



THE HONG KONG
POLYTECHNIC UNIVERSITY

香港理工大學

Pao Yue-kong Library

包玉剛圖書館

Copyright Undertaking

This thesis is protected by copyright, with all rights reserved.

By reading and using the thesis, the reader understands and agrees to the following terms:

1. The reader will abide by the rules and legal ordinances governing copyright regarding the use of the thesis.
2. The reader will use the thesis for the purpose of research or private study only and not for distribution or further reproduction or any other purpose.
3. The reader agrees to indemnify and hold the University harmless from and against any loss, damage, cost, liability or expenses arising from copyright infringement or unauthorized usage.

IMPORTANT

If you have reasons to believe that any materials in this thesis are deemed not suitable to be distributed in this form, or a copyright owner having difficulty with the material being included in our database, please contact lbsys@polyu.edu.hk providing details. The Library will look into your claim and consider taking remedial action upon receipt of the written requests.

**MINING HUMAN INTERACTION SIGNALS FOR
HUMAN AFFECTIVE AND COGNITIVE STATE
DETECTION**

WANG JUN

PhD

The Hong Kong Polytechnic University

2021

The Hong Kong Polytechnic University

Department of Computing

**Mining Human Interaction Signals for Human Affective
and Cognitive State Detection**

Wang Jun

A thesis submitted in partial fulfillment of the requirements
for the degree of Doctor of Philosophy

February 2021

CERTIFICATE OF ORIGINALITY

I hereby declare that this thesis is my own work and that, to the best of my knowledge and belief, it reproduces no material previously published or written, nor material that has been accepted for the award of any other degree or diploma, except where due acknowledgment has been made in the text.

X _____

Wang Jun (Name of student)

Abstract

Human-Aware AI Systems are able to provide timely support to humans in different situations, based on the understanding of their mental state and intentions. As a step towards developing such systems, this thesis focuses on understanding humans' affective state and cognitive process when interacting with computers.

For the affective state understanding, this thesis focuses on studying mental stress, one of the most prevalent negative affective states encountered by users when interacting with computers. Mental stress can affect both users' mental health and the quality of user experience. Previous work often detects mental stress based on bio-signals and physical information collected via intrusive devices, which is not feasible in daily life. Other studies have recently focused on non-intrusive stress detection approaches relying on behavioral signals, especially gaze and mouse behaviors. However, the consistency of users' behavioral patterns has seldom been investigated by previous studies. Our approach proposes a stress detection method that considers the consistency of gaze and mouse behaviors. Based on the result of the analysis on the subjects' behaviors during the experiment, we discover that when a user is stressed, his/her eye gaze behavior patterns are more consistent, and the proposed stress detection method can detect stress efficiently in a common e-Learning evaluation task. To take one step further, we find that most of the previous stress detection methods rely on the knowledge of user interface (UI) layout information, limiting their methods' generalizability, especially for tasks with dynamic UIs. Therefore, MGAttraction, a rotation- and translation-invariant coordinate system, is proposed to model the relative movement between gaze and mouse in this thesis. Based on that, a UI-agnostic stress detection method is proposed, which is able to work in the dynamic UI

environment. We evaluate the performance of our method on a web searching task with dynamic UI. With the gaze location tracked by a commercial eye-tracker, the proposed UI-agnostic stress detection method can successfully detect stress and outperform the performance of state-of-the-art methods. To further generalizability, we explore the feasibility of substituting webcam video in place of eye-tracker gaze locations. The resulting system, using the webcam to estimate the gaze locations, is able to detect mental stress without sacrificing too much accuracy.

For the cognitive process understanding, this thesis studies the process of writing, which is one of the most common activities undertaken on a computer. Given that writing is an intensively cognitive process, it makes sense that users' age and the genre of writing that is being produced would affect the user behaviors. However, only a few studies have explored this relationship. In this thesis, the eye gaze behaviors and the typing dynamics in different writing stages are investigated for subjects in different age-groups: child, college, and the elderly, producing original articles in different genres: reminiscent, logical, and creative. We design both statistics-based features and sequence-based features to infer the cognitive process of writing. Statistics-based features focus on modeling the overall gaze-typing behaviors during the entire writing period, and sequence-based features focus on the transition of the gaze-typing behaviors with the development of the writing. Evaluation results illustrate that both the age-factors and article genres affect the writing behaviors, and our statistics-based and sequence-based features can successfully capture the differences in writing behaviors.

Besides the writing process, this thesis also investigates the process of summarizing. Summarizing is a multitasking process requiring subjects to perform the reading/understanding process and writing process iteratively. In this thesis, we analyze users' cognitive process when carrying out summarizing tasks, as

evidenced through their eye gaze and typing features, to obtain insight into different difficulty levels. Multimodal features are extracted from different summary writing phases, including reading and understanding the source, referring to content from the sources, rereading the already-generated text, typing the generated texts into the computer, and reviewing the already-generated texts. Each phase is determined based on the characteristics of gaze behaviors and typing dynamics. A classifier is constructed based on the multimodal features, which can discriminate the difficulty level of each summary writing in a decent performance and outperforms other models constructed on the part of modalities or a single modality. The potential reasons for the decent performance of multimodal features are also investigated.

Experimental results in this thesis show success in detecting mental stress and writing cognitive process based on gaze and hands behaviors, which implies the effectiveness of behavioral signals used by human-aware AI systems to understand users' affective state and cognitive process.

Publications Arising from the Thesis

- [1] **Wang, J.**, Huang, M. X., Ngai, G., & Leong, H. V. (2017, October). Are you stressed? Your eyes and the mouse can tell. In 2017 Seventh International Conference on Affective Computing and Intelligent Interaction (ACII) (pp. 222-228). IEEE.
- [2] **Wang, J.**, Fu, E. Y., Ngai, G., Leong, H. V., & Huang, M. X. (2019, April). Detecting stress from mouse-gaze attraction. In Proceedings of the 34th ACM/SIGAPP Symposium on Applied Computing (SAC) (pp. 692-700).
- [3] **Wang, J.**, Fu, E. Y., Ngai, G., & Leong, H. V. (2019, July). Investigating Differences in Gaze and Typing Behavior Across Age Groups and Writing Genres. In 2019 IEEE 43rd Annual Computer Software and Applications Conference (COMPSAC) (Vol. 1, pp. 622-629). IEEE.
- [4] **Wang, J.**, Ngai, G., & Leong, H. V. (2020, October). Hand-eye Coordination for Textual Difficulty Detection in Text Summarization. In Proceedings of the 2020 International Conference on Multimodal Interaction (ICMI) (pp. 269-277).
- [5] **Wang, J.**, Fu, E. Y., Ngai, G., & Leong, H. V. Investigating Differences in Gaze and Typing Behavior Across Writing Genres. Accepted and to appear in International Journal of Human-Computer Interaction, under publication.
- [6] **Wang, J.**, Fu, E. Y., Ngai, G., & Leong, H. V. (under review). Detecting Stress from Mouse-Gaze Attraction.

Acknowledgements

I would like to express my sincere appreciation to all the individuals who helped me to complete this degree. The following acknowledgments are by no means exhaustive, for which I apologize.

I am deeply thankful and indebted to the full support and professional guidance of my supervisors, Dr. Grace Ngai and Dr. Hong Va Leong. Without their encouragement, constructive criticism and helpful advice, my thesis work would have been an overwhelming and frustrating pursuit.

I have had great pleasure working with members in CHILab: Dr. Michael Xueling Huang, Dr. Jiajia Li, Dr. Yujun Fu, Zhongqi Yang. The creativity of all my colleagues has been a constant inspiration throughout my time.

Finally, I would like to acknowledge my parents and my girlfriend, who unconditionally support me in all my decisions.

Table of Contents

CERTIFICATE OF ORIGINALITY	III
ABSTRACT	IV
PUBLICATIONS ARISING FROM THE THESIS	VII
ACKNOWLEDGEMENTS	VIII
TABLE OF CONTENTS	IX
LIST OF FIGURES	XV
LIST OF TABLES	XIX
1 INTRODUCTION	1
1.1 BACKGROUND AND MOTIVATION	4
1.1.1 INFERRING USERS' AFFECTIVE STATE BASED ON GAZE AND MOUSE BEHAVIORS	4
1.1.2 INFERRING USERS' COGNITIVE PROCESS BASED ON GAZE AND TYPING BEHAVIORS	6
1.2 STUDY OVERVIEW	10
1.2.1 DETECTING MENTAL STRESS VIA GAZE AND MOUSE BEHAVIORS	11
1.2.2 INFERRING COGNITIVE PROCESS OF WRITING AND SUMMARIZING VIA GAZE AND TYPING BEHAVIORS	13
1.3 THESIS AIMS AND OUTLINE	16
2 LITERATURE REVIEW	19
2.1 GAZE AND HANDS COORDINATION	19

2.2	AUTOMATIC STRESS DETECTION	21
2.2.1	PHYSIOLOGICAL SIGNALS BASED STRESS DETECTION	22
2.2.2	BEHAVIORAL SIGNALS BASED STRESS DETECTION	24
2.2.3	CONTINUOUS DAILY STRESS DETECTION APPROACHES	26
2.3	COGNITIVE PROCESS OF WRITING AND GAZE AND TYPING BEHAVIORS IN WRITING	27
2.4	GAZE AND TYPING BEHAVIORS IN SUMMARIZING	31
3	<u>INFERRING USERS' AFFECTIVE STATE BASED ON GAZE AND MOUSE BEHAVIORS</u>	34
3.1	STRESS DETECTION IN STATIC UI ENVIRONMENT	35
3.1.1	INPUT SIGNALS PREPROCESSING	36
3.1.2	FEATURE EXTRACTION	38
3.1.2.1	Modelling Gaze and Mouse Movement Patterns	38
3.1.2.2	Modelling Gaze-Mouse Coordination	40
3.1.2.3	Modeling Consistency of Gaze and Mouse Movement Behaviors	43
3.1.3	CONSTRUCT DATASET FOR STRESS DETECTION IN STATIC UI ENVIRONMENT	44
3.1.4	QUESTION-LEVEL STRESS DETECTION	46
3.1.5	SESSION-LEVEL STRESS DETECTION	49
3.2	STRESS DETECTION IN DYNAMIC UIs ENVIRONMENT	52
3.2.1	MGATTRACTION: MODELING MOUSE AND GAZE RELATIVE	

MOVEMENT	52
3.2.2 PREPROCESSING MGATTRACTION SIGNALS	56
3.2.3 INFERRING MENTAL STRESS FROM MGATTRACTION SIGNALS	57
3.2.4 CONSTRUCT DATASET FOR STRESS DETECTION IN DYNAMIC UIs	63
ENVIRONMENT	63
3.2.5 EXPERIMENTAL EVALUATION OF STRESS DETECTION ON MGATTRACTION SIGNALS	66
3.3 WEBCAM-BASED STRESS DETECTION VIA GAZE AND MOUSE BEHAVIORS 73	
3.3.1 ESTIMATE GAZE LOCATIONS FROM WEBCAM VIDEO	74
3.3.2 INFERRING MENTAL STRESS BASED ON THE ESTIMATED GAZE LOCATIONS	76
3.4 SUMMARY	82
<u>4 INFERRING USERS' WRITING COGNITIVE PROCESS BASED ON GAZE AND</u> <u>TYPING BEHAVIORS</u>	<u>86</u>
4.1 CONSTRUCTING WRITING COGNITIVE PROCESS DATASETS	87
4.1.1 SUBJECTS' BACKGROUND AND ENVIRONMENT SETTING	87
4.1.2 EXPERIMENT DESIGN	89
4.1.3 OVERVIEW OF DATASETS	90
4.1.4 DATA DISTRIBUTION	93
4.2 INVESTIGATING THE EFFECT OF AGE-FACTORS	96
4.2.1 IDENTIFYING THE THINKING/TYPING PHASES THROUGH GAZE-TYPING	

DYNAMICS 97

4.2.2	EXTRACTING GAZE-TYPING FEATURES	100
4.2.3	FEATURE SELECTION	105
4.2.4	EVALUATION OF AGE-GROUP DETECTION	106
4.3	INVESTIGATING THE EFFECT OF ARTICLE GENRES	109
4.3.1	IDENTIFYING DIFFERENT WRITING PROCESSES	109
4.3.1.1	Types of Thinking Window	110
4.3.1.2	Types of Typing Window	111
4.3.2	EXTRACTING STATISTICS-BASED GAZE-TYPING FEATURES FROM TIME WINDOWS	113
4.3.2.1	Extracting Statistics-based Gaze-typing Features from Thinking Window	114
4.3.2.2	Extracting Statistics-based Gaze-typing Features from Typing Window	117
4.3.2.3	Extracting Statistics-based Gaze-typing Features from Transition Window	120
4.3.2.4	Building Session-level Statistics-based Gaze-typing Features	121
4.3.3	EXTRACTING SEQUENCE-BASED GAZE-TYPING FEATURES FROM SESSION	122
4.3.3.1	Modeling the Behavior Transition within a Session	122
4.3.3.2	Extracting Indicative Patterns from the Behavior Sequence	123
4.3.3.3	Sequence-based Gaze-typing Features	127

4.3.4	EVALUATION OF DETECTING ARTICLE GENRES	128
4.3.4.1	Understanding Statistics-based Gaze-typing Features	128
4.3.4.2	Understanding Sequence-based Gaze-typing Features	134
4.3.4.3	Evaluating the Performance of Writing Genre Detection	138
4.4	SUMMARY	142
5	<u>INFERRING USERS' COGNITIVE PROCESS OF SUMMARIZATION BASED ON GAZE AND TYPING BEHAVIORS</u>	146
5.1	CONSTRUCTING SUMMARIZING TASK DATASETS	147
5.1.1	EXPERIMENT SETTINGS	147
5.1.2	EXPERIMENT DESIGN	149
5.2	EXTRACTING MULTIMODAL FEATURES FROM INPUT SIGNALS	151
5.2.1	PRE-PROCESSING OF INPUT SIGNALS	151
5.2.2	EXTRACTING FEATURES FROM EYE-TRACKING SIGNAL	152
5.2.3	EXTRACTING FEATURES FROM KEYBOARD SIGNAL	160
5.2.4	EXTRACTING DURATION-RELATED FEATURES	161
5.3	EVALUATION OF MULTIMODAL FEATURES	162
5.3.1	FEATURE NORMALIZATION AND FEATURE SELECTION	163
5.3.2	PERFORMANCE OF DIFFICULTY LEVEL DETECTION	164
5.4	EXPLORING THE BENEFITS OF MULTIMODAL FEATURES	168
5.5	SUMMARY	171

6	DISCUSSION	173
6.1	PERFORMANCE EVALUATION FOR SUBJECTIVE EXPERIMENT	173
6.2	USER-INDEPENDENT MODEL VS. USER-DEPENDENT MODEL	174
6.3	INFLUENCES OF VARIOUS USER ENVIRONMENTS IN REAL LIFE	176
6.4	OTHER USEFUL SIGNAL MODALITIES UNDER THE SAME SETTINGS	177
6.5	IMPLICATIONS OF THE FINDINGS BEYOND THIS WORK	179
7	CONCLUSION AND FUTURE WORK	182
7.1	CONCLUSION	182
7.2	LIMITATIONS AND FUTURE WORK	184
	REFERENCES	187

List of Figures

Figure 1-1 The flow of this thesis. This thesis explores non-intrusive stress detection methods in both (a) the static UI environment and (b) the dynamic UI environment; and understanding users' cognitive process in (c) writing and (d) summarizing tasks.....	10
Figure 3-1 From signals to prediction: The system flow chart of feature extraction	35
Figure 3-2 Experiment interface and UI components	37
Figure 3-3 Procedure of extracting movement pattern features.....	40
Figure 3-4 An illustrative example of the gaze-mouse coordination extraction	41
Figure 3-5 Trends of CCR across different <i>lengram</i>	48
Figure 3-6 Procedure of extracting session-level features.....	50
Figure 3-7 Attraction coordinate system showing displacement (dotted arrow), velocity (solid arrow), mouse (blue), and gaze (red) information; solid points are real positions, and hollow points are projections. (a) shows an example of mouse and gaze trajectories. (b) illustrates the origin identification of the coordinate system. (c) shows mouse and gaze velocity decomposition based on attraction coordinate	53
Figure 3-8 Overall pipeline of feature extraction to infer mental stress from MGAttraction signal	57
Figure 3-9 Two example periods of gaze attraction and mouse attraction with a Type P segment (red) and a Type B segment (yellow) and a Type N segment (blue)	58
Figure 3-10 Illustration of features extracted in <i>FP</i>	61

Figure 3-11 Experiment interface for stress detection in dynamic UIs environment: (a) Question page (b) Searching results page and (c) Potential answer page.....	64
Figure 3-12 Experimental environment for stress detection in dynamic UIs environment.....	66
Figure 3-13 Dynamic UIs component detection methods: (a) Heuristic-based and (b) Content-based.....	69
Figure 3-14 Distributions of selected important features.....	72
Figure 3-15 Overall pipeline of estimating gaze locations from webcam video	74
Figure 3-16 Average error in pixels of estimated gaze locations for each subject	75
Figure 3-17 Error analysis with respect to screen regions. The average estimated gaze location error is smaller than <i>threserr</i> in the black areas	78
Figure 3-18 Generating the pupil movement histogram: (a) showing the detected landmarks and pupil center (b) showing the segmentation of the eye image based on the landmarks and (c) showing the 2-D histogram encoding the probability of the pupil center appearing in each zone	79
Figure 4-1 An example of the pop-up candidates box. The user input the Latin text "pin' yin' shu' ru' fa", shown on the top of the candidates box, and five corresponds Chinese words/phrases are generated automatically by the system, shown below the user input. The user can either choose the correct mapping by pressing the number key or press "space" to select the first option.....	88
Figure 4-2 Experiment environment	89
Figure 4-3 Cumulative distribution function of γ for all subjects.....	91

Figure 4-4 Vocabulary usage across different writing genres for touch typists and non-touch typists: percentage of vocabulary in the article belonging to the top N frequently used Chinese characters, where N equals 500, 1000, 1500 and 2000..... 94

Figure 4-5 The three types of time windows and their correspondence with the appearance of the pop-up candidates box. Two adjacent typing-windows are merged if the time gap is less than 750 ms. The duration for each transition window is $2\Delta t$, where Δt is 250 ms. 98

Figure 4-6 Overview of feature extraction. $FThinking$, $FTyping$, FTr are window-level feature vectors and \phiThinking , \phiTyping , ϕTr are session-level feature vectors for each window type. ϕ is the final overall session feature vector that aggregates all information across all window types and all individual windows..... 101

Figure 4-7 Overview of feature extraction of statistics-based gaze-typing features 113

Figure 4-8 Illustration of the features that describe the reread texts 116

Figure 4-9 Illustration of the features that describe the staring point 116

Figure 4-10 Examples of two patterns, which have the same total occurrence times but gave different trend distance weighting 125

Figure 4-11 Generating behavior subsequences from session data 126

Figure 4-12 Top-100 normalized rf weights and Top-100 normalized $rf \cdot td$ weights for touch typists and non-touch typists..... 135

Figure 4-13 Examples of how the td term further distinguishes the discriminating power of patterns with the same td weight... 136

Figure 4-14 Overall performance trends of writing genre detection approach with a different number of partitions $npar$ and number of indicative patterns selected $nselect$ 139

Figure 4-15 Performance trends of writing genre detection approach trained on the touch typists dataset and non-touch typist dataset together	141
Figure 5-1 Experimental interface for summarizing task	148
Figure 5-2 Experimental environment and the overall framework of the multimodal approach	148
Figure 5-3 An example of the pop-up candidates words selection window..	154
Figure 5-4 Examples of two types of <i>Spread</i> sub-scanpath	155
Figure 5-5 Box plot of selected features	166
Figure 5-6 Performances (CCRs) contributed by different modalities	170

List of Tables

Table 3-1 Gaze-mouse coordination features	42
Table 3-2 Performance of stress detection at question-level	47
Table 3-3 Performance of stress detection at session-level.....	52
Table 3-4 <i>FP</i> : Features extracted from the P segment.....	61
Table 3-5 <i>FB</i> : Features extracted from the B segment.....	62
Table 3-6 Classification performance for stress detection based on MGAttraction signals in dynamic UIs environment	67
Table 3-7 Confusion matrix for stress detection based on MGAttraction signals in dynamic UIs environment	68
Table 3-8 Performance of different approaches in dynamic UIs task	70
Table 3-9 <i>FFacial</i> : Facial features extracted from webcam video	75
Table 3-10 Stress detection performance based on estimated gaze locations	76
Table 3-11 Confusion matrix for stress detection based on estimated gaze locations	76
Table 3-12 Stress detection performance based on estimated gaze locations	79
Table 3-13 <i>FPupil</i> : Features extract from I(t) and D(t) signals to describe pupil movement	81
Table 3-14 Performance of different approaches in dynamic UIs task	82
Table 3-15 Confusion matrix for webcam-based stress detection with pupil moment features.....	82
Table 4-1 Detailed composition of the datasets from each age group	92

Table 4-2 Number of words among different writing genres.....	93
Table 4-3 Typing speed in words per minute (WPM) among different writing	93
Table 4-4 Results of the expert review – Detailed ratings of articles written by subjects from different age groups	96
Table 4-5 P-values of ANOVA tests on article genre ratings for different age groups.....	96
Table 4-6 Window-level features extracted from the thinking window	102
Table 4-7 Window-level features extracted from the typing window	103
Table 4-8 Window-level features extracted from the transition window.....	104
Table 4-9 Selected indicative features of capturing differences among different age-groups.....	106
Table 4-10 Detailed performance of the age-group detection	107
Table 4-11 Confusion matrix of the age-group detection	107
Table 4-12 Types of phrases generated in Type L typing window.....	112
Table 4-13 Different types of thinking window and typing window based on gaze and typing activities	112
Table 4-14 <i>FO</i> : Features describing the behavior in Type O thinking window	114
Table 4-15 <i>FR</i> : Features describing the behavior in Type R thinking window	115
Table 4-16 <i>FF</i> : Features describing the behavior in Type F thinking window	117
Table 4-17 <i>FL</i> : Features describing the behavior in Type L typing window .	118

Table 4-18	<i>FU</i> : Features describing the behavior in Type U typing window	119
Table 4-19	<i>FN</i> : Features describing the behavior in Type N typing window	119
Table 4-20	<i>FTr</i> : Window-level features extracted from the transition window	120
Table 4-21	Overview of behavior types for different genres of writing.....	123
Table 4-22	Kruskal Wallis H test and Dunn's test results of significant statistics- based gaze-typing features for touch typists (R: Reminiscent, L: Logical, C: Creative)	131
Table 4-23	Kruskal Wallis H test and Dunn's test results of significant statistics- based gaze-typing features for non-touch typists (R: Reminiscent, L: Logical, C: Creative)	132
Table 4-24	Top-5 selected patterns for both touch typists and non-touch typists	137
Table 4-25	Confusion matrix of the article-category detection for touch typists	139
Table 4-26	Confusion matrix of the article-category detection for non-touch typists	139
Table 4-27	Article-category detection for different age groups.....	142
Table 5-1	Features of clustering <i>Spread</i> sub-scanpaths.....	154
Table 5-2	Summary of sub-scanpath in different categories	156
Table 5-3	<i>Fgazeunderstand</i> : Features extracted from <i>Spunderstand</i> .	157
Table 5-4	<i>Fgazeskim</i> : Features extracted from <i>Spskim</i>	159
Table 5-5	<i>Fkeyboarddynamics</i> : Features extracted from keyboard signal	160

Table 5-6 <i>Fkeyboardprocedure</i> : Features extracted from text generation procedure	161
Table 5-7 <i>Fduration</i> : Duration-related features	162
Table 5-8 Indicative features selected from potential features	164
Table 5-9 Classification performance for difficulty level detection by using multimodal model.....	165
Table 5-10 Confusion matrix for difficulty level detection by using multimodal model	165
Table 5-11 Features selected from each modality	169
Table 5-12 Average of pair-wise R ² for different modalities.....	170

1 Introduction

Human-Computer Interaction (HCI) has progressed from simply designing an interface that fits between humans and computers to taking on a more human-centered perspective [10]. Human-centered computing is an upcoming research field that focuses on designing and developing intelligent systems that can understand human beings through multimodal inputs. Human beings can be understood from different prospects, such as affective state and cognitive process. The affective state is the emotional response of interaction, and the cognitive process reflects the stage of information processing. Once a system is able to infer the affective state and cognitive process of a user, it can intelligently provide corresponding assistance for specific purposes, including improving productivity, healthcare concerns, etc. For example, some studies [8, 98, 106] proposed the cognitive load sense e-learning systems to keep learners' cognitive load is in the ideal range to optimize the learning outcomes. Also, some studies [22, 108, 118] constructed healthcare systems based on the multimodal signals collected from smartphone sensors to continuously track their health condition.

