} z_j w_{jk}$ //y_in_k is the net input to output unit k

$y_k = \text{tansig}(y\_in_k)$ //y_k is the output signal of output unit k

where x = input; v_{0j}=bias on hidden unit j; w_{0k}=bias on output unit k

7.3.3.3 ADAPTIVE NEURO-FUZZY INFERENCE SYSTEM

To test the effectiveness of using adaptive neuro-fuzzy inference system (ANFIS) to measure stress, MATLAB [262] is used in this study. To simplify the explanation on how it works, we illustrate the first fuzzy inference system (FIS) in Figure 7.2, which is used to predict the stress based on the changes of keystroke behaviour. The other FISs are used to predict the stress based on the changes of mouse behaviour *B(M)* that contains 4 inputs, and the unification of both behaviours, *B(M, K)* that contains 6 input features.

First we hypothesize a parameterized model structure of the first FIS as below:

RULE 1: If x_1 is A_1 and x_2 is B_1 then $f_1=p_1x_1+q_1x_2+t_1$
RULE 2: If x_1 is A_2 and x_2 is B_1 then $f_2=p_2x_1+q_1x_2+t_2$
RULE 3: If x_1 is A_3 and x_2 is B_1 then $f_3=p_3x_1+q_1x_2+t_3$
RULE 4: If x_1 is A_1 and x_2 is B_2 then $f_4=p_1x_1+q_2x_2+t_4$
RULE 5: If x_1 is A_2 and x_2 is B_2 then $f_5=p_2x_1+q_2x_2+t_5$

RULE 6: If $x_1$is A_3 and $x_2$is B_2 then $f_6=p_3x_1+q_2x_2+ t_6$
RULE 7: If $x_1$is A_1 and $x_2$is B_3 then $f_7=p_1x_1+q_3x_2+ t_7$
RULE 8: If $x_1$is A_2 and $x_2$is B_3 then $f_8=p_2x_1+q_3x_2+ t_8$
RULE 9: If $x_1$is A_3 and $x_2$is B_3 then $f_9=p_3x_1+q_3x_2+ t_9$

where $\boldsymbol{x}$ = [S_{KS}, S_{KL}] (S_{KS} and S_{KL} are defined in Equation 7.4) and $\{p_i, q_i, t_i\}$ is the parameter set. Note that *f* is a linear function.

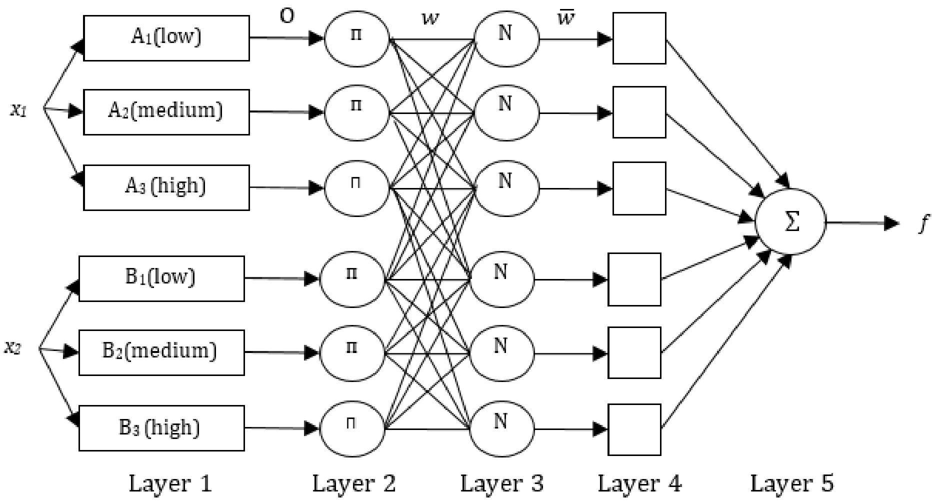


Figure 7.2. ANFIS architecture with 2 inputs

Next, we prepare input/output data into input/output vectors. Each FIS consists of 3 membership functions for all premises. The distribution of training sets and sample sets are described in Table 7.1. The input vector to be fed to the first FIS is $\boldsymbol{x}$ = [S_{KS}, S_{KL}] (produced in Equation 7.4). The input vector for the second FIS is $\boldsymbol{x}$ = [S_{MS}, S_{MID}, $S_{MIO}S_{MCL}$] (produced in Equation 7.4). The input vector for the third FIS is $\boldsymbol{x}$ = [S_{MS}, S_{MID}, S_{MIO}, S_{MCL}, S_{KS}, S_{KL}]. The target output for both networks is the $Y(S_{TD})$, where $Y(S_{TD})$ = -1,0 or 1, as computed in Equation 7.1.

Layer 1 shows three node functions, which are the membership functions (A_i) that specify the degrees to which the given *x* satisfies the quantifier A_i according to symmetric Gaussian function [263], as follows:

$$O_i^1 = \mu_{A_i}(x) = exp\left(-\frac{(x-c)^2}{2\sigma^2}\right), \text{c and } \sigma \text{ are arbitrary real constants} \quad (7.13)$$

Then in Layer 2, the production of incoming signals from Layer 1 is generated, and the output is sent to Layer 3. Since there are two inputs, Layer 1 should produceO_i^1 andO_j^2. The node function of Layer 2 will be:

$$w_{ij} = O_i^1 \times O_j^2, i = 1,2,3;\ j = 1,2,3 \quad (7.14)$$

Layer 3 calculates the ratio of the *i*th rule's firing strength, w_i, to the sum of all rules' firing strengths. The output, which is called normalized firing strengths, is as follows:

$$\bar{w}_i = \frac{w_i}{\sum_i w_i}, i = 1,2,3; n = 3 \quad (7.15)$$

In Layer 4, the subsequent parameters are produced by the following node function:

$$O_i^4 = \bar{w}_i f_i = \bar{w}_i(p_i x + r_i) \tag{7.16}$$

Consider in Layer 5, which is also the output layer, it is a single node that computes the overall output as the summation of all incoming signals from Layer 4, which is:

$$O^5 = \sum_i^n \bar{w}_i f_i \tag{7.17}$$

Thus we have demonstrated how an ANFIS is constructed. The concept to build the other FIS is similar, except that for the one based on *B(M)* has 81 fuzzy rules with 5 parameters (as there are 4 inputs with 3 correspondent membership functions). For example,

RULE 1: If x_1 is A_1 and x_2 is B_1 and x_3 is C_1 and x_4 is D_1 then $f_1=p_1x_1+q_1x_2+r_1x_3+s_1x_4+t_1$

where $\{p_1, q_1, r_1, s_1, t_1\}$ is the parameter set.

As for the FIS based on *B(M, K)*, there will be 729 rules with 7 parameters since it has 6 inputs.

7.4 RESULTS AND ANALYSIS

7.4.1 TEST 1: USING $S_{B(SENSOR)}$ AND S_{TD} TO MEASURE STRESS IN THREE DIFFERENT TASKS

Univariate analysis (ANOVA) is used to test the difference in terms of S_{TD}, and multivariate analysis (MANOVA) [246], [248] is carried out to test the difference in terms of $S_{B(M)}$, $S_{B(K)}$ and $S_{B(M,K)}$ between different tasks. As keystroke dynamics are only involved in the assessment and typing tasks, we separated the analyses into two parts. The first focuses on the effects of all 3 tasks on $S_{B(M)}$ only, while the second tests the effects of Task on $S_{B(M)}$, $S_{B(K)}$ and $S_{B(M, K)}$. Table 7.2 shows the results.

Table 7.2: Univariate and Multivariate Tests on the Effects of Tasks on S_{TD} and $S_{B(M)}$

Effect of Task on	S_{TD}	$S_{B(M)}$					$S_{B(K)}$			$S_{B(M,K)}$
		S_{MS}	S_{MID}	S_{MIO}	S_{MCL}	Effect size	S_{KS}	S_{KL}	Effect size	Effect size
	p-value	*p*-value				Wilks' λ	*p*-value		Wilks' λ	Wilks' λ
All tasks	.382	3×10^{-28}	1×10^{-31}	.193	3×10^{-20}	.971	N/A	N/A	N/A	N/A
Assessment and Typing	.456	.078	.0003	.413	.888	.994	.075	.002	.993	.986

The difference is significant at the level of $p < 0.05$ (2-tail)

The differences between tasks provide no significant effect on S_{TD} at all, but they give a significant effect on $S_{B(M)}$, $S_{B(K)}$ and $S_{B(M, K)}$. Although the effects of different tasks on these $S_{B(M)}$, $S_{B(K)}$ and $S_{B(M,K)}$ are significant, nevertheless high Wilks' lambda values (λ> 0.97) indicate that the effects are very small and could be ignored [251].

7.4.2 TEST 2: PREDICTION OF $S_{B(SENSOR)}$ AND S_{TD}, BY CF, FFBP NEURAL NETWORK AND ANFIS

Based on the sample set given in Table 7.1, the probabilities (*P*) that the $S_{B(Sensor)}$ of a single user will fall within the range of ($S_{TD} - 0.5$, $S_{TD} + 0.5$), considering the interval of S_{TD} is [−1, 1] using CF, FFBP neural net and ANFIS, are shown in Table 7.3. The probabilities of all models for all tasks fall within [0.6559, 0.9892], indicate that if $S_{TD} = 1$, then there is at least 65.59% of chance that $S_{(Sensor)}$ will fall above 0.5 if CF is used. The best result is gained from FFBP neural net, which the overall probability for all $S_{B(Sensor)}$ and all tasks is 0.9553, followed by ANFIS (0.9257), and lastly CF (0.7403).

Table 7.3: The Chance of $S_{B(M)}$ Falls within the Range of (S_{TD} - 0.5, S_{TD} + 0.5)

B	Task/Model	CF			FFBP			ANFIS		
		n	P	overall	n	P	overall	N	P	overall
B(M)	Search (N=3682)	2657	.7216	.7161	3514	.9544	.9567	3546	.9631	.9490
	Assessment (N=630)	450	.7143		599	.9508		581	.9222	
	Typing (N=372)	247	.6640		368	.9892		318	.8548	
B(K)	Assessment (N=630)	572	.9079	.9202	592	.9397	.9391	599	.9508	.9451
	Typing (N=372)	350	.9409		349	.9382		348	.9355	
B(M,K)	Assessment (N=630)	431	.6841	.6737	601	.9540	.9651	519	.8238	.7974
	Typing (N=372)	244	.6559		366	0.9339		280	0.7527	

Next, we examine the conditional probability that $Y(S_{TD})$is normal (see Equation 7.1) given that $Y(S_{B(Sensor)})$ is normal (given in Equation 7.2), i.e. $P(normal(S_{TD})|normal(S_{B(Sensor)}))$. From Table 7.4, the best result is gained from FFBP neural net, which provides the overall probability of0.9093, followed by ANFIS (*p*=0.9037), and lastly CF (*p*=0.7774).

Table 7.4: Chance that Normal S_{TD} Falls within the Normal $S_{B(Sensor)}$

B.	Task / Model	CF			FFBP			ANFIS		
		N=P(normal($S_{B(Sensor)}$))			P=P(normal(S_{TD})|normal($S_{B(Sensor)}$))					
		N	P	overall	N	P	overall	N	P	overall
B(M)	Search	3004 (.8159)	2124 (.7071)	.7388	2503 (.6798)	2278 (.9101)	.9136	2592 (.7040)	2325 (.8970)	.9054
	Assess.	484 (.7683)	424 (.8760)		472 (.7492)	436 (.9237)		531 (.8429)	499 (.9397)	
	Typing	271 (.7285)	229 (.8450)		312 (.8387)	289 (.9263)		291 (.7823)	267 (.9175)	
B(K)	Assess.	571 (.9063)	495 (.8669)	.8701	532 (.8444)	468 (.8797)	.8832	537 (.8524)	469 (.8734)	.8774
	Typing	322 (.8656)	282 (.8758)		324 (.8710)	288 (.8889)		344 (.9247)	304 (.8837)	
B(M,K)	Assess.	464 (.7365)	406 (.8750)	.8620	509 (.8079)	463 (.9096)	.9199	492 (.7810)	458 (.9309)	.9266
	Typing	275 (.7392)	231 (.8400)		278 (.7473)	261 (.9388)		271 (.7285)	249 (.9188)	

7.4.3 TEST 3: THE PERFORMANCE OF CF, FFBP AND ANFIS

Table 7.5 demonstrates the false acceptance rate (FAR), false rejection rate (FRR), the overall accuracy and the equal error rate (EER) for CF, FFBP neural net and ANFIS in the measurement of $Y(S_{B(Sensor)})$ (Equation 7.2) against $Y(S_{TD})$ (Equation 7.1). FAR indicates the chance of the expected normal stress level is incorrectly accepted as non-normal stress. On the other hand, FRR indicates the chance of the expected non-normal stress level is incorrectly accepted as normal stress. From the results, the average FAR and FRR are 19.11% and 79.63% for CF; 13.47% and 29.66% for FFBP neural net; and 12.37% and 34.44% for ANFIS. The 3 models produce an average of 67.25%, 82.88% and 83.60% overall accuracy respectively by CF, FFBP neural net and ANFIS. The average EER for each model is 54.16% by CF, 47.20% by FFBP neural net and 49.83% by ANFIS. In terms of FAR, FRR, overall accuracy and EER, FFBP neural net appears to provide the best results among all models.

Table 7.5: The Performance of CF, FFBP and ANFIS

Model	Task	B(Sensor)	FAR	FRR	Accuracy	Overall Accuracy	EER %
CF	Search	B(M)	456/2580 (.1767)	880/1102 (.7985)	2346/3682 (.6372)	.6725	49.33
	Assessment	B(M)	125/549 (.2277)	60/81 (.6074)	445/630 (.7063)		54.12
		B(K)	54/549 (.0984)	76/81(.9383)	500/630 (.7937)		46.18
		B(M,K)	143/549 (.2605)	58/81 (.7160)	429/630 (.6810)		51.61
	Typing	B(M)	89/318 (.2799)	42/54 (.7778)	241/372 (.6479)		53.58
		B(K)	36/318 (.1132)	40/54 (.7407)	296/372 (.7957)		68.54
		B(M,K)	87/318 (.2736)	44/54 (.8148)	241/372 (.6479)		55.77
FFBP	Search	B(M)	302/2580 (.1171)	225/1102 (.2042)	3155/3682 (.8569)	.8288	48.11
	Assessment	B(M)	113/549 (.2058)	36/81 (.4444)	481/630 (.7635)		29.57
		B(K)	81/549 (.1475)	64/81 (.7901)	485/630 (.7698)		48.53
		B(M,K)	86/549 (.1566)	46/81 (.5679)	498/630 (.7905)		34.41
	Typing	B(M)	29/318 (.0912)	23/54 (.4259)	320/372 (.8602)		57.16
		B(K)	30/318 (.0943)	36/54 (.6667)	306/372 (.8226)		53.58
		B(M,K)	57/318 (.1792)	17/54 (.3148)	298/372 (.8011)		59.03
ANFIS	Search	B(M)	255/2580 (.0988))	267/1102 (.2423)	3160/3682 (.8582)	.8360	49.73
	Assessment	B(M)	81/549 (.1475)	64/81 (.7901)	548/630 (.8698)		40.50
		B(K)	80/549 (.1457)	68/81 (.8395)	482/630 (.7651)		51.70
		B(M,K)	91/549 (.1658)	34/81 (.4198)	505/630 (.8016)		54.12
	Typing	B(M)	51/318 (.1604)	24/54 (.4444)	297/372 (.7984)		55.45
		B(K)	14/318 (.0440)	40/54 (.7407)	318/372 (.8548)		51.22
		B(M,K)	69/318 (.2170)	22/54 (.4074)	281/372 (.7554)		46.10

7.5 DISCUSSION

This preliminary research compares three stress classifiers, which could be effectively used in an online environment due to their simple architecture, to manage uncertainty in the collection of a learner's stress states. To enable stress measurement based on time duration and mouse/keystroke dynamics, the changes of task completion time and mouse/keystroke features of a learner between the current question and the previous question are computed, and produced with the correlation coefficients that relate users' self-evaluated stress perceptions. This method does not only eliminate the high variability of users' habits in using mouse and the time they would spend on a question, and to also allow us to construct a personalized stress measurement. Besides, it also allows us to compare whether the current job is deemed more challenging than the previous job. Most importantly it enables a mechanism to continuously monitor or measure a learner's stress level from time to time using the time-interval-based measurement. For instance, even without the knowledge of task length or task duration in a real-time environment, the learner's stress level could be measured using mouse and keystroke dynamics. Although the correlation coefficients need to be obtained from the past user's survey, nevertheless these values give significant clues about how the timing data and sensors could react to a learner's stress states. These values can be set as constants or parameters that measure the strength of the changes in timing data, as well as the sensor activities of two different tasks, for the initial rule-based stress measurement model. However, future work will identify the process to dynamically generate adaptable set of parameters for personified emotion detection.

7.5.1 THE EFFECTS OF TASKS ON S_{TD} AND $S_{B(SENSOR)}$

To explore a stress measurement method that is context-independent, so that it can be applied to various task carried out by the learners, we compared the effects of 3 different tasks, i.e. search, assessment and typing, on S_{TD} and $S_{B(Sensor)}$. If the effects of the tasks on the stress measurement are significant, this indicates that the accuracy of the measurement could be affected when the user switches between tasks. The result shows that the effect of tasks on S_{TD} is not significant at all. This gives us a very good benchmark on testing $S_{B(Sensor)}$ against S_{TD}. Unfortunately, the effect of different tasks on $S_{B(Sensor)}$ is significant for most features. This significant effect shows that the users may have demonstrated different behaviour during different tasks. In certain activity, such as mental arithmetic, the user's cognitive load is higher than other type of task, such as typing. Secondly, it could be due to typing task requiring fewer mouse/keystroke activities as compared to search. Although the effect of tasks on $S_{B(Sensor)}$ is significant, fortunately the effect size is small, which is considered meaningless and can be ignored [251]. In addition, despite the effect being significant, it would only last temporarily as after the task is switched, the stress measurement is continued by detecting the behavioural changes between 2 consecutive questions or 2 time intervals.

7.5.2 THE PREDICTION OF S_{TD}AND $S_{B(SENSOR)}$ BY CF, FFBG NEURAL NET AND ANFIS

To validate the feasibility to enable continuous stress monitoring by observing mouse/keystroke dynamics alone, we determine the chance of $S_{B(Sensor)}$ would fall close to S_{TD}. This depends on the model being used: there is 65.59 to 98.92% chance that $S_{B(Sensor)}$ of a single user would fall within the range of ($S_{TD} - 0.5$, $S_{TD} + 0.5$), considering the range of S_{TD} is [-1, 1]. Furthermore, when

FFBP neural net is used, there is an overall 90.93% chance that S_{TD} is normal if $S_{B(Sensor)}$ is normal. This indicates that there is a high probability that $S_{B(Sensor)}$ would provide a close estimation as S_{TD}. In other words, the possibility to utilize mouse dynamics alone in the stress measurement system is high if FFBP neural net is used.

7.5.3 THE PERFORMANCE OF THE STRESS CLASSIFIERS

In terms of assessing the effectiveness of the three stress classifiers in measuring stress, namely certainty factors (CF), feedforward back-propagation (FFBP) neural net and adaptive neuro-fuzzy inference system (ANFIS), we examine the classifier that produces the best false acceptance rate (FAR), false rejection rate (FRR), overall accuracy and equal error rate (EER). Although most of the time, the performances of the classifiers are mixed with both positive and negative results, we consider the overall performance is acceptable. The overall FAR is considered low (e.g. 13.47% for FFBG neural net), indicates the chance that the system mistakenly classifies a normal stress level as a non-normal stress level is low. Although the FRRs are generally high for all classifiers, we regard the outcome is still favourable, as the system would not perform any adaptation although the user is actually stressed, as excessive adaptation may annoy the learner.

Among the three classifiers, we consider the FFBP neural network produces best performances. It is easy to be applied in the stress inference system but it requires data to be trained before the application can be implemented. Besides, its overall performance for all three tasks is better than CF and ANFIS. On the other hand, ANFIS overall results are considered as good as FFBP, although its performance is slightly lower than FFBP. Unfortunately, there are two major limitations of using ANFIS. First, pre-application training is required. Second, if the number of inputs and membership functions are high, it could be programming and processing load challenging as it needs high number of rules and fuzzy sets to be built. The last classifier, CF is easy to use and its simple algorithm should not harm the processing performance of the computer. In addition, unlike FFBP or ANFIS, it does not require the data to be trained beforehand. Therefore, it is easily implemented in the web environment. However, the greatest limitation is the reliability of the stress measurement results. Amongst the 3 models, CF achieves lowest overall accuracy and EER, as well as highest FAR and FRR. However, despite of poorest performance, the overall accuracy is 60.22%, which is still considered acceptable for an emotion classifier. We should not forget the fact that the inaccurate results could be due to anomalous behaviour, in which the users might give up the task in shorter time, but the stress level is still high. Furthermore, the utilization of stress measurement based on task duration data provides only an estimation of the expected stress level, but it is not fully reliable.

To examine the best model to be used as the inference engine for the stress measurement system, we tested the accuracy of CF, FFBP neural net and ANFIS in measuring the correct hypothesis of $Y(S_{B(M)})$ against $Y(S_{TD})$. Although mostly used in biometrics research but not in emotion recognition, FAR, FRR and EER can be used as an indicator to know the performance of the stress measurement by the 3 models, instead of relying on overall accuracy itself. From the results, FFBP neural net produces best overall FAR (13.47%), FRR (29.66%), accuracy (82.88%) and EER (47.20%) compared to CF and ANFIS.

7.6 CONCLUSION

As a conclusion, the results of this research demonstrate high feasibility to use mouse and keystroke dynamics alone in stress measurement and classification. The outcome of this research also suggests that feedforward back-propagation (FFBP) neural net could be the best model to construct the stress classifier in the inference engine, followed by adaptive neuro-fuzzy inference system (ANFIS) and lastly certain factors (CF). Overall the stress measurements by CF, FFBP neural net and ANFIS are on a par with the existing research in the area of emotion measurement using keyboard and mouse dynamics [134].

The limitation of this research is it only detects stress. Detecting stress alone may not be enough for affective learning, which requires better understanding of granularity of emotion. However, it is useful to determine the stressor that causes student's unhelpful behaviour in learning. The next chapter will include both mouse and keyboard dynamics in the application of the stress measurement model that we designed in this chapter. First, a construction of automated detection of task demand in an online environment, which could be useful to determine the stressor that caused poor student’s learning behaviour, will be presented. Secondly, the design of an adaptive system that adapts learning materials, in particularly mental arithmetic, using the stress measurement model built on FFBP neural network will be given.

CHAPTER 8: THE APPLICATION OF STRESS MEASUREMENT MODEL IN AFFECTIVE LEARNING USING MOUSE AND KEYSTROKE DYNAMICS

Chapter 7 proposed a stress measurement model using mouse and keystroke dynamics to classify learners' stress levels in a web-based e-learning system. The results showed high potential to use mouse and keystroke dynamics alone in stress measurement and classification. Amongst certainty factors, feedforward back-propagation neural network and adaptive-neuro fuzzy inference system, the neural net achieved the best performance in stress classification. Accordingly, the first research question has been answered. The second research question attempts to look into how the application of the stress measurement model using mouse and keystroke dynamics can be designed and incorporated in an ITS. A prototype of such ITS would be designed and developed. Accordingly, Chapter 8 presents the design of the ITS architecture based on the groundwork conducted out earlier, but no further empirical research will be carried out to validate the effectiveness of the ITS. First, an adaptive assessment in the ITS is constructed based on the mental arithmetic task that was presented in Chapter 5. The adaptive assessment system aims to adapt assessment material when it detects a significant stress increment, or anomalous behaviour of an individual learner. Second, after the assessment is marked, collective feedback will be provided to the examiner to alert her to any mismatched expectation of the task difficulty level. The collective feedback system aims to provide a report that tabulates not only the performances such as the error rates of the tasks, but also to include the learners' stress measurements based on mouse and keystroke dynamics, which could effectively reflect the changes of their cognitive and affective states. For example, a question that produces a high error rate does not necessarily mean the question is demanding. A demanding question usually requires high cognitive load that may over stress the students. On the flip side, a question that was expected easy by the examiner may be deemed challenging for the students.

Section 8.1 presents the overall design and the architecture of the ITS with the application of the stress measurement model using mouse and keystroke dynamics. This includes the detailed designs of the stress inference engine, which is the core of the ITS, the adaptive assessment and interface, and the collective feedback reporting system. Section 8.1 ends with the comparison between the proposed collective feedback report and the report generated by the existing learning management system such as Blackboard™. Lastly, Section 8.2 presents the conclusion.

8.1 A DESIGN OF THE INTELLIGENT TUTORING SYSTEM BASED ON MOUSE AND KEYSTROKE DYNAMICS

This chapter aims to propose two possible extensions in an ITS, by tracking a learner's stress and behaviour. However, the validation of the designs outlined in the chapter is not the main concern of the research. The two main objectives of this chapter are:

1. To design an adaptive learning system that provides adaptation of learning material when user's behaviour is detected as anomalous
2. To design a collective feedback reporting system that provides an examiner with insights on students' performance and their behaviour when answering the questions

Section 8.1.1 explains the general architecture of the ITS. The subsequent sections describe the designs of the adaptive learning system and the collective feedback reporting system in detail.

8.1.1 THE ARCHITECTURE OF THE INTELLIGENT TUTORING SYSTEM

Figure 8.1 illustrates the overall architecture of the proposed ITS. The ITS is built based on model-view-controller design. The *models* consist of `QuestionBank`, `JobPerformance`, `MouseBehaviour`, `KeystrokeBehaviour` and `LearnerProfile`, which are defined in Figure 8.2 (the detailed code is provided in Figure A3.1 in Appendix III). *Controllers* are mainly constructed to work in the inference engine, and the *view* refers to the adaptive interface.

The ITS first requires the examiner to insert a number of questions with different levels of difficulties. The examiner must indicate the level of difficulty of each question. Sample interface is given in Figure 8.4. The questions are then saved in a database table called `QuestionBank`. To setup the assessment, the examiner could choose to distribute the questions randomly by the ITS, or to choose the questions manually. The examiner could also specify the distribution of questions according to the question difficulty. For instance, the examiner could specify that 30% of the questions are easy (Level 1 to Level 3), 40% are at medium difficulty (Level 4 to Level 7), and 30% are difficult questions (Level 8 to Level 10). Figure 8.5 shows the sample interface given to the examiner. Before the students start the assessment, they are required to login to the system so that the calibrations of keystroke dynamics and mouse dynamics can be collected. The reason for performing calibrations is to manage the huge temporal variations of keystroke and mouse dynamics of individual user, and also the high behavioural differences between individuals. The calibration is useful as a benchmark to determine whether the subsequent learning activities are considered significantly more stressful, stable/normal, or less stressful. Figure 8.3 shows the respective login screen for keystroke and mouse data calibration. Once the students start the assessment module, the question will be retrieved from the `QuestionBank` table automatically. The answer, error made, time spent in milliseconds, and the passive attempt (if time constraint is given) of each question that the learners provided are then formulated into the `JobPerformance` model, which is needed by the inference engine for stress measurement and adaptation. The keystroke and mouse loggers continue to collect the sensor data every 10 milliseconds. The collected data are transformed using the log_{10} function for the subsequent stress classification process.

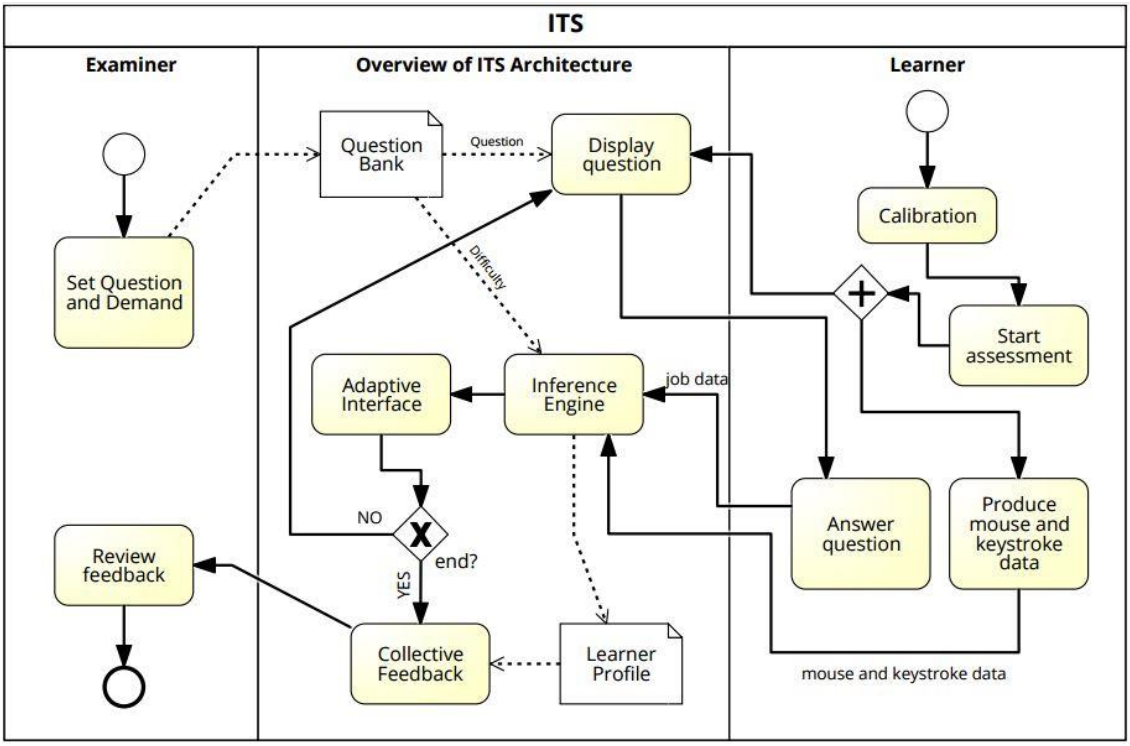


Figure 8.1. The architectural design of the Intelligent Tutoring System

The inference engine takes in-charge of the transformation of `MouseBehaviour`, `KeystrokeBehaviour`, and `JobPerformance` objects into the formation of individual `LearnerProfile`, using a trained feedforward neural network to measure stress. The feedforward neural net was trained based on 4,684 samples, and was identified as the best stress classifier as stated in Chapter 7. Once significant increment of stress level, or anomalous behaviour is found, then the instructional content of the assessment is adapted to improve learning. The adaptive system is also designed to display some words of wisdom to encourage a disengaged learner to continue the next task, whenever necessary. This hopefully could help motivating the learner when he or she is considered significantly stressful, or has demonstrated anomalous behaviour, such as attempting to give up, or not putting concentration on the task. Section 8.1.2 explains the stress inference engine in detail. More examples of the adaptive interfaces are given in Section 8.1.3. At the end of the assessment session, the collective feedback reporting system will gather and analyse all the `LearnerProfile` data and provide recommendations on the question demand for the examiner, based on the learners' job performance and stress measured. Section 8.1.4 provides the detailed design of the collective feedback reporting system.

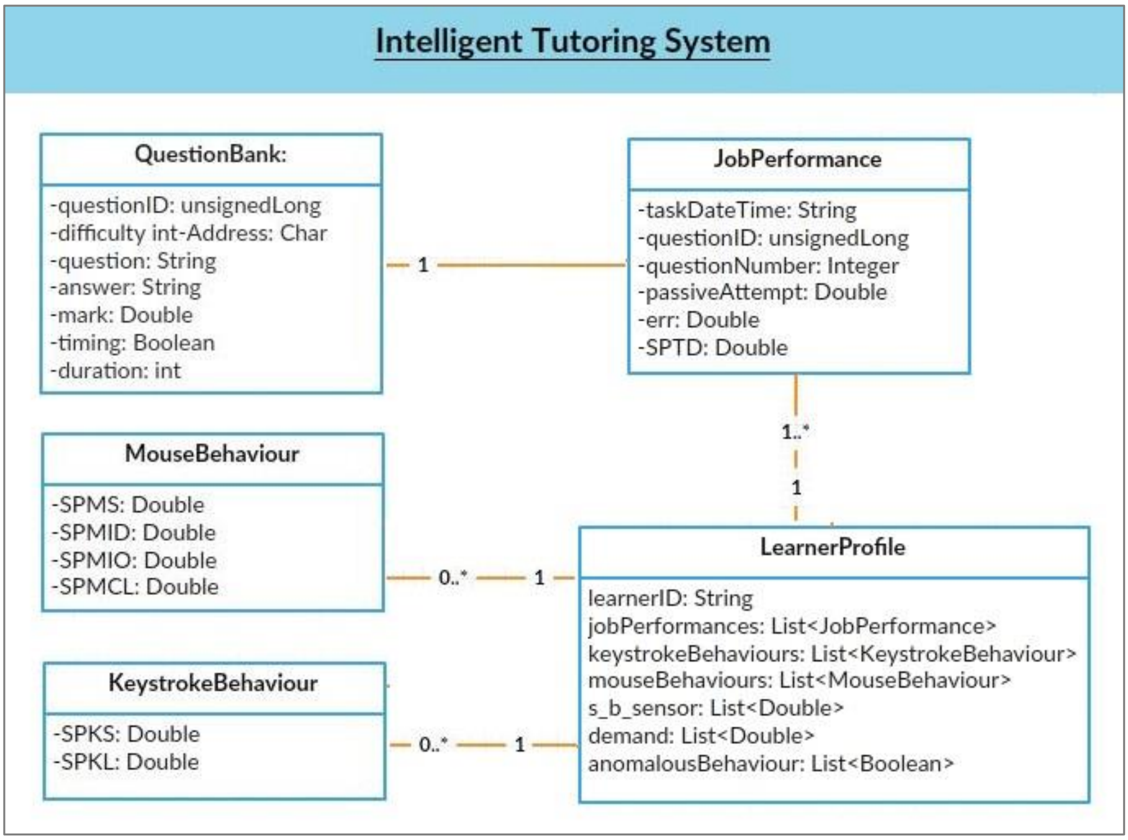


Figure 8.2. The class diagram of the models

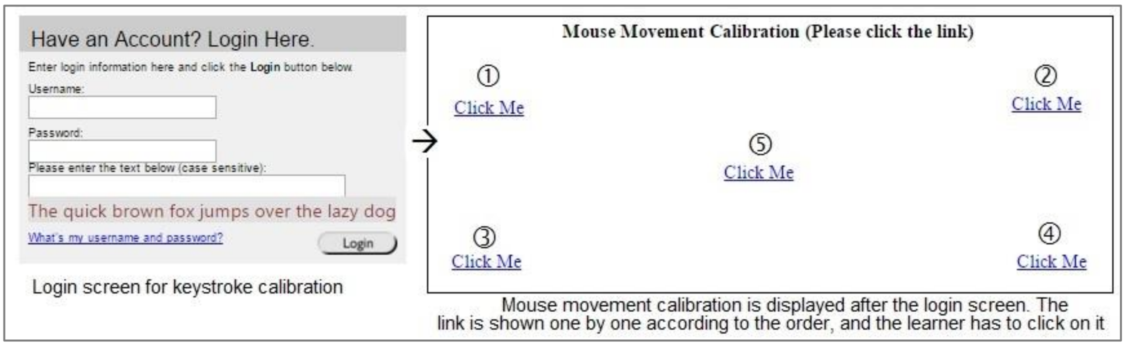


Figure 8.3. Keystroke and mouse movement calibrations required when login

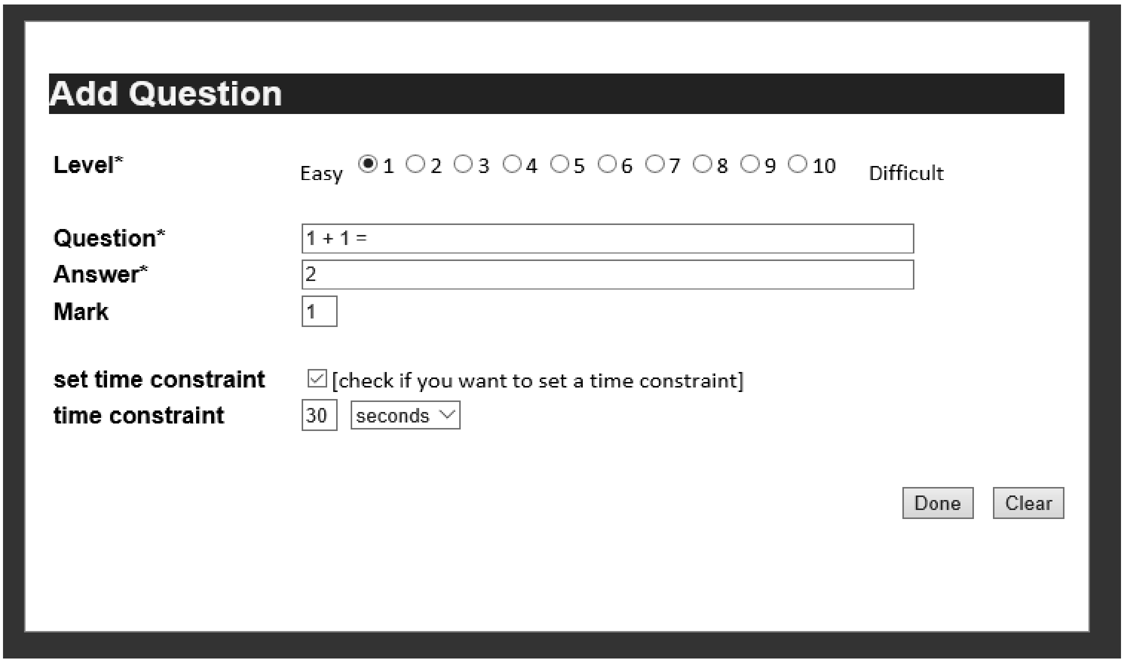


Figure 8.4. Sample interface for the examiner to add question and difficulty level

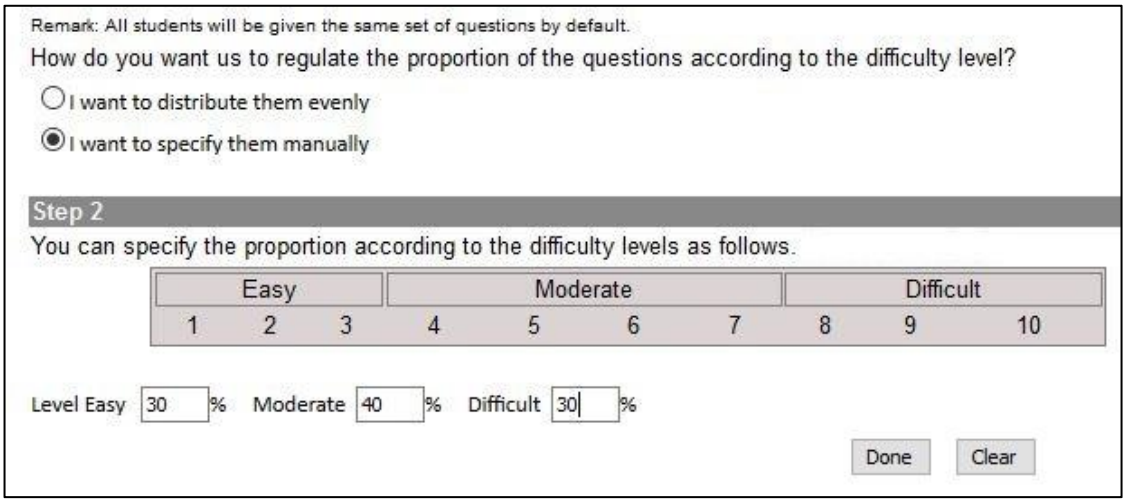


Figure 8.5. Sample interface for the examiner to set up an assessment by specifying the distribution of easy, moderate and difficult questions.

8.1.2 THE INFERENCE ENGINE

There are a few processes involved in the stress inference engine before it produces a stress measurement of a learner, as shown in Figure 8.6. First, it produces `JobPerformance`, `MouseBehaviour` and `KeystrokeBehaviour` data objects through finite state machines, aka finite state automata. According to [264], [265], a state machine is a device that stores the status of something at a given time and can operate on input to change the status and/or cause an action or output to take place for any given change. For instance, the sequence of symbols being read can be thought to constitute the input, while the sequence of symbols being written could be thought to constitute the output. We can also derive output by looking at the internal state of the controller after the input has been read. During the data collection, every raw keystroke and mouse data are collected at intervals of 10 milliseconds (ms). The high velocity of the data collection may result in computer resources overhead. Therefore, implementing finite-state automata in the data collection phase is important to enhance the system performance. Each time a question is completed, the raw data of the task duration, and mouse and keystroke dynamics are transformed into `JobPerformance`, `MouseBehaviour` and `KeystrokeBehaviour` according to the Equation 7.3 and 7.4 in Chapter 7. After that, the `MouseBehaviour` and `KeystrokeBehaviour` data will be fed into the Mouse and Keystroke Unifier to determine which behaviour to be analysed.

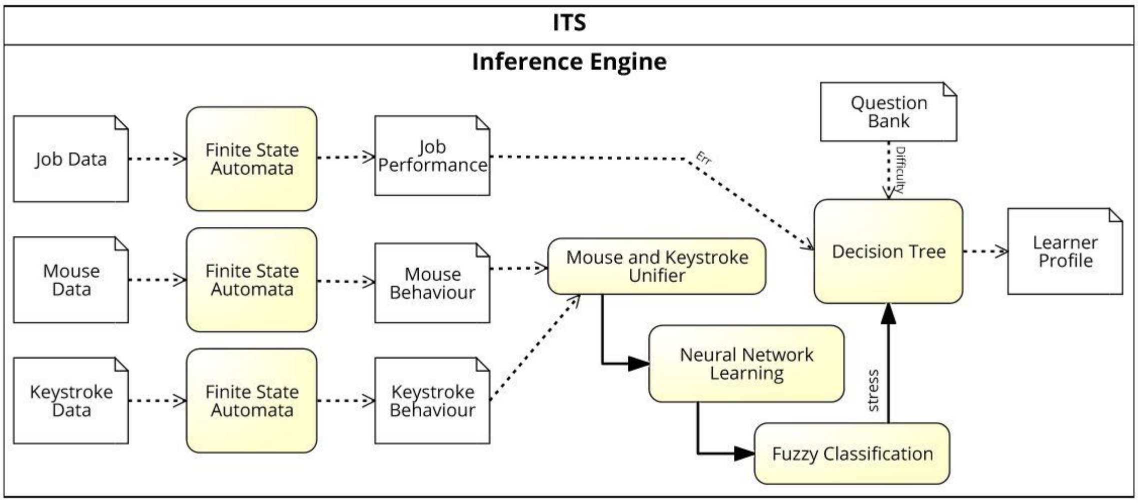


Figure 8.6. The Design of Inference Engine

8.1.2.1 THE MOUSE AND KEYSTROKE UNIFIER

Mouse and keystroke dynamics are used in stress measurement so that they can complement each other, since not all tasks require the use of both devices. For instance, when the user is busy typing, a mouse may become idle and hence no mouse data could be collected. Similarly, an exam that involves questions with multiple-choice selections may only require the use of a mouse but not a keyboard. Therefore, it is crucial to have a mechanism to determine which behaviour should be considered by the neural network to measure stress. Algorithm 8.1 shows how the Mouse and Keystroke Unifier determines the appropriate behaviour to be forwarded to the next process. The algorithm is simple. First it is determined whether the mouse is in use, by simply checking the mouse movement such as speed and left click activity. Even a movement can be tiny but since the mouse data are collected every 10 ms, the movement data will be computed and stored in the `MouseBehaviour` object. Similarly, any key press will be logged and transformed into the `KeystrokeBehaviour` object by the finite state automata process. If both behaviours are detected present, then all mouse and keystroke behaviours will be sent to the neural network to measure stress. Otherwise, only the activated device will be considered. Algorithm 8.2 shows the rules to determine the use of appropriate neural network.

ALGORITHM 8.1. RULES THAT DETERMINE THE PRESENCE OF MOUSE/KEYSTROKE BEHAVIOURS

```
Set MB = False, KB = False, MKB = False
 if(sum(MouseBehaviour.SMS, MouseBehaviour.SMCL) != 0) //if mouse is moved or clicked
        MB = True
if(sum(KeystrokeBehaviour.SKS, KeystrokeBehaviour .SKL) != 0) //if keys are pressed
        KB = True
if(MB == True &&KB == True) //if both behaviours present
        MKB = True                          //then use both behaviours
```

ALGORITHM 8.2. RULES THAT DETERMINE THE USE OF APPROPRIATE NEURAL NETWORK

```
if(MKB == True)
        fire_FFBP_MouseKey_Rules(mousekey_vector)
        //neural network with mouse and keystroke data
else if(MB == True)
        fire_FFBP_Mouse_Rules(mouse_vector)        //neural network with mouse data
else if(KB == True)
        fire_FFBP_Key_Rules(keystroke_vector)      //neural network with keystroke data
```

The C# code of the neural network learning functions above are shown in Appendix III Figure A3.2 to Figure A3.4.

8.1.2.2 STRESS MEASUREMENT USING FEEDFORWARD NEURAL NETWORK

Chapter 7 explained why feedforward back-propagation neural network is chosen and how it can be constructed. Three different neural networks are formed after the back-propagation training. The neural networks are used to predict the stress based on `MouseBehaviour`, `KeystrokeBehaviour`, or the unification of both. The architecture of the feedforward back-propagation neural net was presented in Chapter 7 Section 7.3.3 (see Algorithm 7.1). The numbers of hidden neurons of the networks are correspondent to the numbers of inputs. There is only one hidden layer for each network. The weights and the biases to layer 1 and layer 2 are obtained from the trained networks from Matlab. The actual implementation of the architecture written in C# code with the constant values of weights and biases for stress measurements based on mouse and keystroke dynamics are shown on Figure A3.2 to Figure A3.4 in Appendix III. These networks are able to produce a value that represents the stress measured, referred as $S_{B(Sensor)}$, which should be in the range of [-1, 1]. Then the value will be passed to the next classification process to identify whether the stress is significantly increased, decreased, or maintained stable (normal).

8.1.2.3 STRESS CLASSIFICATION AND FUZZY CLASSIFICATION OF DEMAND

This stage involves classifying the measured stress using either mouse, keystroke or the unification of both dynamics, into one of the crisp sets, i.e. A_1 −increases significantly, A_2 − decreases significantly or A_3 − remains stable (normal). The process will be followed by classifying the measured demand using fuzzy set model. The thresholds of the crisp set and fuzzy set are determined based on a large amount of sample data as presented in Chapter 7. These thresholds are used as default constants, universal to all users at the initial stage. However, as the individual behaviours, i.e. `MouseBehaviour` and `KeystrokeBehaviour`, are kept in the system, therefore a personalized adaptation can be generated in the future to update the neural network architecture as well as the fuzzy logic function so that they can be individualized. Enabling a personalized adaptation is very important as there are huge differences between individuals in terms of mouse behaviour and keystroke behaviour. However, this personalized adaptation design is not implemented in this project but will only be considered for future research. As presented in Chapter 7, the thresholds of the stress levels are determined using one standard deviation (*stdev*) away from the mean of $S_{B(Sensor)}$, ($mean(S_{B(Sensor)})$), to produce the actual output of stress, $Y(S_{B(Sensor)})$, which is activated by the following simple crisp function.

$$Y(S_{B(Sensor)}) = \begin{cases} 1 \ if S_{B(Sensor)} > mean(S_{B(Sensor)}) + \ stdev(S_{B(Sensor)}) \ , indicatesstressincreases \\ -1 \ if S_{B(Sensor)} < mean(S_{B(Sensor)}) - \ stdev(S_{B(Sensor)}), indicatesstressdecreases \\ 0 \qquad\qquad ifotherwise, indicatesstressisstable \ (normal)) \end{cases} \tag{8.1}$$

where $mean(S_{B(M)}) = 0.0354$, $stdev(S_{B(M)}) = 0.1283$ based on a total of 12,144 records of all tasks; $mean(S_{B(K)}) = 0.0245$, $stdev(S_{B(K)}) = 0.0738$, $mean(S_{B(M, K)}) = 0.0245$, $stdev(S_{B(M, K)}) = 0.1820$ based on 2562 records of both assessment and typing tasks. Therefore, there are 9 different crisp sets in total, expressed in Table 8.1below:

Table 8.1: Crisp Sets of Stress Level

Stress measured by mouse, $S_{B(M)}$	Stress measured by keyboard, $S_{B(K)}$	Stress by both mouse and keyboard, $S_{B(M,K)}$
$A_1 = \{x \mid x > 0.1637\}$	$A_1 = \{x \mid x > 0.0983\}$	$A_1 = \{x \mid x > 0.2065\}$
$A_2 = \{x \mid x < -0.0929\}$	$A_2 = \{x \mid x < -0.0493\}$	$A_2 = \{x \mid x < -0.1575\}$
$A_3 = \{x \mid -0.0929 \leq x \leq 0.1637\}$	$A_3 = \{x \mid -0.0493 \leq x \leq 0.0983\}$	$A_3 = \{x \mid -0.1575 \leq x \leq 0.2065\}$

where $A_1 = 1$, $A_2 = -1$ and $A_3 = 0$

Classical logic is chosen since the classification of stress level is pretty straight forward. The system is only required to determine whether the stress has decreased, remained stable or increased significantly, which generates an output of true or false, or more specifically {-1,0,1}. This classification of stress level is required in the next step for decision making.

To compute a task demand using objective measures, fuzzy inference system is utilized to handle the degree of vagueness in demand, with two fuzzy inputs. The fuzzy inference process applying the Mamdani method involves fuzzifying two inputs, and one fuzzy output. The fuzzy output demand consists of 3 membership functions (MF), i.e. *low* demand, *medium* demand and *high* demand. The two fuzzy inputs to the fuzzy inference system are the stress measured by the sensor, $S_{B(Sensor)}$, and stress measured based on time duration, i.e. S_{TD} (see Chapter 7). These two inputs are named as *stress* and *duration* respectively. This fuzzy model is built based on the assumption that Task Demand is correlated to Stress and Time Duration from the preliminary research done earlier [203]. Although error (or score) and passive attempt are also correlated to Task Demand, they are not fuzzy hence they are not placed as part of the fuzzy model. Error rate and passive attempt will be used in the next process for decision making.

Gaussian distribution function is used in all MFs of the input and output variables. The symmetric Gaussian function [263] is defined as follows:

$$\mu_{A_i}(x) = exp\left(-\frac{(x-c)^2}{2\sigma^2}\right) \tag{8.1}$$

where c is the mean and σ is the standard deviation. Both c and σ are the parameters fed to the Gaussian MF.

There are three MFs in each of the fuzzy input and output. The MFs determine the fuzzy membership values of each member in the fuzzy sets. The fuzzy set of stress is denoted by x, as:

$$S = \{x, \mu_S(x) | x \in X\} \tag{8.2}$$

where $\mu_S(x)$ as the membership function (MF) of x in S. Figure 8.7 below depicts the MFs visually.

The fuzzy set of duration is denoted by x, as:

$$T = \{x, \mu_T(x) | x \in X\} \tag{8.3}$$

where $\mu_T(x)$ as the membership function (MF) of x in T. Figure 8.8 visualizes the MFs.

The fuzzy set of demand is denoted by x, as:

$$D = \{x, \mu_D(x) | x \in X\} \tag{8.4}$$

where $\mu_D(x)$ as the membership function (MF) of *x* in *D*. Figure 8.9 visualizes the MFs.

Gaussian distribution function is used in all MFs of the output variable. The ranges of the fuzzy sets could be determined according to the distribution of the question difficulty levels set to the students. For example, assume that the examiner distributed 30% of easy questions (level 1 to level 3), 40% of average questions (level 4 to level 7) and 30% difficult questions (level 8 to level 10) during the setting of the test (see Figure 8.5).