In human-centered computing, the affective state and cognitive process can be accessed through subjects' physiological signals and behavioral signals. Physiological signals include electrodermal activity signal (EDA), heart activity (ECG), blood activity (BVP), and pupil dilation signal (PD). Methods based on the physiological signals focus on extracting features to describe the indicative patterns of signals in different affective states or cognitive processes. Extracted features are then utilized for training a machine learning model, which can discriminate among different affective states and cognitive processes. Usually, methods based on the physiological signals can achieve high performance, but

they use intrusive devices to access users' bio-signals, making them impractical in daily life. Therefore, this thesis focuses on understanding users' affective state and cognitive process via the behavioral signals, especially their gaze behaviors, hands behaviors, and gaze-hand coordination behaviors. The reason for choosing gaze and hands as two primary modalities is that gaze and hands play essential roles during human-computer interaction, where the gaze is used to obtain the screen's information, and hands perform corresponding actions.

We first explore the approaches for inferring users' affective states based on gaze and mouse behaviors. In our work, we focus mainly on a special kind of affective state: mental stress. Although some stress detection approaches exist based on the behavioral signals and achieve good performance in their evaluations, not all of them can be directly utilized in real-world scenarios due to various limitations. One of the prevalent limitations is that most previous approaches rely on the user interface (UI) related information. However, extracting UI-related information in a dynamic UI environment in real-time is computationally consuming, as it often involves using computer vision techniques to recognize different UI components such as menus, buttons, captions and so on, which makes these approaches not practical. Another limitation of many previous approaches is that they require special equipment to collect behavior signals, which hinders their generalizability from being widely exploited by common users. Therefore, three significant challenges we are facing are 1) how to model gaze and mouse behaviors without relying on UI related information, 2) how to infer the affective state of stress based on gaze and mouse behaviors, and 3) how to access the gaze signal modality without relying on any special equipment. We believe this study will open up a new avenue for affective-aware user interfaces and numerous advanced HCI studies.

In addition, we also investigate approaches to infer users' cognitive process based on gaze and typing behaviors. In this part of the study, we focus on understanding the cognitive process in writing and summarizing tasks. To the best of our knowledge, most of the previous studies focus on investigating typing behaviors and gaze behaviors on copy-typing tasks – i.e., each subject is only required to type words from a source prepared in advance, rather than formulating his/her original writing thoughts. However, the type of behaviors in a copy-typing task can be expected to be quite different from daily usage, where the user is generating the content cognitively at the same time as he/she is typing the generated content into the computer. To fill in the gap, we, therefore, focus on exploring the writing/summarizing cognitive process when users are generating their own texts from three perspectives. The first perspective is to explore how age-factors of subjects affect the cognitive process of writing. The second perspective is to explore the influence of the writing cognitive process when subjects are writing different genres of texts based on gaze and typing behaviors, and the third is to investigate the effect of difficulty levels when subjects are performing summary writing. The challenge of understanding users' writing cognitive process is that the cognitive process keeps changing throughout the writing period. Segmenting the writing process so that different types of the cognitive process can be isolated as possible in each segmentation is important for further analysis and feature extraction. Meanwhile, compared with general writing, the cognitive process for summary writing is more complex, as it involves the reading comprehension process, the writing cognitive process, and the typing process. Therefore, this study would also help us understand how gaze and hands behave during a multitasking process involving reading and writing.

1.1 Background and Motivation

1.1.1 Inferring Users' Affective State Based on Gaze and Mouse

Behaviors

Human-centered computing is a relatively novel research area to make the computer be able to recognize users' emotions and respond intelligently to help users recover from the negative affective states [130]. To understand the users' affective state when interacting with computers, we focus on detecting mental stress in our work. Stress can be induced from both environmental sources and user-centric sources, where environmental sources include time pressure, noise level, etc., which are related to the physical environment, social environment, and computational environment. User-centric sources are associated with a user's background and the type of task he/she is performing [19]. Mental stress frequently occurs during interaction with computers. Since when a human interacts with a computer, especially with multimedia interfaces, a high volume of information in various formats can easily result in a high cognitive load, which causes mental stress [46]. A previous study illustrates that psychological stress responses often contain negative emotions, including annoyance, depressed [27]. Constantly exposed to stressful environments links to many health problems such as high blood pressure, diabetes, and cardiovascular problems [8]. In HCI, high-level stress may cause frustration in interacting with computer interfaces and reduce the effectiveness of interacting and the quality of user experience. Therefore, an automatic and intelligent stress detection method, which can continuously and implicitly monitor users' mental stress levels while users are interacting with computers, is valuable and compelling. Only if mental stress can be detected effectively first, a variety of cognitive load alleviation approaches can

further be applied to reduce mental stress.

Conventional stress detection methods rely on the processing of human physiological signals and physical information. EDA, ECG, BVP, and PD signal are often exploited as input modalities to detect mental stress [42, 103, 110, 119]. Although physiological signals-based approaches yield high performance on stress detection, they always require intrusive devices to access users' bio-signals. For example, to measure the EDA signal, two electrically conductive plates are needed to attach on the index and middle fingers, and it is no longer convenient to control the mouse and type on the keyboard with two plated attracted. Hence, such requirements make physiological signals-based approaches impractical in daily life. Moreover, bio-signals are also affected by other factors, including stimulus specificity and initial mental state level [84], which may result in the measurements of bio-signals are inconsistent. Compared with physiological signals-based approaches, non-intrusive stress detection draws more attention. Facial expression/facial cues [42, 81, 113], body postures [57, 63], voice of speech [40, 41], voice of environment [77], and social media engagement [72, 73] have been explored for stress inference. For facial expression and facial cues, these signals are sensitive to the noise and are greatly influenced by the initial state. Gaze and hands are two primary channels to interact with the computer, and their behaviors are often exploited to infer mental stress [44, 49, 55, 109, 114, 116, 120]. However, there are some downsides to these approaches. One of the major downsides is that methods proposed in some of these studies [109] rely on the prior knowledge of UI information to provide the context of interaction, which cannot work in the dynamic UI environment. Also, many of these approaches [49, 55] are only evaluated by simple UI layout tasks with relatively simple operations, so they are not certain about whether they can work effectively in the dynamic UI

environment, where most of the actual applications are carried out. Another downside is that some of these approaches [44, 114, 120] rely on special equipment, such as the force transducer, electromyography system, or capacitive pad, which reduces the generalizability of their methods.

Also, we find that gaze and hands behaviors are considered separately for most of the prior approaches, except StressClick [49]. However, gaze behaviors and hands behaviors are not independent of each other. Many previous studies illustrate that there exists a strong correlation between gaze movements and mouse movements during the interaction, named as the gaze-hand coordination, which depicts the relationship from the information received through the eyes to control, guide, and direct the hands in accomplishing a given task [23]. Therefore, in this thesis, we would like to propose an innovative UI-agnostic stress detection method based on gaze and mouse behaviors and gaze-hand coordination. Moreover, we construct our own dataset to evaluate the performance of our approach of detecting stress in the wild when a user is searching for information online in a dynamic UI environment.

1.1.2 Inferring Users' Cognitive Process Based on Gaze and Typing Behaviors

The cognitive process refers to the mental action or stage to accomplish a task, which involves thinking, knowing, remembering, judging, and problem-solving [37]. In different stages of the cognitive process, users may need different assistance from the computers. Therefore, it is essential for having a system, which is able to infer the cognitive process of users during the interaction. In this thesis, we focus on understanding users' cognitive process of writing, which forms a large proportion of daily computer usage. The cognitive process of writing has been well

explored in the psychology field. One of the popular writing cognitive process models proposed by Flower et al. [30] asserts that the writing process contains three major stages: planning, translating, and reviewing. In the planning stage, subjects are mainly generating and organizing writing ideas. During the planning stage, relevant writing material is retrieved from the long-term memory. In the translating stage, the writing ideas are converted into sentences. Formulated sentences are evaluated and revised in the reviewing stage. During the writing, users will substantially interact with the task environment by rereading previously generated texts and typing on the keyboard, and these interactions are the essential clues to infer the cognitive process. For example, there are some previous works that explored the relationship between the complexity of the writing task and the rereading behaviors [111, 118] and keyboard dynamics [71, 121]. Also, some works investigated the influence of typing skills on the gaze movement behaviors traveling between the screen and keyboard [29, 53, 89]. It is obvious that the cognitive process is different while writing different genres of articles. For instance, when a user is writing a diary, most sentences in the article are narrative to describe happened events. On the other hand, when composing a scientific journal, sentences are usually more logical and formal. Considering these cases by Flower's writing cognitive model, when writing a diary, a user will spend more time on the planning stage to recall the memory, but the scientific journal translating stage may take more time to explain it explicitly. However, to the best of our knowledge, there is no work exploring whether or how different writing genres affect the writing cognitive process. Therefore, in this thesis, we focus on understanding how writing genres affect the writing cognitive process and whether we can determine the differences in the writing cognitive process via gaze and typing behaviors.

In addition, most of the previous studies in writing are conducted in the English language. But English is unique in the sense that there is a direct mapping between the user's actions, which are keys that are typed, and the desired output, which is the texts to be generated. In other words, English texts can be directly inputted letter by letter, which is not the same for some languages such as Chinese, Kanji in Japanese, and Hindi. In these languages, words cannot directly input on a standard keyboard, and they usually need a two-step process: generating and committing. In the generating step, users need to type the phonetic reading of the words on a keyboard to approximate the target words, and in the committing step, users select the target from a group of word candidates with the same phonetic. Even though language modeling algorithms are utilized to adaptively shuffle the most likely options to the front of the candidate list, it is still reasonable to expect that the cognitive process of writing in Chinese would be different from writing in English.

One limitation of previous studies is that most of them focus on investigating gaze and typing behaviors in the copy-typing tasks. In the copy-typing task, subjects just need to type the sentences from a source prepared in advance into the computer without the process of formulating their own writing ideas and converting ideas into sentences. Thus, it can be expected that the gaze and typing behaviors of copy-typing tasks are quite different from the behaviors that subjects generate their own articles. We also notice that typing skill has a great impact on the gaze and typing behaviors of writing [89], which may overshadow the impact of the writing cognitive process. To address those limitations and challenges, we construct two datasets for both touch typists and non-touch typists by asking them to compose an article under a broad topic without any more constraints.

Besides investigating gaze and typing behaviors when users are composing

their own articles, we also explore gaze and typing behaviors when users summarize a document. Summarizing is a complex multitask composed of reading and writing processes, which frequently occur in daily computer interaction. As two common tasks for daily computer interaction, reading and writing's human behaviors are well investigated. However, it is less known how human behaviors change when these two processes are interleaved. Although some previous studies [122, 123] explore the cognitive process of summary writing, analyses in these studies are based on think-aloud protocols or retrospective questionnaires. Both methods are deficient. For example, think-aloud protocols require subjects to verbalize their concurrent thinking process. Such a requirement needs subjects to be aware of their mind thinking during writing, which will increase their cognitive load and affect their writing process [51]. Retrospective questionnaires relying on self-reporting after the experiment to ask subjects to recall all the thinking process details is not possible. Therefore, learning from the experience of investigating the writing cognitive process, we describe an investigation into the cognitive process of summary writing via analysis of subjects' gaze and typing behaviors. We hope that the proposed method could infer the cognitive process in an unobtrusive manner in as natural an environment as possible.

1.2 Study Overview

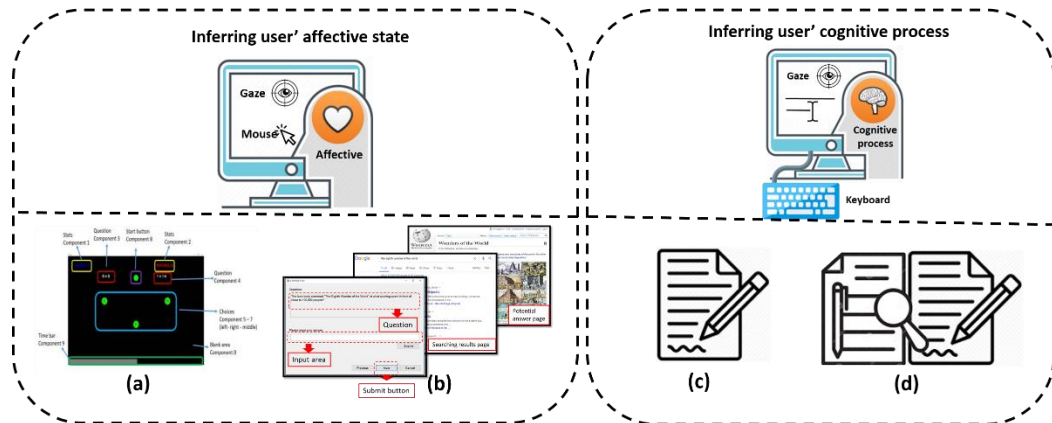


Figure 1-1 The flow of this thesis. This thesis explores non-intrusive stress detection methods in both (a) the static UI environment and (b) the dynamic UI environment; and understanding users' cognitive process in (c) writing and (d) summarizing tasks.

In this thesis, we study both users' affective state and cognitive process. The flow of this thesis is shown in Figure 1-1. In order to understand users' affective state, we focus on inferring the stress state based on gaze and mouse behaviors and their coordination during the interaction with the computer. The study starts from detecting stress in a simple interactive task with static UI. The MGAttraction (mouse-gaze attraction) coordinate system is proposed, which will be introduced comprehensively in Section 3.2.1. MGAttraction allows gaze and mouse behaviors to be modeled without relying on any UI information. An innovative UI-agnostic stress detection approach is designed based on the MGAttraction coordinate system. Finally, we extend our UI-agnostic stress detection approach by substituting the eye-tracker with the webcam to obtain the eye gaze positions on the screen. Another essential part of the content is to understand users' cognitive processes for both writing and summarizing. In our study, we mainly explore how the writing cognitive process is affected by the age-factors and genres of articles shown by the gaze and typing behaviors, also, how the cognitive process is

different when summarizing documents with different difficulty levels.

1.2.1 Detecting Mental Stress via Gaze and Mouse Behaviors

To infer the mental stress state when users are interacting with the computers, we first propose a non-intrusive stress detection method by extracting features from the gaze and mouse movement patterns with regard to UI layout. The features include the consistency of gaze and mouse movement patterns and the gaze-mouse coordination such as the correlation of the position, time of delay, and speed between gaze and mouse as features. Gaze and mouse movement patterns are modeled by the transition sequence of UI components that a user draws attention to, and the similarity among the transition sequences measures the consistency of gaze and mouse movement patterns. As far as we know, there is not too much work in affective computing that thoroughly investigates the consistency of attention transition sequences, and we hope the findings in our work could bring new knowledge to the community.

Since the gaze and mouse movement patterns are constructed based on UI layout, the proposed method can only work in a static UI environment which allows us to obtain the position of each UI component. By considering such a requirement, we build a simple interactive system based on a computerized multiple-choice math quiz. Our interface displays a math question, potential answer choice options, and performance statistics. Stress is induced through environmental means by imposing a time limit for each question and adding background noise. Our method yields an overall performance of mental stress recognition of around 66.4% accuracy on the question level and 82.9% on the session level, where each session contains 25 questions, over 9 different subjects. The ground truth of whether a subject is stressed is achieved through self-reporting.

One obvious limitation of the above method is that it relies on the prior knowledge of UI information to model gaze and mouse behaviors, making the method hardly works in a dynamic UI environment. Because extracting accurate UI related information in a dynamic UI environment is always highly computational consuming. In fact, a UI-agnostic stress detection method, which is able to work in a dynamic UI environment, is highly valuable and necessary since most of the actual applications are carried out with dynamic UIs. To overcome the limitation, we construct a new coordinate system, and in that system, gaze and mouse movements are no longer modeled based on the gaze and mouse on-screen locations relative to the application context. Inspired by previous studies of gaze-hand coordination, the relative movements between gaze and mouse are utilized to model their behaviors, and the mental stress could be inferred via their relative movements.

In order to better formulate the relative movement between gaze and mouse, we propose a new coordinate system, *mouse-gaze attraction*, or MGAttraction. As the name implies, MGAttraction measures the *attraction* between the gaze and mouse. This novel measurement considers both relative speed, position, and moving direction between gaze and mouse. The coordinate system of MGAttraction also has rotation- and translation-invariant characteristics. Therefore, MGAttraction coordinate system can be completely agnostic to the UI layout and invariant to the location and movement direction of the gaze and mouse. To infer mental stress based on the gaze and mouse attraction measured by MGAttraction coordinate system, we first divide the MGAttraction signal into multiple segments and then categorize them into different types based on the shape of the signal. Segment-level features are extracted from different types of segments to describe the changing of attraction between gaze and mouse inside

each segment period. Session-level features are extracted to augment the segment-level features as well as model the overall behaviors during the whole session. Both segment-level and session-level features are learned by the classifier to detect mental stress. To evaluate our agnostic stress detection method's performance, we conduct our human experiment by recruiting subjects to search for information online without considering UIs. Totally, there are 15 subjects recruited in the experiment. Stress is induced by imposing a time limit and adding background noise, and our method achieves 78.8% accuracy of detecting mental stress in the wild, beating the state-of-the-art around 20%, which fully illustrates the effectiveness of our method. Similarly, the ground truth of whether a subject is stressed is achieved by self-reporting.

For the method evaluated above, the gaze modality information is collected by an eye-tracker. However, eye-tracker is still considered as special equipment, which is not popular among common users. Relying on special equipment will significantly reduce the generalizability of our method. Therefore, a more consumer-friendly webcam-based approach with gaze locations estimated and pupil movement features is proposed. Our webcam-based approach achieves an accuracy of 73.7% in correct classification rate (CCR), which closes to the performance of using eye-tracker.

1.2.2 Inferring Cognitive process of Writing and Summarizing

via Gaze and Typing Behaviors

Writing tasks form a large proportion of daily computer usage. As shown by Chukharev-Khudilaynen et al. [21], the activity of writing is composed of the cognitive process and the generative (typing) process. During the cognitive process, a writer formulates his writing ideas and converts writing ideas into

contextual sentences. When a writer is in the typing process, he/she inputs sentences into the computer through the keyboard. These two processes are carried on alternately, and the final output is a piece of text that can be categorized into different genres. The cognitive process of writing has indeed been well studied in psychology. One of the most popular writing models is proposed by Flower et al. [30]. In their model, the writing cognitive process can be further divided into three major parts: planning, translating, reviewing, which can be utilized as a reference in our study.

In order to comprehensively explore gaze and typing behaviors in different stages of writing cognitive process as well as overcome the limitations of previous works, we construct our datasets by conducting human experiments and build datasets fulfill characteristics including: (1) subjects producing their own original articles in different genres, (2) subjects recruited are from different age groups, (3) subjects have different typing skill levels and (4) subjects typing in Chinese. To the best of our knowledge, our datasets are the first datasets in the community with those characteristics simultaneously, and all the following investigations are performed on these datasets.

We first explore whether and how age-factors affect the writing cognitive process shown by gaze and typing behaviors. Inspired by previous findings of Chukharev-Khudilaynen et al. [21], we divide a whole writing period into several thinking-windows and typing-windows, where a thinking-window is a continuous period of time of mainly formulating writing ideas and reviewing generated texts. A typing-window is a continuous period of time of inputting sentences into a computer. We then extract different groups of multimodal features for different time windows to describe gaze and typing behaviors during the time window period to detect the age groups that a subject belongs to. By learning from these

multimodal features, the classifier is able to achieve an age group detection accuracy of 83.3% utilizing a leave-one-subject-out cross-validation evaluation, which is 43% higher than baselines.

In addition to investigating the impact of age-factors, we further explore whether and how writing genres affect the writing cognitive process illustrated by gaze and typing behaviors. Similar to the process of analyzing the age-factors effect, we first divide the whole writing period into thinking-windows and typing-windows. But compared with the impact of age-factors, the effect of writing genres is more complicated and harder to be captured. Therefore, we refine both thinking- and typing-windows by further differentiating them into sub-categories based on the gaze and typing activities. Statistics-based gaze-typing features are then extracted from different time windows, which are multimodal features considering both gaze and typing behaviors in temporal and spatial domains. The purpose of statistical features is to model the macro behaviors of a subject during the writing activity. Besides statistics-based gaze-typing features, we also extract sequence-based gaze-typing features, which capture the change in subject behaviors as the writing activity progresses. Finally, a machine learning model is developed to distinguish the writing genres based on the statistics-based and sequence-based gaze-typing features. Evaluated by the leave-one-subject-out cross-validation, our final model is able to achieve the overall performance of 88.4% correctness of writing genres detection.