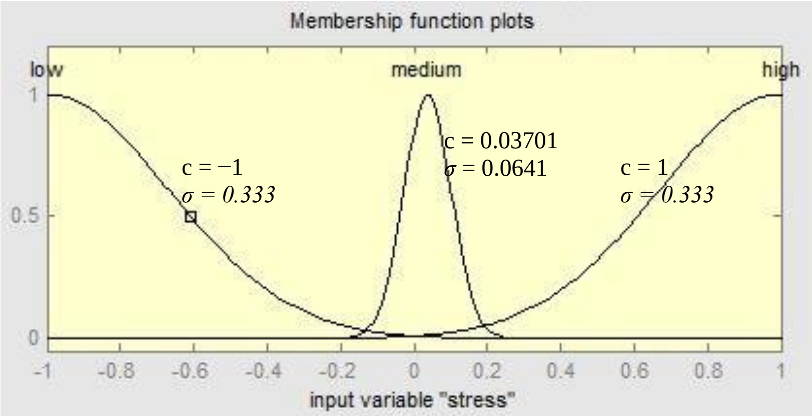


Figure 8.7. The MFs in stress fuzzy input: low(S_1), medium(S_2) and high(S_3)

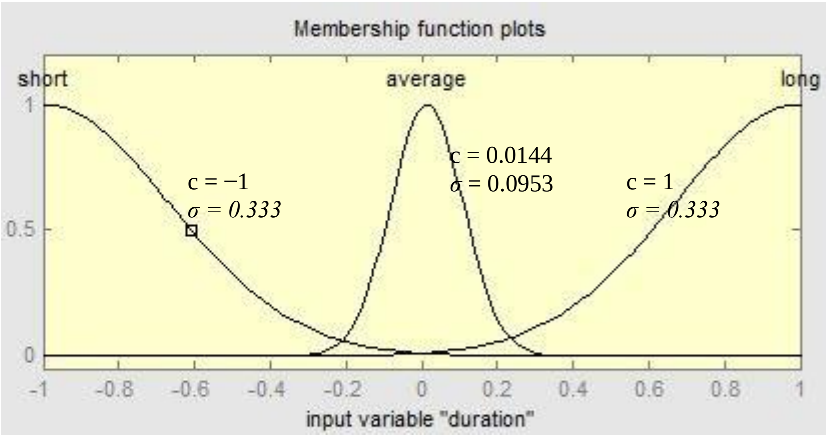


Figure 8.8. The MFs in duration fuzzy input: short(T_1), average(T_2) and long(T_3)

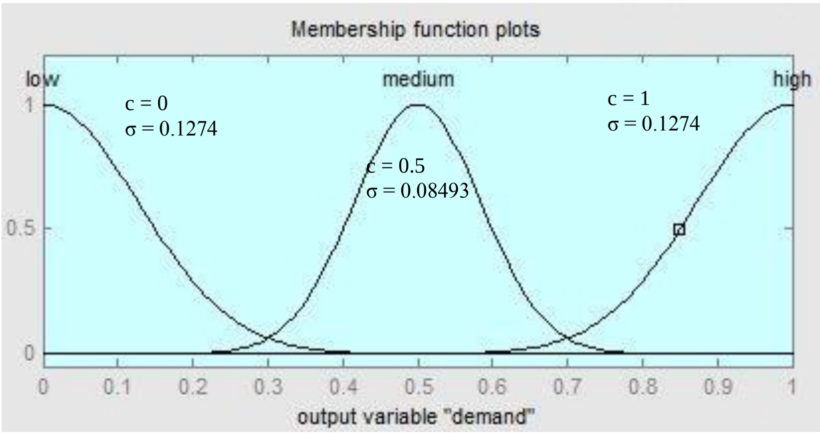


Figure 8.9. The MFs in demand fuzzy input: low(D_1), medium(D_2) and high(D_3)

After all the settings of input variables and output variable, associated with their MFs are done, then fuzzy rules must be defined.

Three fuzzy rules are set as below:

Rule 1: If (stress is low) and (duration is short) then (demand is low)
Rule 2: If (stress is medium) and (duration is average) then (demand is medium)
Rule 3: If (stress is high) and (duration is long) then (demand is high)

The logical AND operator is used to combine both *stress* and *duration* inputs, which is min(S, T). After the aggregation process, centroid defuzzification method will be used to defuzzify the output since it is the most popular method. The defuzzified value produced from the fuzzy inference system could be useful to determine the next difficulty level of the question to be distributed to the students. However, this research mainly uses this value for the analytic feedback function in the last stage. After the classifications of stress and demand are done, the inference process will proceed with making decisions of necessary adaptation on the interface based on a decision tree.

8.1.2.4 THE DECISION TREE

A decision tree is designed to represent the process of making a decision or a series of decisions by the ITS. A decision tree has internal nodes that test some attributes, e.g. the stress level and whether an error is made. Each branch represents the outcome of the test and each leaf node represents a decision outcome after considering all the attributes. The paths from root to leaf represent classification rules[4]. The decision tree is designed manually when creating the ITS prototype. Automatic decision tree induction will be explored in the future in order to refine the design of the decision tree classifier. Based on Figure 8.10, stress outcome, $Y(S_{B(Sensor)})$, which is obtained from the fuzzy classification process, is classified as normal, increased or decreased significantly. The attribute `Difficulty` is obtained from the `QuestionBank` object, and error made (Err) is retrieved from the `JobPerformance` object. The decision rules, after being transformed from the decision tree, are represented by the Algorithm 8.3. Table 8.2 presents the decision table that represents the decision needed for the adaptive interface and collective feedback reporting system. The `LearnerProfile` object is updated at the end of the process, which is needed by the collective feedback reporting system.

[4]https://en.wikipedia.org/wiki/Decision_tree

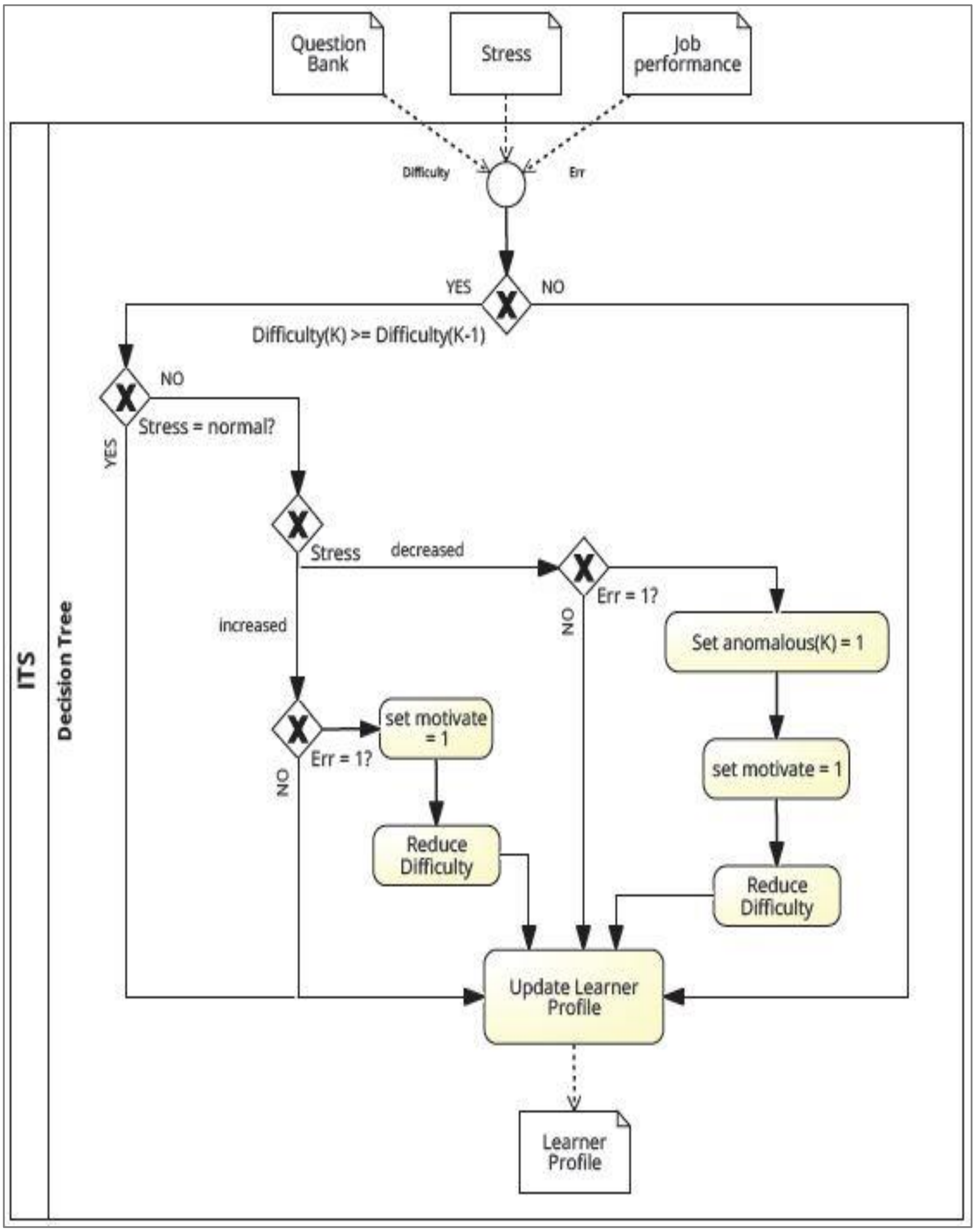


Figure 8.10. Decision tree in the inference engine

```
ALGORITHM 8.3. THE DECISION RULES
IF      Current QuestionBank.Difficulty >= previous QuestionBank.Difficulty
THEN    Check Stress                                        //Difficulty(K) >= Difficult(K-1)
        IF      Stress = "normal"      THEN  Continue
        ELSE    IF      Stress = "increased"
                THEN    check current JobPerformance.Err
                        IF      JobPerformance.Err = 1              //the student made error
                        THEN    Activate motivation                 //then set motivate = 1
                                AND
        QuestionBank.Difficulty(K+1)=QuestionBank.Difficulty(K)−1
                                                            //Reduce next question difficulty
                ELSE    Continue
        ELSE    IF      Stress = "decreased"
                THEN    check current JobPerformance.Err
                        IF      JobPerformance.Err = 1
                        THEN    Set current LearnerProfile.AnomalousBehaviour = 1

 //the user may have demonstrated anomalous behaviour, e.g. give up, not paying attention, etc.
                                AND     Active motivation           //set motivate= 1
                                AND
        QuestionBank.Difficulty(K+1)=QuestionBank.Difficulty(K)−1
                        ELSE    Set current LearnerProfile.Demand = −1
                                        // the demand could be lower than expected
ELSE    Continue

Update LearnerProfile at the end
        //to include the job performance, mouse behaviour and keystroke behaviour and stress
                                                                        classification
```

Table 8.2: The Decision Table that Tabulates Actions According to the Decision Rules

RULES			ACTIONS		
difficulty(K) >= difficult(K-1)	Stress	Err=1	LearnerProfile. AnomalousBehaviour	activate motivation	reduce Difficulty(K+1)
TRUE	normal	TRUE	X	X	X
TRUE	normal	FALSE	X	X	X
TRUE	increased	TRUE	X	YES	YES
TRUE	increased	FALSE	X	X	X
TRUE	decreased	TRUE	1	YES	YES
TRUE	decreased	FALSE	X	X	X
FALSE	normal	TRUE	X	X	X
FALSE	normal	FALSE	X	X	X
FALSE	increased	TRUE	X	X	X
FALSE	increased	FALSE	X	X	X
FALSE	decreased	TRUE	X	X	X
FALSE	decreased	FALSE	X	X	X

8.1.3 THE ADAPTIVE INTERFACE

This section aims to propose a plausible adaptive mechanism in the ITS. However, it does not focus on creating a new method for the adaptation purpose. A relative simple method to adapt the interface and the question difficulty level based on the outcome of the stress inference engine, such as the stress measured and the decision on activating motivation will be outlined here. According to the decision table as shown in Table 8.2, there are only two adaptations needed when a learner makes a mistake, one is when his/her stress is detected as significantly increased, and another is when anomalous behaviour is identified. When the learner makes a mistake, whether he or she feels significantly stressed, or starts losing attention, these are the moments he/she needs to be motivated to continue the next task. When the motivation mode is activated by the decision tree classifier in the inference engine, the learner will see a motivation message, before continue answering the next question. To motivate the learner further, the difficulty of the next question is set one level easier than the current question during the decision tree classification process, which

$$\text{Difficulty}(K+1) = \text{Difficulty}(K) - 1 \tag{8.6}$$

The adaptation of assessment could still be enhanced to accommodate a learner's needs further. For example, the adjustment of the difficulty level of the next question could be associated with the measured stress or demand gained from the inference engine, or the ITS could include adaptive help [260] so the system can adapt hints according to the line of problem that the student is currently following. Furthermore, assessment in e-learning does not limit to only numeric arithmetic problem. There are other problems, such as arithmetic word problem [266], and problem solving [267]. The system could also adapt according to learning style [268]. Figure 8.11 shows a sample motivation message displayed to the learners, which is picked randomly by the ITS. The collection of the job performance, including the time duration, will be paused until the learner is ready to go for the next question. When the NEXT button is clicked, the new question will be displayed, with difficulty level reduced by one as compared to the previous question.

Figure 8.11. The adaptive interface that shows motivation message when needed

8.1.4 COLLECTIVE FEEDBACK REPORTING SYSTEM

The existing e-learning systems, such as Blackboard [5] and Moodle [6],offer test analysis functions that provide statistics on overall test performance and individual test questions. One key feature of the functions provides discriminative information that helps examiners to recognize questions that might be poor discriminators of learner-performance. With this information, the examiners shall be able to improve questions for future test administrations or to adjust credit on current attempts. This feature is certainly good in helping the examiners to identify which questions are considered good, fair or poor (or easy, medium or hard in terms of difficulty). Questions that are considered good and fair are better at differentiating between students with higher and lower levels of knowledge, while poor questions, either easy or hard, are recommended for review. However, their analyses rely heavily on the learners' scores of the given test. This is certainly not enough for the examiners to identify the mistakes made by the students whether is due to the question is stressful in terms of cognitive load, or the students simply do not pay attention, or they attempt to give up. It is important to note that emotions, attention and engagement are key drivers for learning [59]. If analytics of learner-states such as emotions are introduced, the examiners will be able to track which learning activities the students are following, and whether they are distracted, simply guessing answers to quiz tests, or really engaged in learning [1].

A prototype of the collective assessment feedback reporting system is designed to enable a learner's cognitive and emotion states analysis. There are a few data generated by the inference engine are being used in the collective feedback reporting system. First, each question answered by the learner, which is stored into the `JobPerformance` object, allows the examiner not only to study the overall performance but also to review the progress of each individual learner. Job performance includes the time duration spent on a question, the error of the question, and the passive attempt produced by the learner. Second, the stress level measured by the neural network gives hint on how the question affects the stress state of the learner. A difficult question is expected to increase the stress level at a steady pace if the level of difficulty is adjusted on a stable basis. However, if the stress level increased or decreased significantly, then the question may be considered affecting the learner’s cognitive states or emotion state significantly. Third, the question demand that is computed by the decision tree is important in indicating which questions have a significant increment of demand, that may cause a learner to make mistakes. Lastly, the anomalous behaviour observed by the decision tree classifier gives hint to the examiners on which question the students are possibly demotivated, distracted, or simply guessing answers. To ease understanding, the items displayed on the analytical report are defined as follows:

Question: the order of the question asked

Initial Difficulty: the difficulty level set by the examiner, based on his/her personal assumption.

Demand: The variable `Demand` produced during the fuzzy classification process, based on stress level and time duration. According to the fuzzy sets as shown in Figure 8.9, an index between 0 and 0.3 indicates low demand, medium demand if the value is between 0.3 and 0.7, and high demand if it falls above 0.7.

Stress: The variable $S_{B(Sensor)}$produced during the neural network function. Positive value indicates increment of stress if compared to previous question, and vice versa.

Time Duration: Stress measured based on time difference between 2 consecutive tasks, as according to Equation 7.3. Positive value indicates increment of duration if compared to the previous question, and vice versa.

Error: The average error made for a particular question of all students

Passive Attempt: The average rate of passive attempt by all learners for a particular question. Passive attempt refers to the attempt to wait until the time is up if a time constraint is given.

Anomalous Behaviour: The average value of the variable named`Anomalous` used during the Decision Tree classification.

Discrimination Index: Each of the items in the summation is required to be scaled to equal range so that the summation of the maximum values of Demand, Stress, Time Duration, Error, Passive Attempt and Anomalous Behaviour is equal to one. Accordingly, the discrimination index represents the impact of the question on a learner's behaviour, where Discrimination Index = Demand + Stress + Time Duration + Error + Passive Attempt + Anomalous Behaviour. Any value in between 0.1 to 0.4 shows that the question is fair. Values below 0.1 shows that the demand could be much easier than expected. Values above 0.4 indicates that the question could be harder than expected. Values above 0.7 shows that the question could be extremely demanding. The question or the marking should be reviewed if the index is above 0.4 or below 0.1.

Revision of Question: It is a simple reference to the examiner if he or she does not understand the significance of the discrimination index. A tick (✓) will be shown to the examiner if the question is flagged for

	revision, when the discrimination index is greater or equal to 0.4 or below 0.1.
What was asked:	The actual question displayed to the learners.

Figure 8.12 and Figure 8.13 demonstrate the sample outputs of the collective feedback reports. The collected results are based on the same set of questions, which all or majority of the students have attempted. The first sample as shown in Figure 8.12 shows the results from Group 000, i.e. the group that was not given any time constraint. The first column of the table illustrates the order of the questions displayed. The actual contents of the questions are shown in the last column. The difficulty of the question is varied based on the maximum digits per number and the amount of numbers in a question, as well as the use of summation, deduction and multiplication operation (see Table 3.2 in Section 3.3.2). The examiner may assume that the question difficulty increases from Question 1 to Question 10 accordingly, which is shown in the second column. The setting of the initial difficulty is based on examiner's knowledge and assumption. However, the expectation could be wrong. This is the reason why the ITS is designed to recommend the necessary review of the question difficulty based on the discrimination index in the report, so that any mismatched expectation could be revised. The third column shows the average demand values of each question, measured from the fuzzy classification function. For instance, the demand values of Question 1 to Question 3 are more or less the same, but the demand has been increased to 0.61 in Question 4, indicates the possibility of change of question style. The demand is then seemed becoming lower for the subsequent question but it rises again in Question 8.

The conventional learning management system such as Blackboard mainly uses scores in calculating the discrimination index and question difficulty. Besides considering the score or the error rate, the proposed ITS has additional features, which include Stress, Time Duration, Passive Attempt and Anomalous Behaviour. For the first question, the stress has increased but the time used is lesser than the calibration/login task. The increment of stress may indicate that the question requires more cognitive load if compared to the calibration task. However, in Question 2, a slight drop of 2.88% is observed, indicating that the stress level remains stable. Question 4 shows rises of stress and time duration, and students also start making errors from Question 4 onward. Question 8 demonstrates significant rises of stress and time duration, indicating that the question difficulty is levelled up significantly. The examiner could also observe that all students have produced wrong answers for Question 10, but there is no increment in terms of demand, but stress and time duration have decreased unexpectedly. The examiner could also notice 23.33% of the learners behaved anomalously on the last question, indicating that the students may not answer the question properly, which could possibly due to their losing motivation. In this case, they should consider revise the marking of Question 10, or to review the teaching method of this particular question in the future. In terms of the discrimination index, Question 8 to Question 10 produce high values, which are more than 0.4, that indicate a high possibility of a mismatch of expectation. Therefore, the questions are recommended for review.

Comparing the outcome of this report to the exam analysis by the existing LMS such as Blackboard, this proposed ITS provides a better discrimination index that is not only based on score, but also from the inferred demand, stress, time duration, passive attempt and anomalous behaviour from the inference engine. For example, Blackboard may rate Question 5 and Question 8 as the same difficulty since they provide the same score or error rate. However, the inference system from the proposed ITS rates them differently, which Q8 is considered harder than Q5 as its discrimination index is higher.

The next case study shows the analytical report from Group 110, which the learners were given 30 seconds time constraint and a digital clock display on the screen. A comparison between the report generated based on the Blackboard Item Analysis [5] and the analytical report based on the proposed ITS are given. Slightly different from Figure 8.12, the analysis shown in Figure 8.13 is collected from Group 110. Since the group was given a time constraint, the item of Passive Attempt is taken into account. The results of the analysis are quite similar to Group 000. However, the examiner could observe that the discrimination indexes from Question 1 to Question 7 are slightly lower compared to Group 000, which shows that the impact of these questions to the learners' motivation or behaviour could be lower. However, the discrimination indexes rise at Question 8 and Question 9, which become higher than Group 000, and indicate that the impacts on Group 110 are higher. These changes could be due to the time constraint given, which affect the passive attempt and errors made by the students in Group 110. Interestingly, the indices show no difference for Question 10. All the students from both groups did not answer the question correctly, nevertheless only 13% of them from Group 110 demonstrated anomalous behaviour when answering the question, compared to 23% from Group 000. Low value in the Anomalous Behaviour column indicates that most of the students may have attempted the question properly. As the final conclusion to the examiner for Group 110, reviewing the assessment of Question 8 to Question 10 would be needed as their discrimination indices exceed 0.4, which the questions could be slightly more demanding than the expected level, especially when a short time constraint is given.

Review Assessment: Collective Feedback Report

The test started on	14 April 2014 Time: 5:00PM
Details	**Group 000**
Assessment is timed	This test has NO time limit.
Multiple Attempts	Not allowed. This Test can only be taken once.
No. of Participants	30
Self Assessment	Student answers and score are not visible to the instructor.

Question	Initial Difficulty	Demand	Stress	Time Duration	Error	Anomalous Behaviour	Discrimination Index	Revision of Question	What was asked
			Changes from previous task (%)		Percentage of students (%)		Revise if Index > 0.4		
1	1	0.45	0.19	-0.25	0	0	**0.33**		6+2
2	2	0.48	-0.03	-0.02	0	0	**0.29**		9*4
3	3	0.48	0	0.02	0	0	**0.3**		6*5-1
4	4	0.6	0.07	0.19	0.03	0	**0.34**		(8+9)*2
5	5	0.55	0.08	0.11	0.2	0.07	**0.38**		7-8*10
6	6	0.44	-0.81	-0.26	0	0	**0.13**		58+20*(8-6)
7	7	0.51	-0.01	0.08	0.03	0	**0.31**		67-2*(4+2)
8	8	0.67	0.18	0.29	0.2	0.03	**0.42**	✓	(880+12+50-520)*2
9	9	0.53	0.01	0.1	0.37	0.03	**0.39**		105+83*5-3*60
10	10	0.43	-0.12	-0.11	1	0.23	**0.51**	✓	561-81*5+3*610

Figure 8.12. The collective feedback report for Group 000 (without time constraint)

Review Assessment: Collective Feedback Report

The test started on	15 April 2014 Time: 5:00PM
Details	**Group 110**
Assessment is timed	This test has 30 seconds time limit. A clock is shown on the screen.
Multiple Attempts	Not allowed. This Test can only be taken once.
No. of Participants	30
Self Assessment	Student answers and score are visible to the instructor.

Question	Initial Difficulty	Demand	Stress	Time Duration	Error	Passive Attempt	Anomalous Behaviour	Discrimination Index	Revision of Question	What was asked
			Changes from previous task (%)		Percentage of students (%)			Revise if Index > 0.4		
1	1	0.42	0.16	-0.29	0	0	0	**0.22**		6+2
2	2	0.48	-0.02	-0.02	0	0	0	**0.24**		9*4
3	3	0.54	0.04	0.08	0	0	0	**0.27**		6*5-1
4	4	0.5	-0.01	0.08	0	0	0	**0.26**		(8+9)*2
5	5	0.6	0.14	0.16	0.13	0	0	**0.31**		7-8*10
6	6	0.52	-0.03	0.11	0.2	0.07	0.03	**0.31**		58+20*(8-6)
7	7	0.42	-0.07	-0.08	0.13	0	0.03	**0.25**		67-2*(4+2)
8	8	0.7	0.12	0.31	0.47	0.43	0.03	**0.47**	✓	(880+12+50-520)*2
9	9	0.51	-0.01	0.04	0.6	0.57	0	**0.45**	✓	105+83*5-3*60
10	10	0.44	-0.05	-0.07	1	0.57	0.13	**0.51**	✓	561-81*5+3*610

Figure 8.13. The collective feedback report for Group 110 (with 30s time constraint and a clock display)

Table 8.3 is generated based on the computation set by Blackboard [5]. The discrimination index is calculated with the Pearson correlation coefficient between the scores of each student on a question and the scores of each student on the assessment. The question is considered *Good* if the value falls greater than 0.3, *Fair* if it is in between 0.1 and 0.3, and *Poor* if it is less than 0.1. Discrimination is listed as *Cannot Calculate* when 100% students either did the question right or wrong, or when all of them receive the same score on the same question. Questions fall into *Good* and *Fair* categories are considered better at differentiating between students with good knowledge and students with poor knowledge of a subject. Questions in the Poor category are recommended for review as they cannot differentiate these two groups of students. It is also stated that high difficulty values do not assure high levels of discrimination. The Item Analysis also provides an indicator that is known as Difficulty. Difficulty refers to the percentage of students who answered the question correctly. Difficulty values are ranged from 0% to 100%, with a high percentage (greater than 80%) means the question is *Easy*, while low percentage (lower than 30%) indicates the question is *Hard*, otherwise the difficulty is considered *Medium*. Questions in the *Easy* or *Hard* categories are recommended for review and they be flagged with a red circle for examiner's attention. Two other statistical indicators, which is the standard deviation (Std. Dv.), measures of how far the scores deviate from the mean, and the standard error (Std. Err) provides a measure of the statistical accuracy of the estimated amount of variability in a student's score due to chance.

As shown in Table 8.3, the Item Analysis in the Blackboard could not discriminate Question 1 to Question 4, as well as Question 10, since all of the students did those questions either correctly or wrongly. Compared to the analytical report as shown in Figure 8.13, the proposed ITS could discriminate these questions more precisely. It can also show the level of difficulty is in fact increasing slightly from Question 1 to Question 4, and Question 10 is almost double the difficulty of those questions. Besides, the Blackboard Item Analysis rates Question 7 as Easy question since 87% of the students answered the question correctly. However, this information could mislead the examiner, as he or she may think that the question is easier than his or her expectation. Similarly, Blackboard rates both Question 5 and Question 7 as Easy, but it could not further differentiate the level of difficulty. For Question 8 and Question 9, although they are identified by the proposed ITS to be difficult questions that are recommended for revision, but the Blackboard only flags Question 10. Blackboard also flags Question 1 to Question 5 for revision as the questions are considered easy, although their difficulties are within the examiner's expectation.

Table 8.3: Part of the Item Analysis based on Blackboard [5]

Q.	Discrimination Index	Discrimination	Average Score	Difficulty	Std. Dv.	Std. Err
1●	cannot calculate	-	1	easy	0	0
2●	cannot calculate	-	1	easy	0	0
3●	cannot calculate	-	1	easy	0	0
4●	cannot calculate	-	1	easy	0	0
5●	0.60	good	87%	easy	0.35	0.06
6	0.77	good	80%	medium	0.41	0.07
7●	0.75	good	87%	easy	0.35	0.06
8	0.44	good	53%	medium	0.51	0.09
9	0.60	good	40%	medium	0.50	0.09
10●	cannot calculate	-	0	hard	0	0

Questions that are recommended for examiner's review are flagged with red circles.

8.2 CONCLUSION

It is certainly not enough to track a learners' performances by referring only to number facts such as time spent and scores of a test. Teachers or LMS should take into account learners' emotions, motivation and engagement while learning, so that personalized learning can be enabled, and fairer assessment can be done. A prototype of such ITS using an affective computing method is proposed. The proposed ITS is built using simple algorithms, based on the motivation to build a cheap and effective method for an online affective learning system. The architecture of the ITS is presented, and the details of its inference engine are given. The design of the collective feedback reporting system is proposed. Inference engine is the core module in the ITS that produces inferences on a learner's behaviour, stress level, job performances, and the decision of adaptation that motivates learner. Stress measurement is done based on learners' mouse and keystroke dynamics. Adaptation of assessment material is done when significant stress increment or anomalous behaviour of individual is detected. At the end of the session, a collective feedback is sent to the examiner to assess the possibility that a task contains mismatched expectation of difficulty levels. Analytics of learner states such as stress and anomalous behaviour are also introduced in the system, so that the examiners are able to track which questions their students are following, and whether they are distracted, simply answering the test without effort, or really engaged in learning.

Two case studies were given, which the learners from Group 000 and Group 110 are compared. These two groups of learners demonstrated slightly different behaviours when answering the questions, which learners from Group 110 are believed having better motivation since the discrimination indexes are slightly lower than Group 000 in general. However, Question 8 to Question 10 could be more demanding for Group 110 as they were given only 30 seconds to answer those questions. The collective feedback report generated by the proposed ITS is also compared with the Item Analysis Report by Blackboard. The ITS analytical reports managed to overcome some limitations of the Blackboard methods, such as unable to calculate the discrimination index if all the students scored the same on a question. Unlike Blackboard, the ITS would only recommend the examiners to review a question if it does not match with his or her initial expectation. Lastly, Blackboard generates its discriminative factors such as discrimination index, difficulty, standard deviation and standard error solely based on learners' scores. On the flip side, the ITS uses the factors such as measured demand, stress, time duration, passive attempt and detection of anomalous behaviour, on top of scores, to provide a better discrimination index that allows the examiner to understand how much easier or how much harder a question is compared to the previous one.

Although the proposed ITS provides useful features to the users, it is not without limitations. There are three areas that the ITS that need improvements in the future. First, constant values, such as weights and biases are used in the trained neural network, as well as the fuzzy classification in the inference engine. These values are good enough to be used at the initial stage where learners' data are missing. However, to build a personalized affective learning system, the neural network learning and the fuzzy set shall be adapted too based on individual learner's historical data. Therefore, the mouse and keystroke behaviour objects are included in the `LearnerProfile` dataset for this purpose. Second, more experiments needed to be conducted to validate the design of the proposed ITS. Physiological methods, such as blood pressure and heart beat measurement will be considered for validation purpose, although using this equipment also means large sample data is hard to obtain. Third, the decision tree was designed manually and done based on our own assumptions. More heuristic data should be collected from experts to produce more optimal decisions. Lastly, the applications of the stress measurement model based on mouse and keystroke dynamics are only limited to the adaptive interface, adaptive assessment and the collective feedback reporting system. However, we strongly believe that there are more applications can in fact be built. This solution is considered cheap, ubiquitous and less intrusive, and could be reliably reflect the changes of learners' behaviour and stress state.

CHAPTER 9: CONCLUSION

It would be desirable to have a means of assessing a learner's stress levels in a task-independent way through an online platform. The underlying intent of this research, to detect relevant learner's states such as stress from mouse dynamics and keystroke dynamics in the context of e-learning system, made a contribution in affective and adaptive learning system design and development. This research was set out to produce a cheap, task independent, ubiquitous and less obtrusive means of estimating users' stress levels using mouse and keystroke dynamics. There are many valuable application areas in affective learning research and development if stress can be measured automatically. The study sought to answer two research questions for achieving the desired solution. First, how an effective construct that measures a learner's cognitive states and stress levels can be developed by using mouse and keystroke dynamics? Second, how the construct that measures users' cognitive states and stress levels using mouse and keystroke dynamics can be applied in an intelligent tutoring system? Due to a lack of literature supporting such a solution, before the two questions could be answered, three preliminary research experiments were conducted. These experiments were carried out to study the relationships between task demand, external psycho-physiological stimuli, learners' stress perceptions, cognitive states, and their mouse and keystroke behaviours. Significant impacts and correlations found between those factors shed light on constructing the means of measuring learners' states using mouse and keystroke dynamics. Accordingly, a stress measurement model using artificial intelligence methods, and an ITS that applies the affective measurement have been proposed. The objectives of the research are then considered achieved. Hence, this chapter concludes the work that have been carried out to answer the two research questions. Section 9.1 concludes the research aims and objectives. Section 9.2 critically review the limitations of the study. Section 9.3 presents the contributions to cognitive researchers and the e-learning developers. Section 9.4 states the potential future work. Lastly Section 9.5 provides the summative conclusion of the thesis.

9.1 STRESS MEASUREMENT FOR AFFECTIVE E-LEARNING SYSTEM

Much existing research related to affective learning adopted emotions defined by psychological research, e.g. the four quadrants of learning emotions as proposed by Kort et al [7], the Positive and Negative Affect Schedule (PANAS) scale by Watson et al [8], or Russell's Circumplex Model of Affect [9]. However, enabling automated detection of rich granularity of emotions in an online environment is extremely challenging. As measuring emotions in large scale is difficult, this study aims to measure only stress instead of other emotions. Stress can degrade reception and cause inefficient learning [10], [41]. If stress can be detected automatically, it could be useful for affective computing developers to build an effective e-learning system that helps to identify the stress factors that cause poor learning behaviour or performance. Examples of stress factors may include mismatched demand by the teachers, frustrating resources, or even bad user interface and usability design, which could negatively impact user experience during e-learning.

The applications of stress measurement in educational technology research prompted two important research questions to be solved: (1) how an effective construct that measures a learner's cognitive states and stress level can be developed? (2) How the construct that measures users' cognitive states and stress level can be applied in an ITS? The emerging affective computing research in mouse and keystroke dynamics-based analyses show potential implementation of automated emotion detection, but most of them studied the methods separately. We strongly believe that the analyses of mouse and keystroke dynamics should be unified, since not all tasks require the use of single device.

To answer the first research question, a means that can effectively and quantitatively measure a learner's cognitive states and stress levels, possibly with mouse and keystroke dynamics, must be identified. Unfortunately, there is a lack of affective computing studies that examine the correlations of emotional stress or cognitive states to user's mouse and keystroke dynamics. Using a self-report survey to collect learner's self-perception of stress could be easy, but it is not suitable to be applied in the ITS. Physiological method such as heart-beat, blood pressure or cortisol measurement could be more accurate but special setup of equipment is needed. Furthermore, it is more complicated to measure human cognitive states than emotional stress, as cognitive load usually involves process working with short-term and long-term memory, attention, motivation, behaviour [18]–[22]. To explore the correlations between tasks, external stimuli, learner's stress and cognitive states, and his/her mouse and keystroke dynamics in an e-learning environment, the MADB Model proposed by Wang [22] is adopted and adapted according to e-learning environment. Wang demonstrated how the complicated human emotional and perceptual phenomena can be rigorously modelled and formally treated based on cognitive informatics theories and denotational mathematics. The model allows us to define formally and quantitatively the relationship between emotion, motivation, attitude and behaviour. The detailed application of the MADB model in the research was explained in Chapter 3. To simulate the tasks that are usually carried out in the e-learning environment, a mock-up of an e-learning system was built, and the learners were required to do three different tasks in the experiments, i.e. search for a learning material, assessment, and typing.

To validate the feasibility of building an ITS that enables stress measurement using mouse and keystroke dynamics, three hypotheses below are important.

1. Direct instruction (such as assessment and typing demand), indirect instruction (such as search requirement), and external stimuli (such as menu design, time pressure, clock and/or countdown timer displays) affect stress perception and motivation.
2. The correlations between direct instruction, indirect instruction, external stimuli, stress perception, motivation, rational motivation, attitude, decision, andbehaviour are significant.
3. Behavior affects mouse behaviour and keystroke behaviour.

We argued that if the hypotheses above are accepted, then mouse or keystroke dynamics could be considered as sensors that can sense the changes of the learner's cognitive and stress level when task demand is changed significantly or when the stimuli is induced. To answer the three hypotheses above, three separate experiments were conducted according to the three aforementioned tasks, i.e. search, assessment and typing. The first task studied the effects of search instructions and menu design on learners' stress and motivation. The second task studied the effects of cognitive load demand using mental arithmetic, and external stimuli such as time pressure, clock display and timer display, on a learner's stress perception and motivation. The last task identified the impacts of task demand varied by text length and language familiarity, and external stimuli such as time pressure, clock display and timer display, on a learner's stress perception and motivation. The search task studied the effects of mouse behaviour while the rest of the tasks examined the effects on both mouse and keystroke behaviours. Although these three tasks had different objectives, the computation of MADB was consistent, except for the attention measurement in the search task was calculated based on the attempt to revisit the question, instead of the attempt to wait till the time is up. The details of the results were discussed in Chapter 4, Chapter 5 and Chapter 6 respectively. Significant effects and similar correlations found on the three tasks shed light on building a construct that measures learners' cognitive states and stress level by using mouse and keystroke dynamics, which can be cheap, ubiquitous, less intrusive and task-independent.

The acceptance of the three hypotheses above enables us to proceed with the development of the construct that measures a learner's cognitive states and stress levels using mouse and keystroke dynamics. Accordingly, a stress measurement model based on mouse and keystroke dynamics was constructed and tested using three different stress classifiers, namely certainty factor (CF), feedforward back-propagation (FFBP) neural network, and adaptive neuro-fuzzy inference system (ANFIS). The details and the critical evaluations of these three classifiers were given in Chapter 7. Close estimation of stress based on mouse/keystroke dynamics against the estimation of stress based on time duration showed promising possibility to use solely mouse and keystroke dynamics in measuring a learner's cognitive states and stress. From the experiments, FFBP neural net could be the best model to construct the stress classifier for the inference engine of the intelligent tutoring system.

To proceed with the second research question, the proposed ITS was designed to enable automated stress classification. A learner's stress level was classified into one of the three groups, i.e. increased, decreased or stable stress. The classification was done with finite state automata, which transform raw data into behaviour models, mouse and keystroke unifier, feed-forward neural network, fuzzy classification and decision tree. Besides producing inferences on a learner's stress level, the inference engine, which is the central component of the ITS, also makes inferences on job performances, anomalous behaviour, and the decision of adaptation that motivates learner. The detailed processes in the stress inference engine were discussed in Chapter 8. The stress measurement model based on mouse and keystroke dynamics were then applied in two modules: adaptive learning; and collective feedback report. The first module focused on identifying the need to adapt interface in order to motivate a learner when he or she loses motivation to continue the task. This was done by automated computation of anomalous behaviour by the inference engine. The second module provided an analytical report to the examiners about their learners' collective performance, and the discrimination factors that distinguish extremely challenging questions (or vice versa). Different from the existing learning management systems such as Moodle and Blackboard, the analytical report of the proposed ITS was done based on the learners' states, such as stress levels and anomalous behaviours, on top of their scores. This enables the teachers to track the questions that are engaged by the learners, and the discriminative questions that lead to bad performances possibly due to losing attention or motivation. The ITS analytical report is believed has overcome the limitations of Blackboard™ Item Analysis Report [5]. For instance, the proposed ITS would recommend the examiners to review a question only when necessary, i.e. if the students' performance did not match with the expectation. Besides, Blackboard uses only learners' scores in determining its discrimination factors. However, the proposed ITS uses more factors, such as measured task demand, stress value, task duration, passive attempt and the detection of anomalous behaviour, to determine the discrimination factor that causes undesirable result. In addition, the ITS is able to produce a better discrimination index that allows the examiner to understand how much easier or how much harder is a question compared to the previous one, although the scores of two questions are close with each other. Accordingly, the analytical report generated by the ITS definitely provides a better insight to the examiners on how effectively the assessment will improve learning.

9.2 LIMITATION

Although a few contributions are made, this research has limitations. First, this research is set only to detect stress. Detecting stress alone may not be enough for affective learning, which requires a better understanding of a granularity of emotions. However, detecting stress can be useful to determine the stressor that causes student's unhelpful behaviour in learning. In our case, only those stress factors that are related to the design of the system and job demands have been included into considerations. Personal stress or stress related to well-being issues, were not measured. Secondly, the sample size is small and the data are collected from the students who come from a higher learning institution, with narrow range of ages and disciplines. This sample does not allow the findings of this research to be generalized. Third, the experiment subjects' skills in mental arithmetic and typing are not assessed prior to the experiments and hence their skill levels might be varied, which could affect the results of the studies. Fourth, the limited capabilities of the keystroke and mouse loggers that are built by us generated incomplete data from a number of participants. Accordingly, some records are either removed or imputed, which could affect the analyses of the experiments. The limitations of the instruments are due to the technical constraints by the operating system and the programming languages used, as well as a bug that was not discovered earlier during the pilot test. Besides, the mouse and key-loggers are constructed separately using two different languages for the experiments, due to limited knowledge and short time constraint. The mouse logger is built using Java, while the key logger is built with VB.NET.A more robust instrument needs to be developed in the future to ensure more complete data can be collected, and both loggers can be unified into a single solution.

There are also some flaws in the experimental design. During the preliminary studies that examine the effects of external stimuli, the timer versus clock was conflated with invasive and distracting flashing. There is no significant evidence from the study for the hypothesis that timers are proved to be more stressful than clocks. Further experiments with more controlled salience of timer and clock displays need to be conducted to identify the significant effects of these external stimuli on learners. Although some socio-demographic factors that might affect the results have been considered for the experiments, such as age and specialization, there are other factors that we do not control. These uncontrolled factors include gender and non-disclosed disability. More control factors should be considered, especially for the students who are from different education background, cultures and races. Furthermore, prior knowledge and skills of solving a given problem, such as mental arithmetic and typing, should be assessed before the participants taking part in the experiments. The experiments are also conducted on different sessions with different group of learners. Some external factors that might affect learners' stress levels are not controlled. These external uncontrolled factors include: some students having a mid-term test right after the experiments; some having just finished the test before taking part in the experiments; some having attended long day of classes; and some having to come early in the morning for the experiments. These uncontrolled factors might have affected the preliminary studies related to the learners' self-report stress survey. However, the impacts of these factors to the dataset needed for the inference engine at the later stage are small. Besides, the uncontrolled factors that may create external stress to the users before taking up the task have been handled by the calibration process during the login session in the system. The calibrated mouse and keystroke data provide us a useful baseline of the non-stress condition.

The limitations of the experiments also include the content of the search task instructions, which the questions given to the participants to look for specific learning materials are designed with mixture of straight-forward instructions and indirect instructions. Straight-forward instruction is the question that specifies clearly of what item to search, while indirect instruction is the question that intentionally provides ambiguous information, which a learner has to guess the item to search. The effect of the search instruction content is not included in the studies, but it might affect the results of the job performance. Besides, the stress measurement is validated based on the learners' self-reported stress perceptions, and the time duration spent on a question. User self-survey is easy but it could be unreliable as human has problem quantifying thoughts and feeling accurately. The correlation between time duration spent on a task and stress is not rigorously founded by existing psycho-physiological research. More research must be carried out to validate the proposed stress measurement model based on mouse/keystroke dynamics.

Finally, although the design of the ITS is working well to produce a personalized adaptation and a collective feedback report for the examiners, it is not rigorously tested nor validated. Therefore, it is important to test the effectiveness and efficiency of the ITS in the future with different groups of learners. The stress measurement model in the inference engine was also developed using the correlation coefficients, means and standard deviations obtained from the collected dataset. However, these constants are fixed. Although they might be useful as the default setting for the new users, nevertheless these constants should be varied according to individuals, as different people have different behaviours when using the mouse and keyboard under varied conditions. Besides, the learning theory that focuses on the understanding of students’ behaviour during learning were not studied. Finally, the main aims of the research are to design a stress measurement model that is able to detect learners’ stress levels automatically using a low cost and unobtrusive method, and to suggest plausible applications of such stress measurement model in an ITS. Therefore, the adaptation of curriculum design and content delivery were not included in our research scope, although they are important in e-learning.

9.3 CONTRIBUTION

There are two major implications sought from the research that might worth further research by the cognitive researchers and neuroscientists. First, the research added new theoretical and empirical knowledge of stress measurement models in an e-learning environment using mouse and keystroke dynamics. The few signals produced by mouse and keystroke dynamics could be reliably used, cost effective, less intrusive and can be implemented ubiquitously as part of a normal system. To achieve that, the feasibility of using mouse and keystroke dynamics in measuring user's stress levels is tested based on different activities in an e-learning environment, such as searching for a desired learning material, assessment and typing. The learner's stress and cognitive states are computed based on the adapted Motivational Attitude-driven Behaviour model as proposed by Wang [22].

A total of 190 undergraduate students voluntarily participated in the experiments. The datasets generated from these preliminary research experiments are not only useful in helping us to develop the stress measurement model, but also for future research. The results reported in Chapter 4 to Chapter 6 show that the correlations of learners' affective and cognitive states to their mouse behaviour and keystroke behaviour are significant. This sheds light on the possibility of producing a cost-effective, unobtrusive, task-independent and objective method to measure user's states. This stress measurement model based on mouse and keystroke dynamics also enables continuous stress monitoring in an online environment, by computing the differences of task durations and mouse behaviours between 2 tasks or 2 time intervals, which are discussed in Chapter 7.

The second contribution from this research is to add a theoretical framework of a stress inference engine to the affective learning system developers. Accordingly, a theoretical framework of the inference engine for the ITS development is proposed, which consists of three major components. The first component is a feed-forward neural network that measures a learner's stress level by comparing his/her current mouse/keystroke behaviours during the current task and previous task. The second component contains fuzzy classifications that classify the learner's stress level, and the correspondent task demand into different classes. The last component in the inference engine is a decision tree that identifies the anomalous learning behaviour, and decides whether an adaptation is needed.

The output of the inference engine helps in improving several learning processes. First, it enables useful adaptation, such as adaptive interface, personalized learning content or customized assessment materials according to learner's behaviour. For instance, the proposed adaptation module presented in Chapter 8 produces an adaptive interface that is designed to re-engage the learner to continue the next task, if significant stress increment or anomalous behaviour is detected. Secondly, it is useful in producing collective feedback for the examiners to identify effectively the possibly mismatched expectation of task demand. Section 8.1.4 illustrates the design of the collective feedback report, which consists of the computed task demand, stress level, time duration, error rate, passive attempt, anomalous behaviour and discrimination index. This could provide more precise feedback that takes learners' emotion and learning behaviour into account, rather than relying only the assessment scores that are done by the existing learning management systems, such as Blackboard and Moodle. A prototype is built accordingly based on the existing dataset, from the mental arithmetic test, as a proof of concept to demonstrate its feasibility. Although there are still many challenges and difficulties in the sense of technologies that need to be solved, this research managed to propose a solution that is easy and feasible to be implemented in an online environment.

9.4 FUTURE WORK

The main limitation of the current research is that the applications of the proposed stress measurement model in the ITS are not rigorously validated. Our future work shall continue to validate the model and its applications using physiological approach, such as cortisol, blood pressure or heart-beat measurements. However, using physiological methods may also mean that the experiment sample size may be small since special equipment is required. The current research is also limited to using pre-trained network and constant parameters in the stress inference engine. Future research will look into algorithms adapting these constants in order to produce a more personalized adaptive learning system. Since a cheap, task independent, ubiquitous and less obtrusive means of estimating users' stress levels can be produced based on automated mouse and keystroke dynamics analyses, we strongly believe that many valuable applications in affective computing can be developed. Our future research will also look into more applications of the proposed stress estimation model in usability testing, personalized games and adaptive web, in addition to many other useful areas in affective learning.

9.5 CONCLUSION

A cheap, task-independent, ubiquitous and less obtrusive means of estimating users' stress levels using mouse and keystroke dynamics was proposed, which could bring many useful application areas in affective learning research and development. For instance, the system could adapt learning materials or provides analytical information to teachers related to learner states, such as stress and motivational behaviour. Two research objectives of the research have been achieved. First, we aimed to design an effective construct that measures a learner's cognitive states and stress level using mouse and keystroke dynamics. Second, we proposed two applications of stress measurement using mouse and keystroke dynamics in an ITS. The inference engine, which was the core element of the ITS that measures learners' states, is built based on feedforward back-propagation neural network, fuzzy classification and a decision tree. Stress estimated using mouse and keystroke data, can be classified into three classes dependent on the demand of a task, i.e. increased significantly, decreased significantly, or remained stable (normal). Accordingly, the significantly high task demand, which causes the learner to disengage from learning, and hence making mistakes and experiencing significantly higher stress levels, could be determined. At this point, adaptation such as giving the disengaged learner a short pause by displaying a motivational message, and reducing the difficulty of next task, can be activated to re-engage him or her to continue learning. Although not rigorously tested, the production of the ITS that incorporates adaptive interface for individual learner and the analytical feedback to the examiners, based on learner’s mouse and keystroke dynamics data, has achieved the objectives of the research.

REFERENCES

[1] J. Sharples, M., Adams, A., Alozie, N., Ferguson, R., FitzGerald, E., Gaved, M., McAndrew, P., Means, B., Remold and L. Rienties, B., Roschelle, J., Vogt, K., Whitelock, D. & Yarnall, "Innovating Pedagogy 2015," Milton Keyne, 2015.

[2] Y. Rogers, H. Sharp, and J. Preece, *Interaction Design: Beyond Human-Computer Interaction*, 2nd ed. West Sussex: John Wiley & Sons Ltd, 2007.

[3] A. Zafar and N. Ahmad, "Towards Adaptive e-learning: Technological Challenges and Enabling Technologies," in *Annual IEEE India Conference*, 2006, pp. 1–6.

[4] F. Trif, C. Lemnaru, and R. Potolea, "Identifying the user typology for adaptive e-learning systems," in *IEEE International Conference on Automation Quality and Testing Robotics (AQTR)*, , 2010, vol. 3, pp. 1–6.

[5] Blackboard, "Item Analysis," *Blackboard Help*, 2016. [Online]. Available: https://en-us.help.blackboard.com/Learn/9.1_2014_04/Instructor/110_Tests_Surveys_Pools/140_Item_Analysis. [Accessed: 25-Mar-2016].

[6] Moodle, "Quiz statistics report," *Moodle Docs 3.0*, 2016. [Online]. Available: https://docs.moodle.org/30/en/Quiz_statistics_report. [Accessed: 25-Mar-2016].

[7] B. Kort, R. Reilly, and R. W. Picard, "An affective model of interplay between emotions and learning: reengineering educational pedagogy-building a learning companion," in *IEEE International Conference on Advanced Learning Technologies*, 2001, pp. 43–46.

[8] D. Watson, L. A. Clark, and A. Tellegen, "Development and validation of brief measures of positive and negative affect: the PANAS scales.," *J. Pers. Soc. Psychol.*, vol. 54, no. 6, pp. 1063-- 1070, 1988.

[9] J. A. Russel, "A Circumplex Model of Affect," *J. Pers. Soc. Psychol.*, vol. 39, no. 6, pp. 1161–1178, 1980.

[10] J. A. LePine, M. A. LePine, and C. L. Jackson, "Challenge and hindrance stress: relationships with exhaustion, motivation to learn, and learning performance.," *J. Appl. Psychol.*, vol. 89, no. 5, p. 883, 2004.

[11] R. Duncko, B. Cornwell, L. Cui, K. R. Merikangas, and C. Grillon, "Acute exposure to stress improves performance in trace eyeblink conditioning and spatial learning tasks in healthy men," *Learn. Mem.*, vol. 14, no. 5, pp. 329–335, 2007.

[12] H. Selye, *The stress in life*. McGraw-Hill, 1956.

[13] R. S. Lazarus and S. Folkman, *Stress, Appraisal, and Coping*. Springer Publishing Company, 1984.

[14] OxfordDictionaries, "Stress," *Oxford University Press*, 2016. [Online]. Available: http://www.oxforddictionaries.com/definition/english/stress. [Accessed: 30-Mar-2016].

[15] AIS, "What is Stress?," *The Americal Institute of Stress*, 2016. [Online]. Available: http://www.stress.org/what-is-stress/. [Accessed: 30-Mar-2016].

[16] G. Hajcak, J. P. Dunning, and D. Foti, "Neural response to emotional pictures is unaffected by concurrent task difficulty: an event-related potential study.," *Behav. Neurosci.*, vol. 121, no. 6, p. 1156, 2007.

[17] R. Crandall, "Validation of self-report measures using ratings by others," *Sociol. Methods Res.*, vol. 4, no. 3, pp. 380–400, 1976.

[18] J. Sweller, "Cognitive load theory, learning difficulty, and instructional design," *Learn. Instr.*, vol. 4, no. 4, pp. 295–312, 1994.

[19] Y. Wang, S. Patel, and D. Patel, "The cognitive process and formal models of human attentions," *Int. J. Softw. Sci. Comput. Intell.*, vol. 5, no. 1, pp. 32–50, 2013.

[20] P. A. Kirschner, "Cognitive load theory: Implications of cognitive load theory on the design of learning," *Learn. Instr.*, vol. 12, no. 1, pp. 1–10, 2002.

[21] F. Paas, A. Renkl, and J. Sweller, "Cognitive load theory: Instructional implications of the interaction between information structures and cognitive architecture," *Instr. Sci.*, vol. 32, no. 1, pp. 1–8, 2004.

[22] Y. Wang, "On the Cognitive Processes of Human Perception with Emotions, Motivations, and Attitudes," *Int. J. Cogn. Informatics Nat. Intell.*, vol. 1, no. 4, pp. 1–13, 2007.

[23] W. Szwoch, "Using physiological signals for emotion recognition," in *Human System Interaction (HSI)*, 2013, pp. 556–561.

[24] A. Anastasi, *Psychological testing* . Macmillan, 1954.

[25] R. A. Karasek, "Job demands, job decision latitude and mental strain: implications for job design," *Adm. Sci. Q.*, no. 24, pp. 285–308, 1979.

[26] A. E. Rijk, P. M. Le Blanc, W. B. Schaufeli, and J. Jonge, "Active coping and need for control as moderators of the job demand--control model: Effects on burnout," *J. Occup. Organ. Psychol.*, vol. 71, no. 1, pp. 1–18, 1998.