Besides writing, summarizing a document is also a common task for daily computer usage, but without being thoroughly investigated. In this thesis, we focus on analyzing whether and how the difficulty levels affect the cognitive process of summarizing shown by gaze and typing behaviors. Summarizing can be considered as a complex task that requires a person to multitask between reading

and writing. Compared with simple writing, more processes need to be considered, including reading and comprehending the document's key points and referencing the summarized document during writing. Therefore, we first identify the different cognitive stages based on eye gaze scanpath and keyboard typing activities, where the cognitive stages include understanding the document to be summarized, referencing the document while summarizing, rereading generated texts as typing. Then, features that model reading and writing behaviors in different cognitive processes are extracted. We then exploit these features to construct a classifier to predict the difficulty level of the summarization task. The performance of the classifiers is evaluated in our summary writing dataset. Finally, our classifiers are able to achieve 91.0% accuracy at determining difficulty levels.

In order to better model reading and writing behaviors, the multimodal approach [133] has been applied in this study. The input signals include eye-tracker signal, keyboard event signal, and screen video recording. Evaluation results illustrate that the multimodal approach outperforms other models that use only a single modality. For this finding, we also explore potential reasons for the performance improvement of the multimodal approach so that the knowledge can be generalized to other related problems.

1.3 Thesis Aims and Outline

The aims of this thesis, as outlined in the study overview, are as follows:

- To propose a non-intrusive stress detection method by using the consistency of gaze and mouse behaviors and investigate how mental stress influence the consistency of gaze and mouse behaviors
- To propose an innovative MGAttraction coordinate system to quantify the gaze and mouse movements without depending on any UI related information

and propose a UI-agnostic stress detection approach based on the MGAttraction coordinate system, which can detect stress in a dynamic UI environment

- To explore the cognitive process of writing by investigating how age-factors and writing genres affect gaze and typing behaviors during a writing process and identify a set of statistical- and sequence-based gaze-typing features, which can be used to discriminate different age-groups and writing genres

- To investigate the cognitive process of the summary writing task through gaze and typing behaviors and build multimodal features fusing information from different input channels, which are able to determine the difficulty level of the summary task

Chapter 2 presents the literature reviews on the research works about gaze-hand coordination, stress detection, the cognitive process of writing, and summarizing.

Chapter 3 introduces two stress detection approaches for both the static and dynamic UI environment as well as MGAttraction, a rotation- and translation-invariance coordinate system to measure the attraction between gaze and mouse. This study shows that the proposed approaches can successfully detect mental stress based on gaze and mouse behaviors. At the end of this chapter, a webcam-based stress detection without relying on any special equipment such as the eye-tracker is proposed to further improve our stress detection approach's generalizability.

Chapter 4 investigates how the age-factors and writing genres affect the writing cognitive process based on gaze-typing behaviors. Statistics-based and sequence-based gaze-typing features are designed to capture the differences in writing behaviors in different writing phases. The evaluation results illustrate that both age-factors and writing genres affect the writing cognitive process in different

ways, and the classifiers constructed based on the extracted features can discriminate the age group of a subject and the genre of the article being written by a subject.

Chapter 5 explores the cognitive process of a user when performing the summarization task in different difficulty levels. Multimodal features are extracted from both the reading/understanding phase and the writing phase to model the user's eye-gaze behaviors and typing dynamics with the development of summary writing and successfully capture the differences of behaviors when summarizing texts in different difficulty levels.

Chapter 6 presents the discussion of this thesis and chapter 7 shows the limitations of this thesis and the potential future work.

2 Literature Review

This thesis aims to understand the users' affective state and cognitive process when interacting with computers via gaze and hands behaviors. More specifically, we focus on understanding a special kind of affective state: stress state and the cognitive processes of writing and summarizing. Therefore, this chapter begins with a literature review about gaze and hands coordination during the interaction. To better understand the stress state and the stress detection approaches, we first review the studies about background knowledge about stress and the risk of stress followed by the stress detection approaches based on the physiological signals, the behavioral signals, and the continuous daily stress detection approaches. This chapter also reviews the studies about the cognitive process of writing, the gaze and typing behaviors during the writing, and how typing skills affect the gaze and the mouse behaviors while writing. At the end of the chapter, the previous works about the cognitive process of summary writing are presented. Based on that, this chapter outlines the rationales for the proposed studies.

2.1 Gaze and Hands Coordination

Gaze and hands coordination captures the relationship between eye gaze movements and hands movements when a user interacts with a computer, which is essential for daily human-computer interaction. The reason for eye gaze movements and hands movements are correlated to each other is that a user first receives the on-screen information through his/her eyes, and then hands are controlled, guided, and directed in accomplishing a given task based on the processed information [23]. Gaze and hands coordination generally can be divided into gaze and type coordination and gaze and cursor coordination. For the gaze and type coordination, Inhoff et al. [50] point out that the gaze is expected to be

four characters to the right of (leading) the typed character in the copy-type task based on the 19000 observations. Xu et al. [124] use the gaze and type coordination to predict users' visual attention on the graphical user interfaces with windows, icons, menus, and pointer. One of the frequent behavior patterns they find in their experiment is that a user's eye gaze always focuses near the caret when he/she is editing texts. Papoutsaki et al. [89] extend findings by divide subjects into touch typists and non-touch typists, and they find that the average distance between the caret and the gaze position is 192 *pixels* across all subjects. For touch typists, the average distance is 160 *pixels*, smaller than the average distance of non-touch typists, which is 352 *pixels*. In their study, they also examine that the closest distance between the caret and the gaze position is 210 *ms* after the keypress for touch typists compared with 540 *ms* after the keypress for non-touch typists. Jiang et al. [52] further expand the scope of gaze and type coordination from the computer interaction to the smartphone interaction. In their experiment, they observe that touch typists spend around 60% of the time focusing on the touchscreen keyboard, which is significantly larger than the percentage of time of focusing on the physical keyboard, which is about 20% of the time. Also, they find that the frequency of gaze shifts from the content area to the keyboard area is much higher in mobile typing, which is about 3.8 times more frequent than physical keyboard typing.

For the gaze and cursor coordination, Bieget al. [16] find that there are two main gaze and cursor coordination strategies that serve for different scenarios in the search and selection tasks. First, if a subject wants to select a target whose approximate location is known, he/she moves the mouse directly to the target without gaze guidance. Second, if the target's approximate location is unknown, he/she parallelizes searching and pointer movements to minimize the amplitude of

the acquisition movement. Rodden et al. [97] analyze the gaze and cursor coordination on the web searching result pages. They report that mouse moving is mainly for click for most cases, but *following the eye horizontally* and *highlighting a particular result* indicates that a user is processing the content. Liebling et al. [29] investigated the gaze and cursor coordination in realistic task settings, and they illustrate that the gaze is leading the mouse for about 60% of the cases, which is affected by the kind of target interacted with and the pre-experience of the task. They also explore the coordination of the gaze and cursor at different moments before the click and show that the largest distance between the gaze and cursor is 171 pixels appears at 1000 *ms* before the click, and then the distance keeps decreasing to 74 pixels until 250 *ms* before the click. Finally, the distance expands to 89 pixels at the click moment. Deng et al. [25] illustrate a similar finding that the gaze movements lead the cursor about 2.96 ± 1.94 (mean \pm STD) degrees in the tracing task, and the distance between gaze and cursor shows no significant difference among different shapes of trajectories. Weill-Tessier et al. [123] extend the gaze and cursor coordination to the tablet interaction and show that gaze leads the finger about 356 *ms* to the touching target, and the distance between gaze and finger is around 159 pixels.

2.2 Automatic Stress Detection

Stress is first documented by Selye [60] in 1956 that stress can be considered as a response of the body to external stimulus. External stimulus, also known as stressors, leads to the internal body's chemical and hormonal changes to produce the stress response. Stressors can be divided into two categories: environmental stressors and user-centric stressors, where environmental stressors include the time, temperature, luminance, noise level, etc. and user-centric stressors contain the

user's profile, such as age, gender, social status, and the kinds of activities are performed by the user [19]. From a high-level point of perspective, stress can be categorized into two types: acute stress [65], which is a short-term response to a stressful event, and chronic stress [91], which is a response to pressure for a chronic period of time. Both types of stress can lead to various emotional and physical health problems [7, 104]. Hence, it is valuable to explore automatic stress detection methods. This section starts with an overview of the stress detection approaches based on the physiological signals followed by the approaches based on the behavioral signals and the continuous daily stress detection approaches.

2.2.1 Physiological Signals Based Stress Detection

Conventional stress detection is based on the analysis of physiological signals and physical information. The prevalent input modalities include electrodermal activity signals, heart activity, blood activity, pupil dilation, and the multimodal approach to achieve better performance. Healey et al. [42] detected stress during a driving task by continuously processing electrocardiogram, electromyogram, skin conductance, and respiration signals taken over 5-minute windows. Statistical features, spectral power features, and features to characterize orienting responses were extracted to construct the model, which achieves over 97% accuracy in real-world driving tasks. Wagner et al. [119] exploit the same physiologic modalities to infer four classes of emotion: joy, anger, sadness, and pleasure, which is triggered by music. They explore a variety of feature selection methods and feature reduction methods with different machine learning models. They achieve up to 92% accuracy, which is around 12% improvement compared with the performance achieved without using any feature selection or reduction methods. Sun et al. [110] present the activity-aware mental stress detection based on electrocardiogram,

galvanic skin response, and accelerometer signals. Stress in their study is induced by mental arithmetic tasks with time limit pressure, while subjects are in different activities: sitting, standing, and walking, and they obtain 80.9% accuracy without requiring the controlled laboratory setting. In their work, they also point out that the accelerometer signal is essential in stress detection to help determine different physical activity conditions, which has a strong impact on spectrum features of physiological signals. Sierra et al. [103] propose a fuzzy expert system to determine an individual's stress level by analyzing galvanic skin response and heart rate signals. In their experiment, stress is induced through hyperventilation and talk preparation. Their system achieves over 90% accuracy for the 3-5 seconds signals acquisition period and 99.5% accuracy for the 10-second period. Barreto et al. [11] and Ren et al. [96] show that the pupil diameter, which is controlled by the autonomic nervous system, provides a strong indication of stress. They show that after involving the pupil dilation signal into stress recognition, the performance of detection can be improved significantly compared with only used physiological signals. It is obvious that most of the physiological signals based stress detection methods can achieve decent performance around 90% accuracy of recognizing stress, but one significant drawback always inherited by these methods is that they are intrusive methods and special equipment is required to attach to the users, which may cause psychological side effects.

For non-intrusive stress detection, facial expressions are one of the prevailing modalities for stress detection. Bosch et al. [67] detect affective states, including boredom, confusion, delight, engagement, and frustration when interacting with the educational game. Facial action units and head pose are extracted from the video as features and achieve an accuracy of 65% for the overall classification of affect. In their study, they note that the length of the time window, which is used

to extract features, needs to be various for different affective states to achieve optimal performance. For example, confusion detection works better with a larger window length comparing with delighted. Viegas et al. [113] build a dataset contains 114 different subjects. Each subject accomplishes three typing phases before the stressor, after the stressor, and after relaxation, and each phase lasts 15-minute. The stressor is a multitasking exercise with social evaluation. They extract 18 facial action units and build a random forest classifier to determine different phases from videos and obtain an average accuracy of over 97% and 50% accuracy for the subject dependent and independent models. Abouelenien et al. [1] detect acute stress through thermal imaging. They extract thermal features from thermal facial images of each subject. Thermal features are used to describe the distribution of colors in the Hue Saturation Value space. Their experiment results illustrate that by fusing with thermal features, the relative accuracy can improve 26.6% performance over the heart rate and skin conductance features and 38.2% performance over the respiration rate features.

2.2.2 Behavioral Signals Based Stress Detection

Besides physiological signals and physical information, human behaviors are also linked to the level of stress. Haak et al. [39] observe that when a human is in stressful situations, he/she tends to show a higher frequency of eye blinks. With further investigation, they discover that the occipital lobe, a specific brain area, is highly activated while blinking. Hernandez et al. [44] show that under the stressed condition, most of the users (>79%) in their experiments consistently type the keyboard with more forceful typing pressure and click the mouse with a greater amount of mouse contact. Ciman et al. [22] measure the differences in smartphone interaction between users' relaxed and stressed states. In their study, they mainly

investigate four kinds of smartphone interactions, including *scroll*, *swipe*, *touch* and *text input*, Where scroll, swipe and text input behaviors can be utilized for stress classification. They find that scroll features, such as the length, the duration, the speed, and the delta speed of scrolling and swipe features, such as the length, the duration, touch size, and touch pressure have weak correlations with the stress state, but the stress assessment model built based on these features can achieve decent performance, where F-measure of the scroll is 0.79 and F-measure of the swipe is 0.85. For text input features as writing speed and the error rate show significant correlations with stress level. Paredes et al. [90] model the human arm while driving by using the mass spring damper (MSD) model and find that when people are driving under stress, the arm's muscle tension is significantly higher than the calm state. A similar finding is also found by Sun et al. [109]. They apply the MSD model to the arm while holding the mouse. After analyzing the mouse trajectories for *point – and – click*, *drag – and – drop*, and *steering* mouse operations, they suggest that the arm muscle is stiff under the stressed condition.

StressClick, proposed by Huang et al. [49], is the closest work to our study. StressClick detects stress based on the mouse and gaze behaviors around each mouse click. They illustrate that when a subject clicks a target, the closest fixation duration preceding/during a click and the reaction latency after a click are negatively correlated to whether under the stressed condition, since under the stressed condition, a subject tends to conduct operation more rapidly. A stress detection system is constructed based on the findings, which achieves 74.0% accuracy. However, StressClick is only evaluated under a static UI environment, which may not be effective in the dynamic UI environment. Another drawback of StressClick is that it only considers a small time window around each click but

ignores other time periods. It means that most of the information is discarded without being utilized by their method, which results in their method may not be robust enough to work efficiently in the task with complex interactions.

2.2.3 Continuous Daily Stress Detection Approaches

As mentioned at the beginning of this section, stress can be categorized into acute stress and chronic stress from a high-level perspective. Methods overviewed above are closer to detect acute stress during a specific task. However, there is another research direction, which is continuous daily stress detection. For continuous daily stress detection, the optimal goal is to detect stress levels of every period of time for individuals in their non-restricted daily life. Data is collected from non-intrusive wearable devices or smartphones integrated with different sensors. Labels of each segment are determined by assessment prompts, and the stress level is detected based on the data in each segment. Gjoreski et al. [33, 34] propose a context-based stress detection method, which is composed of three machine learning components: lab stress detector, activity recognizer, and context-based stress detector. The lab stress detector is used for detecting short-term stress (every 2 min) trained by laboratory data. The activity recognizer is designed for continuously determining the user's activity. Input data is first fed into the lab stress detector and activity recognizer to achieve the prediction of current stress level and current user's activity. The bio-signals, stress level, and activity are then fed into the context-based stress detector to discriminate between stress in real life and many other situations, which have similar physiological arousal. In their experiment, they collect 55 days of real-life data from 5 subjects, and their method achieves 70% accuracy of stress detection. Hovsepian et al. [47] present a continuous stress assessment, namely cStress, and they collect respiration and

electrocardiogram signals from a chest belt and successfully detect daily stress. cStress obtains a recall of 89% accuracy of stress detection with only 5% false-positive rate in the lab environment and 72% accuracy of stress detection in the real-life environment. Adams et al. [2] collect various signals from a small group of subjects during their real-life activities. In their study, they focus on understanding and comparing different types of signals whether they are effective or not in multiple contexts and find that EDA-based signals are not effective in physical discomfort context and voice-based signals are ineffective in quiet or noisy spaces. By utilizing EDA- and voice-based signals together can provide less invasive and reasonably robust stress detection in real-world environments.

2.3 Cognitive Process of Writing and Gaze and Typing

Behaviors in Writing

Writing on the computer is a complex task, which is composed of both cognitive and physical processes. Both eye gaze and hand movements are involved in the writing task. In this section, we will first review some relative works of writing cognitive process followed by the related studies of gaze and typing behaviors while writing on the computer.

In the 1980s, Flower et al. [30] propose the first cognitive process model of writing, shifting from the traditional sequential models to the hierarchical models to represent the recursive nature of writing. Their model can be divided into three main parts: the writing process, the long-term memory, and the writing environment, where the writing process contains three major phases: planning, translating, and reviewing. The planning phase involves recalling to access the long-term memory to retrieve writing information and creative thinking to formulate writing ideas. The translating phase is mainly converting the writing

ideas into sentences based on the context logic, and the reviews phase is mainly evaluating and revising the generated texts. During writing, three phases of writing processes are conducted alternately. Bereiter et al. [14] focus on analyzing the writing cognitive process for compare, diagnose and operate (CDO) procedure, and they find that subjects' diagnostic skills will be improved if they are provided with evaluative comments or tactical cues for revision work. They also try to provide some cues to the subjects in the planning phase and observe that these cues will help them increase reflective thinking. Kellogg et al. [59] interpret the basic parts of their writing model in three parts of the process. In the first part of the process, subjects mainly formulate, involves planning and translate rhetorical goals into texts; In the second part, subjects generate the text, either by hand or typing into computers. And in the final part of the process, subjects reread and revise the generated texts. All processes are operated simultaneously, which greatly affects the capacity of working memory. They also point out that expert writers always have a larger overall capacity of working memory. Alamargot et al. [5] show that maturity and practice are able to develop expertise in writing. Through the practices, writers may become more familiar with the writing topics, which can help them retrieve the writing information more easily from the long-term memory and construct it into an effective structure. Also, maturity enables writers to covert writing ideas into sentences more fluently and more automatically. Therefore, the working memory space during writing can be utilized more efficiently.

When it comes to eye gaze and hand movements in the writing task, there has been some previous work along this line. Butsch et al. [18] contribute the first study to investigate the eye-hand behaviors of typewriting. They find that the gaze is always approximately 5–7 characters ahead of hands. Inhoff et al. [50] also

illustrate a related observation that the eye gaze location is always three character-spaces before the actual character, which is being currently typed. Logan et al. [76] expand the findings by determining three kinds of *span*, or attention of foci in typing: stopping, eye-hand, and copying. The stopping span is for committing text and the eye-hand span is the temporal or pixel difference between the locations of the eye gaze and hand execution for activities such as mouse movements and keypresses. A special case of the eye-hand span when 40-odd characters were involved is also identified and named the copying span. However, all these findings are obtained from copy-typing tasks in which a subject simply copies words from a pre-prepared source. Compared with producing original texts on the computer, the copy-typing task omits the cognitive process of producing contextual sentences based on the writing goal, which would be expected to affect gaze and hand behaviors.

Feit et al. [29], Johansson et al. [53], and Papoutsaki et al. [89] take another step in investigating the differences of gaze and typing behaviors across touch typists and non-touch typists while producing their own texts. Feit et al. [29] place 26 anatomical landmarks on each hand to track hands movements and explore the motor of typing and gaze deployment based on each finger's movement during writing. They illustrate that even non-touch typists who spend significantly more time fixating at the keyboard but there is no significant performance (average entry rate and uncorrected error rates) difference between touch typists and non-touch typists, which is conflicted with previous findings. For the motion analysis, they find that touch typists utilize a greater number of fingers than non-touch typists. Specifically, touch typists input by using different fingers of the same hand compared with non-typists, who prefer inputting by using the same finger for successive keystrokes. Johansson et al. [53] further divide the writers into three

groups based on the interplay between typing texts and rereading texts that already generated, including monitor gazers (spend more percentage of time looking at the screen), keyboard gazers (spend more percentage of time looking at the keyboard) and mixed-strategy writers (spend a similar percentage of time on the keyboard and the screen). By analyzing 28 subjects writing data, they discover that monitor gazers are more productive writers with better typing skills. They always reread the generated texts in parallel with typing. Keyboard gazers use left and right cursor keys significantly frequently to revise their texts sequentially. Papoutsaki et al. [89] also investigate how gaze movement behaviors are different between the touch typists and non-touch typists. They also develop a classifier to discriminate between touch typists and non-touch typists based on the gaze behaviors and achieve the performance of 74.5% accuracy by using the eye tracker and 62.5% accuracy by using a webcam. They also encode their findings into the webcam-based gaze estimation method: WebGazer [88] by adding typing as a cue to help predict the on-screen gaze positions, improving the tracking accuracy for both touch typists and non-touch typists.

There are also some studies that explore the relationship between gaze and typing behaviors with the complexity and the quality of writing tasks. Torrance et al. [111] discover that subjects will spend more time rereading previously generated material while producing complex texts, and their fixation duration becomes longer for lexical processing. Waes et al. [118] conduct an experimental writing task in which subjects are asked to correct an embedded error and also complete a sentence. As the task increases in complexity, subjects tend to complete the sentence and then correct errors, even though sometimes they have already noticed the presence of the error. The cognitive load of subjects also increases, and they fixate less on the partial sentence while reading. Likens et al. [71] use fractal

analysis to model the inter-keystroke intervals as a time series. Their findings suggest that writing pieces with higher quality are generated by typing processes with a higher degree of autocorrelation in the inter-keystroke intervals.

Most previous studies investigating writer's typing behaviors have been done in the context of English typing and relatively little attention has been paid to non-English typing. Zheng et al. [132] collected over 54 million error-correction operations in Chinese typing with Pinyin input method. and discovered that the errors caused by omitting some letters are always (around 50%) corrected by deletions (re-typing). Common errors include transposition errors caused by messing the typing order of the left and right hands, and substitution errors caused by mistyping phonic representations which are similar to and close to the correct ones on the keyboard, such as "m vs. n", and "z vs. c vs. s". Meena et al. [85] and Joshi et al. [54] focused on Hindi typing. They found that the large number of letters, complex characters in Hindi language, and special structure of Indic scripts increase the difficulty of typing Hindi on QWERTY keyboards. Users thus need much more training to type Hindi. Samura et al. [102] explored keyboard dynamics of typing free texts in Japanese. Their results suggested that keypress duration is an important feature for individual identification. our study focuses on gaze-typing behaviors in Chinese typing, through which we use to determine the genre of the article which is being written. To the best of our knowledge, this is the first time that this problem has been investigated.