[27] P. Ekman and W. V Friesen, "Facial action coding system: A technique for the measurement of

facial movement. Palo Alto.” CA: Consulting Psychologists Press, 1978.
[28] P. Branco, P. Firth, L. M. Encarnação, and P. Bonato, “Faces of emotion in human-computer interaction,” *Work*, pp. 1236–1239, 2005.
[29] H. Crawford, “Keystroke dynamics: Characteristics and opportunities,” in *Eighth Annual International Conference on Privacy Security and Trust (PST)*, 2010, pp. 205–212.
[30] P. Zimmermann, S. Guttormsen, B. Danuser, and P. Gomez, “Affective Computing – Measuring Mood with Mouse and Keyboard,” *Int. J. Occup. Saf. Ergon.*, vol. 9, no. 4, pp. 2–5, 2003.
[31] G. Tsoulouhas, D. Georgiou, and A. Karakos, “Detection of Learnerś Affetive State Based on Mouse Movements,” *J. Comput.*, vol. 3, no. 11, pp. 9–18, 2011.
[32] A. Landowska, M. Szwoch, W. Szwoch, M. R. R. Wróbel, A. Kołakowska, and A. Kolakowska, “Emotion Recognition and Its Applications,” in *Human-Computer Systems Interaction: Backgrounds and Applications 3*, Springer, 2014, pp. 51–62.
[33] J. R. M. Filho and E. O. Freire, “On the equalization of keystroke timing histograms,” *Pattern Recognit. Lett.*, vol. 27, no. 13, pp. 1440–1446, 2006.
[34] J. L. Plass, R. Moreno, and R. Brünken, *Cognitive load theory*. Cambridge University Press, 2010.
[35] Y. M. Lim, A. Ayesh, and K. N. Chee, “Socio-demographic Differences in the Perceptions of Learning Management System (LMS) Design,” *Int. J. Softw. Eng. Appl.*, vol. 4, no. 5, pp. 15–35, 2013.
[36] S. L. Beilock and G. Ramirez, “On the Interplay of Emotion and Cognitive Control: Implications for Enhancing Academic Achievement,” *Psychol. Learn. Motiv. Res. Theory*, vol. 55, p. 137, 2011.
[37] D. Westen, *Psychology. Mind, Brain, & Culture*, 2nd ed. New York: Second Edition. John Wiley & Sons., 1999.
[38] M. Pantic, A. Pentland, A. Nijholt, and T. Huang, “Human Computing and Machine Understanding of Human Behavior: A Survey,” *Artifical Intell. Hum. Comput.*, vol. 4451, pp. 47–71, 2006.
[39] Y. Wang, Y. Wang, S. Patel, and D. Patel, “A layered reference model of the brain (LRMB),” *Syst. Man, Cybern. Part C Appl. Rev. IEEE Trans.*, vol. 36, no. 2, pp. 124–133, 2006.
[40] Y. Wang, “The OAR Model of Neural Informatics for Internal Knowledge Representation in the Brain,” *Int. J. Cogn. Informatics Nat. Intell.*, vol. 1, no. 3, pp. 66–77, 2007.
[41] H. F. O’Neil and C. D. Spielberger, *Cognitive and affective learning strategies*. Academic Pr, 1979.
[42] J. Anania, *The effects of quality of instruction on the cognitive and affective learning of students*. University of Chicago, 1981.
[43] R. P. Sloan, J. B. Korten, and M. M. Myers, “Components of heart rate reactivity during mental arithmetic with and without speaking,” *Physiol. Behav.*, vol. 50, no. 5, pp. 1039–1045, Nov. 1991.
[44] R. W. Picard, S. Papert, W. Bender, B. Blumberg, C. Breazeal, D. Cavallo, T. Machover, M. Resnick, D. Roy, and C. Strohecker, “Affective learning-a manifesto,” *BT Technol. J.*, vol. 22, no. 4, pp. 253–269, 2004.
[45] L. Shen, M. Wang, and R. Shen, “Affective e-Learning: Using‘ Emotional’ Data to Improve Learning in Pervasive Learning Environment.,” *Educ. Technol. Soc.*, vol. 12, no. 2, pp. 176–189, 2009.
[46] A. Landowska, “Affective computing and affective learning--methods, tools and prospects,” *Stara strona magazynu EduAkcja*, vol. 5, no. 1, 2013.
[47] S. Ssemugabi, R. de Villiers, R. D. Villiers, and R. de Villiers, “A comparative study of two usability evaluation methods using a web-based e-learning application,” in *Proceedings of the 2007 annual research conference of the South African institute of computer scientists and information technologists on IT research in developing countries*, 2007, pp. 132–142.
[48] K. O’Regan, “Emotion and E-learning,” *J. Asynchronous Learn. Netw.*, vol. 7, no. 3, pp. 78–92, 2003.
[49] J. S. Eccles and A. Wigfield, “Motivational beliefs, values, and goals,” *Annu. Rev. Psychol.*, vol. 53, no. 1, pp. 109–132, 2002.
[50] R. Sj. Baker, S. K. D’Mello, M. M. T. Rodrigo, and A. C. Graesser, “Better to be frustrated than bored: The incidence, persistence, and impact of learners’ cognitive--affective states during interactions with three different computer-based learning environments,” *Int. J. Hum. Comput. Stud.*, vol. 68, no. 4, pp. 223–241, 2010.
[51] R. W. Picard, *Affective computing / Rosalind W. Picard*. Cambridge (MA) : MIT Press, 1997.
[52] J. Sweller, P. Ayres, and S. Kalyuga, *Cognitive load theory*, vol. 1. Springer, 2011.
[53] R. Bixler, S. D’Mello, F. Hall, N. Dame, and S. D. Mello, “Detecting boredom and engagement during writing with keystroke analysis, task appraisals, and stable traits,” in *Proceedings of the*

2013 international conference on Intelligent user interfaces, 2013, pp. 225–234.
[54] G. Tsoulouhas, D. Georgiou, and A. Karakos, “Detection of Learner’s Affetive State Based on Mouse Movements,” *J. Comput.*, vol. 3, no. 11, pp. 9–18, 2011.
[55] L. M. Vizer, L. Zhou, and A. Sears, “Automated stress detection using keystroke and linguistic features: An exploratory study,” *Int. J. Hum. Comput. Stud.*, vol. 67, no. 10, pp. 870–886, 2009.
[56] S. Scotti, M. Mauri, R. Barbieri, B. Jawad, S. Cerutti, L. Mainardi, E. N. Brown, and M. A. Villamira, “Automatic Quantitative Evaluation of Emotions in E-learning Applications,” in *Engineering in Medicine and Biology Society, 2006. EMBS ’06. 28th Annual International Conference of the IEEE*, 2006, pp. 1359–1362.
[57] W. Liao, W. Zhang, Z. Zhu, and Q. Ji, “A Real-Time Human Stress Monitoring System Using Dynamic Bayesian Network,” in *Proceedings of the 2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, 2005, p. 70.
[58] C. Puri, L. Olson, I. Pavlidis, J. Levine, and J. Starren, “StressCam: Non-contact Measurement of Users’ Emotional States through Thermal Imaging,” in *CHI05 extended abstracts on Human factors in computing systems*, 2005, pp. 1725–1728.
[59] A. H. Azni, A. A. Baker, N. Shah, and H. A. Hamid, “Factors influencing knowledge sharing in higher learning,” 2010, pp. 1606–1609.
[60] J. Kalbach, *Designing web Navigation*. Sebastopol: O’Reilly Media, 2007.
[61] R. H. Kay and S. Loverock, “Assessing emotions related to learning new software: The computer emotion scale,” *Comput. Human Behav.*, vol. 24, no. 4, pp. 1605–1623, 2008.
[62] J. Tidwell, *Designing Interfaces*, 2nd ed. Sebastopol: O’Reilly Media, 2011.
[63] J. Lazar, K. Bessiere, I. Ceaparu, J. Robinson, and B. Shneiderman, “Help! I’m Lost: User Frustration in web Navigation,” *web Navig.*, vol. 1, no. 3, pp. 18–26, 2003.
[64] M. Jenkins, T. Browne, and R. Walker, “A longitudinal perspective between March 2001, March 2003 and March 2005 for higher education in the United Kingdom,” 2005.
[65] T. Browne, R. Hewitt, M. Jenkins, and R. Walker, “2008 survey of Technology Enhanced Learning For Higher Education in the UK,” 2008.
[66] C. Bee and R. Madrigal, “Outcomes are in the eye of the beholder: The influence of affective dispositions on disconfirmation emotions, outcome satisfaction, and enjoyment,” *J. Media Psychol. Theor. Methods, Appl.*, vol. 24, no. 4, pp. 143–153, 2012.
[67] U. R. Hülsheger, H. J. E. M. Alberts, A. Feinholdt, and J. W. B. Lang, “Benefits of Mindfulness at Work: The Role of Mindfulness in Emotion Regulation, Emotional Exhaustion, and Job Satisfaction,” *J. Appl. Psychol.*, vol. 98, no. 2, pp. 310–325, Mar. 2013.
[68] M. P. Penna, V. Stara, and M. De Rose, “The failure of e-learning: why should we use a learner centred design,” *J. e-Learning Knowl. Soc.*, vol. 3, no. 2, 2009.
[69] N. Avouris, N. Tselios, C. Fidas, and E. Papachristos, “website evaluation: A usability-based perspective,” in *Advances in Informatics*, Springer, 2003, pp. 217–231.
[70] S. Cohen, R. C. Kessler, L. U. Gordon, S. M. Monroe, and J. M. Kelley, *Measuring Stress: A Guide for Health and Social Scientists*. Oxford University Press USA, 1997.
[71] P. Ekman and W. Friesen, “Facial Action Coding System: A Technique for the Measurement of Facial Movement,” *Consult. Psychol. Press*, 1978.
[72] J. R. Cowell and A. Ayesh, “Extracting subtle facial expression for emotional analysis,” in *IEEE International Conference on Systems, Man and Cybernetics*, 2004, vol. 1, pp. 677–681.
[73] W. L. Cheong, C. M. Char, Y. C. Lim, S. Lim, and S. W. Khor, “Building a computation savings Real-Time Face Detection and Recognition System,” in *ICSPS 2010 - Proceedings of the 2010 2nd International Conference on Signal Processing Systems*, 2010, vol. 1, pp. V1815–V1819.
[74] R. Cowie, E. Douglas-Cowie, N. Tsapatsoulis, G. Votsis, S. Kollias, W. Fellenz, and J. G. Taylor, “Emotion recognition in human-computer interaction,” *Signal Process. Mag. IEEE*, vol. 18, no. 1, pp. 32–80, 2001.
[75] R. Hazlett and J. Benedek, “Measuring emotional valence to understand the user’s experience of software,” *Int. J. Hum. Comput. Stud.*, vol. 65, no. 4, pp. 306–314, 2007.
[76] P. Zaharias and A. Poylymenakou, “Developing a Usability Evaluation Method for e-Learning Applications : Beyond Functional Usability,” *Int. J. Hum. Comput. Interact.*, vol. 25, no. 1, pp. 75–98, 2009.
[77] T. Lindberg, R. Näsänen, and K. Müller, “How age affects the speed of perception of computer icons,” *Displays*, vol. 27, no. 4–5, pp. 170–177, 2006.
[78] D. Dhar and P. Yammiyavar, “Design Approach for E-learning Systems: Should it be User Centered or Learner Centered,” in *IEEE Fourth International Conference on Technology for Education (T4E)*, , 2012, pp. 239–240.
[79] T. Carey, K. Harrigan, A. Palmer, and J. Swallow, “Scaling up a learning technology strategy: Supporting student/faculty teams in learner- centred design,” *ALTJ*, vol. 7, no. 2, pp. 15–26, 1999.

[80] B. Han, X.-W. Hao, and C.-F. Liu, “The design and implementation of user behavior mining in E-learning system,” in *International Conference on Automatic Control and Artificial Intelligence (ACAI 2012)*, 2012, pp. 2078–2081.
[81] G. Savic and Z. Konjović, “Learning style based personalization of SCORM e-learning courses,” in *SISY ’09. 7th International Symposium on Intelligent Systems and Informatics*, 2009, pp. 349–353.
[82] T. Swinke, “A unique, culture-aware, personalized learning environment,” in *15th International Conference on Interactive Collaborative Learning (ICL)*, 2012, pp. 1–7.
[83] R. Zhou and K. Rechert, “Personalization for Location-Based E-Learning,” in *NGMAST ’08. The Second International Conference on Next Generation Mobile Applications, Services and Technologies*, 2008, pp. 247–253.
[84] A. Al-Dujaily and H. Ryu, “A Study on Personality in Designing Adaptive e-Learning Systems,” in *ICALT ’08. Eighth IEEE International Conference on Advanced Learning Technologies*, 2008, pp. 136–138.
[85] E. Soloway, M. Guzdial, and K. E. Hay, “Learner-centered design: The challenge for HCI in the 21st century,” *interactions*, vol. 1, no. 2, pp. 36–48, 1994.
[86] Q. Gu and T. Sumner, “Support Personalization in Distributed E-Learning Systems through Learner Modeling,” in *ICTTA ’06. 2nd International Conference on Information and Communication Technologies*, 2006, vol. 1, pp. 610–615.
[87] P. Q. Dung and A. M. Florea, “An Architecture and a Domain Ontology for Personalized Multi-agent e-Learning Systems,” in *Third International Conference on Knowledge and Systems Engineering (KSE)*, 2011, pp. 181–185.
[88] M. V Judy, U. Krishnakumar, and A. G. H. Narayanan, “Constructing a personalized e-learning system for students with autism based on soft semantic web technologies,” in *IEEE International Conference on Technology Enhanced Education (ICTEE)*, 2012, pp. 1–5.
[89] M. K. Khribi, M. Jemni, and O. Nasraoui, “Automatic Recommendations for E-Learning Personalization Based on web Usage Mining Techniques and Information Retrieval,” in *ICALT ’08. Eighth IEEE International Conference on Advanced Learning Technologies*, 2008, pp. 241–245.
[90] N. Pandey, S. Sahu, R. K. Tyagi, and A. Dwivedi, “Learning algorithms For intelligent agents based e-learning system,” in *Advance Computing Conference (IACC), 2013 IEEE 3rd International*, 2013, pp. 1034–1039.
[91] J. Henkel and L. Bauer, “What is adaptive computing?,” *SIGDA Newsl.*, vol. 40, no. 5, pp. 1, 2010.
[92] A. C. Moorman and J. Cates D.M., “A complete development environment for image processing applications on adaptive computing systems,” in *ICASSP ’99. IEEE International Conference on Acoustics, Speech, and Signal Processing*, 1999, vol. 4, pp. 2159–2162 vol.4.
[93] A. A. Firte, C. V Bratu, and C. Cenan, “Intelligent component for adaptive E-learning systems,” in *ICCP 2009. IEEE 5th International Conference on Intelligent Computer Communication and Processing*, 2009, pp. 35–38.
[94] S. V Kolekar, S. G. Sanjeevi, and D. S. Bormane, “Learning style recognition using Artificial Neural Network for adaptive user interface in e-learning,” in *IEEE International Conference on Computational Intelligence and Computing Research (ICCIC)*, 2010, pp. 1–5.
[95] S. Hammami, H. Mathkour, and E. A. Al-Mosallam, “A multi-agent architecture for adaptive E-learning systems using a blackboard agent,” in *ICCSIT 2009. 2nd IEEE International Conference on Computer Science and Information Technology*, 2009, pp. 184–188.
[96] M. Leontidis, C. Halatsis, and M. Grigoriadou, “MENTORing Affectively the Student to Enhance his Learning,” in *ICALT 2009. Ninth IEEE International Conference on Advanced Learning Technologies*, 2009, pp. 455–459.
[97] K. Chatzara, C. Karagiannidis, and D. Stamatis, “Students Attitude and Learning Effectiveness of Emotional Agents,” in *IEEE 10th International Conference on Advanced Learning Technologies (ICALT)*, 2010, pp. 558–559.
[98] I. Hatzilygeroudis, C. Koutsojannis, and N. Papachristou, “Adding adaptive assessment capabilities to an e-learning system,” in *SMAP ’06. First International Workshop on Semantic Media Adaptation and Personalization*, 2006, pp. 68–73.
[99] S. U. Khalid, A. Basharat, A. A. Shahid, and S. Hassan, “An adaptive E-learning Framework to supporting new ways of teaching and learning,” in *ICICT ’09. International Conference on Information and Communication Technologies*, 2009, pp. 300–306.
[100] A. Kardan and H. Monkaresi, “Developing a Novel Framework for Effective Use of Implicit Feedback in Adaptive e-Learning,” in *ITNG 2008. Fifth International Conference on Information Technology: New Generations*, 2008, pp. 955–960.
[101] K. Takano and K. F. Li, “An Adaptive e-Learning Recommender Based on User’s web-Browsing

Behavior," in *International Conference on P2P, Parallel, Grid, Cloud and Internet Computing (3PGCIC)*, 2010, pp. 123–131.
[102] S. Garruzzo, D. Rosaci, and G. M. L. Same, "MASHA-EL: A Multi-Agent System for Supporting Adaptive E-Learning," in *ICTAI 2007. 19th IEEE International Conference on Tools with Artificial Intelligence*, 2007, vol. 2, pp. 103–110.
[103] H. Mahdi and S. S. Attia, *MASCE: A Multi-Agent System for Collaborative E-Learning*. IEEE, 2008. *In* AICCSA 2008. IEEE/ACS International Conference on Computer Systems and Applications, 2008. pp. 925-926
[104] Z. Liu and Y. Liu, "An adaptive personalized E-learning model based on agent technology," *WTOS*, vol. 7, no. 12, pp. 1443–1452, 2008.
[105] S. S. AlZahrani, A. Ayesh, and H. Zedan, "Multi-agent based dynamic e-learning environment.," *Int. J. Inf. Technol. web Eng.*, vol. 4, no. 2, pp. 61–77, 2009.
[106] C. Jing and L. Quan, "A Complex Adaptive E-Learning Model Based on Semantic web Services," in *KAM '08.International Symposium on Knowledge Acquisition and Modeling*, 2008, pp. 555–559.
[107] R. A. Calvo and S. D. Mello, "Affect Detection : An Interdisciplinary Review of Models , Methods , and Their Applications," *IEEE Trans. Affect. Comput.*, vol. 1, no. 1, pp. 18–37, 2010.
[108] C. E. Izard, "Basic emotions, natural kinds, emotion schemas, and a new paradigm," *Perspect. Psychol. Sci.*, vol. 2, no. 3, pp. 260–280, 2007.
[109] H. Selye, *Stress in health and disease*. Butterworth-Heinemann, 1946.
[110] L. E. Bourne Jr and R. A. Yaroush, *Stress and cognition: A cognitive psychological perspective*, National Aeronautics and Space Administration, 2003.
[111] J. V. Johnson and E. M. Hall, "Job strain, work place social support and cardiovascular disease: a cross sectional study of a random sample of the Swedish working population," *Am. J. Public Health*, vol. 78, no. 10, pp. 1336–1342, 1988.
[112] R. S. Lazarus, "Theory-Based Stress Measurement," *Psychol. Inq.*, vol. 1, no. 1, pp. 3–13, 1990.
[113] J. Hernandez, P. Paredes, A. Roseway, and M. Czerwinski, "Under pressure: sensing stress of computer users," in *Proceedings of the 32nd annual ACM conference on Human factors in computing systems*, 2014, pp. 51–60.
[114] P. Ekman, "An argument for basic emotions," *Cogn. Emot.*, vol. 6, no. 3–4, pp. 169–200, 1992.
[115] C. Busso, Z. Deng, S. Yildirim, M. Bulut, C. M. Lee, A. Kazemzadeh, S. Lee, U. Neumann, and S. Narayanan, "Analysis of emotion recognition using facial expressions, speech and multimodal information," in *Proceedings of the 6th international conference on Multimodal interfaces*, 2004, pp. 205–211.
[116] Z. Zeng, M. Pantic, G. I. Roisman, and T. S. Huang, "A survey of affect recognition methods: Audio, visual, and spontaneous expressions," *Pattern Anal. Mach. Intell. IEEE Trans.*, vol. 31, no. 1, pp. 39–58, 2009.
[117] S. D'Mello, R. W. Picard, and A. Graesser, "Toward an affect-sensitive AutoTutor," *IEEE Intell. Syst.*, no. 4, pp. 53–61, 2007.
[118] N. M. Yusoff and S. S. Salim, "SCOUT and affective interaction design: Evaluating physiological signals for usability in emotional processing," in *2nd International Conference on Computer Engineering and Technology (ICCET)*, 2010, vol. 1, pp. V1-201-V1-205.
[119] J. Zhai, A. B. Barreto, C. Chin, and C. Li, "User stress detection in human-computer interactions.," 2005.
[120] M. Heiden, E. Lyskov, M. Djupsjöbacka, F. Hellström, and A. G. Crenshaw, "Effects of time pressure and precision demands during computer mouse work on muscle oxygenation and position sense," *Eur. J. Appl. Physiol.*, vol. 94, no. 1, pp. 97–106, 2005.
[121] I. J. Bennett, M. A. Motes, N. K. Rao, and B. Rypma, "White matter tract integrity predicts visual search performance in young and older adults," *Neurobiol. Aging*, vol. 33, no. 2, pp. 433--e21, Feb. 2012.
[122] C. Setz, B. Arnrich, J. Schumm, R. La Marca, G. Troster, and U. Ehlert, "Discriminating Stress From Cognitive Load Using a Wearable EDA Device," *Inf. Technol. Biomed. IEEE Trans.*, vol. 14, no. 2, pp. 410–417, Mar. 2010.
[123] B. Pang and L. Lee, "Opinion mining and sentiment analysis," *Found. trends Inf. Retr.*, vol. 2, no. 1–2, pp. 1–135, 2008.
[124] J. Lazar, J. H. Feng, and H. Hochheiser, *Research Methods in Human-Computer Interaction*. West Sussex: John Wiley & Sons Ltd, 2010.
[125] A. Cavoukian, "Privacy by design," *Rep. Inf. Priv. Comm. Ontario, Canada*, 2012.
[126] W. Maehr, *eMotion: Estimation of User's Emotional State by Mouse Motions*. VDM Verlag, 2008.
[127] B. Schuller, M. Lang, and G. Rigoll, "Multimodal emotion recognition in audiovisual communication," in *ICME'02. IEEE International Conference on Multimedia and Expo*, 2002,

vol. 1, pp. 745–748.
[128] C. Epp, M. Lippold, and R. L. Mandryk, “Identifying emotional states using keystroke dynamics,” *Proc. 2011 Annu. Conf. Hum. factors Comput. Syst. - CHI ’11*, p. 715, 2011.
[129] A. Alhothali, “Modeling User Affect Using Interaction Events,” University of Waterloo, 2011.
[130] H. Lee, Y. S. Choi, S. Lee, and I. P. Park, “Towards unobtrusive emotion recognition for affective social communication,” in *Consumer Communications and Networking Conference (CCNC)*, 2012, pp. 260–264.
[131] P. Khanna and M. Sasikumar, “Recognising emotions from keyboard stroke pattern,” *Int. J. Comput. Appl.*, vol. 11, no. 9, pp. 8887–8975, 2010.
[132] H.-R. Lv, Z.-L. Lin, W.-J. Yin, and J. Dong, “Emotion recognition based on pressure sensor keyboards,” in *IEEE International Conference on Multimedia and Expo*, 2008, pp. 1089–1092.
[133] P. Shukla and R. Solanki, “web Based Keystroke Dynamics Application for Identifying Emotional State,” *Int. J. Adv. Res. Comput. Commun. Eng.*, vol. 2, no. 11, pp. 4489–4493, 2013.
[134] A. Kolakowska, “A review of emotion recognition methods based on keystroke dynamics and mouse movements,” in *HSI*, 2013, pp. 548–555.
[135] C. Shen, Z. Cai, and X. Guan, “Continuous authentication for mouse dynamics: A pattern-growth approach,” in *42nd Annual IEEE/IFIP International Conference on Dependable Systems and Networks (DSN)*, 2012, pp. 1–12.
[136] C. Shen, Z. Cai, X. Guan, and J. Cai, “A hypo-optimum feature selection strategy for mouse dynamics in continuous identity authentication and monitoring,” in *IEEE International Conference on Information Theory and Information Security (ICITIS)*, 2010, pp. 349–353.
[137] F. Bergadano, D. Gunetti, and C. Picardi, “User authentication through keystroke dynamics,” *ACM Trans. Inf. Syst. Secur.*, vol. 5, no. 4, pp. 367–397, Nov. 2002.
[138] D. Chudá and P. Krátky, “Mouse Usage Biometrics in eLearning Systems: Detection of Impersonation and User Profiling,” *Int. J. Hum. Cap. Inf. Technol. Prof.*, vol. 6, no. 1, pp. 39–50, Jan. 2015.
[139] L. M. Vizer, “Detecting cognitive and physical stress through typing behavior,” *Proc. 27th Int. Conf. Ext. Abstr. Hum. factors Comput. Syst. CHI EA 09*, p. 3113, 2009.
[140] P. Zimmermann, P. Gomez, B. Danuser, and S. Schär, “Extending usability: putting affect into the user-experience,” *Proc. Nord.*, pp. 27–32, 2006.
[141] P. Khanna, “Recognising Emotions from Keyboard Stroke Pattern,” vol. 11, no. 9, pp. 1–5, 2010.
[142] D. Gunetti and C. Picardi, “Keystroke analysis of free text. ACM Transactions on Information and System Security,” vol. 8, no. 3, pp. 312–347, 2005.
[143] N. Harun, W. L. Woo, and S. S. Dlay, “Performance of keystroke biometrics authentication system using artificial neural network (ANN) and distance classifier method,” in *International Conference on Computer and Communication Engineering (ICCCE)*, 2010, pp. 1–6.
[144] S. Salmeron-Majadas, O. C. Santos, and J. G. Boticario, “An evaluation of mouse and keyboard interaction indicators towards non-intrusive and low cost affective modeling in an educational context,” *18th Int. Conf. Knowl. Based Intell. Inf. Eng. Syst. - KES2014*, vol. 35, pp. 691–700, 2014.
[145] K. Revett, F. Gorunescu, M. Gorunescu, M. Ene, S. Tenreiro De Magalhaes, D. Santos, M. Henrique, S. Magalhaes, and H. Santos, “A machine learning approach to keystroke dynamics based user authentication,” *Int. J. Electron. Secur. Digit. Forensics*, vol. 1, no. 1, p. 55, 2007.
[146] S. Z. Syed Idrus, E. Cherrier, C. Rosenberger, and P. Bours, “Soft biometrics for keystroke dynamics: Profiling individuals while typing passwords,” *Comput. Secur.*, vol. 45, pp. 147–155, 2014.
[147] J. C. Checco, “Keystroke Dynamics and Corporate Security,” *WSTA Ticker magazine*, 2006.
[148] D. Gunetti and C. Picardi, “Keystroke analysis of free text,” *ACM Trans. Inf. Syst. Secur.*, vol. 8, no. 3, pp. 312–347, 2005.
[149] P. Kang and S. Cho, “Keystroke dynamics-based user authentication using long and free text strings from various input devices,” *Inf. Sci. (Ny).*, vol. 308, pp. 72–93, 2014.
[150] J.-D. Marsters, “Keystroke dynamics as a biometric,” University of Southampton, 2009.
[151] L. C. F. Araujo, J. Sucupira L.H.R., M. G. Lizarraga, L. L. Ling, and J. B. T. Yabu-Uti, “User authentication through typing biometrics features,” *Signal Process. IEEE Trans.*, vol. 53, no. 2, pp. 851–855, 2005.
[152] M. S. Obaidat and B. Sadoun, “Verification of computer users using keystroke dynamics.,” *IEEE Trans. Syst. man Cybern. Part B Cybern. a Publ. IEEE Syst. Man Cybern. Soc.*, vol. 27, no. 2, pp. 261–269, 1997.
[153] J. A. Robinson and V. M. Liang, “Computer User Verification Using Login String Keystroke Dynamics,” *IEEE Trans. Syst. man Cybern. A Syst. Humans*, vol. 28, no. 2, pp. 236–241, 1998.
[154] P. S. Teh, S. Yue, and A. B. J. Teoh, “Improving keystroke dynamics authentication system via multiple feature fusion scheme,” in *International Conference on Cyber Security, Cyber Warfare*

and Digital Forensic (CyberSec), 2012, pp. 277–282.
[155] R. Giot, M. El-Abed, and C. Rosenberger, “web-based benchmark for keystroke dynamics biometric systems: a statistical analysis,” in *Eighth International Conference on Intelligent Information Hiding and Multimedia Signal Processing (IIH-MSP)*, 2012, pp. 11–15.
[156] F. Monrose and A. Rubin, “Authentication via keystroke dynamics,” in *Proc. of the 4th ACM Conf. on Computer and Communications Security*, 1997, pp. 48–56.
[157] T. Shimshon, R. Moskovitch, L. Rokach, and Y. Elovici, “Clustering di-graphs for continuously verifying users according to their typing patterns,” in *IEEE 26th Convention of Electrical and Electronics Engineers in Israel (IEEEI)*, 2010, pp. 445–449.
[158] Y. Nakkabi, I. Traore, and A. A. E. Ahmed, “Improving Mouse Dynamics Biometric Performance Using Variance Reduction via Extractors With Separate Features,” *Syst. Man Cybern. Part A Syst. Humans, IEEE Trans.*, vol. 40, no. 6, pp. 1345–1353, 2010.
[159] B. Shneiderman and C. Plaisant, *Designing the User Interface*, 4th ed. Pearson Education, Inc, 2005.
[160] C. Shen, Z. Cai, X. Guan, H. Sha, and J. Du, “Feature analysis of mouse dynamics in identity authentication and monitoring,” in *Proceedings of the IEEE international conference on Communications*, 2009, pp. 673–677.
[161] M. Pusara and C. E. Brodley, “User re-authentication via mouse movements,” in *Proceedings of the 2004 ACM workshop on Visualization and data mining for computer security*, 2004, pp. 1–8.
[162] J. Wahlström, M. Hagberg, P. W. Johnson, J. Svensson, and D. Rempel, “Influence of time pressure and verbal provocation on physiological and psychological reactions during work with a computer mouse.,” *Eur. J. Appl. Physiol.*, vol. 87, no. 3, pp. 257–263, 2002.
[163] F. Monrose and A. D. Rubin, “Keystroke dynamics as a biometric for authentication,” *Futur. Gener. Comput. Syst.*, vol. 16, no. 4, pp. 351–359, 2000.
[164] A. A. E. Ahmed and I. Traore, “A New Biometric Technology Based on Mouse Dynamics,” *Dependable Secur. Comput. IEEE Trans.*, vol. 4, no. 3, pp. 165–179, 2007.
[165] P. Bours and C. J. Fullu, “A Login System Using Mouse Dynamics,” in *IIH-MSP ’09. Fifth International Conference on Intelligent Information Hiding and Multimedia Signal Processing*, 2009, pp. 1072–1077.
[166] C. Shen, Z. Cai, X. Guan, Y. Du, and R. Maxion, “User Authentication through Mouse Dynamics,” *Inf. Forensics Secur. IEEE Trans.*, vol. PP, no. 99, p. 1, 2012.
[167] C. Shen, Z. Cai, X. Guan, and J. Wang, “On the effectiveness and applicability of mouse dynamics biometric for static authentication: A benchmark study,” in *5th IAPR International Conference on Biometrics (ICB)*, 2012, pp. 378–383.
[168] C. Shen, Z. Cai, X. Guan, and R. Maxion, “Performance evaluation of anomaly-detection algorithms for mouse dynamics,” *Comput. Secur.*, vol. 45, pp. 156–171, 2014.
[169] C.-C. Lin, C.-C. Chang, and D. Liang, “A New Non-intrusive Authentication Approach for Data Protection Based on Mouse Dynamics,” in *International Symposium on Biometrics and Security Technologies (ISBAST)*, 2012, pp. 9–14.
[170] D. Chudá and P. Krátky, “Usage of computer mouse characteristics for identification in web browsing,” in *Proceedings of the 15th International Conference on Computer Systems and Technologies*, 2014, pp. 218–225.
[171] P. Van Schaik and J. Ling, “The effects of frame layout and differential background contrast on visual search performance in web pages,” *Interact. Comput.*, vol. 13, no. 5, pp. 513–525, 2001.
[172] R. Pearson and P. van Schaik, “The effect of spatial layout of and link colour in web pages on performance in a visual search task and an interactive search task,” *Int. J. Hum. Comput. Stud.*, vol. 59, no. 3, pp. 327–353, 2003.
[173] J. Ling and P. Van Schaik, “The effect of text and background colour on visual search of web pages,” *Displays*, vol. 23, no. 5, pp. 223–230, 2002.
[174] J. Ling and P. Van Schaik, “The influence of font type and line length on visual search and information retrieval in web pages,” *Int. J. Hum. Comput. Stud.*, vol. 64, no. 5, pp. 395–404, 2006.
[175] S. Murugesan, “web application development: Challenges and the role of web engineering,” in *web engineering: modelling and implementing web applications*, G. Rossi, Ed. Springer, 2008, pp. 7–32.
[176] Y. M. Lim, A. Ayesh, M. Stacey, and K. N. Chee, “Designing Learning Management System to Encourage E-Learning Sustainability,” 2013, pp. 76–83.
[177] K.-K. Shieh, M.-T. Chen, and J.-H. Chuang, “Effects of color combination and typography on identification of characters briefly presented on VDTs,” *Int. J. Hum. Comput. Interact.*, vol. 9, no. 2, pp. 169–181, 1997.
[178] I. Humar, T. Turk, others, M. Gradisˇar, and T. Turk, “The impact of color combinations on the legibility of a web page text presented on CRT displays,” *Int. J. Ind. Ergon.*, vol. 38, no. 11, pp.

885–899, 2008.
[179] K.-C. Huang, “Effects of computer icons and figure/background area ratios and color combinations on visual search performance on an LCD monitor,” *Displays*, vol. 29, no. 3, pp. 237–242, 2008.
[180] K.-K. Shieh and M.-T. Chen, “Effects of screen color combination, work-break schedule, and workpace on VDT viewing distance,” *Int. J. Ind. Ergon.*, vol. 20, no. 1, pp. 11–18, 1997.
[181] P. van Schaik and J. Ling, “The effect of link colour on information retrieval in educational intranet use,” *Comput. Human Behav.*, vol. 19, no. 5, pp. 553–564, Sep. 2003.
[182] E. A. Krupinski, J. Johnson, H. Roehrig, J. Nafziger, J. Fan, and J. Lubin, “Use of a human visual system model to predict observer performance with CRT vs LCD display of images,” *J. Digit. Imaging*, vol. 17, no. 4, pp. 258–263, 2004.
[183] M. L. Bernard, B. S. Chaparro, M. M. Mills, and C. G. Halcomb, “Comparing the effects of text size and format on the readibility of computer-displayed Times New Roman and Arial text,” *Int. J. Hum. Comput. Stud.*, vol. 59, no. 6, pp. 823–835, Dec. 2003.
[184] C. B. Mills and L. J. Weldon, “Reading Text from Computer Screens,” *ACM Comput. Surv.*, vol. 19, no. 4, pp. 329–357, 1987.
[185] G. S. E. Briem, “How to arrange text on web pages,” *Comput. Typogr.*, vol. 2, pp. 10–20, 2002.
[186] J. G. Hollands and P. M. Merikle, “Menu organization and user expertise in information search tasks,” *Hum. Factors J. Hum. Factors Ergon. Soc.*, vol. 29, no. 5, pp. 577–586, 1987.
[187] J. E. McDonald, J. D. Stone, and L. S. Liebelt, “Searching for items in menus: The effects of organization and type of target,” in *Proceedings of the Human Factors and Ergonomics Society Annual Meeting*, 1983, vol. 27, no. 9, pp. 834–837.
[188] B. Mehlenbacher, T. M. Duffy, and J. Palmer, “Finding information on a menu: Linking menu organization to the user’s goals,” *Human-Computer Interact.*, vol. 4, no. 3, pp. 231–251, 1989.
[189] E. MacGregor, James and Lee, J. MacGregor, and E. Lee, “Menu search: random or systematic?,” *Int. J. Man. Mach. Stud.*, vol. 26, no. 5, pp. 627–631, 1987.
[190] A. J. Hornof and D. E. Kieras, “Cognitive modeling reveals menu search in both random and systematic,” in *Proceedings of the ACM SIGCHI Conference on Human factors in computing systems*, 1997, pp. 107–114.
[191] J. M. Spool, *web site usability: a designer’s guide*. Morgan Kaufmann, 1999.
[192] A. Dillon, J. Richardson, C. McKnight, E. Grove, and L. Le, “The effects of display size and text splitting on reading lengthy text from screen,” *Behav. Inf. Technol.*, vol. 9, no. 3, pp. 215–227, 1990.
[193] K. O’Hara, A. Sellen, and A. O’Hara, Kenton and Sellen, “A comparison of reading paper and on-line documents,” in *Proceedings of the ACM SIGCHI Conference on Human factors in computing systems*, 1997, pp. 335–342.
[194] W. J. Hansen and C. Haas, “Reading and writing with computers: a framework for explaining differences in performance,” *Commun. ACM*, vol. 31, no. 9, pp. 1080–1089, 1988.
[195] A. Piolat, J.-Y. Roussey, and O. Thunin, “Effects of screen presentation on text reading and revising,” *Int. J. Hum. Comput. Stud.*, vol. 47, no. 4, pp. 565–589, Oct. 1997.
[196] M. C. Dyson and M. Haselgrove, “The influence of reading speed and line length on the effectiveness of reading from screen,” *Int. J. Hum. Comput. Stud.*, vol. 54, no. 4, pp. 585–612, Apr. 2001.
[197] B. Bridgeman, M. Lou Lennon, and A. Jackenthal, “Effects of screen size, screen resolution, and display rate on computer-based test performance,” *Appl. Meas. Educ.*, vol. 16, no. 3, pp. 191–205, 2003.
[198] Y. M. Lim, A. Ayesh, and M. Stacey, “The Effects of Menu Design on Users’ Emotions, Search Performance and Mouse Behaviour,” in *IEEE 13th Int’l Conf. on Cognitive Informatics & Cognitive Computing (ICCI*CC’14)*, 2014, pp. 541–549.
[199] A. M. Owen, K. M. McMillan, A. R. Laird, and E. Bullmore, “N-back working memory paradigm: A meta-analysis of normative functional neuroimaging studies,” *Hum. Brain Mapp.*, vol. 25, no. 1, pp. 46–59, 2005.
[200] I. Imbo and A. Vandierendonck, “Do multiplication and division strategies rely on executive and phonological working memory resources?,” *Mem. Cognit.*, vol. 35, no. 7, pp. 1759–1771, 2007.
[201] A. Weinberg, J. Ferri, and G. Hajcak, “Interactions between Attention and Emotion,” *Handb. Cogn. Emot.*, p. 35, 2013.
[202] Y. M. Lim, A. Ayesh, and M. Stacey, “Detecting cognitive stress from keyboard and mouse dynamics during mental arithmetic,” in *Science and Information Conference, SAI 2014*, 2014, pp. 146--152.
[203] Y. M. Lim, A. Ayesh, and M. Stacey, “Using Mouse and Keyboard Dynamics to Detect Cognitive Stress During Mental Arithmetic,” in *Intelligent Systems in Science and Information 2014*, vol. 591, K. Arai, S. Kapoor, and R. Bhatia, Eds. Switzerland: Springer, 2015, pp. 335–

350.
[204] S. Tobias, T. Abramson, T. Sigmund, and A. Theodore, “Interaction among anxiety, stress, response mode, and familiarity of subject matter on achievement from programmed instruction.,” *J. Educ. Psychol.*, vol. 62, no. 4, p. 357, 1971.
[205] C. Hulme, S. Maughan, and G. D. A. Brown, “Memory for familiar and unfamiliar words: Evidence for a long-term memory contribution to short-term memory span,” *J. Mem. Lang.*, vol. 30, no. 6, pp. 685–701, 1991.
[206] J. E. Davidson and R. J. Sternberg, *The psychology of problem solving*. Cambridge university press, 2003.
[207] G. C. Boechat, J. C. Ferreira, others, and E. Carvalho Filho, “Authentication personal,” in *ICIAS 2007. International Conference on Intelligent and Advanced Systems*, 2007, pp. 254–256.
[208] N. Eswari, S. Sundarapandiyan, P. Vennila, R. Umamaheswari, and G. Jothilakshmi, “Keystroke Biometrics with number-pad input using hybridization of adaboost with random forest,” in *International Conference on Advances in Engineering, Science and Management (ICAESM)*, 2012, pp. 105–109.
[209] R. Giot, C. Rosenberger, and B. Dorizz, “Can Chronological Information be Used as a Soft Biometric in Keystroke Dynamics?,” in *Eighth International Conference on Intelligent Information Hiding and Multimedia Signal Processing (IIH-MSP)*, 2012, pp. 7–10.
[210] T. Shimshon, R. Moskovitch, L. Rokach, and Y. Elovici, “Clustering di-graphs for continuously verifying users according to their typing patterns,” in *IEEE 26th Convention of Electrical and Electronics Engineers in Israel (IEEEI)*, 2010, pp. 445–449.
[211] Y. M. Lim, A. Ayesh, and M. Stacey, “Detecting Emotional Stress during Typing Task with Time Pressure,” in *Science and Information Conference 2014*, 2014, pp. 329--338.
[212] Y. M. Lim, A. Ayesh, and M. Stacey, “The Effects of Typing Demand on Emotional Stress, Mouse and Keystroke Behaviours,” in *Intelligent Systems in Science and Information 2014*, vol. 591, K. Arai, S. Kapoor, and R. Bhatia, Eds. Switzerland: Springer, 2015, pp. 209–225.
[213] E. H. Shortliffe, “MYCIN: Computer-based medical consultations.” Elsevier, New York, 1976.
[214] D. E. Heckerman and E. H. Shortliffe, “From certainty factors to belief networks,” *Artif. Intell. Med.*, vol. 4, no. 1, pp. 35–52, 1992.
[215] E. H. Shortliffe and B. G. Buchanan, “A model of inexact reasoning in medicine,” *Math. Biosci.*, vol. 23, no. 3, pp. 351–379, 1975.
[216] D. Heckerman, “Probabilistic interpretations for MYCIN’s certainty factors,” *arXiv Prepr. arXiv1304.3419*, 2013.
[217] L. V Fausett, *Fundamentals of Neural Networks: Architectures, Algorithms, and Applications*. Prentice-Hall, 1994.
[218] D. Svozil, V. Kvasnicka, and J. Pospichal, “Introduction to multi-layer feed-forward neural networks,” *Chemom. Intell. Lab. Syst.*, vol. 39, no. 1, pp. 43–62, 1997.
[219] H. Ishibuchi and M. Nii, “Fuzzy Neural Networks Techniques and Their Applications,” in *Fuzzy Logic and Expert Systems Applications*, C. T. Leondes, Ed. Academic Press, 1998, pp. 1–56.
[220] J.-S. Jang, “ANFIS: adaptive-network-based fuzzy inference system,” *Syst. Man Cybern. IEEE Trans.*, vol. 23, no. 3, pp. 665–685, 1993.
[221] L. A. Zadeh, “Fuzzy sets,” *Inf. Control*, vol. 8, no. 3, pp. 338–353, 1965.
[222] Mathworks, “Foundations of Fuzzy Logic,” *Mathworks R2016a Documentation*, 2016. [Online]. Available: http://uk.mathworks.com/help/fuzzy/foundations-of-fuzzy-logic.html#bp78l70-2. [Accessed: 06-Apr-2016].
[223] J. Jantzen, *Foundations of Fuzzy Control: A Practical Approach*, 2nd ed. John Wiley & Sons, 2013.
[224] Mathworks, “Comparison of Sugeno and Mamdani Systems,” *Mathworks R2016a Documentation*, 2016. [Online]. Available: http://uk.mathworks.com/help/fuzzy/comparison-of-sugeno-and-mamdani-systems.html. [Accessed: 06-Apr-2016].
[225] E. H. Mamdani and S. Assilian, “An experiment in linguistic synthesis with a fuzzy logic controller,” *Int. J. Man. Mach. Stud.*, vol. 7, no. 1, pp. 1–13, 1975.
[226] L. A. Zadeh, “Outline of a new approach to the analysis of complex systems and decision processes,” *IEEE Trans. Syst. Man Cybern.*, no. 1, pp. 28–44, 1973.
[227] Mathworks, “What Is Mamdani-Type Fuzzy Inference?,” *Mathworks R2016a Documentation*, 2016. [Online]. Available: http://uk.mathworks.com/help/fuzzy/what-is-mamdani-type-fuzzy-inference.html. [Accessed: 06-Apr-2016].
[228] S. N. Sivanandam, S. Sumathi, and S. N. Deepa, *Introduction to Fuzzy Logic using MATLAB*. Springer Berlin Heidelberg, 2006.
[229] Mathworks, “Defuzzification Methods,” *Mathworks R2016a Documentation*, 2016. [Online]. Available: http://uk.mathworks.com/help/fuzzy/examples/defuzzification-methods.html. [Accessed: 06-Apr-2016].

[230] Y. Bai and D. Wang, "Fundamentals of Fuzzy Logic Control—Fuzzy Sets, Fuzzy Rules and Defuzzifications," in *Advanced Fuzzy Logic Technologies in Industrial Applications*, Springer, 2006, pp. 17–36.
[231] P. S. Kumar and B. M. Harif, "Fuzzy Modeling of Perceived Stress, And Cortisol Responses to Awakening Using Distance for Fuzzy Sets," *Int. J. Sci. Res. Publ.*, vol. 4, no. 11, pp. 1–6, 2014.
[232] R. Likert, "A technique for the measurement of attitudes.," *Arch. Psychol.*, vol. 22, no. 140, p. 55, 1932.
[233] B. H. Kantowitz, H. L. Roediger, and D. G. Elmes, *Experimental Psychology*. Cengage Learning, 2008.
[234] F. J. Gravetter and L.-A. B. Forzano, *Research Methods for the Behavioral Sciences*, 5th ed. Cengage Learning, 2015.
[235] N. A. Weiss, *Elementary Statistics*, 6th ed. Addison-Wesley, 2004.
[236] H. Jahankhani, A. Al-Nemrat, and A. Hosseinian-Far, "Cyber Crime and Cyber Terrorism Investigator's Handbook," in *Cyber Crime and Cyber Terrorism Investigator's Handbook*, B. Akhgar, A. Staniforth, and F. Bosco, Eds. Syngress, 2014, pp. 149–164.
[237] B. Venners, "Java's security architecture: An overview of the JVM's security model and a look at its built-in safety features," *JavaWorld*, 1997. [Online]. Available: http://www.javaworld.com/article/2076989/core-java/java-s-security-architecture.html?page=2. [Accessed: 02-Nov-2011].
[238] Windows Dev Center, "GetAsyncKeyState function," *Microsoft MSDN*, 2015. [Online]. Available: https://msdn.microsoft.com/en-us/library/windows/desktop/ms646293(v=vs.85).aspx. [Accessed: 01-Sep-2015].
[239] Windows Dev Center, "GetKeyState function," *Microsoft MSDN*, 2015. [Online]. Available: https://msdn.microsoft.com/en-us/library/windows/desktop/ms646301(v=vs.85).aspx. [Accessed: 01-Sep-2015].
[240] Windows Dev Center, "About Keyboard Input," *Microsoft MSDN*, 2015. [Online]. Available: https://msdn.microsoft.com/en-us/library/windows/desktop/ms646267(v=vs.85).aspx. [Accessed: 01-Sep-2015].
[241] R. Mariappan and B. Parthasarathy, "PROCESSING AND PERFORMANCE OF TEXT FILE FORMAT ON DIFFERENT DATA STORAGE SYSTEMS," *Int. J. Power Control Signal Comput.*, vol. 2, no. 1, pp. 34–37, 2011.
[242] Windows Dev Center, "About Mouse Input," *Microsoft MSDN*, 2015. [Online]. Available: https://msdn.microsoft.com/en-us/library/windows/desktop/ms645601(v=vs.85).aspx. [Accessed: 02-Sep-2015].
[243] S. Teilhet, *Subclassing and hooking with Visual Basic*. O'Reilly Media, Inc., 2001.
[244] Microsoft Support, "Mouse wheel events do not work in the Visual Basic 6.0 IDE," *Microsoft*, 2012. [Online]. Available: https://support.microsoft.com/en-us/kb/837910. [Accessed: 02-Sep-2015].
[245] Processing.org, "Class PApplet," *Processing*, 2015. [Online]. Available: http://processing.org/reference/javadoc/core/processing/core/PApplet.html. [Accessed: 02-Sep-2015].
[246] IBM and IBM Knowledge Center, "GLM Univariate Analysis," *IBM SPSS Statistics Information Center*, 2011. [Online]. Available: http://pic.dhe.ibm.com/infocenter/spssstat/v20r0m0/index.jsp?topic=/com.ibm.spss.statistics.help/idh_glmu.htm.
[247] J. W. Grice and M. Iwasaki, "A truly multivariate approach to MANOVA," *Appl. Multivar. Res.*, vol. 12, no. 3, pp. 199–226, 2007.
[248] IBM Knowledge Center and IBM, "Multivariate General Linear Modeling," *IBM SPSS Statistics Information Center*, 2012. [Online]. Available: http://pic.dhe.ibm.com/infocenter/spssstat/v21r0m0/index.jsp?topic=/com.ibm.spss.statistics.cs/glmm_intro.htm.
[249] R. B. Darlington, *Regression and Linear Models*. McGraw-Hill, 1990.
[250] F. Gravetter and L. Wallnau, *Statistics for The Behavioral Sciences*, 10th ed. Cengage Learning, 2015.
[251] G. M. Sullivan and R. Feinn, "Using effect size-or why the P value is not enough," *J. Grad. Med. Educ.*, vol. 4, no. 3, pp. 279–282, 2012.
[252] J. L. Szalma, J. S. Warm, G. Matthews, W. N. Dember, E. M. Weiler, A. Meier, and F. T. Eggemeier, "Effects of sensory modality and task duration on performance, workload, and stress in sustained attention," *Hum. Factors J. Hum. Factors Ergon. Soc.*, vol. 46, no. 2, pp. 219–233, 2004.
[253] K. McGarry, E. Rovira, and R. Parasuraman, "Effects of task duration and type of automation support on human performance and stress in a simulated battlefield engagement task," in

Proceedings of the Human Factors and Ergonomics Society Annual Meeting, 2003, vol. 47, no. 3, pp. 548–552.
[254] B. Burle and L. Casini, “Dissociation between activation and attention effects in time estimation: implications for internal clock models.,” *J. Exp. Psychol. Hum. Percept. Perform.*, vol. 27, no. 1, p. 195, 2001.
[255] I. S. Penton-Voak, H. Edwards, A. Percival, and J. H. Wearden, “Speeding up an internal clock in humans? Effects of click trains on subjective duration.,” *J. Exp. Psychol. Anim. Behav. Process.*, vol. 22, no. 3, p. 307, 1996.
[256] M. Wittmann, “The inner experience of time.,” *Philos. Trans. R. Soc. Lond. B. Biol. Sci.*, vol. 364, no. 1525, pp. 1955–1967, 2009.
[257] L. H. Peters, E. J. O’Connor, A. Pooyan, and J. C. Quick, “Research note: The relationship between time pressure and performance: A field test of Parkinson’s Law,” *J. Organ. Behav.*, vol. 5, no. 4, pp. 293–299, 1984.
[258] O. Svenson and A. J. Maule, *Time pressure and stress in human judgment and decision making*. Sringer, 1993.
[259] R. Bolle, *Guide to Biometrics*. Springer, 2004.
[260] M. Arevalillo-Herráez, D. Arnau, L. Marco-Giménez, J. A. González-Calero, S. Moreno-Picot, P. Moreno-Clari, A. Ayesh, O. C. Santos, J. Boticario, M. Saneiro, and others, “Providing personalized guidance in arithmetic problem solving,” in *PALE 2014. Personalization Approaches in Learning Environments*, 2014, pp. 42–48.
[261] Mathworks, “tansig: Hyperbolic tangent sigmoid transfer function,” *Mathworks R2015a Documentation*, 2015. [Online]. Available: http://www.mathworks.com/help/nnet/ref/tansig.html. [Accessed: 17-Apr-2015].
[262] Mathworks, “Neuro-Adaptive Learning and ANFIS,” *Mathworks Documentation*, 2015. [Online]. Available: http://uk.mathworks.com/help/fuzzy/neuro-adaptive-learning-and-anfis.html. [Accessed: 14-Apr-2015].
[263] Mathworks, “Gaussmf (Gaussian Curve Membership Function),” *Mathworks Documentation*, 2015. [Online]. Available: http://uk.mathworks.com/help/fuzzy/gaussmf.html. [Accessed: 14-Apr-2015].
[264] R. M. Keller, “Finite-State Machines,” in *Computer Science: Abstraction to Implementation*, Harvey Mudd College Department of Computer Science, 2001.
[265] J. A. Anderson and T. J. Head, *Automata theory with modern applications*. Cambridge University Press, 2006.
[266] D. Arnau, M. Arevalillo-Herráez, L. Puig, and J. A. González-Calero, “Fundamentals of the design and the operation of an intelligent tutoring system for the learning of the arithmetical and algebraic way of solving word problems,” *Comput. Educ.*, vol. 63, pp. 119–130, Apr. 2013.
[267] K.-E. Chang, Y.-T. Sung, and S.-F. Lin, “Computer-assisted learning for mathematical problem solving,” *Comput. Educ.*, vol. 46, no. 2, pp. 140–151, Feb. 2006.
[268] M. Rani, R. Nayak, and O. P. Vyas, “An ontology-based adaptive personalized e-learning system, assisted by software agents on cloud storage,” *Knowledge-Based Syst.*, vol. 90, pp. 33–48, Dec. 2015.
[269] C.-L. Liu, C.-J. Tsai, T.-Y. Chang, W.-J. Tsai, and P.-K. Zhong, “Implementing multiple biometric features for a recall-based graphical keystroke dynamics authentication system on a smart phone,” *J. Netw. Comput. Appl.*, vol. 53, pp. 128–139, 2015.
[270] D. Chudá, P. Krátky, and J. Tvarožek, “Mouse Clicks Can Recognize web Page Visitors!,” in *Proceedings of the 24th International Conference on World Wide web*, 2015, pp. 21–22.
[271] Mowrer, O. (1960). *Learning theory and behavior*. APA PsycNET.