2.4 Gaze and Typing Behaviors in Summarizing

Summary writing is a multitasking process requiring reading comprehension, content acquisition, and writing [126]. Most of the related work is done in the linguistics field. Keck [58] illustrates that summarizing writing can be divided into

four levels: near copy, minimal revision, moderate revision, and substantial revision. Among these four levels, novel writers are more willing to produce the summary in the near copy level compared with expert writers. Brown et al. [17] show similar findings in their results that skilled writers tend to rearrange material in the original text when summarizing text. Kirkland et al. [61] investigate the relationship between cognitive load and summary writing, and they point out that cognitive load is determined by the internal and external constraints, where external constraints include familiarity with the genre of the document, the complexity of the document and the length of the document need to be summarized. Internal constraints contain the reading skills, writing skills, comprehension level, and critical thinking skills of writers. Yu et al. [129] compare the performances and perceptions of summarization in both Chinese and English. In their study, they recruit 157 Chinese undergraduate students to complete summary writings in both English and Chinese, and all the subjects have been learned English at least for eight years. They illustrate that the type of language greatly influences the performance of summary: 1) subjects produce significantly longer summaries in Chinese (their first language) because of their proficiency in Chinese; 2) Chinese summarization can be better reflect the reading abilities of subjects. Based on these findings, they conclude that the summarization performance is only determined by the reading comprehension, which may disaccord with common sense. Similar findings are also discovered by Li [70]. In his study, he examines the summarization procedure of 64 Chinese college students, and regression analysis results show that English writing ability can significantly contribute to the prediction of summary writing performance, but English reading ability cannot. Li [69] also investigates how genres (narrative and expository) of documents impact the performance and perception of summary writing. The questionnaire surveys

collected by 86 undergraduate students indicate that subjects show better performance on narrative text summarization. However, most subjects consider that the narrative text summarization is more difficult than the expository text summarization. Such perceived contradiction associates with the factors of internal constraints of subjects.

Yang et al. [127], Yang [126], Yi [128] investigate the cognitive process of summary writing. Yang et al. [127] use the thinking aloud method to access the subjects' cognitive process. They request the subjects to verbalize their mental process while summarizing. Five different cognitive stages are defined by them, including planning content, referring to sources, generating texts, rereading, and reviewing the generated texts. They also find that subjects in different writing levels spend different portions of time on each cognitive stage. Yang [126] applies the exploratory factor analysis and builds the latent variable model to predict the final outcome of summarizing according to the performance of each cognitive stage. Yi et al. [128] construct a dataset covering 50 subjects' gaze trajectory data collected by the eye-tracker for automatic text summarization. According to the gaze movement behaviors, they find that gaze movement patterns of reading significantly differ from gaze movement patterns of summaries.

3 Inferring Users' Affective State Based on Gaze and Mouse Behaviors

We start our study of understanding users' affective state based on gaze and mouse behaviors. We focus on detecting mental stress when users are interacting with computers. This chapter first presents our non-intrusive stress detection approach based on gaze and mouse behaviors with regard to UI layout. Compared with other stress detection approaches based on behavioral signals, we are the first work to investigate the relationship between the consistency of gaze and mouse behavioral patterns and the stress state. Our approach is evaluated in the static UI environment by doing mental math calculations. From the results, our approach shows decent performance in recognizing mental stress in the static UI environment, and we also understand how the stress state impacts the consistency of the gaze and mouse behaviors when interacting with computers.

However, for most real applications, they are executed in a dynamic UIs environment. That may limit the generalizability of our proposed stress detection approach. To address this challenge, we propose the MGAttraction coordinate system to model gaze and mouse behaviors without relying on the UI related information. Also, a UI-agnostic stress detection method is proposed, which is built based on the MGAttraction coordinate system. We conduct human experiments to recognize mental stress in the wild while searching for information online without considering UI. Details of our methods and results of evaluations are presented in Section 3.2.

To further improve our UI-agnostic stress detection method's generalizability, we decide to use the webcam to substitute for the eye-tracker by estimating the gaze locations from the webcam video. Combining with the pupil movement

features proposed by us, the performance of the webcam-based stress detection method is close to the performance of using the eye-tracker. Section 3.3 presents the details of how to estimate the gaze locations from the webcam video, as well as the evaluation procedures.

The rest of this chapter is organized as follows. Section 3.1 describes the non-intrusive stress detection approach adopted for the static UI environment stress detection, its evaluation experiments, as well as exploring the relationship between mental stress and the consistency of gaze and mouse behaviors. Section 3.2 introduces the MGAttraction coordinate system and the UI-agnostic stress detection method with its evaluation experiments. Section 3.3 describes procedures of how we estimate the gaze locations from the webcam video and extract the pupil movement features and build a webcam-based stress detection approach. Finally, this chapter is concluded in Section 3.4.

3.1 Stress Detection in Static UI Environment

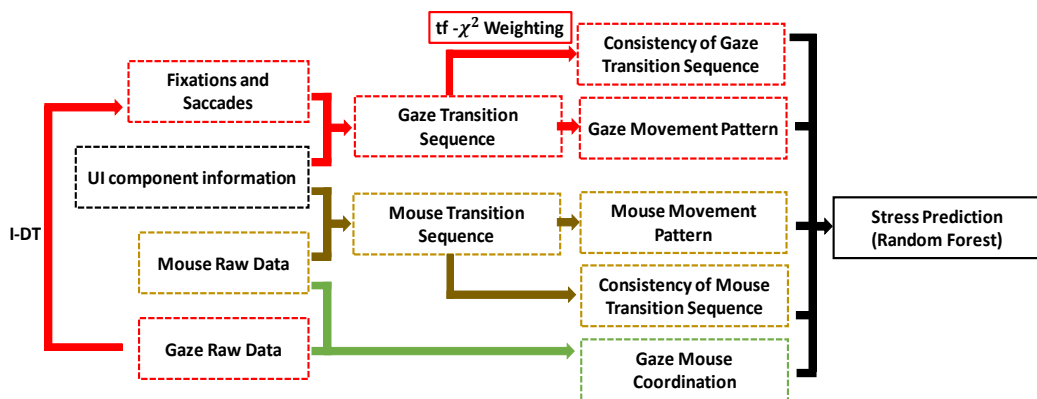


Figure 3-1 From signals to prediction: The system flow chart of feature extraction

Many prior studies have shown that affective states can influence gaze and mouse behaviors. We believe that mental stress as a kind of mental state can impact gaze and mouse behaviors as well. Therefore, we want to extract features to

describe the gaze movement patterns, mouse movement patterns, and gaze-mouse coordination as indicators of mental stress. Figure 3-1 shows the whole feature extraction procedure from input data: gaze raw data, mouse raw data, and UI component information to the final stress prediction model. The stress prediction model is built based on the gaze movement patterns, mouse movement patterns, and gaze-mouse coordination, where gaze and mouse movement patterns can be considered as the indicative gaze and mouse attention transition sequences with regard to UI layout that is representative of stressed or relaxed conditions and gaze-mouse coordination describes the correlation between gaze and mouse in the space, time and speed domains. Before introducing the gaze and mouse movement patterns and gaze-mouse coordination deeply, we start with input signal preprocessing.

3.1.1 Input Signals Preprocessing

According to the procedure shown in Figure 3-1, input signals contain the eye gaze data, the mouse data, and the UI component information. In this study, Tobii EyeX is utilized to obtain the eye gaze data, which encodes the user's gaze on-screen locations with corresponding timestamps. To eliminate the impulse noise, the two-phase heuristic filter [107] is used on the eye gaze data. From the preprocessed eye gaze data, fixations and saccades can be detected easier, where a fixation refers to a period of time that the user's gaze maintains within a single area, and a saccade stands for a short and quick movement between two successive fixations. Then, the Dispersion-Threshold Identification (I-DT) algorithm [101] is utilized to detect fixations with the dispersion set to 35 pixels and the minimum time of fixation as 170 *ms*. Data between every two fixations were considered as saccades. The mouse data is obtained by a C++ script developed by us to log the

mouse coordinates in every 10 *ms* and the linear interpolation is utilized to resample the mouse data to align with the gaze data.

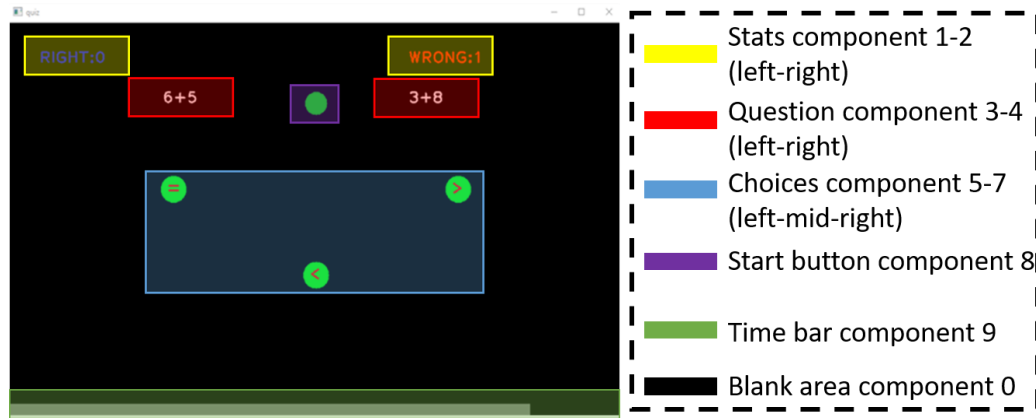


Figure 3-2 Experiment interface and UI components

To better model gaze and mouse behaviors, we extract UI component information from the current UI interface. The screen area is first divided into components according to the user interfaces' functionalities, shown in Figure 3-2. Then we construct a gaze transition sequence by mapping each fixation inside the gaze point sequence into a UI component, and the UI component area is the area that the fixation location belongs to. For example, a gaze transition sequence $8 \rightarrow 1 \rightarrow 2$ indicates that the subject first looks at component 8 (start button), component 1 (left stats), then on component 2 (right stats), where each type of UI component with its corresponding label is presented in Figure 3-2. If a user fixates at a screen region without being labeled as any particular components, then that fixation is mapped to 0. We can also generate the mouse transition sequence by following the same procedure as the gaze transition sequence. Therefore, after the data preprocessing, we achieve the following information for each mental math question: 1) timestamped gaze coordinates; 2) a gaze transition sequence; 3) timestamped mouse coordinates and mouse events (clicking, scrolling, and dragging) and 4) a mouse transition sequence.

3.1.2 Feature Extraction

3.1.2.1 Modelling Gaze and Mouse Movement Patterns

```
function Seqs2grams(seqs, n)           % seqs are all the transition sequences
{  potential_grams ← []              % n is the length of the gram
  for seq in seqs                     % iterate all the transition sequences
    for i = n to len(seq) step 1 do
      g ← seq[i-n...i]                % extract sub-sequence
      if g is not in potential_grams
        potential_grams.append(g)
      end if
    end for
  end for
  return potential_grams
}
```

Algorithm 3-1 Generating potential N-grams from transition sequences

In this part, we first introduce the procedures to model the gaze and mouse movement patterns based on the transition sequences and identify the indicative movement patterns that represent stressed or relaxed conditions. Then, the way we are modeling the gaze-mouse coordination is presented based on the timestamped gaze and mouse coordinates, followed by the behavioral consistency features.

Our gaze and mouse movement patterns are extracted from the gaze transition sequences and the mouse transition sequences, respectively. For generalizability, we break down the transition sequences into n-gram subsequences by following the procedure in Algorithm 3-1. For instance, when $n = 3$, a transition sequence $3 \rightarrow 4 \rightarrow 5 \rightarrow 6 \rightarrow 2$ will be broken into three subsequences, which are $3 \rightarrow 4 \rightarrow 5$, $4 \rightarrow 5 \rightarrow 6$ and $5 \rightarrow 6 \rightarrow 2$, and movement patterns are the subsequences representing stressed or relaxed conditions so that they can be used to be the indicators of mental stress.

The aim of involving movement patterns is that we want to differentiate whether a given transition sequence is engendered under the stress condition. To be more specific, we want the selected patterns to be abundant in one category but seldom appear in other categories. In order to satisfy such a requirement, the $tf \cdot \chi^2$ weighting scheme [24] is applied to calculate the weighting of a subsequence, where the value of $tf \cdot \chi^2$ weight can reflect the representativeness of the subsequence. A subsequence with a large weighting means that the number of occurrences for that subsequence is significantly different between the stressed and relaxed groups. $tf \cdot \chi^2$ weighting of a subsequence can be calculated by the multiplication between tf term and χ^2 term. tf term is the subsequence frequency in the considered condition and χ^2 term can be computed in the following manner:

$$\chi^2 = N_{seq} \cdot \frac{(n_r \cdot n_{\bar{s}} - n_s \cdot n_{\bar{r}})}{(n_r + n_s) + (n_{\bar{s}} + n_{\bar{r}}) + (n_r + n_{\bar{r}}) + (n_s + n_{\bar{s}})} \quad 3-1$$

Where:

- n_r : the number of transition sequences contains the given subsequence in the relaxed condition
- $n_{\bar{r}}$: the number of transition sequences does not contain the given subsequence in the relaxed condition
- n_s : the number of transition sequences contains the given subsequence in the stressed condition
- $n_{\bar{s}}$: the number of transition sequences does not contain the given subsequence in the stressed condition
- N_{seq} : total number of transition sequences

We calculate $tf \cdot \chi^2$ weighting of each subsequence (n-gram) generated from all transition sequences for both stressed and relaxed conditions. After sorting

all the weightings in descending order, we select the top K ($K = 5$) subsequences for both conditions to be the movement patterns, and the number of times each selected movement pattern appears in the transition sequence is counted. The movement pattern features are the count of each movement pattern normalized by the length of the transition sequence. This whole procedure of extracting movement pattern features is shown in Figure 3-3.

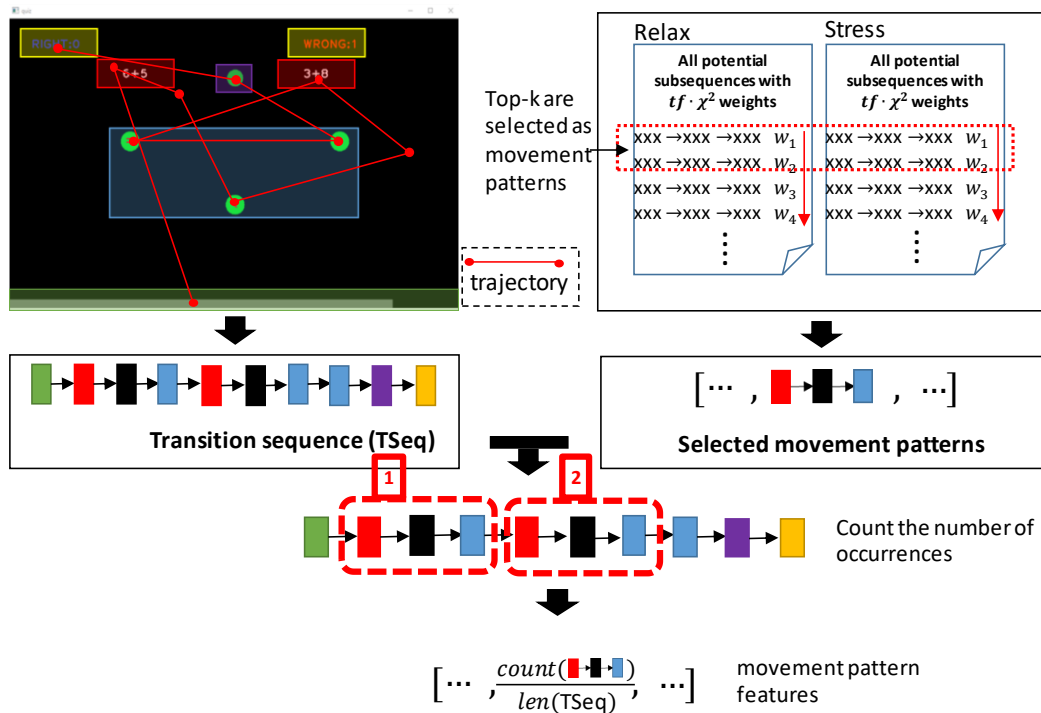


Figure 3-3 Procedure of extracting movement pattern features

3.1.2.2 Modelling Gaze-Mouse Coordination

Besides the movement pattern features, we also extract features to model gaze-mouse coordination. As shown by the previous studies introduced in Section 2.1, that gaze and mouse movements are strongly correlated with each other, and their relationship can be used to infer the affective and cognitive state of users. This study also attempts to explore whether the relative movements between gaze and mouse is able to reflect mental stress. Therefore, our gaze-mouse coordination

features are computed based on the correlation between the gaze and mouse movements in the spatial, time, and speed domains.

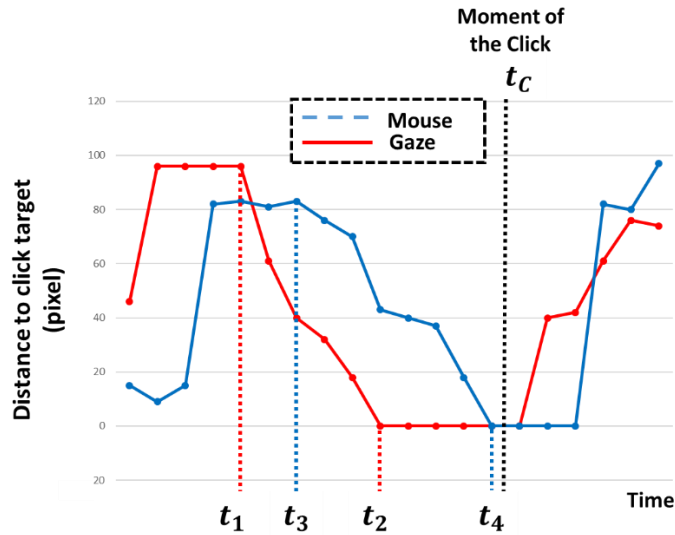


Figure 3-4 An illustrative example of the gaze-mouse coordination extraction

Figure 3-4 presents the process of extracting the gaze-mouse coordination features in an illustrative case. In the figure, the distances between the target of the mouse click and the gaze position and mouse cursor position for each timestamp are shown by the red and blue curves, respectively. t_c is the moment of the click. t_1 and t_3 are the moments that gaze and mouse start to move toward the mouse clicking target and t_2 and t_4 are the moments that gaze and mouse reach the mouse clicking target.

For each question, we consider the click event that answers the question – i.e., the click event on one of the choices ">", "<" or "=". First, we check whether a click occurred inside a fixation. If this is the case, then the beginning of the fixation is marked as t_2 . If not, we mark the moment of click as t_2 . Then, we find the last fixation right before the click and the ending moment of that fixation as t_1 . The interval between t_1 and t_2 is a saccade between two fixations.

Feature	Meaning	Formulation
f_1^{GMC}	Average distance in pixel between mouse cursor and gaze point	$Mean(D_{GM}^{t_i})$
f_2^{GMC}	Time difference of gaze and mouse start moving (positive for gaze leading mouse)	$t_3 - t_1$
f_3^{GMC}	Time difference of gaze and mouse end moving (positive for gaze leading mouse)	$t_4 - t_2$
f_4^{GMC}	Total time of feature extraction interval	$Max(t_2, t_4) - Min(t_1, t_3)$
f_5^{GMC}	Average speed of gaze	$\frac{euclidean(P_G^{t_1}, P_G^{t_2})}{t_2 - t_1}$
f_6^{GMC}	Average speed of mouse	$\frac{euclidean(P_M^{t_3}, P_M^{t_4})}{t_2 - t_1}$

Table 3-1 Gaze-mouse coordination features

We use similar procedures to identify t_3 and t_4 , which correspond to the interval of mouse movement surrounding a mouse click -i.e., the period between the two mouse hovers before and after the click. The gaze and mouse coordination features are extracted from the time interval $[Min(t_1, t_3), Max(t_2, t_4)]$, which contains a totally of 6 features. Meanings and formulations are shown in Table 3-1, where $D_{GM}^{t_i}$ is the distance between gaze position and mouse cursor position at t_i and $P_G^{t_i}, P_M^{t_i}$ are the positions of gaze and mouse cursor respectively at t_i . According to the gaze-mouse coordination features shown in Table 3-1, f_1^{GMC} captures the average distance that gaze leading/catching the mouse in the space domain, f_2^{GMC} , f_3^{GMC} capture the relationship in the time domain and f_5^{GMC} , f_6^{GMC} capture the speed information for both of gaze and mouse.

Both movement pattern features and gaze-mouse coordination features are designed to model gaze and mouse behaviors, and we hope these features are able to capture the differences of gaze and mouse behaviors when a user is in relaxed and stressed conditions. Besides exploring the differences of gaze and mouse behaviors, we are also interested in investigating whether the consistency of uses'

behavioral patterns is different when they are in the different affective states: stress and relax.

A prior study [32] shows that when a user is under stress, his/her alertness may also be increased, which results in that they are more concentrated on their current works. Therefore, we hypothesize that when a user is in a stressed condition, he/she will be more motivated to stay on the task, and their behaviors will be more close-ended and consistent, which will manifest itself in more similar behaviors.

3.1.2.3 Modeling Consistency of Gaze and Mouse Movement Behaviors

In this study, the consistency of behaviors is measured by the similarity of transition sequences. We use Dynamic time wrapping (DTW) [100] to measure the average distance among transition sequences with regard to UI components for the same channel, including gaze and mouse movements. If the average distance among transition sequences is small, it means that the behaviors in the periods of these transition sequences are consistent and vice versa. The cost function between the UI component uic_1 and uic_2 are defined as below (eq. 3-2), where UI components marked in the same color belong to the same component group as presented in Figure 3-2.

$$cost(uic_1, uic_2) = \begin{cases} 0, & \text{if } uic_1 \text{ and } uic_2 \text{ are same} \\ 1, & \text{if } uic_1 \text{ and } uic_2 \text{ are in the same component group} \\ 2, & \text{otherwise} \end{cases} \quad 3-2$$

We prefer DTW over Euclidean distance is because DTW is more robust to noise and the missing or irrelevant patterns, which is particularly useful for gaze behaviors since surrounding factors can easily influence gaze movements.

Finally, the whole feature vector extracted for recognizing the mental stress includes movement pattern features, gaze-mouse coordination features, and

behavioral consistency features. We use the random forest algorithm as the classifier constructed on the extracted feature vector. Normally, the random forest algorithm, as one of the ensemble algorithms, has the characters of fast training, good generalized accuracy, and robustness to overfitting.

3.1.3 Construct Dataset for Stress Detection in Static UI

Environment

In order to evaluate the mental stress detection method, we build a simple interactive system based on computerized multiple-choice math quizzes. The system's interface is displayed on a 22" monitor at 1680×1050 resolution in the full-screen mode. Previous studies prove that recursive mental math calculations [3, 78, 109, 116] and setting a time limit for the task [19, 63, 120] can efficiently induce mental stress. Therefore, our study utilizes recursive mental math calculation as a common stress factor and uses time pressure and noise to adjust the stress level, which belong to user-centric sources (relate to a user's background and the type of task a user is doing) and environmental sources (physical/social/computational environment), respectively.

Figure 3-3 presents the graphic interface of our experiment system. On the top of the screen, there were two math expressions in the form of $A \text{ op } B$, where A and B were two numbers, either 1-digit or 2-digit, and op is an operation that could either be "+, -, ×, ÷". Subjects were expected to mentally compute the results for both left and right expressions and decide if the outcome of the left expression is ">", "<" or "=" relative to the outcome of the right expression. Subjects had to press the start button labeled as Component 8, which allowed the choice components to be shown, and then subjects could make their choices by pressing. For each question, the order of choices was assigned arbitrarily. The

subject's current scores were presented in Component 1 and 2 in respect of the number of correct answers and incorrect answers.