Appendix I

Part A: Affective Computing Research Involving Mouse and Keystroke Dynamics

Table A1.1: Affective Computing Research Involving Mouse and Keystroke Dynamics

No	Author(s)	Emotion(s) analyzed	Mood induction techniques	Feature(s)
1	Lee, Tsui & Hsiao, 2015	positive, negative, neutral	Stimuli was induced by 63 sounds selected from the IADS-2 database	Keystroke
2	Shukla & Solanki 2013	emotion recognition in the case of naturalistic	No mood induction, based on fixed text typed by users	Keystroke
3	Bixler et al. 2013	boredom and engagement	Essay writing on three topics: (a) academics topics (b) socially charged issues (c) personal emotion experiences	Keystroke
4	Lee et al. 2012	happiness, surprise, anger, disgust, sadness, fear, neutral	No mood induction, based on tweeter message sent by user when he/she feels a certain emotion	Keystroke
5	Epp et al. 2011	anger, boredom, confidence, distraction, excitement, focused, frustration, happiness, hesitance, nervousness, overwhelmed, relaxation, sadness, stress, tired	No mood induction. Participant's experiences are recorded periodically in real-time during their daily activities.	Keystroke
6	Alhothali 2011	confusion, boredom and frustration in the case of negative valence; delight and neutral in the case of positive valence	No mood induction. Participants are asked to answer questions based on the selected topic during the tutoring session.	Keystroke
7	Khanna 2010	positive, negative, neutral	Reading a small paragraph	Keystroke
8	Vizer et al. 2009; Vizer 2009b	cognitive stress and physical stress	Cognitive stress induction: mental multiplication and three-back, or Lag-2, number recall Physical stress induction: cardiovascular exercise and resistances exercise	Keystroke
9	Lv et al. 2008	anger, fear, happiness, sadness, surprise and neutral	Listening/watching a short story for each of the six emotions	Keystroke
10	Tsoulouhas et al. 2011	boredom	Learning objects: (short, medium, long) text with images, short text, video, multiple choice questions, exercise	Mouse
11	Maehr 2008	sadness, happiness, neutral	Watching 3 videos that induce the 3 emotions	Mouse
12	Schuller et al. 2002	surprise, joy, anger, fear, disgust, sadness, neutral	3 types of speeches: speeches to control an internet browser (b) sample phrases from radio plays (c) acted emotions	Mouse
13	Salmeron-Majadas et al. 2014	Valence (pleasure vs displeasure) and arousal (high activation vs low activation)	Answering some tricky and awkward personal questions, and watching eight affective images	Mouse and Keystroke
14	Hernandez et al. 2014	stress	(1) Text transcription (stressful environment is induced by timer and progress bar, faster blinking of cursor, decreased font readability and loud traffic noise)	Mouse and Keystroke

No	Author(s)	Emotion(s) analyzed	Mood induction techniques	Feature(s)
			(2) Expressive writing of recent past memory (in relaxed or stressful condition) (3) Mouse clicking (done after stress condition of text transcription and exptressive writing) (to capture spillover-effects of stress from the previous tasks)	
15	Kolakowska 2013	literature review on the use of mouse and keystroke dynamics in emotion detection	Not applicable	Mouse and Keystroke
16	Zimmermann et al. 2006	positive, negative, high, low, neutral, valence, arousal	Six 8-11 minutes long movie clips	Mouse and Keystroke
17	Zimmermann et al. 2003	Neutral, positive valence/high arousal, positive valence/low arousal, negative valence/high arousal, negave valence/low arousal	Six 7 – 11 minutes long film clips	Mouse and Keystroke

Part B: Literature Review of Existing Research involving Keystroke Dynamics-based Analysis

Table A1.2: Summary of Existing Research Papers based on Keystroke Dynamics-based Analyses

Research by	Data collection	No of participants	Classification/clustering Techniques	Type of Text	Testing period	FAR	FRR	EER or other accuracy rate	Remark
Monrose & Rubin 1997	keystroke latencies (digraph), keystroke durations and typing speed	42 then 31	clustering: maximin distance; classification: Euclidean distance, non-weighted probability and weighted probability	fixed text and free text	7 weeks	9.3% for fixed text; 77% for free text	-	-	11 profiles were eliminated due to erroneous timing results; Weighted-probability performed the best
Obaidat & Sadoun 1997	key duration (digraph) (average 7 characters)	15	neural network and pattern recognition: fuzzy ARTMAP, radial basis function networks (RBFN), learning vector quantization (LVQ) neural network, backpropagation with a sigmoid transfer function (BP, Sigm), hybrid sum-of-products (HSOP), sum-of-products (SOP), potential function and Bayes' rule	fixed text	8 weeks	-	-	average mis-classification error = 0.8%	the research found that hold durations are more effective than key latencies; Fuzzy ARTMAP, RBFN, and LVQ neural network paradigms gave 0% misclassification error
Robinson, Liang 1998	interkey time and keyhold time.	140 student userids, 10 used for forgery. 10 imposters.	nearest-neighbour hierarchical clustering to see the effect of interkey and keyhold dimensions. To recognize between true and forged samples, Minimum intra-class distance (MICD) classifier, non-linear classifier, and inductive learning are used.	Fixed text (login string) – average length: 6.4 chars	Sampling took place during their routine use.	Inductive learning with Interkey and hold times combine-9%	Inductive learning with Interkey and hold times combined-10%		For MICD and nonlinear classifiers, hold times alone performed better. Each of 10 imposters attempted each 10 userids 10 times.

Research by	**Data collection**	**No of participants**	**Classification/clustering Techniques**	**Type of Text**	**Testing period**	**FAR**	**FRR**	**EER or other accuracy rate**	**Remark**
Monrose & Rubin 2000	keystroke latencies (digraph), keystroke durations	63	clustering: K-nearest neighbour; classification: Euclidean distance, non-weighted probability and weighted probability	fixed text	11 months	12.82%	-	-	Weighted-probability performed the best
Bergadano et al. 2002	keystroke latencies (trigraph)	154	classification: mean distance measure	fixed text	1 month	0.01%	4%	-	keys used are only the 26 lower-case letters, space, full stop, comma, apostrophe and the carriage return keys.
Araujo et al. 2005	key code, Up-Down time, Down-Down time ,and key duration (min 10 characters)	30	classification: statistical classifier	fixed text	-	1.89%	1.45%	-	Results obtained based on 10 samples
Gunetti & Picardi 2005	keystroke duration (average 800 characters)	205	classification: Adopted distance measure (Relative and Absolute distances) using n-graph	free text	6 months	0.00489%	4.83%	-	Results obtained based on 14 samples
Filho & Freire 2006	Down-down time (from 4 words to ±500 keystrokes)	47	timing histogram equalization with 1) Bleha's algorithm (fixed-text); 2) Monrose and Rubin's algorithm (fixed-text); 3) Monrose and Rubin's algorithm (free-text); 4) 2D histogram (free-text)	fixed text and free text	around1 month	-	-	EER 1) 6.2 - 7.5%; 2) 10-12.5%; 3) 19.9%; 4) 12.7%	The research argues that single memoryless non-linear mapping of time intervals can improve the performance of the existing algorithms

Research by	Data collection	No of participants	Classification/clustering Techniques	Type of Text	Testing period	FAR	FRR	EER or other accuracy rate	Remark
Boechat et al. 2007	durations of keystrokes, and latencies between keystroke	30 patterns	Criterion of Separation (Threshold) using the using the Weighted probability measure	Fixed text (user's full name)	-	0%.	4.08%		
Revett, Gorunescu, Gorunescu, Ene, Magalhaes, et al. 2007	digraph, trigraph, entry time, speed (6 - 15 characters)	50	Probabilistic Neural Network and MLFN back-propagation neural network	fixed text	14 days	-	-	*FAR/FRR = 3.7% for edit distance; FAR/FRR = 4.2% for derived attributes	PNN performed better; analysis with derived attributes such as digraph/trigraph times, speed and edit distance are more effective than primary attributes, such as keystroke duration and key code
Lv, Lin et al. 2008	Pressure sequence, keystrokes (characters), key down time, key up time	50 individuals with 3000 samples	Average filter to remove noise; normalization to set 0 as mean value and 1 as standard deviation of the pressure. Classifier fusion technique to combine 3 methods: global features, dynamic time warping and traditional keystroke dynamics	Fixed text based on 10 utterances	-	-	-	Error rates: Neutral: 5.8 Anger: 6.6 Fear: 6.4 Happiness: 14.4 Sadness: 14.4 Surprise: 4.4 Average: 6.6	Required pressure sensor keyboards. 6 emotions: neutral, anger, fear, happiness, sadness and surprise. Traditional keystroke dynamics can distinguish happiness and sadness better than other methods although it does not perform well alone
Marsters 2009	keystroke durations and keystroke latencies (min 300 keystrokes)	10	classification: superviser learning - BayesNet classifier, K-Star classfier and RandomForest	fixed text	18 months	-	-	EER = 0.27%	BayesNet performed fastest and the best; RandomForest used longest time
Vizer 2009a; Vizer et al. 2009	timing features, key features, text features	24	classification using machine learnig: Decision Tree (DT), Support Vector	free text	median time span = 9 days	ANN (physical stress)	ANN (physical stress)	*classification rate (ROC) = 62.5% for the	The resarch is to detect changes in typing associated with stress,

Research by	Data collection	No of participants	Classification/clustering Techniques	Type of Text	Testing period	FAR	FRR	EER or other accuracy rate	Remark
	(±1200 characters)		Machine (SVM), k-Nearest Neighbor (kNN), AdaBoost, Artificial Neural Network (ANN).			0.375; kNN (cognitive stress) 0.187	0.375; kNN (cognitive stress) 0.313	physical stress condition (ANN) and 75% for the cognitive stress condition (kNN)	by analyzing keystroke and linguistic features
Harun et al. 2010	Down-down time	47	Classification: Artificial Neural Network (ANN) (multi-layer perceptron (MLP) with back-propagation and Radial Basis Function (RBF)) and Distance Classifier (Euclidean distance, Mahalanobis & Manhattan distance)	fixed text and free text	around1 month	-	-	EER (fixed text) = 2% (MLP); EER (free text) = 22.9% (MLP) ; EER(free text) = 4% with manhanttan distance classifier	all databases used are based on the work of Filho & Freire (2006); All the databases were normalized using an equalization histogram which is a nonlinear transformation
Shimshon, Moskovitch et al. 2010	di-graphs(two consecutive keystrokes) and and their corresponding interval times	10 legitimate users and 15 imposters	clustering *di*-graphs based on temporal features (e.g. latency) and multi-class classification. The results were improvised with superior k (35) and k = 1. Second experiment is done based on ensemble approach.	Free text (email with different length, with the mean of 433 to 1034 keystrokes)	Each one types 15 real emails (session).	0.41%	0.63%	EER: 0.53% AUC= 0.0013	The ensemble classifier consists of five classifiers. These five classifiers were generated using 24, 27, 30, 35 and 53 *di*-graphs in each cluster
Khanna 2010	Typing speed, number of characters typed in 5 seconds interval, total typing time, backspace, idle time	41	Simple Logistics, SMO, Multilayer Perceptron, Random Tree, J48, BF Tree	Fixed text – paragraphs of 8 to 9 lines	4 to 5 months	-	-	-	Recognition rates for 2 emotional categories (negative and positive) using various classification algorithms ranged from 62.66% to 89.02%

Research by	Data collection	No of participants	Classification/clustering Techniques	Type of Text	Testing period	FAR	FRR	EER or other accuracy rate	Remark
Alhothali 2011	Timestamps and key-codes, type speed rate, key latency, key duration, deletion rate, capitalization, spaces per response, punctuation rate, unrelated key rate, response quality (e.g. typo, completeness)	20	Correlation analysis Discriminant Analysis (LDA, PCA and QDA) Naïve Bayes (Gaussian naïve Bayes & Kernel naïve Bayes) k-Nearest Neigbour Decision Trees Artificial Neural Network	Fixed text – around 100 words	Around 2 weeks. Session 2 of the experimen t took around 45 minutes	-	-	-	Correlation analysis did not show any significant correlation between features and user's emotion, but session duration is significantly correlated to emotion. Classification accuracy based on various classification algorithms ranged from 30.05% to 53.89% for emotion, and from 57.05% to 82.82% for valence
Epp et al. 2011	Diagraphs, trigraphs, Number of key events that were part of the graph, keystroke duration, key latency, mistakes (backspace + delete), key codes	12	Decision trees	Fixed text	4 weeks	-	-	-	Keystroke dynamics can accurately classify at least 2 levels of 7 emotional states with classification accuracy rate from 77.4% to 87.8%

Research by	Data collection	No of participants	Classification/clustering Techniques	Type of Text	Testing period	FAR	FRR	EER or other accuracy rate	Remark
Lee et al. 2012	typing speed, frequency of pressing a specific key, maximum text length, erased text length, and touch count; device shake count; environment condition (location, time zone, weather)	1 (with 314 datasets)	Bayesian Network classifier	Free text – tweet to Twitter when she/he feels a certain emotion	-	-	-	-	Typing speed has the highest correlation to emotions; inputted text lengths, shaking of the device, or user location were also important features for emotion recognition. An average classification accuracy rate of 67.52% for 7 emotional states is achieved.
Teh, Yue et al. 2012	Dwell time, flight time (down-down time, up-down time, up-up time)	100 (50 Phase 1, 100 Phase 2) with total of 1000 keystroke timing data	Gaussian Probability Density Function (GPD) and Direction Similarity Measure (DSM). Three fusion scheme (Single Layer Single Expert, Single Layer Multiple Expert, Multiple Layer Multiple Expert) to merge the scores paring with six fusion rules (Sum Rule, Weighted Sum Rule, Product Rule, Max Rule, OR Voting rule, AND Voting Rule)	Fixed text (“the brown fox”)	2 phases separated by an interval of 4 months	-	-	EER: 1.401%	Subjects are required to type the text without typing error for 10 times. Best result is to combine dwell time and flight time (up-down time) with MLME coupled with AND rule.

Research by	**Data collection**	**No of participants**	**Classification/clustering Techniques**	**Type of Text**	**Testing period**	**FAR**	**FRR**	**EER or other accuracy rate**	**Remark**
Eswari, Sundarapandiyan et al. 2012	keyhold time, flight time(down-down time, up-down time)	28	Adaboost and random forest	10-digit number	4 sessions		12.5%	EER: 8.60% Accuracy: 99.54%	The subjects are required to use only 1 finger to type the password for 50 times in each session.
Giot, El-Abed et al. 2012	Flight time (up-down time, up-up time, down-down time and down-up time)	83 students and produced 5185 genuine samples, 5754 impostor samples, 5439 imposed samples	Gaussian distribution (distance computation); Kruskal-Wallis (KW) test (Statistical validation); simple feature authentication; score fusion	Fixed text: userid and password	4 sessions			Simple feature: EER: 10.00% on imposed dataset (up-up time for login only) EER: 8.87% for combination of login and password using all features fusion	No typing error is allowed. Using login userid is better than using passwords; use all features during the fusion improves results; the size and the entropy of the password has impact on the performance
Giot, Rosenberger et al. 2012	Comparison score between query and the number of times the user has typed the password	Set 1: 51 users x 400 samples Set 2: 100 users x 60 samples	Normalize with z-score. Q-stack classifier. SVM with three folds cross validation scheme	Fixed text -password	Set 1: 8 sessions; Set 2: 5 sessions			EER: Group 1: 1.43% Group 2: 1.06% Group 3: -6.36%	Group 1: Users having no correlation between time and recognition score Group 2: Users having a very small correlation • Group 3: Users having more or less correlation

Research by	Data collection	No of participants	Classification/clustering Techniques	Type of Text	Testing period	FAR	FRR	EER or other accuracy rate	Remark
Bixler et al. 2013	Relative timing (e.g. session time), keystroke verbosity (e.g. backspace), keystroke timing (e.g. latency), pausing behavior; Stable traits; Task appraisals	44	J48, Naïve Bayes, Bayes Net, SMO, Decision Table, One R, Random Forest, Random Tree, and REP Tree	Free text (essay writing)	3 topics: 1 topic 10 minutes	-	-	-	The Kappa rates between 2 or 3 affects ranged from 0.021 to 0.374. The best classification is between boredom and engagement using the features of keystroke/timing + task appraisals + stable traits, but the accuracy drops significantly when they classify between neutral and other affects.
Shukla & Solanki 2013	keystroke latency, key holdduration, typing speed, frequency of error, pause rate, capitalization rate; diagraph, trigraph; session time	-	Discriminate Analysis, Bayesian Analysis, k-Nearest Neighbor, Artificial neural network and Decision Trees	Fixed text	About 4 weeks	-	-	-	This paper presents techniques to recognize the emotional state of the user through analyzing the keystroke patterns of the user. No results are given in the paper.
Syed Idrus et al. 2014	down-down time (PP, diagraph), up-up time (RR), down up time (PR), up-down time (RP)	110	Support Vector Machine (SVM) with data fusion (majority voting and score fusion)	Fixed text and free text	-	-	-	-	To recognize 4 soft categories: hand category (type with one or two hands) (recogntion rate > 90%), gender (recogntion rate > 79%), age (recogntion rate > 72%), handedness (left or righ

Research by	Data collection	No of participants	Classification/clustering Techniques	Type of Text	Testing period	FAR	FRR	EER or other accuracy rate	Remark
									hand) (recogntion rate > 83%)
Kang & Cho 2014	Down-down time (diagraph)	35	One-class classification • the mean and variance equality test (MV test), • Kolmogorov–Smirnov statistic (K–S statistic), • Cramér–von Mises criterion (C–M criterion), • the distance between two digraph matrices (digraph distance; DD), • R measures (R); • A measures (A), • the linear combination of the R and A measures (R+A), • the product combination of the R and A measures (RA), • Gaussian density estimator (Gauss), • Parzen window density estimator (Parzen), • k-nearest neighbor detector (k-NN), • support vector data description (SVDD)	Free text (minimum 3000 characters)	-	-	-	Test length (1000) Traditional keyboard – 5.64% Soft keyboard – 14.10% Touch keyboard (1 hand) – 12.42% Touch keyboard (2 hands) – 16.62%	Examine the difference between 3 types of keyboards: traditional PC keyboard, soft keyboard with stylus pen, and touch keyboard. EER decreased when test size or refererence length increased. With R+A or RA measures, a near zero error rate could be achieved for PC keyboard, but not for other keyboard types. For virtual or soft keyboard, Parzen, k-NN and SVDD with C-M criterion was found to be the best model for larger reference-test length

Research by	Data collection	No of participants	Classification/clustering Techniques	Type of Text	Testing period	FAR	FRR	EER or other accuracy rate	Remark
Liu et al. 2015	Down-Up time, down-down time, up-down time and up-up time, pressure, size and angle	113	Statistical classifier (distance)	4 to 9 pattern lock buttons for adroid lock screen	1st phase 10 training samples from each user; 2nd phase collects 10 samples 7 weeks later as test samples	3.03% (all features combined) (training size=10)	2.92% (all features combined) (training size=10)	3.03% (all features combined) (training size=10)	Use pattern lock layout for collecting the user keystroke dynamics features toward improved authenti-cation practices. Employ knowledge-based and biometric-based authentication by combining graphical password and keystroke dynamics
Lee, Tsui & Hsiao, 2015	Keystroke duration and keystroke latency	52	Descriptive Statistics such as mean and ANOVA	fixed text "748596132"	Each participant takes 63 trials from 63 sounds	-	-	-	Stimuli was induced by 63 sounds selected from the IADS-2 database. Affective state was collected using affective space, the Self-Assessment Manikin (SAM), based on the affective rating system devised by Lang. Their results support the hypothesis that both keystroke duration and keystroke latency are influenced by emotional states, specifically, influenced by the arousal (low, medium, high). Results show that negative emotion leads to slower keystroke speed

Part C: Literature Review of Existing Research involving Mouse Dynamics-based Analysis

Table A1.3: Summary of Recent Mouse Dynamics-Based Research Papers

Research by	Data collection	No of participants	Classification/clustering Techniques	testing period	FAR	FRR	EER	Remark
Schuller et al. 2002	Mouse or touch screen gesture, speech	15	DTW algorithm with Itakura local constraints and Euclidean distance metric	-	-	-	-	Acceptance tests with 15 users showed a classification potential of >80% recognition rate, using multimodal fusion. The classification is mainly based on speech signal, with the combination of user profiling on haptic interaction using touch-screen or mouse signal.
Zimmermann et al. 2003	Click rate per min, average duration of mouse clicks, mouse total movement distance, average distance per single movement, pause length, pause rate, number of "heavy mouse movements", max/min/average mouse speed, keystroke rate, average keystroke duration, performance	96	-	1.5 – 2 hours	-	-	-	The preliminary results show that film clips are effective in inducing the expected mood changes.
Pusara & Brodley 2004	cursor movements (distance, angle and speed) and mouse events (i.e. button clicks and wheel events)	18 to 11	supervised learning and decision tree classifier, with smoothing filter	average 2 hours	1.75%	0.43%	-	seven users were excluded due to data sets contain few mouse events

Research by	Data collection	No of participants	Classification/clustering Techniques	testing period	FAR	FRR	EER	Remark
Zimmermann et al. 2006	All mouse and keyboard types of action (e.g. mouse button down, mouse position x and y coordinates, which key pressed)	96 and 33	ANOVA and MANOVA	-	-	-	-	The groups that have seen affective film clips are significantly different from the neutral group. However, there is no significant differences between groups with different combination of positive, negative, high, low, valence and arousal emotions.
Ahmed & Traore 2007	the type of action (mouse move, drag-and-drop, click, silence) , distance, elapsed time, and movement direction	22	artificial neural networks	9 weeks (around 13 hours per participant)	2.4649%	2.4614%	2.46%	The research explores multiple sets of conditions, for instance, on imposing greater control on environmental variables and also imposing less control on environmental variables.
Maehr 2008	Mouse acceleration, movement precision, smoothness, speed	39	ANOVA, t-Test, descriptive statistics	One experiment took around 10 minutes	-	-	-	There is a significant correlation between the arousal films shown and the mouse movement speed. No significant differences between 4 emotion groups. Different levels of arousal lead to significantly different mouse motions.
Shen et al. 2009	Type of action (click, double click), silence periods, elapsed time, movement speed, travelled distance, cursor position distribution	10	PCA and ISOMAP	2 months	1.48% (PCA) 0.55% (ISOMAP)	5.33% (PCA) 3.00% (ISOMAP)	-	The empirical research shows that variations are obvious in mouse activities. To tackle the problem of variability, they propose a

Research by	Data collection	No of participants	Classification/clustering Techniques	testing period	FAR	FRR	EER	Remark
								dimensionality reduction based approach
Bours & Fullu 2009	cursor location, speed	28	distance metrics (Edit Distance)	6 days	-	-	>40%	In the experiment the participants needed to perform a pre-defined task called ”follow the maze”. The result was not promising
Shen et al. 2010	mouse action (single click, double click, drag and drop, mouse-wheel, mouse silence), distance, direction, speed	20	Support Vector Machine (SVM)	2 months	1.86%	3.46%	-	the optimum combination of features only contains 14 features, most of which (12 out of 14) are computable online from observed mouse activities
Nakkabi et al. 2010	speed, direction, type of action, travelled distance, elapsed time	48	Variance Reduction (VR); Unsupervised learning - Learning Algorithm for Multivariate Data Analysis (LAMDA)	284 hours	0.0%	0.36%	-	The results obtained fulfil the European standard for access control

Research by	Data collection	No of participants	Classification/clustering Techniques	testing period	FAR	FRR	EER	Remark
Tsoulouhas et al 2011	Total Mouse Speed (TMS); Latest Mouse Speed (LMS); Mouse Inactivity Occurrences Before Asked (MIN) ; Average Duration of Mouse Inactivity Before Asked (DMIN); Movements to Total Movements Ratio – Horizontal (HRZ), vertical (VRT) and diagonal (DGNL). Average Movement Speed per Movement Direction (MDA). Interval b= {10, 20, 30, 40} seconds	136	Statistical classifier, decision trees as classification algorithm; *SimpleKmeans* with *Euclidean* distance as clustering algorithm.	45 minutes	2.7586 % when b=10	-	-	To test boredom of students. Use Weka as analysis tool Tested on medium text with images; short text; short text with images; long text with images; video; multiple choice questions; and exercise. TMS vs LMS – significant different for boredom. MIN & DMIN – significant difference between bored and non-bored users. HRZ, VRT and DGNL are connected to user's behaviour (especially VRT). MDA for some directions significantly increased for bored users.
Chao Shen, Cai & Guan 2012	Action type (mouse move or mouse click), screen area, window position, the timestamp when the event occurred. Application type. Features: -click elapsed time -movement speed -movement -relative position of extreme speed	28 students	Kernel density estimation to estimate the probability density function (PDF) of a random variable. Detection method: 1) One-class 2) Nearest-neighbor 3) Neural network (single hidden layer) – p inputs, 1 output and 2p/3 hidden nodes 4) Support vector machine	thirty minutes for each of the 30 sessions (interval 24 hrs) for a total of 30 to 60 days	One-class SVM 0.37% (30 mins) 7.78% (5 mins)	One-class SVM 1.12% (30 mins) 9.45% (5 mins)	5%	System runs as a background job, They employ one-class learning methods to perform the task of continuous user authentication,