In the experiment, every subject was expected to complete two sessions conducted under relaxed and stressed conditions, respectively, and for each session, there were 25 questions. Compared with the relaxed condition sessions, questions in the stressed condition sessions were more difficult (1-digit problems were used in the relaxed sessions, and 2-digit problems were used in the stressed sessions). We also imposed a time limit for each question with a countdown bar shown at the bottom of the screen. Background noise was also utilized in the stressed sessions to further induce stress. Subjects were asked to answer the question within the allocated time limit as possible. Otherwise, the system would immediately proceed to the next question, and the unanswered question was considered as a wrong answer.

To prevent frustration, the duration of each session was kept short, which was about 5 minutes. In order to alleviate adaption to experiment conditions, the number of experiments that each subject could participate in one day was also constrained. A pre-experiment was carried out for each subject to determine an appropriate time limit. Otherwise, either stress could not be induced successfully, or the subject could not have adequate time to solve the question. The first question was served as a warm-up question, and data was collected starting from the second question.

As previous studies have shown that individuals generally stay in a consistent mental state when given similar tasks [49], we consider the stress level constant across all questions in the same session. The question is then whether our methods to induce stress were indeed successful. To ensure that stress was indeed induced, the subjects were asked to complete a brief report on his/her stress level on a 5-

point Likert scale after each session. We believe that self-reporting is a feasible way to achieve the ground truth that whether a subject is stressed, as self-reporting has been used in many previous works as an efficient way to access the stress level [49, 90]. As, when a user is stressed, the heartbeat rate of that user may increase, or he/she may become more alert and more focused on the task, therefore we presume that it is not hard for him/her to be aware that he/she is under stress.

Data in the constructed dataset were collected from 9 subjects (*Age* 18-32). After removing the questions that subjects failed to answer within the allocated time (less than 10%) and the stressed sessions whose reported stress levels were less than 3, we totally achieved 836 questions in 37 sessions, where 17 of them were labeled as being conducted under the stress condition.

3.1.4 Question-level Stress Detection

We first evaluate the performance of our approach to detecting mental stress on the question-level.

The wrapper method is adopted with the best-first search. Final selected features are determined when there is no improvement for 10 consecutive searches. Our feature extraction process gives us one instance per question, where each instance is represented by a feature vector containing 26 features, where 10 of them are the gaze movement pattern features, 10 features are mouse movement pattern features, and 6 of them are gaze-mouse coordination features. We do not extract behavioral consistency feature at this stage since at the question level, each instance only contains one gaze and mouse transition sequence, meaning that consistency features would be meaningless.

In real applications, a user-independent model is highly valuable, since a user-independent model would presumably be able to detect the mental stress for

a new user who has never been seen before, as contrasted to a user-dependent model, which can only work for a specific user for whom the behavior is known. Therefore, leave-one-subject-out cross-validation is applied to evaluate the performance of our model. Specifically, we iteratively select one subject and use his/her data as the testing data to evaluate the stress detection model, which is trained on the data from other subjects. We repeat this process until all the subjects have been selected to test the model, and the average correct classification rate (CCR) are reported as the overall evaluation performance.

Table 3-2 presents the detailed performance of our approach for question-level stress detection. Based on the results shown in Table 3-2, It is obvious that the mouse movement pattern features alone have no contribution to the stress

	Gaze Moment Patterns	Mouse Moment Patterns	Gaze-mouse Coordination Features	All Features
Accuracy	59.7%	52.8%	59.6%	66.4%
Precision (Relax)	0.62	0.55	0.62	0.69
Precision (Stress)	0.57	0.53	0.59	0.64
Recall (Relax)	0.61	0.21	0.61	0.67
Recall (Stress)	0.59	0.83	0.56	0.66

Table 3-2 Performance of stress detection at question-level

detection since the model built based on the mouse movement pattern features tends to classify all the instances into majority class, which is *relax* in this study. One possible reason that the mouse movement pattern features are not effective is that most subjects do not move the mouse until they know the answer, and then

the mouse is directly moved to the location of the correct answer. This is unlike the gaze movements, which are complex since the gaze will travel among the positions of the questions and answers. Since mouse movements are simple and straightforward, that is why they are not affected by user stress. The contribution of the gaze movement pattern features and gaze-mouse coordination features are also presented. The best performance is achieved when these two sets of features are utilized together.

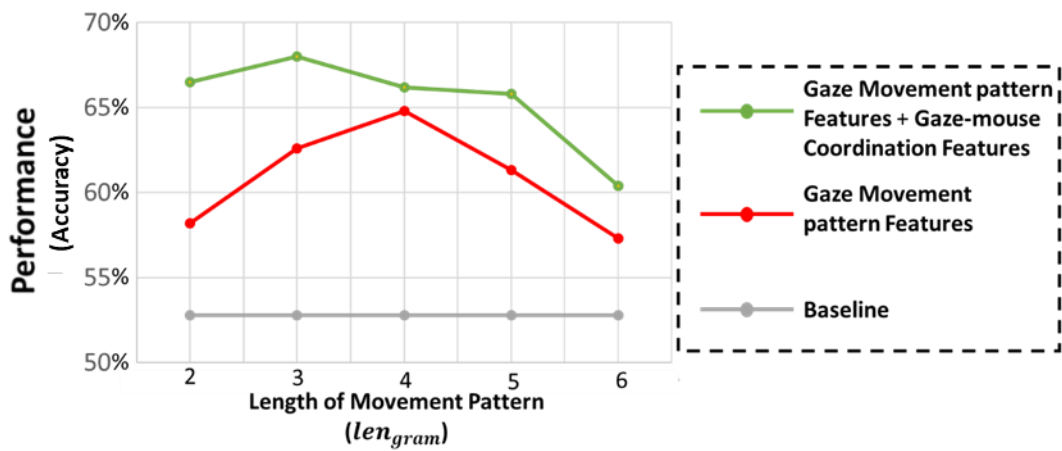


Figure 3-5 Trends of CCR across different len_{gram}

We also investigate the impact of the length of the gaze movement patterns on the performance. Figure 3-5 illustrates the CCR curves for different pattern lengths (len_{gram}). According to the results, we find that CCR curves consistently above the baseline performance of 52.8%, which is achieved by the zeroR classifier, which simply classifies every instance into the majority class. With the increase of len_{gram} (2 to 6), the performances of models built on gaze movement pattern features only and on gaze movement pattern along with gaze-mouse coordination features are both increasing first and then dropping. The best performance is achieved when $len_{gram} = 3$ or 4. This is in line with our expectations. When len_{gram} is too small, each pattern is too short and hard

general to be representative, but when len_{gram} becomes larger, the longer movement patterns may contain too much information, which is not clear. The generalizability is also affected as longer movement patterns are too specific to appear in multiple instances.

In order to investigate how gaze movement patterns capture the difference of gaze movements under different conditions, the selected gaze movement patterns are analyzed, and we find most of them can be categorized into two groups. The first group is *recurrent focuses* – subjects fixate on the same UI component repeatedly. For example, a gaze movement pattern $4 \rightarrow 0 \rightarrow 4 \rightarrow 0$ indicates that a subject reads the question many times. It can be understood that when a subject is relaxed, he/she may not focus entirely on the task, and to remember the content, he/she needs to view each component repeatedly. Patterns in the second group are blended with the UI components, which are not relevant to solve the question, such as the start button (component 8) or stats labels (component 1 and 2). Such behavior can be understood by the fact that only if a subject is relaxed, it is acceptable for him/her to be distracted and drawn by the less important components.

For the gaze-mouse coordination features, we apply a t-test for each of them, and results show that f_3^{GMC} , f_5^{GMC} , f_6^{GMC} are statistically significantly different between relaxed and stressed conditions (p-value is less than 0.05). It suggests that the subjects move their gaze and mouse quickly, and saccades and mouse moves tend to end simultaneously when they are under the stress condition.

3.1.5 Session-level Stress Detection

The above evaluation results illustrate that the movement pattern features and the gaze-mouse coordination features can effectively detect mental stress for the

question-level. We then evaluate our model's performance at the session-level. Totally, there are 37 instances from 9 subjects and 17 of them are labeled as stress, where each instance stands for a process during which the subject completes 24 questions in a row.

The session-level feature extraction procedure is shown in Figure 3-6. The session-level features can be divided into two groups. The first group is the statistical feature, which is built based on question-level stress detection results. For example, for a session $sess_i$, which contains 24 questions, can be represented as $sess_i = \{Q_j^{sess_i} \mid j \in [1,24]\}$. For each question $Q_j^{sess_i}$ in the $sess_i$, we can achieve the predicted result by processing through the question-level mental stress

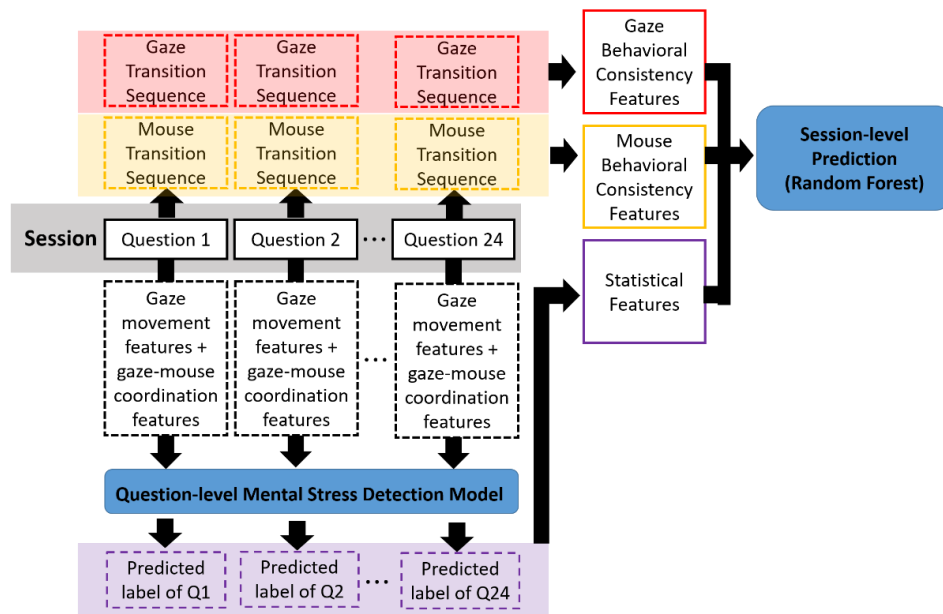


Figure 3-6 Procedure of extracting session-level features

detection model, which is either 0 (predicted as *relaxed* class) or 1 (predicted as *stressed* class). Then statistical feature is the percentage of questions predicted as the stressed class. Another group of session-level features contains gaze and mouse behavioral consistency features, which measure the consistencies of gaze transition sequences and mouse transition sequences among 24 questions. As same

as the question-level model, the leave-one-subject-out cross-validation is applied to evaluate the session-level stress detection model, and the detailed results are presented in Table 3-3, where BCF stands for the behavioral consistency features.

Results shown in Table 3-3 suggest that the behavioral consistency features, especially the gaze behavioral consistency features, are helpful to detect mental stress. Therefore, we further investigate how the consistencies of gaze and mouse behaviors are different between relaxed and stressed conditions. For each subject, we compute the average DTW distances of gaze and mouse transition sequences for both conditions. Averaging across all subjects, we achieve the mean distance of gaze transition sequences across all the subjects for both relaxed and stressed conditions, which are 21.1 and 14.5, respectively. We also compute the mean distance of mouse transition sequences for both relaxed and stressed conditions in the same manner, which are 4.2 and 4.1, respectively. The Wilcoxon signed-rank test result illustrates that the difference in the DTW distances between relax and stress is significant for the gaze channel, and the p-value is 0.002. This indicates that when a user is stressed, his/her gaze behaviors tend to be more similar and consistent.

	All Features	Stat	Gaze + Mouse BCF	Gaze BCF	Mouse BCF	Stat + Gaze BCF	Stat + Mouse BCF	Stress-Click
Accuracy	80.0%	77.1%	77.1%	72.2%	57.1%	82.9%	74.3%	65.7%
Precision (Relax)	0.83	0.76	0.79	0.77	0.56	0.84	0.75	0.66
Precision (Stress)	0.77	0.79	0.75	0.67	0.56	0.81	0.73	0.64
Recall (Relax)	0.79	0.84	0.79	0.68	0.79	0.84	0.79	0.67
Recall (Stress)	0.81	0.69	0.75	0.75	0.31	0.81	0.69	0.62

Table 3-3 Performance of stress detection at session-level

It is promising to note that except for the model constructed exclusively on the mouse movement distance features, all other models significantly outperform the baseline, which is 52.8% achieved by the ZeroR classifier.

3.2 Stress Detection in Dynamic UIs Environment

In the last section, we propose a non-intrusive stress detection method based on gaze and mouse behaviors. The proposed method takes advantage of the features to describe representative attention transition sequences with regard to the UI components and the features relate to the gaze and mouse coordination to successfully detect mental stress in an interactive task with the static UI. However, this method inherits a limitation, which is that it cannot be simply applied to the tasks with dynamic UIs, since it is hard to model the UI component transition sequence when the UI layout keeps changing. Therefore, the MGAttraction, a coordinate system to model gaze and mouse behaviors with regard to their relative movement without relying on UI related information, is proposed to overcome that limitation. Then a new UI-agnostic mental stress detection approach built based on the MGAttraction system will be introduced in this section.

3.2.1 MGAttraction: Modeling Mouse and Gaze Relative

Movement

The name of the MGAttraction stands for the mouse-gaze attraction, which is to model the relative movement between gaze and mouse in the dynamic UIs environment. Hence, the proposed coordinate system is needed to be rotation- and translation-invariant. To achieve rotation- and translation-invariant properties, we first transform the movement signals from the screen coordinate to the attraction

coordinate system.

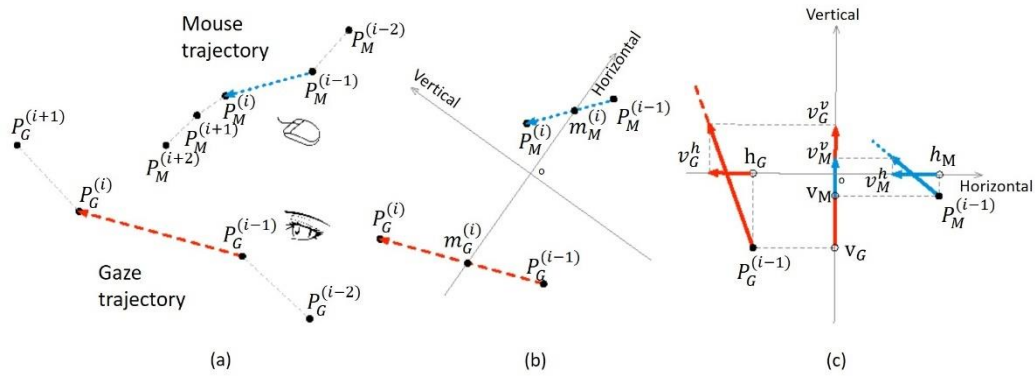


Figure 3-7 Attraction coordinate system showing displacement (dotted arrow), velocity (solid arrow), mouse (blue), and gaze (red) information; solid points are real positions, and hollow points are projections. (a) shows an example of mouse and gaze trajectories. (b) illustrates the origin identification of the coordinate system. (c) shows mouse and gaze velocity decomposition based on attraction coordinate

As the name suggests, the MGAttraction coordinate system measures the *attraction* between the gaze and mouse, which can be interpreted as the intensity and tendency of the gaze and mouse's relative movement, which is measured from consecutive samples in the gaze and mouse modality streams. The gaze and mouse modality streams are the sequences of on-screen coordinates of the gaze attention location and the mouse location, respectively. The attraction is a scalar quantity, which can be either positive or negative, where a positive attraction indicates the tendency of approaching and a negative attraction means the tendency of departing. For instance, if the gaze is moving toward the mouse direction at high speed, then at this moment, the attraction of the gaze relative to the mouse is in a large positive value. On the other hand, if the gaze is leaving the mouse at high speed, then at this moment, the attraction of the gaze relative to the mouse is in a large negative value.

In our experimental setup, the sequence of on-screen mouse positions is

collected by a C++ program at 100 *Hz*, and the sequence of on-screen gaze positions is captured by the eye-tracker Tobii EyeX at 60 *Hz*. Similar to the preprocessing procedures for mental stress detection in the static UI environment, we first apply a two-phase heuristic filter [107] to remove the impulse noise from the gaze signal to remove artifacts from eye blinks. Then, the mouse signal is downsampled to 60 *Hz* and synchronized with the gaze signal by using the linear interpolation approach. After signal preprocessing, we obtain the sequence of on-screen mouse locations $\mathcal{M} = \langle P_M^{(0)}, P_M^{(1)}, \dots, P_M^{(n)} \rangle$ and the sequence of on-screen gaze locations $\mathcal{G} = \langle P_G^{(0)}, P_G^{(1)}, \dots, P_G^{(n)} \rangle$, which can be illustrated in Figure 3-7(a). The relative movement of gaze and mouse is then modeled via the midpoint of each successive pair of coordinates. $m_G^{(i)} = \frac{1}{2}(P_G^{(i)} - P_G^{(i-1)})$ and $m_M^{(i)} = \frac{1}{2}(P_M^{(i)} - P_M^{(i-1)})$, respectively (Figure 3-7(b)). We then construct the attraction coordinate system, whose *x-axis* is the vector $\overrightarrow{m_M^{(i)} m_G^{(i)}}$ and *y-axis* is the orthogonal vector to $\overrightarrow{m_M^{(i)} m_G^{(i)}}$. The origin is the midpoint of $m_G^{(i)}$ and $m_M^{(i)}$ (Figure 3-7(c)). This attraction coordinate system is constructed based on the relative movement of the gaze and mouse and is independent of the screen UI layout.

The attraction encodes the intensity of the relative movement between gaze and mouse, inspired by the gravity measurement, which should be negatively correlated with their distance and positively to their movement speed. The overall idea is to leverage the relative velocity to delineate the attraction between gaze and mouse over time. For example, if gaze and mouse cursor locations are close and approaching each other at high speed, they exhibit a strong positive attraction. If the mouse *chases* the gaze at a higher velocity than the velocity of the gaze *escaping* from the mouse, then the mouse exerts a larger positive attraction while

the gaze experiences a smaller negative attraction. If the gaze and mouse are departing from each other at high speed, then both of them have a strong negative attraction.

Specifically, the overall attraction between gaze and mouse consists of the attractions exerted by the gaze, $attr_G$, and exerted by the mouse, $attr_M$. We resolve the velocities of gaze and mouse in vector form into the x - and y -components (or horizontal (h) and vertical (v) components). $attr_G$ and $attr_M$ can be formulated in a symmetric manner:

$$attr_G = \frac{\alpha_G V_{G|M}^h |V_G^h|}{D} + \frac{\beta_G V_{G|M}^v |V_G^v|}{D} \quad 3-3$$

$$attr_M = \frac{\alpha_M V_{M|G}^h |V_M^h|}{D} + \frac{\beta_M V_{M|G}^v |V_M^v|}{D} \quad 3-4$$

Where $D^{(i)}$ is the Euclidean distance between $m_G^{(i)}$ and $m_M^{(i)}$. V_G^h, V_G^v, V_G^h , and V_G^v are the horizontal and vertical component velocities of gaze and mouse in the attraction coordinate. $V_{G|M}^h$ and $V_{G|M}^v$ indicate velocity components of gaze relative to mouse and $V_{M|G}^h$ and $V_{M|G}^v$ indicate those of mouse relative to gaze. The relative velocity components can be computed as:

$$V_{G|M}^h = V_G^h - V_M^h; \quad V_{G|M}^v = V_G^v - V_M^v \quad 3-5$$

$$V_{M|G}^h = V_M^h - V_G^h; \quad V_{M|G}^v = V_M^v - V_G^v \quad 3-6$$

Finally, α and β denote the signs of the attraction components:

$$\alpha_G = \text{sgn}(V_{G|M}^h(h_M - h_G)); \quad \beta_G = \text{sgn}(V_{G|M}^v(v_M - v_G)) \quad 3-7, 3-8$$

$$\alpha_M = \text{sgn}(V_{M|G}^h(h_G - h_M)); \quad \beta_M = \text{sgn}(V_{M|G}^v(v_G - v_M)) \quad 3-9, 3-10$$

Where h_G , v_G , h_M and v_M are the current gaze and mouse location projected on the two axes. In other words, the sign of a component is positive, while the mouse and gaze are moving relatively towards each other. Otherwise, it is negative.

3.2.2 Preprocessing MGAttraction Signals

The purpose of the MGAttraction coordinate system is to transform the gaze and mouse movement signals from the screen coordinate to the MGAttraction signals. The method aims to detect mental stress based on the relative movement between gaze and mouse. Therefore, we are more interested in the periods that both gaze and mouse can be detected. While the position of the mouse can always be detected during the experiment, the gaze cannot be detected by the eye-tracker during eye blinks, and when the user's gaze is off-screen.

We handle off-screen eye periods in two ways. Since it is known that a human eye-blink is usually shorter than 150 *ms* [20], we discard time periods longer than 150 *ms* during which the gaze cannot be captured by the eye-tracker. For the remaining time periods, we estimate missing gaze locations using linear interpolation. We then compute the MGAttraction signals $Attr_G$ and $Attr_M$ and they can be represented as $Attr_G = [attr_G^1, attr_G^2, \dots, attr_G^n]$ and $Attr_M = [attr_M^1, attr_M^2, \dots, attr_M^n]$, where $attr_G^i$ and $attr_M^i$ stands for the i^{th} gaze attraction value and the i^{th} mouse attraction value in that period, respectively.

Based on the definition of $attr_G$ and $attr_M$ are positively correlated to the magnitude of gaze and mouse velocity. However, the speed of gaze is normally much faster than the mouse, which leads to a much larger range for $attr_G$ than $attr_M$. To facilitate the following analysis, we normalize the magnitude of $attr_G$ and $attr_M$ to bring them into the range $[-1, 1]$. Specifically, for $Attr$, which can

either be $Attr_G$ and $Attr_M$, we first find the maximum value of all positive $attr$ values in $Attr$ as max_p and the minimum value of all negative $attr$ values in $Attr$ as min_n . Then for each $attr$ in $Attr$, if $attr$ is larger than 0, $attr$ is transformed into $attr/max_p$. If $attr$ is smaller than 0, then $attr$ is transformed into $-1(attr/min_n)$ and $attr$ keeps the same when $attr$ equals 0.

3.2.3 Inferring Mental Stress from MGAttraction Signals

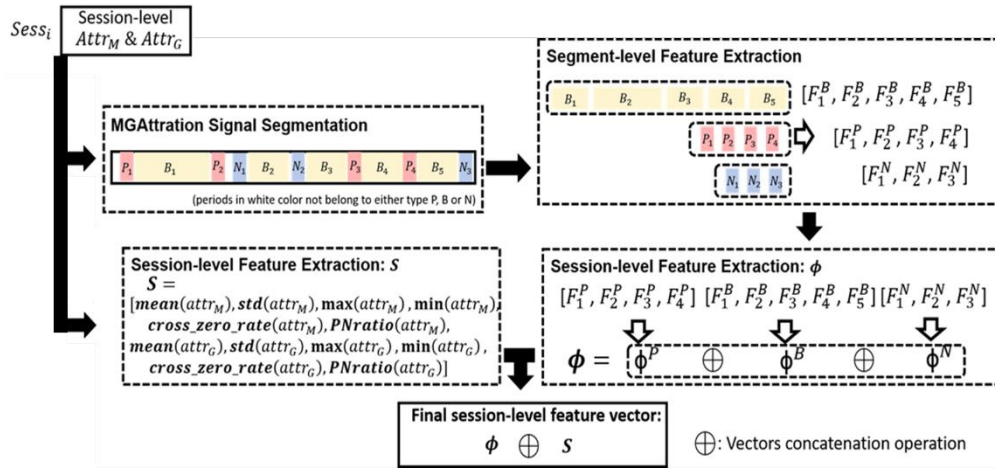


Figure 3-8 Overall pipeline of feature extraction to infer mental stress from MGAttraction signal

Figure 3-8 illustrates the overall pipeline of feature extraction from MGAttraction signals. The final session-level feature vector is composed of two parts: ϕ and S . ϕ contains the segment-level features extracted from each type of MGAttraction signal segment, which will be used to describe the relative movement between gaze and mouse during the segment periods. On the other hand, S contains the statistical features extracted from the preprocessed session-level MGAttraction signals $Attr_G$ and $Attr_M$, which will be used to model the macro behaviors of gaze and mouse over the session period. In the following parts of this section, we will introduce the way of signals segmentation and how we extract features in ϕ and S , respectively.

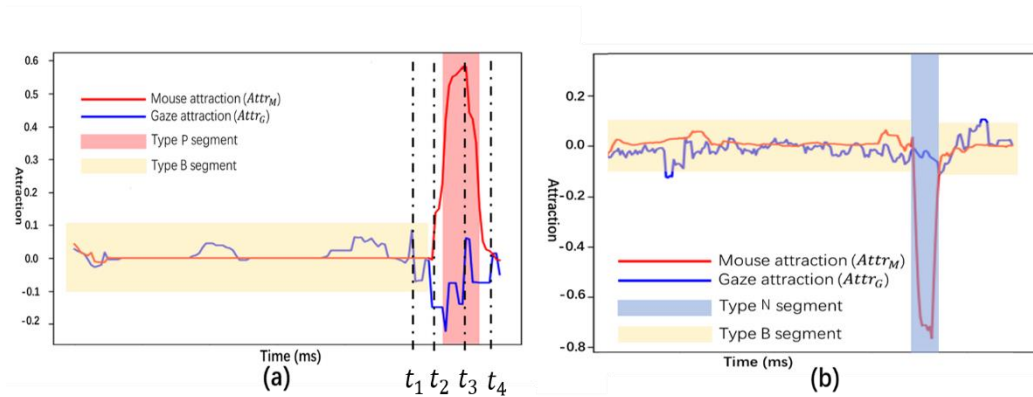


Figure 3-9 Two example periods of gaze attraction and mouse attraction with a Type P segment (red) and a Type B segment (yellow) and a Type N segment (blue)

Figure 3-9 illustrates two example time periods. We observe that the shape of $Attr_M$ resembles that of a pulse signal, in that the amplitude of a signal changes intensively from a baseline value to a higher or a lower value, accompanied by a fast return to the baseline value. The baseline value of $Attr_M$ is 0, which is measured when the mouse is at rest. A positive pulse period indicates that the mouse is approaching the gaze and a negative pulse indicates the opposite (mouse departing from gaze). In the example shown in Figure 3-9(a), both the gaze and mouse are stable with no relative movement before Time t_1 . At t_1 , the gaze starts to move to a new target, thus leaving the mouse. The mouse starts moving at Time t_2 to catch up with the gaze. After Time t_3 , the intensity of the relative movement between the gaze and mouse decreases till Time t_4 , when both gaze and mouse are stable again. A similar interpretation can be made for the example period shown in Figure 3-9(b).

To better model the relative behaviors of gaze and mouse behaviors in a time period, we segment the period based on the behavior pattern exhibited by $Attr_M$. Three types of segments are defined. A Type P segment is a period during which $Attr_M$ shows a positive pulse, a Type B segment is a period during which the

magnitude of $Attr_M$ stays around the baseline value, and a Type N segment is a period during which $Attr_M$ shows a negative pulse. Examples of Type P, B, and N segments are shown in Figure 3-9.

```

function Signal_Segmentation(Attr_M)      % Attr_M : mouse MGAttraction signal
{ Ps ← []                                % initialization
  Ns ← []
  Attr_Mpositive ← {e ∈ Attr_M | e > 0}    % Get all positive values in Attr_M
  Attr_Mnegative ← {e ∈ Attr_M | e < 0}    % Get all Negative values in Attr_M
  % Get the threshold of peaks
  thres_p ← mean(Attr_Mpositive) + 3 × std(Attr_Mpositive)
  thres_n ← mean(Attr_Mnegative) + 3 × std(Attr_Mnegative)
  % Get the threshold of valleys
  is_p ← [idx | Attr_M[idx] == thres_p]
  is_n ← [idx | Attr_M[idx] == thres_n]
  for every two consecutive values i and j in is_p
    if Attr_M[v] ≥ thres_p for ∀ v ∈ [i, j]
      s ← first index that Attr_M[s] == 0 and s ≤ v
      e ← first index that Attr_M[e] == 0 and e ≥ v
      Ps.insert(Attr_M[s : e])      % Attr_M [s : e] is in Type P
    end if
  end for
  for every two consecutive values i and j in is_n
    if Attr_M[v] ≤ thres_n for ∀ v ∈ [i, j]
      s ← first index that Attr_M[s] == 0 and s ≤ v
      e ← first index that Attr_M[e] == 0 and e ≥ v
      Ns.insert(Attr_M[s : e])      % Attr_M [s : e] is in Type N
    end if
  end for
  return Ps, Ns
}

```

Algorithm 3-2 Automatic MGAttraction signal segmentation for Type P and N

Algorithm 3-2 presents the procedures to identify the segments in Type P and

N segments. By definition, Type B segments are the time periods that mouse is at rest. Specifically, mouse speed is less than 75 pixels/sec, and the cursor is within a circle with a radius of 10 pixels for more than 1.2 seconds [117].

The second step of the feature extraction process is the segment-level feature extraction. As shown in Figure 3-8, segments with the same type are considered together. For each segment type, we then extract segment-level features (F^P, F^B, F^N) to describe the relative movement behaviors between gaze and mouse in the Type P, B, and N segments.

Type P segments mainly involve behavior exhibited when a subject moves the mouse towards the gaze point. The segment-level features F^P quantifies behaviors such as how vigorously the mouse approaches the gaze and how much the gaze leads the mouse. Both of these behaviors have been shown to be indicative of different mental states [49].

Table 3-4 shows the extracted features from the segment in Type P. $f_1^p - f_4^p$ quantify the overall MGAttraction level for both gaze and mouse, and $f_5^p - f_6^p$ describe the time needed for the gaze and mouse to reach the largest attraction. When a user is stressed, the speed of movement for both gaze and mouse tend to increase (this phenomenon is found when we detect the stress in a static UI environment), which can be reflected by features $f_1^p - f_6^p$.

$f_5^p - f_8^p$ capture the latency information in the coordination between the gaze and mouse movements, such as the time difference between when the gaze and mouse start moving and when they reach their largest attraction value. Figure 3-10 gives an example of what these features would look like in a sample Type P segment, where the x-axis indicates the timeline, and the y-axis indicates the attraction of gaze and mouse.

Feature	Meaning	Formulation
f_1^P, f_2^P	Mean, max of mouse attraction	Mean and max of $attr_M$ in the segment period
f_3^P, f_4^P	Mean, max of gaze attraction	Mean and max of $attr_G$ in the segment period
f_5^P	Time that mouse has the largest absolute attraction	Time of $attr_M$ shows the maximum absolute value
f_6^P	Time that gaze has the largest absolute attraction	Time of $attr_G$ shows the maximum absolute value
f_7^P	Gaze attraction at the beginning	$attr_G$ at the beginning of the segment period
f_8^P	Latency of peaks	Time difference between the maximum absolute value of $attr_G$ and the maximum absolute value of $attr_M$
f_9^P	Duration of the segment	Total time duration of the segment

Table 3-4 F^P : Features extracted from the P segment

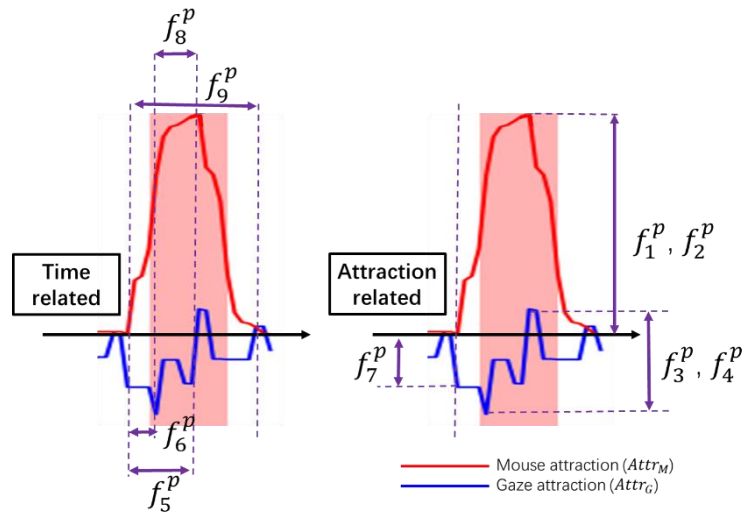


Figure 3-10 Illustration of features extracted in F^P

Compared to the Type P segment, the Type N segment describes a time period during which the mouse departs from the gaze. It can be seen as the upside-down version of the Type P segment. Therefore, we extract the same features F^N as F^P from the Type N segment, shown in Table 3-4.

Feature	Meaning	Formulation
f_1^B, f_2^B, f_3^B	Mean, max, min of gaze attraction	Mean, max, min of $attr_G$ in the segment period
f_4^B, f_5^B	Mean of positive gaze attraction	Mean of all positive and negative values of $attr_G$ in the segment period
f_6^B, f_7^B	Duration of gaze shows a positive and negative attraction	Accumulated sum of time duration that $attr_G$ is positive and negative in the segment period
f_8^B, f_9^B	Power of positive and negative gaze attraction	Accumulated power of positive and negative gaze attraction
f_{10}^B	Duration of the segment	Total time duration of the segment

* **Power** stands for integral of attraction over time

Table 3-5 F^B : Features extracted from the B segment

Type B segments are those where the mouse is stationary for the entire segment period. Hence, we only extract features from $attr_G$, and detailed meaning and formulation of extracted features are presented in Table 3-5. $f_1^B - f_3^B$ depict the overall intensity of gaze movement attraction, and $f_4^B - f_9^B$ are designed to model the movement by which gaze is approaching or departing from the mouse. When a user has a clear idea about the next target and moves the mouse purposefully toward it, then $Attr_G$ should show only one negative pulse with a large amplitude. However, if a user does not have a clear idea about the next target, he/she likely to look around, which will generate a couple of negative and positive pulses with a small amplitude.

According to the feature extraction pipeline from Figure 3-8, the final session-level feature vector is constructed by concatenating the ϕ and S components. ϕ contains the aggregated features constructed from segment-level feature vectors, and S consists of statistical features that model the overall trend of $attr_G$ and $attr_M$ for the entire session.

The first part of the session-level feature vector ϕ contains two statistical features extracted from the generated segment-level feature vectors. The first feature is the average behavior among all the segments and the second one captures the variation of behavior among all the segments. Specifically, suppose a session $Sess$ consists of k segments in Type P. For the i^{th} Type P segment in $Sess$, we can extract a segment-level feature vector F_i^P , where $i \in [1, k]$. ϕ^P is the aggregated feature vector extracted from F_i^P and $i \in [1, k]$ by computing the mean value and the standard deviation of each f_j^P and $j \in [1, 9]$. By following the same procedure, we can also generate ϕ^B , ϕ^N and ϕ by concatenating ϕ^P , ϕ^B and ϕ^N together.

For the second part of the session-level feature vector S , we extract statistical features from the session-level $attr_G$ and $attr_M$ signals, including mean, standard deviation, max, min, cross zero rate (per second) (i.e., the number of times per second that the signal moves from positive to negative, and vice versa) and $NPratio$, where $NPratio$ is computed as the accumulative duration that the signal is negative divided by the accumulative duration which the signal is positive. Our expectation is that these statistical features can capture the overall trend of signals for the whole session. Therefore, the final session-level feature vector is built by concatenating ϕ and S .

3.2.4 Construct Dataset for Stress Detection in Dynamic UIs

Environment

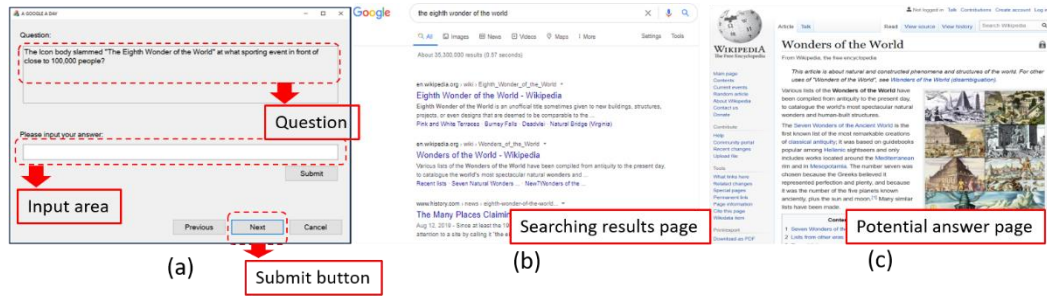


Figure 3-11 Experiment interface for stress detection in dynamic UIs environment: (a) Question page (b) Searching results page and (c) Potential answer page

The purpose of constructing this dataset is to detect stress in a real-world scenario. Hence, the constructed dataset should satisfy requirements, including 1) the task used to evaluate should be a commonly-encountered computer interaction task 2) with dynamic UIs. Given that web searching is one of the most ubiquitous activities, we use a web search task for our study. In our experiment, subjects were required to answer some questions randomly selected from the question-answering game "A Google a Day".

Figure 3-11(a) shows the question-answering interface. We first confirmed that the subject did not know the answer to the posed question in advance. If the subject already knew the answer, a new question was re-selected. Subjects were asked to type their answers in the input area. The submit button was used to check the correctness of the inputted answers. Subjects could repeatedly submit attempts until they found the correct answer, or (in stress sessions) they reached the 5-minute time limit. The time limit for each session was determined through multiple pre-experiments and we found 5 minute an efficient setting to balance between inducing stress and having adequate time to answer a question.

The questions in the "A Google A Day" task were formulated such that it was not allowed to find the answer by simply copying and pasting the question into the

search query. This means that subjects had to iteratively rephrase and refine the searching keywords according to the question and the information they achieved from previous searches. Throughout this process, subjects would be led to different webpages, which they might browse through or even follow links off, to obtain the final answer. Since these webpages would have different UI layouts, and it was impossible for us to forecast which keywords would be used and which websites would be visited. This gives us a dynamically changing UI environment. Some example webpages are presented in Figure 3-11(b) and Figure 3-11(c).

In the experiment, each subject was required to accomplish 12 sessions. Each session required the subject to find the correct answer to one "A Google A Day" question, where 6 of them, we termed as *relaxed*, did not have a time constraint, and subjects could take as long as they liked to finish the task. The other six sessions subjected the experiment subjects to a 5-minute time limit per session to induce stress. Many previous works [64, 79] show that time pressure and background noise are effective ways to induce the mental state. To further ensure that the stress level is indeed increased, a sound cue countdown was included.

The order of the relaxed and stressed sessions was determined randomly to even out the fatigue factor. The experiment started with one warm-up session to familiarize the subject with the experimental procedure and the experimental settings. After each session, subjects were required to report their stress level on a 5-point Likert scale, with 1 being *totally not stressed* to 5, *fully stressed*. A 15-minute break was introduced between every two sessions to allow subjects to relax and recalibrate the eye-tracker.

Totally, there were 15 subjects involved in this experiment. Stress sessions during which the subject self-reported a stress score lower than 3 were discarded. Our final dataset contains 175 sessions, 90 of which were labeled as stress.

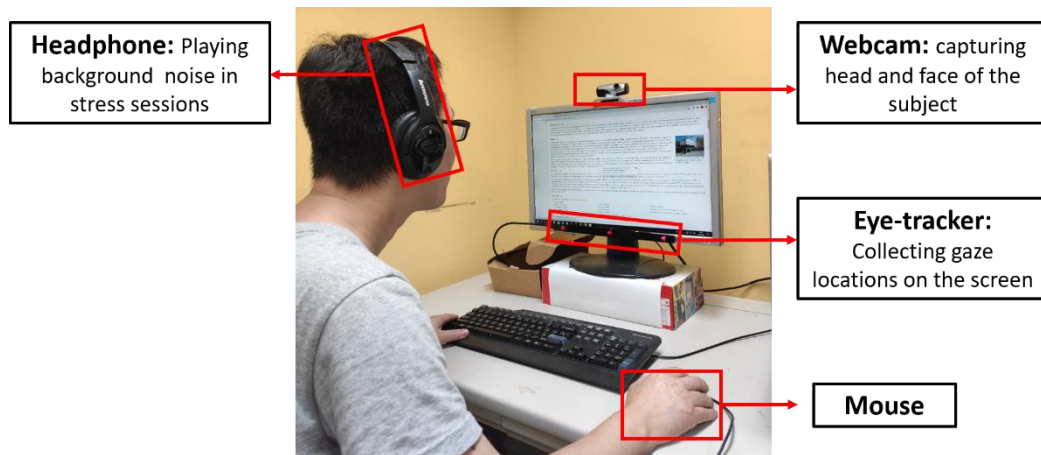


Figure 3-12 Experimental environment for stress detection in dynamic UIs environment

The experiment was conducted in a conventional office setting, which is shown in Figure 3-12. The setup was composed of a 22" LCD monitor at 1600×1000 resolution, a full-size QWERTY keyboard, and a standard optical mouse. Subjects sit around 60 cm away from the screen with their preferable chair heights and screen heights. Three different modalities of data were collected during the experiment: 1) Eye gaze location data captured by the Tobii EyeX eye-tracker at 60 *Hz*, which was attached to the bottom of the display, 2) mouse location data captured by a C++ program at 120 *Hz* and 3) video of subjects' face and upper body captured by a standard webcam fixed on the top of the at 30 *Hz*. All the data collecting programs were running in the background without disturbing subjects' interactions.

3.2.5 Experimental Evaluation of Stress Detection on

MGAttraction Signals

This section will evaluate the performance of detecting mental stress via the gaze and mouse behaviors modeled by the MGAttraction coordinate system on the dataset constructed for stress detection in the dynamic UIs environment. In real

applications, a model should be able to detect stress for a new user never seen before. Hence, the leave-one-subject-out mechanism is utilized to evaluate our method in the dynamic UIs environment. Specifically, we iteratively select one subject and his/her data to be the testing data to evaluate the stress detection model, which is trained on the data from other subjects. We repeat this process until all the subjects have been selected to test the model and report the average correct classification rate (CCR) as the overall evaluation performance.

We follow the procedures shown in 3.2.3 to extract 66 features (12 features in S and 54 features in \emptyset) for each session. A Random Forest (RF) is used to discriminate between the stressed session and relaxed session based on the extracted features. RF has been used in many similar contexts [28, 113, 125, 131] and has the advantages that it is able to 1) handle non-linear data, 2) be somewhat robust to outliers, 3) produce a low bias and moderate variance result, and 4) quantify the relative importance of the features, which helps with interpreting the model.

Performance Class	Precision	Recall	F-measure
Relax	0.82	0.73	0.77
Stress	0.77	0.84	0.80
Weighted Average	0.79	0.79	0.79

Table 3-6 Classification performance for stress detection based on MGAttraction signals in dynamic UIs environment

Predicted as \ Ground truth	Relax	Stress	Total
Relax	62	23	85
Stress	14	76	90
Total	76	99	175

Table 3-7 Confusion matrix for stress detection based on MGAttraction signals in dynamic UIs environment

Table 3-6 shows the method's detailed performance, and the confusion matrix is presented in Table 3-7. The average CCR achieved for two classes is 78.8%, which is significantly higher than the baseline of 51.4% achieved by classifying every instance as the majority class (Stress). The false alarm rate of our approach is less than 0.25, which suggests that our approach can balance between over-reporting possible stress and the danger of missing reporting. Moreover, our approach's weighted average F-measure is close to the weighted average precision and recall, which indicates that our approach does not sacrifice either one of precision or recall for the other. Overall, the results indicate that our approach can successfully detect stress in a real-world scenario based on the MGAttraction signals.

We further study the achieved performance by comparing it with other state-of-the-art, dynamic UI-based approaches in the web search task, in contexts where the experiment subject is required to complete an entire task, and the stress level is measured on the level of the overall task. However, some state-of-the-art approaches rely on UI related features, such as the dwell duration of gaze and mouse within a particular UI area or the speed and frequency of gaze and mouse travel between each UI component and gaze and mouse transition sequence among UI components. In order to evaluate these methods on our dataset, we implement

a module to extract the UI information from the current webpage, which is provided to the approaches that we are comparing ours against. We experiment with two methods for extracting the UI component information dynamically: heuristic and content-based.



Figure 3-13 Dynamic UIs component detection methods: (a) Heuristic-based and (b) Content-based

The heuristic-based method divides the whole UI interface into several sub-area based on the heuristic knowledge of browsers' standard UI design. As shown in Figure 3-13(a), We first extract the top and bottom sub-areas and then further evenly divided the middle area into 4×4 sub-areas. Different UI components will appear in each sub-area with different frequencies. For example, in the Google result page, links in the text form often appear in two left columns of the sub-areas, and users always pay more attention to the top two rows of the sub-areas. On the Wikipedia page, pictures often appear in the right two columns.

The content-based method extracts UI information based on computer vision techniques. The canny edge detection algorithm is adopted to segment different UI component areas, including button area, input area, text-content area, and picture-content area. An example of the UI components division result is presented in Figure 3-13(b).