Research by	Data collection	No of participants	Classification/clustering Techniques	testing period	FAR	FRR	EER	Remark
Lin et al. 2012	Mouse movement that lasts for at least 1 second (recorded every 250ms) Mouse features: Velocity, acceleration, curvature.	20 (only 11 provided sufficient samples)	Use Dimensionality reduction (ISOMAP/LFDA) to reduce the feature vector. Use 2-class base-classifier (KNN, DT and SVM) to train the reduced vector. Use vote scheme to improve the accuracy.	2 weeks	Set A ~6 %	Set A ~5%		Set A for complete file-related operations in Windows Explorer. For comparison, Set B for operating Windows Explorer and Set Cforoperating the computer. The proposed approach iseffective on Set A because users havesimilar mouse behaviour patterns, while Set C is not
Chao Shen, Cai, Guan, et al. 2012	movement direction, distance Mouse features: distance, time, speed and acceleration	26	time warping edit distance to calculate the distance vector. Classifier: 1) Nearest-Neighbour 2) Neural Network (single hidden layer)– p inputs, 1 output and 2p+1 hidden nodes 3) Support Vector Machine	2 times per day for at least 15 days	4.76% (118.14 seconds - 160 moves) 0.001 (1173.95 seconds- 1600 moves)	0.67% (118.14 seconds - 160 movements) 0 (1173.95 seconds- 1600 movements)	2.64%	the technique is able to meet the European standard for commercial biometric technology if a longer authentication time is allowed
C Shen et al. 2012	movement direction, movement distance, and click type	37	Distance metrics and kernel PCA to obtain a distance-based eigenspace. Detection method: 1) One-class 2) Nearest-neighbor 3) Neural network (single hidden layer) – p inputs, 1 output and 2p/3 hidden nodes 4) Support vector machine	15 days and 60 days	8.74% (11.8 seconds)	7.69% (11.8 seconds)	-	their results show that the Nearest Neighbor (Manhattan) detector has the lowest error rates on the data

Research by	Data collection	No of participants	Classification/clustering Techniques	testing period	FAR	FRR	EER	Remark
Shen et al. 2014	Movement directions, movement distance , Movement offset, movement time, mouse speed (x and y-speed), speed [170]against distance, mouse accelaeration (x y aceleration), acceleration aginst distance and click type	58	Euclidean distance Mahalanobis Outlier counting Nearest neigbour K-Means Neural Network Support Vector Machine (one class)	2 rounds, next collection must be at least one day later	-	-	EER = 8.81% (Manhattan nearest neighbour)	The paper evaluates 17 types of anomaly-detection algorithms. EER ranged from 8.81% to 69.46% from the 17 classifiers
Salmeron-Majadas et al. 2014	43 keyboard indicators (such as key press, latency, count of key pressed, key code, digraph, trigraph, etc.); 96 mouse indicators (e.g. clicks, scroll movements, button press, distance, etc)	75	C4.5, Naïve Bayes, Bagging, Random Forest and AdaBoost	-	-	-	Best prediction rate is 59% by Random Forest and AdaBoost using the combination of keyboard and mouse indicators	Both mouse and keyboard indicators have higher correlation with the valence dimension of the affective state reported by the participants than with their arousal dimension.
Chudá & Krátky 2014	Mouse click, silence/leaving, mouse movement (velocity, pace, acceleration, direction, angular velocity, curvature) Mouse scroll (velocity, pace, acceleration)	28	Nearest neighbour (Manhattan, Manhattan with std. deviation, Euclidean, Mahanabolis and t statistics of Welch's test)	-	-	-	Accuracy rate of 87.5 % is gained using *t statistic*when complete user model consisting of 15 characteristics is used	The paper proposes a user modelling process specialized for user identification in browsing the web using mouse dynamics patterns, which might be useful for personalized e-shopping system

Research by	Data collection	No of participants	Classification/clustering Techniques	testing period	FAR	FRR	EER	Remark
Chudá et al 2015 [270]	pause to click, click duration, pause after click, mouse down/up, mouse movement	Controlled set: 20 Uncontrolled set: 180,700	statistic of Kolmogorov-Smirnov test	Controlled: 10 minutes Uncontrolled: 21 days	-	-	Controlled: Success rate 96% for 100 clicks, success rate 44% for 100 movement strokes Uncontrolled: Success rate 85% for 50 clicks. To recognize a user in a pool of 1500 users is 51% accuracy	The paper presents an approach to user identity recognition on the web based on mouse dynamics. It could be useful to accurate recognize users in small groups to improve user-oriented services. It may also enables relatively accurate recognition of a user in a large user pool

Remarks:

FAR	False Acceptance Rate
FRR	False Rejection Rate
EER	Equal Error Rate, where FAR=FRR; also known as cross-over error rate
Accuracy	classification accuracy
ROC	Receiver Operating Characteristic curves
Keystroke latencies	time between the interaction (release and depress) of two keys (referred as digraph if involves two consecutive keys, or trigraph if three consecutive keys)
Keystroke duration	time each key is held down (Down-Up time)

This page is intentionally left blank

Appendix II

Part A: Experiment Consent Form

Evaluation of Stress Effect on E-Learning

Thank you for taking your valuable time to complete this research experiment. The purpose of the study is to evaluate the stress effect on E-Learning. Your participation will also help us to understand the effects of stress to the user's mouse movements and keyboard dynamics.

Procedures

This survey should only take about 30 to 40 minutes of your time. Before you start, please read the following carefully.

1. In Section A, you are required to search for a feature (it's a hyperlink) in each test later. There are 64 cases (challenges) all together. In each challenge, if you could not find the link, you may click the "GIVE UP" button on the top right corner. In case you need to review the question again, please click the "RESTART" button.

   Remarks: Some of the pages are purposely designed with low usability, such as inappropriate combination of text colour and background colour, smaller font size, etc., which may cause eye fatigue or eye strain. If you feel uncomfortable with the page and could not proceed, you may click the "GIVE UP" button on the top right corner and skip to next challenge.
2. In Section B, you are required to answer 10 arithmetic questions. As this test is to evaluate the effect of cognitive stress to user's behaviour, we require you to calculate the answer using only YOUR BRAIN (i.e. no calculator, etc.). In any case, you may click the "GIVE UP" button on the top right corner to quit the challenge as you wish (and skip to the next one). Once you finish a question, please indicate how stress did you feel when you were answering the question.
3. In Section C, you are required to type in the given text into a textbox. As this test is to evaluate the effect of text length and familiarity, there are 6 questions with various text length - 3 questions in English and 3 in German. In any circumstance, you may click the "GIVE UP" button on the top right corner to quit the challenge as you wish (and skip to the next one). Once you finish a question, please indicate how stress did you feel when you were answering the question.
4. At the end of the experiment, you are required to fill up a page of questionnaire regarding your perception of your stress level according to different setting.
5. All information provided by the respondents will be kept with the strictest confidence and will be used only for educational purposes.
6. You may withdraw from this survey at any time.
7. This is an anonymous survey. Please do not write any identifying information (such as your name) anywhere on this survey.
8. If you have any questions, you may contact me at any time.
   - Contact person: Ms. Lim Yee Mei
   - E-mail address 1: limyeemei@gmail.com

If you agree with the terms above, please check the checkbox below. Then click "NEXT" to start the challenge. If you do not agree, you may exit/close this page. Thank you very much for your participation.

☐ I have read the above and agree to the terms

Part B: Menu Design for the Search Task Experiment

Table A2.1: 64 Menu Designs Varied by 6 Factors

Question	colour	font size	text length	hyperlink	organization	scrolling
1	good	big	short	clear	categorized	none
2	bad	big	short	clear	categorized	none
3	good	small	short	clear	categorized	none
4	bad	small	short	clear	categorized	none
5	good	big	long	clear	categorized	none
6	bad	big	long	clear	categorized	none
7	good	small	long	clear	categorized	none
8	bad	small	long	clear	categorized	none
9	good	big	short	ambiguous	categorized	none
10	bad	big	short	ambiguous	categorized	none
11	good	small	short	ambiguous	categorized	none
12	bad	small	short	ambiguous	categorized	none
13	good	big	long	ambiguous	categorized	none
14	bad	big	long	ambiguous	categorized	none
15	good	small	long	ambiguous	categorized	none
16	bad	small	long	ambiguous	categorized	none
17	good	big	short	clear	random	none
18	bad	big	short	clear	random	none
19	good	small	short	clear	random	none
20	bad	small	short	clear	random	none
21	good	big	long	clear	random	none
22	bad	big	long	clear	random	none
23	good	small	long	clear	random	none
24	bad	small	long	clear	random	none
25	good	big	short	ambiguous	random	none
26	bad	big	short	ambiguous	random	none
27	good	small	short	ambiguous	random	none
28	bad	small	short	ambiguous	random	none
29	good	big	long	ambiguous	random	none
30	bad	big	long	ambiguous	random	none
31	good	small	long	ambiguous	random	none
32	bad	small	long	ambiguous	random	none
33	good	big	short	clear	categorized	scroll
34	bad	big	short	clear	categorized	scroll
35	good	small	short	clear	categorized	scroll
36	bad	small	short	clear	categorized	scroll
37	good	big	long	clear	categorized	scroll
38	bad	big	long	clear	categorized	scroll
39	good	small	long	clear	categorized	scroll
40	bad	small	long	clear	categorized	scroll
41	good	big	short	ambiguous	categorized	scroll
42	bad	big	short	ambiguous	categorized	scroll
43	good	small	short	ambiguous	categorized	scroll
44	bad	small	short	ambiguous	categorized	scroll
45	good	big	long	ambiguous	categorized	scroll
46	bad	big	long	ambiguous	categorized	scroll
47	good	small	long	ambiguous	categorized	scroll
48	bad	small	long	ambiguous	categorized	scroll
49	good	big	short	clear	random	scroll
50	bad	big	short	clear	random	scroll
51	good	small	short	clear	random	scroll
52	bad	small	short	clear	random	scroll
53	good	big	long	clear	random	scroll
54	bad	big	long	clear	random	scroll
55	good	small	long	clear	random	scroll
56	bad	small	long	clear	random	scroll

Question	colour	font size	text length	hyperlink	organization	scrolling
57	good	big	short	ambiguous	random	scroll
58	bad	big	short	ambiguous	random	scroll
59	good	small	short	ambiguous	random	scroll
60	bad	small	short	ambiguous	random	scroll
61	good	big	long	ambiguous	random	scroll
62	bad	big	long	ambiguous	random	scroll
63	good	small	long	ambiguous	random	scroll
64	bad	small	long	ambiguous	random	scroll

Part C: 64 instructions given to the participants in the Search Task

Question 1

Assume that the course AACS4214 Database Systems is no longer offered in the current semester, but you wish to download some notes of this course. Search and click the link where do you think you can find these documents.

Question 2

Assume that you are looking for the assignment guidelines for AACS1314 Program Design offered in the current semester. Search and click the link where do you think can find this document.

Question 3

Assume that you wish to search some tutorial materials offered by MSDN Academic Alliance (MSDNAA). Search and click the link where do you think you can download the documents.

Question 4

Assume that you are looking for the notes of AACS1123 Information Systems for your reference. Search and click the link where do you think you can find these reports.

Question 5

Assume that you are currently involved in the course AACS5078 Industrial Training. You wish to download the industrial training report template. Search and click the link where do you think you can download this document.

Question 6

assume that you are searching for a document provided by Department of Quality of Assurance (DQA). Search and click the link where do you think you can find it.

Question 7

Assume that you wish to find a staff's email address. Search and click the link where do you think you can find this information.

Question 8

You are looking for a guideline regarding TARC Business Intelligence (BI) Competition. Search and click the link where do you think you can find this document.

Question 9

Assume that you wish to attend training to learn using the features available in the College e-Learning System (CeL). Search and click the link where do you think you can find this information.

Question 10

Assume that you are searching for information to understand the College e-learning System (CeL) and the features it offers. Search and click the link where do you think you can find this information.

Question 11

Assume that you are a new to the institution. You wish to know the types of support provided by the CITC, such as Internet service. Search and click the link where do you think you can find this information.

Question 12

Assume that you are new to the College E-learning System. You are looking for relevant support provided by CITC, such as beginner's guide to learn how to use the system. Search and click the link where do you think you can find this information.

Question 13

Assume that you are looking for the guidelines to develop Final Year Project/ Dissertation in current semester. Search and click the link where do you think you can find this document.

Question 14

Assume that you are looking for past students' final year project/dissertation reports for your reference. Search and click the link where do you think you can find these reports.

Question 15

Assume that you have joined a group formed by the Computer Science (CS) Department, which enables communication amongst users and resources sharing related to computer science area. Search and click the link where do you think you can find it.

Question 16

Assume that you have recently joined Programming Special Interest Group formed by the Computer Science Department. You are looking for a document shared to all group members. Search and click the link where do you think you can find this document.

Question 17

Assume that you are searching for an assignment template for AACS4064 Programming. Search and click the link where do you think you can find it.

Question 18

assume that you wish to search some practical guidelines of AACS3012 IS Development. Search and click the link where do you think you can download the document.

Question 19

Assume that you wish to check your coursework marks for AACS4134 Internet Programming. Search and click the link where do you think you can find this information.

Question 20

Assume that you are looking for the teaching materials of AACS5124 Project Management. Search and click the link where do you think you can find these reports.

Question 21

Assume that you are currently involved in the internal competition called Imagine Cup 2013/14. You wish read the guidelines. Search and click the link where do you think you can find this information.

Question 22

Assume that you wish to download the notes of AACS5414 Electronic Commerce for SME, which is offered in the current semester. Search and click the link where do you think you can find this document.

Question 23

You are looking for an assignment template for ABMD1022 Tamadun Islam dan Asia. Search and click the link where do you think you can find this document.

Question 24
Assume that the course AAMS1244 Management Mathematics is no longer offered in the current semester, but you wish to download some notes of this course. Search and click the link where do you think you can find these documents.

Question 25
Assume that you are a student from AIA1, and you are requested to evaluate the courses that you study in the current semester. Search and click the link where do you think you can complete this request.

Question 26
Assume that you are a student from AIA2, and you are requested to evaluate the courses that you study in the current semester. Search and click the link where do you think you can complete this request.

Question 27
Assume that you are searching for the seminar reports done by the students from other faculty. Search and click the link where do you think you can find these reports.

Question 28
assume that you are looking for the template to prepare your seminar report. Search and click the link where do you think you can find this document.

Question 29
assume that you would like to find out some information about the lecturers who teach English Language for the Profession. Search and click the link where do you think you can find this information.

Question 30
Assume that you are specialized in IT. You are required to take an English course according to your profession. Search and click the link where do you think you can find this course.

Question 31
Assume that you are involved in AACS4024 Research Methodologies. You wish to see an announcement posted by the lecturer. Search and click the link where do you think you can find it.

Question 32
assume that currently you are taking a course called AACS4024 Research Methods. Search and click the link where do you think you can find this course.

Question 33
Assume that you wish to download the tutorial materials of AAMS1613 Pre-Calculus, which is offered in the current semester. Search and click the link where do you think you can find this document.

Question 34
Assume that you are looking for the course named AACS3103 Java Programming. Search and click the link where do you think you can find the link.

Question 35
assume that you wish to search some tutorial materials of AACS5274 web Services. Search and click the link where do you think you can download the documents.

Question 36

Assume that the course AACS4094 Operating Systems is no longer offered in the current semester, but you wish to download some notes of this course. Search and click the link where do you think you can find these documents.

Question 37

You are looking for the coursework plan of AACS4124 Software Engineering Practice. Search and click the link where do you think you can find this document.

Question 38

Assume that you are currently involved in the course AACS5144 Object-oriented Programming Techniques. You wish to download the report template. Search and click the link where do you think you can download this document.

Question 39

Assume that AACS3143 web-based Multimedia Applications was offered in the previous semester. Search and click the link where do you think you can download the lecture notes of this course.

Question 40

assume that you are looking for AACS3423 Fundamental of Computer Networks which was offered in the previous semester. Search and click the link where do you think you can find it.

Question 41

Assume that you are one of the AIB1 students. You are requested to evaluate all your lecturers in the current semester. Search and click the link where do you think you can complete this request.

Question 42

Assume that you are one of the AIB2 students. You are requested to evaluate all your lecturers in the current semester. Search and click the link where do you think you can complete this request.

Question 43

Assume that you are searching for the course named AACS1093 web Page Design. Search and click the link where do you think you can find this link.

Question 44

Assume that you have been enrolled into a course named AACS1083 web Page Design. Search and click the link where do you think you can find this link.

Question 45

assume that you are looking for the assignment guidelines of AACS2132 Analysis and Design of IS Case Study. Search and click the link where do you think you can download this document.

Question 46

Assume that you wish to download the notes and case studies samples of AACS2142 Analysis and Design of IS. Search and click the link where do you think you can find these documents.

Question 47

Assume that you are searching for a course named AAMS5144 Science and Engineering Mathematics V. Search and click the link where do you think you can find this link.

Question 48

Assume that you are searching for a course named AAMS5244 Science and Engineering Mathematics VI. Search and click the link where do you think you can find this link.

Question 49

Assume that you wish to download the course plan of AAMS3153 Discrete Mathematics. Search and click the link where do you think you can find this document.

Question 50

Assume that you are searching for a document related to AAMS3163 Algebra. Search and click the link where do you think you can find it.

Question 51

Assume that the course AEMS3513 Moral is no longer offered in the current semester, but you wish to download some notes of this course. Search and click the link where do you think you can find these documents.

Question 52

Assume that you are looking for a course named AACS1003 Information Technology. Search and click the link where do you think you can find this link.

Question 53

Assume that you are currently enrolled into the course named AACS3123 Database Development and Applications. Search and click the link where do you think you can find it.

Question 54

Assume that you wish to find a course named AACS4204 Data Structures and Algorithm. Search and click the link where do you think you can find this link.

Question 55

Assume that you wish to search some tutorial materials of AAMS4124 Numerical Analysis and Mathematics. Search and click the link where do you think you can download the documents.

Question 56

Assume that you are looking for a course named AACS5014 Computer Operating System that is offered in the current semester. Search and click the link where do you think you can find this document.

Question 57

Assume that you are looking for AAMS4124 Mathematics IV, which was offered in the previous semester. Search and click the link where do you think you can find this link.

Question 58

Assume that you wish to find a document of AAMS4214 Mathematics VI, which is offered in the current semester. Search and click the link where do you think you can find it.

Question 59

Assume that you have enrolled into AACS1192 E-business course. Search and click the link where do you think you can find this course.

Question 60

Assume that you like to check the announcement posted by the AACS1193 E-business lecturer 2 weeks ago. Search and click the link where do you think you can find it.

Question 61

Assume that you wish to view the assignment of AACS1074 Programming Concept and Design I that you submitted to the system in the previous semester. Search and click the link where do you think you can find this course.

Question 62

Assume that you wish to submit the assignment to AACS1084 Programming Concept and Design II. Search and click the link where do you think you can find this course.

Question 63

Assume that you are enrolled into ABMS1123 Fundamental Macroeconomics. Search and click the link where do you think you can find this course.

Question 64

Assume that you are enrolled into ABMS1133 Fundamental Microeconomics. Search and click the link where do you think you can find this course.

Appendix III

Part A: The Main Models (Classes) used in the Intelligent Tutoring System

```
class QuestionBank
{
ulong questionID; //unsigned long integer
int difficulty; // level of difficulty, from 1 to 10
string question;
string answer;
double mark;
Boolean timing; // true if time constraint is set
int duration; // time limit in seconds
}
```

```
class MouseBehaviour
{
double SPMS;
double SPMID;
double SPMIO;
double SPMCL;
} //SPMS, SPMID, SPMIO, SPMCL are computed based on Equation 7.4 in Chapter 7
```

```
class KeystrokeBehaviour
{
double SPKS;
double SPKL;
} //SPKS and SPKL are computed based on Equation 7.4 in Chapter 7, and produced
by the inference engine
```

```
class JobPerformance
{
string taskDateTime; //the date and time the task is taken
ulong questionID; //the question stored in the QuestionBank
int questionNumber; //question number displayed during the assessment
double passiveAttempt; //attempt to wait till the time is up
double err; //err = 1 if the answer is wrong, else 0
double SPTD;
}//SPTD is computed based on Equation 7.3 in Chapter 7, and produced by the
inference engine
```

```
class LearnerProfile
{
string learnerID;
List<JobPerformance> jobPerformances;
List<KeystrokeBehaviour> keystrokeBehaviours;
List<MouseBehaviour> mouseBehaviours;
List<int> s_b_sensor; // the stress level measured by the sensor
List<double> demands; //the adjustment of the task demand
List<Boolean> anomalousBehaviours; //true if anomalous behaviour is observed
} //the attributes except learnerID are computed by the inference engine
```

Figure A3.1. The classes for the intelligent tutoring system models used in the C# program

```
public double fire_FFBP_Mouse_Rules(double[4,1] vector_x)
{     //vector_x is the individual mouse dynamics input formed by MS, MID,
MIO,MCL
       /* Data of weights and biases for Layer 1 & 2 are gained from Matlab */

/* initialize v_0j=bias on hidden unit j; w_0k=bias on output unit k  */
       //bias for Layer 1
double[,] v_0j = new double[,] { { -2.0056 }, { -0.93825 }, { 0.88249 }, { -
              2.1779 } };
       //bias for Layer 2
double[,] w_0k = new double[,] { { 0.27447 } };

/* initialize trained weights, v_ij and w_jk */
       //weights for Layer 1
double[,] v_ij = new double[4, 4] { { 0.0038071, -0.80601, -0.90731, 1.349 }, {
              0.37451, -1.7009, 0.66942, -0.17974 }, { 0.66034, 1.9674, 0.55542,
              -0.04512 }, { -1.5216, -0.5351, 0.5106, 0.78847 } };
       //weights for Layer 2
double[,] w_jk = new double[,] { { 0.35892, -0.15104, 0.44673, 0.26204 } };

double[,] z_j;
double[,] y_in_k;
double[,] y_k;

/* activation in layer 1 */
       // the net input to the hidden unit j (Z_in_j);
double[,] z_in_j= matrixSummation(v_0j, matrixMultiplication(v_ij, vector_x));
       // the output signal of Zj
double[,] z_j=tansig(z_in_j);

/* activation in layer 2 */
       //y_ink is the net input to output unit k
double[,] y_in_k= matrixSummation(w_0k, matrixMultiplication(w_jk, z_j));
       //yk is the output signal of output unit k
y_k = tansig(y_in_k);

//to get the final value S(B(mouse))
s_b_mouse = y_k[0,0];
}
```

Figure A3.2. Stress measurement model based on mouse dynamics using FFBP neural net architecture

```
public void fire_FFBP_Key_Rules(double[2,1] vector_x)
{
//vector_x is the individual key dynamics input formed by KS and KL
       /* Data of weights and biases for Layer 1 & 2 are gained from Matlab */

/* initialize v_0j=bias on hidden unit j; w_0k=bias on output unit k  */
       //bias for Layer 1
double[,] v_0j = new double[,] { { 0.78143 }, { 3.2839 }, { -1.6231 }, { 1.1536
              } };
       //bias for Layer 2
double[,] w_0k = new double[,] { { 0.33595 } };

       /* initialize trained weights, v_ij and w_jk */
       //weights for Layer 1
double[,] v_ij = new double[4, 2] { { -1.7342, -0.42877 }, { -3.3646, 3.3185 },
              { -2.5225, 0.72427 }, { 1.2735, 0.87515 } };
       //weights for Layer 2
double[,] w_jk = new double[,] { { -0.92105, 0.57006, -0.029867, -0.37036 } };

/* activation in layer 1 */
       // the net input to the hidden unit j (Z_in_j);
double[,] z_in_j = matrixSummation(v_0j, matrixMultiplication(v_ij, vector_x));
              double[,] z_j = tansig(z_in_j);    // the output signal of Zj

       /* activation in layer 2 */
       //y_ink is the net input to output unit k
double[,] y_in_k = matrixSummation(w_0k, matrixMultiplication(w_jk, z_j));
       //yk is the output signal of output unit k
double[,] y_k = tansig(y_in_k);

       //to get the final value from the S(B(K))
s_b_key = y_k[0, 0];
 }
```
Figure A3.3. Stress measurement model based on keystroke dynamics using FFBP neural net architecture

```
Public double fire_FFBP_MouseKey_Rules(double[6,1] vector_x)
{     //vector_x is the individual mouse dynamics input formed by MS, MID,
MIO,MCL
       /* Data of weights and biases for Layer 1 & 2 are gained from Matlab */

/* initialize v_0j=bias on hidden unit j; w_0k=bias on output unit k  */
       //bias for Layer 1
double[,] v_0j = new double[,] { { 1.7808}, { 1.4335 }, { 0.46728}, { 1.2888 },
              { 1.2246 }, { -2.4101 } };
       //bias for Layer 2
double[,] w_0k = new double[,] { { -0.51432 } };

/* initialize trained weights, v_ij and w_jk */
       //weights for Layer 1
double[,] v_ij = new double[4, 4] { { 1.4866, 0.057706, 0.11305, -1.365,       -
              0.02595, -0.30187 }, { 0.33451, -1.3325, 1.624, 0.48431, 1.0687,
              -1.7739 }, { -0.45822, -0.99703, 0.21882, 0.36153, 0.78456, 1.1514
              }, { -0.087941, 2.4065, 0.54953, 0.17444, 0.024405, -0.29707 }, { -
              0.10071, 2.2259, -0.19631, -0.53681, 0.029268, -0.78465 }, {
              1.2943, 0.48779, 0.899, -0.56317, -0.35981, 0.41524 } };
       //weights for Layer 2
       double[,] w_jk = new double[,] { { 1.7084, -0.0038696, -0.37506, 0.23839,
              -0.4541, 0.29879 } };

double[,] z_j;
double[,] y_in_k;
double[,] y_k;

/* activation in layer 1 */
       // the net input to the hidden unit j (Z_in_j);
double[,] z_in_j= matrixSummation(v_0j, matrixMultiplication(v_ij, vector_x));
       // the output signal of Zj
double[,] z_j=tansig(z_in_j);

/* activation in layer 2 */
       //y_ink is the net input to output unit k
double[,] y_in_k= matrixSummation(w_0k, matrixMultiplication(w_jk, z_j));
       //yk is the output signal of output unit k
y_k = tansig(y_in_k);

//to get the final value from the S(B(M,K))
s_b_mousekey = y_k[0,0];
  }
```
Figure A3.4. Stress measurement model based on mouse and keystroke dynamics using FFBP neural net architecture