Performance Method	Accuracy (CCR)	Precision (Stress)	Recall (Stress)
StressClick	58.9%	0.60	0.62
Movement patterns + Heuristic-based UI	62.8%	0.62	0.71
Movement patterns + Content-based UI	67.1%	0.64	0.82
UI-agnostic Stress Detection on MGAttraction	78.8%	0.77	0.84

* Movement patterns include movement features and gaze-mouse coordination features

Table 3-8 Performance of different approaches in dynamic UIs task

With the help of the dynamic UIs information extraction module, the gaze and mouse transition sequences can be constructed in the form of transition among different extracted areas. The state-of-the-art approach: StressClick [49], the movement pattern based stress detection method proposed by us for the static UI environment with heuristic-based and content-based dynamic UIs component detection methods and our new UI-agnostic stress detection method are evaluated on the web search task dataset, and the performance of each approach is presented in Table 3-8.

The results suggest that our UI-agnostic stress detection approach achieves the best performance, which is around 20% improvement over the state-of-the-art: StressClick and more than 10% improvement from the movement pattern based stress detection method with the content-based UI extraction module. One possible reason that StressClick does not perform as well as our method is that StressClick only considers the gaze behaviors relative to the mouse within a small time-window around each mouse click, which may not be sufficient enough to detect

the mental state in a complex task with dynamic UIs. Our hypothesis is borne out by the observations that the movement pattern based stress detection method yields better performance than StressClick, especially when it is provided with detailed UI information (with content-based dynamic UIs component detection method), which allows it to take into account more information related to the gaze and mouse movement behaviors. However, it also tends to generate many false positive (stress) instances and is fairly sensitive to the quality of extracted UI information. This can be seen from the fact that when the heuristic-based module, which is not able to accurately analyze the UI, is used, the performance drops to a CCR of 62.8%. This is a limitation of these kinds of methods, as real-time extraction of UI components in dynamic UIs tasks is expensive since it usually requires heavy image processing computation. In conclusion, our results illustrate that our UI-agnostic stress detection method based on the MGAttraction signals can successfully detect mental stress in a real-world scenario with balanced precision and recall and a low false alarm rate with less computation cost.

To better understand the features and how they work to detect stress, we output the top 5 most important features considered by RF. Figure 3-14 presents the distributions of each important feature across the relax and stress groups. Each bar stands for a distribution (green for relaxed and red for stressed), where the yellow line marks the mean value of each distribution. The box covers the first quartile to the third quartile, and the whiskers cover the range from minimum to maximum, except for outliers, which are shown by hollow circles. A t-test is adopted to determine whether there is a significant difference between the means of the two groups. If the p-value of the t-test is less than 0.01, which indicates that the difference of means is statistically highly significant, it is annotated with "**" at the top of the figure. If the p-value is in [0.01,0.05), which means the

difference is statistically significant, and it is marked with "*". If the p-value is greater or equal to 0.05, the difference is not significant, and it is marked with "X".

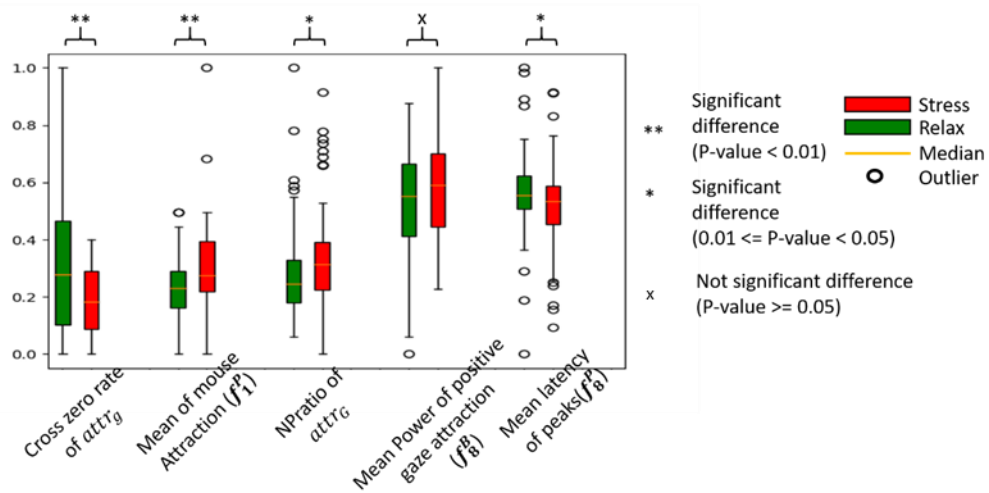


Figure 3-14 Distributions of selected important features

Figure 3-14 suggest that the important features have different distributions between the relax and stress groups. The t-test shows that four of them are shown the significant or highly significant difference in means between the two groups. $attr_G$ of the stressed group shows a lower cross zero rate. In physical terms, every time $attr_G$ across the boundary between positive and negative (or vice versa), it indicates a change in the direction of the gaze movement (e.g., from leaving the mouse to approaching it). Therefore, a higher cross-zero rate implies that the gaze is moving back and forth without a clear target in mind. We also note that $attr_G$ exhibits a higher NPratio. This implies that the gaze is directly moving toward the target and leading the mouse in most cases together with the lower zero crossing rate, suggests that gaze movement is more consistent when the subject is stressed.

Finally, it can be seen that when the user is stressed, the value of the *mean latency of peaks* is smaller. This indicates that the distance (in the time domain) between gaze and mouse is smaller. The *mean of mouse attraction* value is larger,

which means that the mouse has a greater tendency to move. Putting together, this suggests that when under stress, the mouse catches up with the gaze more quickly and with less delay. One possible reason for the above behaviors is that when subjects are stressed, their alertness may also be increased [32]. In situations where the stress is caused by the imposition of a time limit (such as in our study), this alertness discourages distractions and encourages more focus on the task at hand. A similar kind of gaze and mouse coordination has also been found when a user is in a state with a high cognitive load [49].

3.3 Webcam-based Stress Detection via Gaze and Mouse Behaviors

In the previous sections, we propose the movement pattern based stress detection method for static UI tasks and the UI-agnostic stress detection based on the MGAttraction signals for dynamic UIs tasks. Both stress detection methods rely on the gaze location data, which is directly collected by the eye-tracker. However, the eye-tracker is still considered as special equipment that most common users cannot access to it and largely reduce the generalizability of our methods. To address such limitation, we propose to use the webcam, a kind of standard device, to replace the eye tracker by estimating the gaze locations from the webcam video frames and extend our UI-agnostic stress method so that it can detect stress efficiently based on the estimated gaze location without relying on the eye-tracker.

3.3.1 Estimate Gaze Locations from Webcam Video

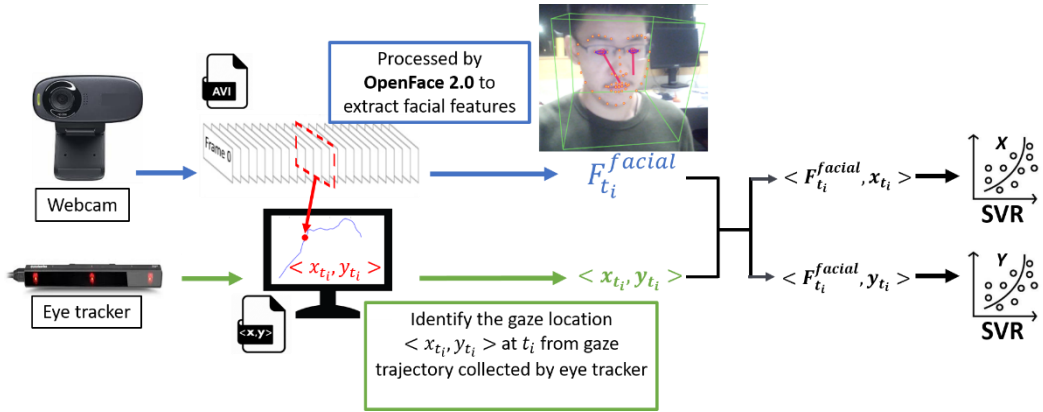


Figure 3-15 Overall pipeline of estimating gaze locations from webcam video

Figure 3-15 shows the whole process of estimating gaze locations from the webcam video. We treat the video as a sequence of frames recording the subjects' face and upper body. The webcam camera is fixed in the middle of the top of the display, which is about 60 *cm* away from the subject. In order to estimate the gaze locations from the webcam video, we exploit the state-of-the-art OpenFace 2.0 [9] to extract facial features related to the head pose and the eye gaze direction. Extracted features with their meanings and formulations are shown in Table 3-9. For each valid $frame_i$ at t_i in the video, where valid is defined by the ability of the OpenFace algorithm to capture the subject's head and face, a facial feature vector $F_{t_i}^{Facial}$ is extracted, which contains 12 features. At the same time, we also record the gaze location $\langle x_{t_i}, y_{t_i} \rangle$ on the screen at t_i as captured by the eye-tracker, to be the ground truth for training purposes. Two Support Vector Regression (SVR) models are then trained to construct the mapping from the facial feature vectors to the estimated x and y gaze locations on the screen: $F^{Facial} \rightarrow x_{estimated}$ and $F^{Facial} \rightarrow y_{estimated}$. We evaluate the performance of our gaze location estimation method using the leave-one-subject-out cross-validation. The

average error in pixels of estimated gaze locations among all subjects is around 125 pixels, and detailed average errors for each subject are shown in Figure 3-16.

Feature	Meaning	Formulation
$head_{Tx}, head_{Ty}, head_{Tz}$	Location of the head	Location of the head corresponding to webcam in millimeters and positive Z is the direction away from the camera
$head_{Rx}, head_{Ry}, head_{Rz}$	Rotation of the head	Pitch (Rx), yaw (Ry), and roll (Rz) of the head with webcam being the origin
$L_gaze_x, L_gaze_y, L_gaze_z$	Left eye gaze direction	Left eye gaze direction vector in the webcam coordinates with webcam being the origin
$R_gaze_x, R_gaze_y, R_gaze_z$	Right eye gaze direction	Right eye gaze direction vector in the webcam coordinates with webcam being the origin

Table 3-9 F^{Facial} : Facial features extracted from webcam video

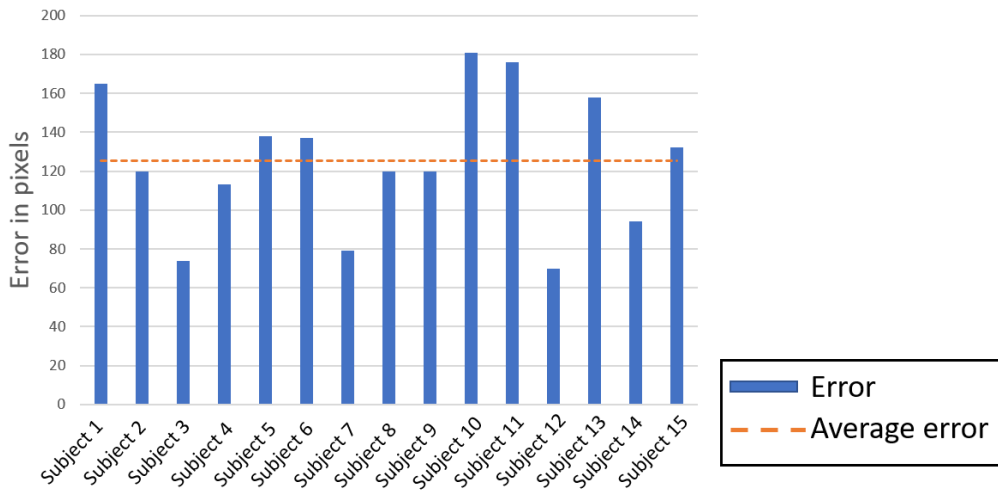


Figure 3-16 Average error in pixels of estimated gaze locations for each subject

3.3.2 Inferring Mental Stress Based on the Estimated Gaze

Locations

Our gaze estimation process described in the last section gives us a sequence of on-screen estimated gaze locations, which forms the estimated gaze trajectory $\mathcal{GE} = \langle P_{GE}^{(0)}, P_{GE}^{(1)}, \dots, P_{GE}^{(n)} \rangle$. We follow the same procedures to generate estimated gaze attraction $attr_{GE}$ and estimated mouse attraction $attr_{ME}$ from the sequence of on-screen mouse locations \mathcal{M} and the sequence of on-screen estimated gaze locations \mathcal{GE} . Session-level features $(\phi_E \oplus S_E)$ are extracted from $attr_{GE}$ and $attr_{ME}$ to discriminate between relax and stress sessions. Similarly, the performance of stress detection based on the estimated gaze locations is evaluated by the leave-one-subject-out mechanism.

Performance \ Class	Precision	Recall	F-measure
Relax	0.64	0.65	0.64
Stress	0.66	0.66	0.66
Weighted Average	0.65	0.65	0.65

Table 3-10 Stress detection performance based on estimated gaze locations

Predicted as \ Ground truth	Relax	Stress	Total
Relax	55	30	85
Stress	31	59	90
Total	86	89	175

Table 3-11 Confusion matrix for stress detection based on estimated gaze locations

Table 3-10 presents the results of stress detection based on the estimated eye gaze locations. The confusion matrix is presented in Table 3-11. The overall CCR for stress detection based on the estimated gaze locations is 65.1%, a drop of

around 14% from the performance achieved when the eye-tracker is used to capture the eye gaze locations.

The average error of the estimated eye gaze locations is about 125 pixels on a 1600×1000-pixel screen, but this performance degradation may or may not be uniform across the entire surface. We, therefore, conduct a deeper analysis to investigate the relationship between the CCR performance in different screen regions and the inherent error in the estimated eye gaze locations. We first evenly divide the whole screen area into 16×10 sub-areas, and each sub-area contains around 100×100 pixels. For each sub-area, we compute the average error of estimated gaze locations based on the following equation:

$$error^j = \frac{\sum_i^{C_j} \sqrt{(P_G^{(i)} - P_{GE}^{(i)})^2}}{|C_j|} \quad 3-11$$

Where $error^j$ is the average error of gaze estimation in the j^{th} sub-area, C_j is the set of gaze locations P_G collected by the eye-tracker within the j^{th} sub-area, $P_{GE}^{(i)}$ is the corresponding estimated gaze of $P_G^{(i)}$ and $|\cdot|$ returns the size of the set. We then set a threshold ($thres_{err}$) and classify all the sub-areas into two categories: those which exhibit an average error greater than $thres_{err}$, and those which exhibit an error less than $thres_{err}$.

Figure 3-17 shows the results of our analysis based on different thresholds. The region in black shows the area in which the average estimated gaze location error is smaller than the threshold. It can be seen that the estimated gaze locations in the central area of the screen are more accurate (i.e., have lower errors) compared to locations at the edge of the screen.

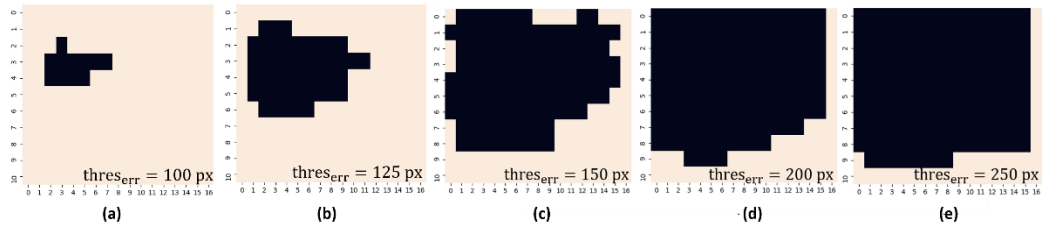


Figure 3-17 Error analysis with respect to screen regions. The average estimated gaze location error is smaller than $thres_{err}$ in the black areas

Therefore, we build separate models that specialize in handling user behaviors in particular parts of the screen. Specifically, we construct five models for each $thres_{err}$: 125 pixels, 150 pixels, 200 pixels, 250 pixels and ∞ (no $thres_{err}$). Based on the estimated location of the gaze, we then trigger the model with the lowest $thres_{err}$ that covers the corresponding part of the screen. The leave-one-subject-out mechanism is used to evaluate the performance, and the results are shown in Table 3-12. It is obvious that our webcam-based stress detection model can achieve better performance when the gaze estimation is more accurate unless there are no adequate estimated gazes achieved inside the corresponding part of the screen.

In order to improve the performance of our webcam-based model despite the error in the estimated gaze location, we investigate the possibility of adding additional information directly from the webcam video without going through the gaze location estimation process. The webcam video shows the eyes and the surrounding structures, and as such, pupil movements can be easily extracted from the video. We, therefore, investigate whether adding this information would improve our stress detection performance.

Performance <i>thres_{err}</i>	Precision (Stress)	Recall (Stress)	F-measure (Stress)	Accuracy
100 px	Not evaluated for the size of the area is too small			
125 px	0.64	0.58	0.61	64.5%
150 px	0.70	0.69	0.69	68.6%
200 px	0.67	0.72	0.70	67.4%
250 px	0.67	0.66	0.66	65.7%
no <i>thres_{err}</i>	0.66	0.66	0.66	65.1%

Table 3-12 Stress detection performance based on estimated gaze locations

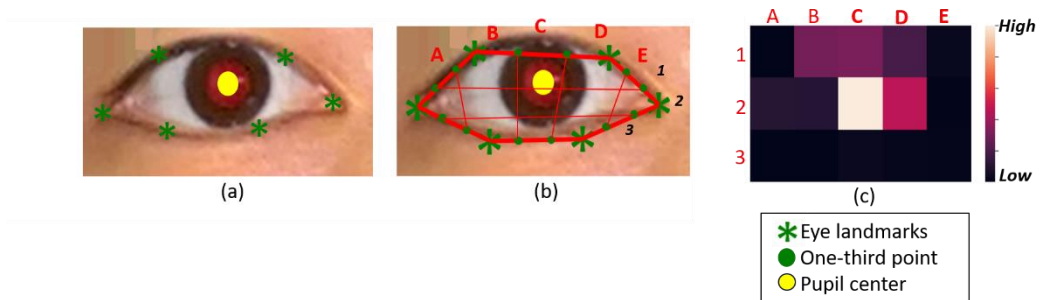


Figure 3-18 Generating the pupil movement histogram: (a) showing the detected landmarks and pupil center (b) showing the segmentation of the eye image based on the landmarks and (c) showing the 2-D histogram encoding the probability of the pupil center appearing in each zone

To extract the pupil movements, we utilize state-of-the-art Dlib [58] and OpenFace [9] to detect eye landmarks and each eye's pupil center positions for each eye frame in the webcam video. An example of landmarks and corresponding pupil center positions is shown in Figure 3-18(a). Based on the positions of landmarks, we divide the eye area into 5×3 sub-zones, and the layout of sub-zones is presented in Figure 3-18(b). We then represent the center of the eye pupil by the column-row ID of the sub-zone that it currently resides in. For example, Figure 3-18(b) shows a pupil center residing in Zone *C1*.

This method of encoding the eye pupil locations allows us to create a sequence of Zone IDs that represent the eye pupil movement for the entire session.

This sequence can be further encoded into a 2-D histogram by considering the pupil location IDs in all the frames recorded for that session, which can be shown in Figure 3-18 (c). The 2-D histogram thus encodes the probability that the pupil center appears in each zone.

The 2-D histograms allow us to analyze the relative positions of the eye pupil center. Our observations suggest that there is always one zone that has a significantly higher probability than the others. This suggests that the subjects' pupils often remain in one specific for most of the time in a session period. This phenomenon makes sense since each subject has a *comfortable resting spot* for the eyes when reading. Given a choice, he/she would either move the content window inside a *preferred reading region* of the screen, or he/she would move his/her head such that his/her eyes will rest in the comfort zone [92].

By this definition, the zone of the highest probability, which we call the *ground zone*, indicates when the subject is reading content in his/her most comfortable reading position. If the next reading target results in the pupil center being outside of the ground zone, he/she will move the webpage's position to the preferred reading region or move his/her head. Either way, at this moment, his/her pupil centers will return to the ground zone.

To model this behavior, two kinds of signals $I(t)$ and $D(t)$ are generated from the webcam video. $I(t)$ captures whether the pupil center is inside the ground zone at time t and $D(t)$ is the Euclidean distance between the pupil center and the center of the ground zone at time t . $I(t)$ and $D(t)$ are formulated as below:

$$I(t) = \begin{cases} 1 & \text{if pupil center within the ground zone} \\ 0 & \text{otherwise} \end{cases} \quad 3-12$$

$$D(t) = euclidean(P_{pc}^{(t)}, P_g)$$

3-13

Where $P_{pc}^{(t)}$ is the position of pupil center at time t and P_g is the position of the ground zone center. After generating the $I(t)$ and $D(t)$ signals, we then extract features F^{Pupil} from $I(t)$ and $D(t)$. Meanings and formulations of features in F^{Pupil} are shown in Table 3-13.

Feature	Meaning	Formulation
f_1^{Pupil}, f_2^{Pupil}	Mean of signal $I(t)$ and $D(t)$	$Mean(I(t))$ and $Mean(D(t))$
f_3^{Pupil}, f_4^{Pupil}	Standard deviation of signal $I(t)$ and $D(t)$	$Std(I(t))$ and $Std(D(t))$
f_5^{Pupil}, f_6^{Pupil}	Max and Min of signal $D(t)$	$Max(D(t))$ and $Min(D(t))$
f_7^{Pupil}	Number of activations of signal $I(t)$	Number of changes from 0 to 1 of signal $I(t)$ divided by the total duration of the session
$f_8^{Pupil}, f_9^{Pupil}, f_{10}^{Pupil}$	Mean, Max and Min of activation duration of signal $I(t)$	Mean, Max and Min duration of signal $I(t)$ equals to 1 consecutively

Table 3-13 F^{Pupil} : Features extract from $I(t)$ and $D(t)$ signals to describe pupil movement

For each session's webcam video, two pupil feature vectors F_L^{Pupil} and F_R^{Pupil} are extracted for the left and right eyes, respectively. Then, we merge F_L^{Pupil} and F_R^{Pupil} together into F^{Pupil} by computing the mean of the respective features from F_L^{Pupil} and F_R^{Pupil} .

Performance Model	Accuracy (CCR)	Precision (Stress)	Recall (Stress)
Eye-tracker based	78.8%	0.77	0.84
Webcam-based	65.1%	0.66	0.66
Webcam-based + Pupil Moment	73.7%	0.73	0.78

Table 3-14 Performance of different approaches in dynamic UIs task

Similar to previous evaluations, the leave-one-subject-out mechanism is used to evaluate the performance of the stress detection model based on the webcam-estimated eye gaze locations, augmented with the pupil movement features. Evaluation results are shown in Table 3-14, and the confusion matrix is presented in Table 3-15. Results show that, with the help of the pupil movement features, the webcam-based model can improve more than 8% accuracy, which brings its performance close to the eye-tracker based model but without the need for any special equipment.

Predicted as Ground truth	Relax	Stress	Total
Relax	59	26	85
Stress	20	70	90
Total	79	96	175

Table 3-15 Confusion matrix for webcam-based stress detection with pupil moment features

3.4 Summary

In this chapter, we focus on investigating how to infer the affective state, especially mental stress from gaze and mouse behaviors. This study proposes two

mental stress detection approaches: the movement pattern based stress detection approach mainly for static UI tasks and the UI-agnostic stress detection approach based on MGAttraction signals for dynamic UIs tasks. To improve the UI-agnostic stress detection approach's generalizability, we estimate the on-screen gaze locations from the webcam video to replace eye-trackers with webcams without relying on any special devices.

For the first part of the study, we explore the possibility of using gaze and mouse behaviors, especially movement patterns and gaze-mouse coordination for stress detection in an interactive application modeled upon a common e-Learning evaluation task. We use an approach that extracts representative movement patterns from the gaze transition sequence and extracts gaze-mouse coordination features to successfully detect stress at the question-level. Aggregating the result at the question level and adding features that consider the consistency of the gaze and mouse movements across multiple questions allows us to detect the user stress level of an evaluation session with a performance of over 40% over the baseline. By analyzing the data, we find that when a subject is relaxed, he/she tends to revisit the same UI component more frequently and be attracted by the irrelevant information easier than they are stressed. Also, when a subject is stressed, he/she is apt to move their mouse and eyes more rapidly, and their behaviors are more consistent when he/she performs the same operations.

For the second part of the study, we have demonstrated the feasibility of detecting stress on a common user interaction task based on gaze and mouse behaviors in a dynamic UI environment. Unlike most other state-of-the-art approaches, which model the correlation between gaze or mouse and the UI components separately, our approach considers both gaze and mouse information into the MGAttraction coordinate system, which is translation- and rotation-

invariant. This alleviates the need for accurate detection/identification of UI components. Our MGAttraction also allows for interpretation of the gaze and mouse behaviors in physical terms - in other words, the tendency of gaze and mouse to approach to or depart from each other. This allows the qualitative investigation of behaviors that are more important for stress detection. Our observations suggest that a subject tends to be more focused on his/her task and exhibits more consistent gaze movement, more intensity mouse movement, and less time latency between gaze and mouse under the stressed condition. We believe the MGAttraction coordinate system would benefit future studies in the human computer interaction area.

Our UI-agnostic stress detection approach by utilizing the MGAttraction coordinate system achieves the best performance compared with other state-of-the-art stress detection approaches. StressClick [49] is the system that is closest to ours. However, StressClick only considers the movement of the gaze before and after each mouse click. Our approach expands upon theirs by considering both gaze and mouse behaviors during the entire session. Results show that considering both gaze and mouse information together can build a better-performing stress detection model. It also suggests that features extracted in a single modality within a short time-window may not be powerful enough to detect stress in a more open-ended and complex task.

We then explore the feasibility of using the webcam to estimate gaze locations on the screen and detect stress based on gaze estimation, which makes our approach more feasible for consumer applications than approaches relying on the use of a specialized eye-tracker. We first find that the accuracy of gaze estimation strongly affects the performance of the stress detection. We also find that the accuracy of the webcam-based gaze estimation varies according to the

position of gaze on the screen with some regions (the upper-middle part screen when the webcam is placed on top of the monitor in the middle). Both findings illustrate that it may be beneficial for UI designers to better exploit this area of the screen since gaze information in that area can be estimated with higher confidence just using a standard webcam. To further improve the performance of our webcam-based stress detection model, we involve another modality of information, which is the pupil movement. Our final webcam-based stress model considers both the gaze and mouse relative movement behaviors and the pupil movement. This multimodal approach helps to improve our model's performance.

4 Inferring Users' Writing Cognitive Process Based on Gaze and Typing Behaviors

Chapter 3 presents the behavioral signals based approaches to infer the affective state, especially users' mental stress. Based on gaze and mouse behaviors, two important modalities, we successfully detect the mental stress in both static UI environment and dynamic UI environment. In this chapter, we investigate another important component of the cognitive state, which is inferring users' cognitive processes in daily computer interaction tasks. Different tasks may involve different cognitive processes, and in this thesis, we focus on analyzing the cognitive processes of writing.

Unlike the previous studies, gaze and typing behaviors are investigated during copy-typing tasks. In these tasks, subjects just need to type the texts from sources prepared in advance. This type of behavior is not the same as typing the texts composed on their own. Another weakness of previous research is that English is the language used by most of the prior studies to investigate gaze and typing behaviors.

To address those limitations, we construct our datasets by collecting data from subjects from different age groups with different typing skills. Three articles in different genres (reminiscent, logical, and creative) are required to be composed by each subject, and all the articles are written in Chinese. A detailed description of datasets is introduced at the beginning of this chapter. Based on the constructed datasets, we first investigate that whether and how age-factors affect the cognitive process of writing in Section 4.2 and then we explore that whether and how writing genres affect the writing cognitive process for users with different typing skills in Section 4.3, followed by the summary in Section 4.4.

One of the significant challenges of the study is how to identify the time window in different types based on gaze and typing behaviors so that different writing processes can be isolated as clear as possible. Another challenge is to design the appropriate features to capture the differences in the writing cognitive process. To overcome the challenges, we design three different types of time windows to divide the writing process into three different phases: thinking phase, typing phase, and transition phase between the thinking and typing phases for investigating age-factors' effect. Statistical-based gaze-typing features are extracted from each time window to model the behaviors. Because the impact of writing genres is more complicated than age-factors, we further categorize the thinking window and the typing window into three different subtypes, respectively, and statistical-based and sequence-based gaze-typing features are extracted. The concredited definitions of each time window and procedures we extract features are introduced in Section 4.2 and 4.3.

4.1 Constructing Writing Cognitive Process Datasets

4.1.1 Subjects' Background and Environment Setting

This study aims to explore whether and how age-factors, writing skills, and writing genres affect the writing cognitive process based on gaze-typing behaviors. Therefore, we establish datasets that satisfy the requirements, including 1) texts generated originally in different genres by each subject, 2) subjects recruited from diverse age groups with different typing skills, and 3) subjects writing articles in their first language: Chinese. As far as we know, our datasets are the first datasets that satisfy all these requirements at the same time.

Data included in the datasets are collected from 46 subjects. 18 subjects belong to the child age group (*Ages* 8 - 12, *Mean* = 9.85, *STD* = 1.46), 10

subjects to the college students age group (*Ages 22 - 29, Mean = 24.6, STD = 2.46*) and 18 subjects to the elder age group (*Ages 55 - 67, Mean = 60.75, STD = 4.05*). A pre-experiment survey is conducted by each subject before participates in the official experiment. The pre-experiment results suggest that all the participating subjects are acquainted with using computers and able to type in two hands. All the recruited subjects in the study are native Chinese speakers, and they all type in Chinese by using the Chinese Pinyin input method.

When typing Chinese in the Pinyin input method, as illustrated in Figure 4-1, a list of potential Chinese words/phrases corresponding to the Latin texts inputted by a user are presented in a pop-up candidates box under the caret. The user then chooses a certain choice by using the number key or pressing the space to pick the first candidate. After selecting the targeted word/phrase, the candidate box will disappear, and the chosen word/phrase is appended to the position of the caret. Then the caret moves to the end of the word/phrase that has just been produced.



Figure 4-1 An example of the pop-up candidates box. The user input the Latin text "pin' yin' shu' ru' fa", shown on the top of the candidates box, and five corresponds Chinese words/phrases are generated automatically by the system, shown below the user input. The user can either choose the correct mapping by pressing the number key or press "space" to select the first option.

As shown in Figure 4-2, our experiment was performed in a traditional office setting. The setup included a 22" LCD display at 1680 × 1050 resolution with Microsoft Word executing in full-screen mode. A Tobii EyeX eye-tracker was attached to the bottom of the screen, and a full-size QWERTY keyboard was used

for input. During the experiment, subjects were expected to sit around 60 *cm* away from the screen in a comfortable typing position. The subject's eye gaze on-screen locations, which were obtained by the EyeX eye-tracker, was logged at 60 *Hz*. The mouse cursor position was also captured at 100 *Hz*. All keypress events were also logged. Screen recordings were taken at 30 *Hz*.

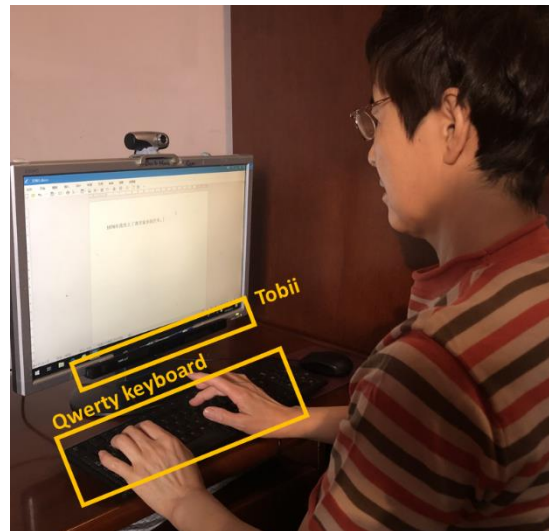


Figure 4-2 Experiment environment

4.1.2 Experiment Design

We hypothesize that during the writing process, the subjects' cognitive process and load are correlated to the genres of the text that they are working on. For instance, the cognitive processes of editing a scientific article and writing a narrative essay are completely different, which may manifest in different gaze-typing behaviors. Therefore, three articles in different genres are required by subjects to complete. Details are shown below:

- **Reminiscent** – Write an article describing an unforgettable event that occurred several years ago or so. With the intention of bringing readers back in time to witness the event, the event should be described clearly and specifically.

- **Logical** – Write a guideline to instruct readers on a new skill. The instruction steps should be connected by the connective words but not written in the list form. Examples are the rules of playing bridge or the procedures of multiplying or dividing a 2-digit number. The presumption is that the reader has no previous experience of this skill.
- **Creative** – Write an essay on a fantasy event, such as a day in the far future, or the life in a moon colony.

The subjects were instructed to write an article that would be about one page in length and given around 30 minutes for each task. If a subject could not finish within 30 minutes, he/she would be reminded of the time, but the experiment would continue till he/she finished writing the article. The font size was set to 18 DenXian with triple-line spacing so that the eye tracker could locate fixations and saccades accurately. Every subject was given sufficient time to adapt to the equipment and experimental settings. There was a 15-minute break between every two tasks to prevent exhaustion. After each break, the eye-tracker was recalibrated. Experiment sessions in which the subjects wrote too little or otherwise did not meet our length requirement were removed.

4.1.3 Overview of Datasets

Our experiments resulted in 138 instances, each of them representing around 30 minutes of composing and typing activity from 46 subjects (18 in child age group, 10 in college student age group, 18 in elder age group). Among all instances, 46 instances are labeled as reminiscent, 46 as logical, and 46 instances as creative.

Many previous studies [52, 53, 89] illustrate that one of the most significant impacts on the gaze-typing behaviors comes from the typing skill, and users are always categorized into touch typists and non-touch typists. Where touch typing

is a style of typing in which the subject relies on muscle memory to locate the keys, and non-touch typing is the other style of typing that requires a subject to look at the keyboard to locate the keys. Therefore, the gaze and typing data of non-touch typists exhibit more dramatic displacements along the y-axis and lower typing speed. These differences in the gaze-typing behaviors are far more marked than the differences induced by the different cognitive processes.

We, therefore, separate the collected writing data based on typing skills for a cleaner analysis. For each subject, we measure the time spent typing by the subject while looking at the keyboard by summing up all the typing periods during which the subject's eye gaze is away from the screen and compute the ratio γ of that time to the sum of all typing periods.

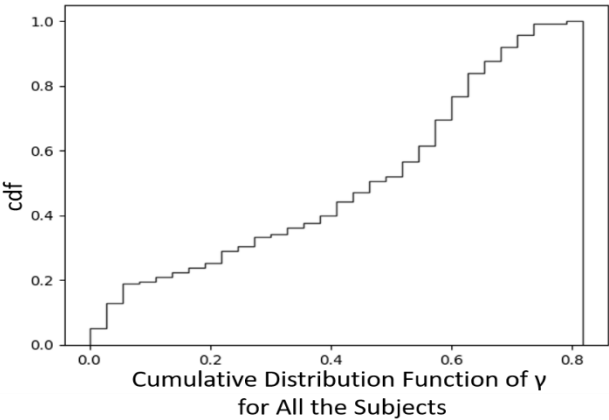


Figure 4-3 Cumulative distribution function of γ for all subjects

Figure 4-3 shows the cumulative distribution function (CDF) of γ for all the subjects. In this study, we choose $\gamma = 0.5$ as the threshold to distinguish touch typists and non-touch typists. If $\gamma \geq 0.5$, implying that the subject needs to look at the keyboard while typing more than half of the time, then the subject is considered as a non-touch typist. Otherwise, the subject is classified as a touch typist.

We finally obtain a total of 69 instances from 23 touch typist subjects and 69 instances from 23 non-touch-typing subjects. The detailed composition of each dataset is shown in Table 4-1. As expected, most of the subjects in the child age group are non-touch typists, and most of the subjects in the college age group are categorized as touch typists. Around 67% of subjects in the elderly age group are classified as touch typists, as far as we know, they worked with computers before they retired.

Dataset	Num. of subjects in Child age group	Num. of subjects in College age group	Num. of subjects in Elderly age group
Touch typist	1	10	12
Non-touch typist	17	0	6
Total	18	10	18

Table 4-1 Detailed composition of the datasets from each age group

Every data instance in our datasets representing the typing activity over one article includes eye gaze and mouse positions in the form of a series of $\langle t_{gaze}, x_{gaze}, y_{gaze} \rangle$ and $\langle t_{mouse}, x_{mouse}, y_{mouse} \rangle$ tuples and keyboard events in the form of a series of $\langle t_{key}, key_{name} \rangle$ tuples, where t_{gaze} , t_{mouse} and t_{key} are the timestamps, x_{gaze} , y_{gaze} , x_{mouse} , y_{mouse} are the on-screen coordinates and key_{name} is the specific key pressed by a subject.

A two-phase heuristic filter [107] is applied to eliminate the impulse noise from the eye-tracking data. Eye gaze fixations are then detected from the processed eye-tracking positions by utilizing the Dispersion-Threshold Identification algorithm [101] with the dispersion as 35 *px* and minimum fixation duration as 170 *ms*, which are presented in the form of a series $\langle t_{fix}, dur_{fix}, x_{fix}, y_{fix} \rangle$ tuples, where t_{fix} is the timestamp when the fixation starts, dur_{fix} is the duration of the fixation and x_{fix}, y_{fix} is the coordinate of the on-screen fixation

position. A series of caret positions are also extracted in the form of $\langle t_{caret}, x_{caret}, y_{caret} \rangle$, which encodes that at moment t_{caret} , the caret is located at $\langle x_{caret}, y_{caret} \rangle$ on the screen. This pre-processed data is used to model the gaze-typing behaviors at a given moment t . At moment t , based on the keyboard events data, we can determine whether a subject is thinking about what is going to write or typing on the keyboard. Keyboard events can also be used to model typing dynamics. Combining with fixation and caret positions allows us to deduce whether a subject is rereading the previously written texts, or just staring at a place and thinking.

4.1.4 Data Distribution

During the experiment, three articles in different genres are generated by each subject. In order to not disturb the subjects' cognitive process, we did not impose many detailed constraints, such as word choices, sentence length, or the number of paragraphs. During the writing, the subjects were also permitted to remove or edit the already-generated texts. To better understand the data, this section presents some descriptive statistical analysis of our writing datasets.

Datasets	Num. words in Reminiscent	Num. words in Logical	Num. words in Creative	Num. words in All genres
Touch typist	266.5	222.0	227.3	237.4
Non-touch typist	237.2	199.0	208.3	216.3

Table 4-2 Number of words among different writing genres

Datasets	Typing speed (WPM) Reminiscent	Typing speed (WPM) Logical	Typing speed (WPM) Creative	Typing speed (WPM) All genres
Touch typist	46.9	44.4	42.2	44.5
Non-touch typist	16.2	15.5	16.7	16.1

Table 4-3 Typing speed in words per minute (WPM) among different writing

Table 4-2 shows the length of articles in different genres generated by touch typists and non-touch typists. According to the table, touch typists tend to generate articles in a longer length. For different genres of articles, reminiscent articles are the longest ones, and creative articles are the shortest ones. Table 4-3 illustrates the typing speed across different groups. Obviously, touch typists type much faster than non-touch typists. Figure 4-4 shows the vocabulary usage for both the touch typists and the non-touch typists. Based on the results, it can be concluded that there is no noticeable difference in the vocabulary usage between touch typists and non-touch typists, and most of the words used are from the top 1000 most-frequently-used Chinese characters *. One interesting observation is that logical writing tends to need a more diverse vocabulary than the other genres of writing.

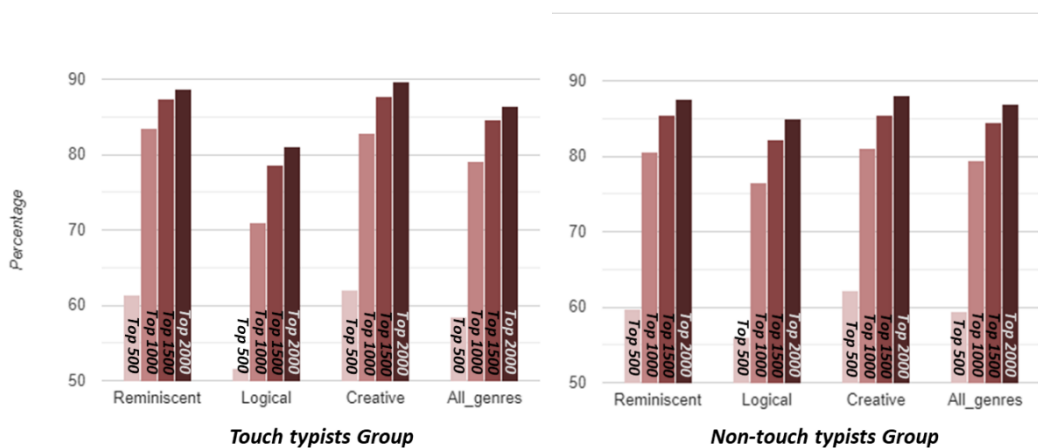


Figure 4-4 Vocabulary usage across different writing genres for touch typists and non-touch typists: percentage of vocabulary in the article belonging to the top N frequently used Chinese characters, where N equals 500, 1000, 1500 and 2000

In our experiment, subjects were asked to write one article in each of the three genres: reminiscent, logical, and creative, respectively. For proper analysis, it is important to know whether the subjects were actually able to follow instructions

*: <http://lingua.mtsu.edu/chinese-computing/statistics/char/list.php>

and generate the appropriate articles for the requested genres, i.e. the content of the articles match the expected genre. This is especially important in the case of the child subjects, who may have less experience writing logical articles. To this end, we recruited two experts, who are high-school Chinese teachers from mainland China, to review the articles written by our subjects. We asked the experts to read the articles and label each of them as reminiscent, logical, creative, or hard to decide. Given an article, the experts labeled it only based on its content without any prior knowledge, such as the expected genre. The articles are presented in random order to the experts.

It is not hard to imagine that different parts of an article may be categorized as different genres. For example, an article presents detailed instructions for cooking a dish (logical), may also include content which touches on the writer's memory and life experience (reminiscent), such as "when I first tried this, I ...". To take this into consideration, in addition to giving one overall genre label, we also asked the experts to rate the genres (reminiscent, logical, creative) for each article by distributing 5 points across the three genres. For example, if an expert thinks that a given article contains about 80% of logical content, and roughly 20% of reminiscent content, he/she would be expected to give the article the overall label of logical and rate the genres as Reminiscent: 1, Logical: 4, and Creative: 0.

Table 15 presents the results of the expert review. Our results suggest that all the articles were correctly written for the requested genres, even for the articles written by our child subjects. Content-wise, most of the children and college students write logical articles with instructions for mathematical operations such as 2-digit number multiplications or divisions, while most of the older adults write articles on cooking or card games. According to the experts, all the logical articles are well written with reasonable and clear logical steps. It is also interesting to

observe that reminiscent and creative articles also have some logical content, as can be seen in Table 4-4, where around 15% of the content in the reminiscent and creative articles was judged as belonging to the logical genre by the experts. This may be due to the fact that the writers feel the need to systematically present their narrations in order to make them believable or convincing.

	Reminiscent (R.)			Logical (L.)			Creative (C.)		
	R.	L.	C.	R.	L.	C.	R.	L.	C.
All	4.21	0.73	0.06	0.13	4.79	0.08	0.02	0.95	4.03
Child	4.24	0.76	0.00	0.04	4.86	0.10	0.00	0.88	4.12
College	4.09	0.60	0.31	0.00	4.91	0.09	0.01	0.85	4.14
Elder	4.25	0.75	0.00	0.28	4.66	0.06	0.00	0.87	4.13

Table 4-4 Results of the expert review – Detailed ratings of articles written by subjects from different age groups

Table 4-5 shows the p-values of one-way analysis of variance (ANOVA) tests [38] on the article genre ratings of different age groups. The resulted p-values suggest that there is no significant difference ($p > 0.05$) in the genre ratings across different age groups. In other words, the experts judge that all the subjects, including the children and elderly, were able to generate the appropriate articles in the requested genres.

	Reminiscent	Logical	Creative
P-value	0.13	0.11	0.15

Table 4-5 P-values of ANOVA tests on article genre ratings for different age groups

4.2 Investigating the Effect of Age Factors

In the last section, we introduce how we construct the writing cognitive

process datasets. Based on the constructed datasets, we first want to explore whether and how age-factors affect the writing cognitive process shown by the gaze-typing behaviors and whether we can detect the age group of a user based on the gaze-typing behaviors. The reason we expect that the age-factor may affect the behaviors is that previous studies [13, 35, 36] find that age-factors tend to affect the cognitive capacity, especially the capacity of working memory, which plays an essential role in writing indicated by the writing model proposed by Flower et al. [30]. In order to thoroughly investigate the gaze-typing behaviors for each cognitive process of writing, we identify time windows in different types to isolate the writing process. Different features to model the gaze-typing behaviors are extracted from different types of the time window, which are analyzed and utilized to detect the age group of each subject.

4.2.1 Identifying the Thinking/Typing Phases through Gaze-typing Dynamics

Based on linguistics and psychology [30], the writing cognitive process involves multiple thinking phases, including planning, translating, and reviewing. The planning phase includes extracting relevant information from long term memory and creative thinking to formulate writing ideas. Then, in the translating phase, the writing ideas are converted into the language by following the context logic in the translating phase, and the translating phase primarily works in short term memory. Finally, the generated sentences are evaluated and revised in the reviewing phase, which can result in a new cycle of planning and translating.

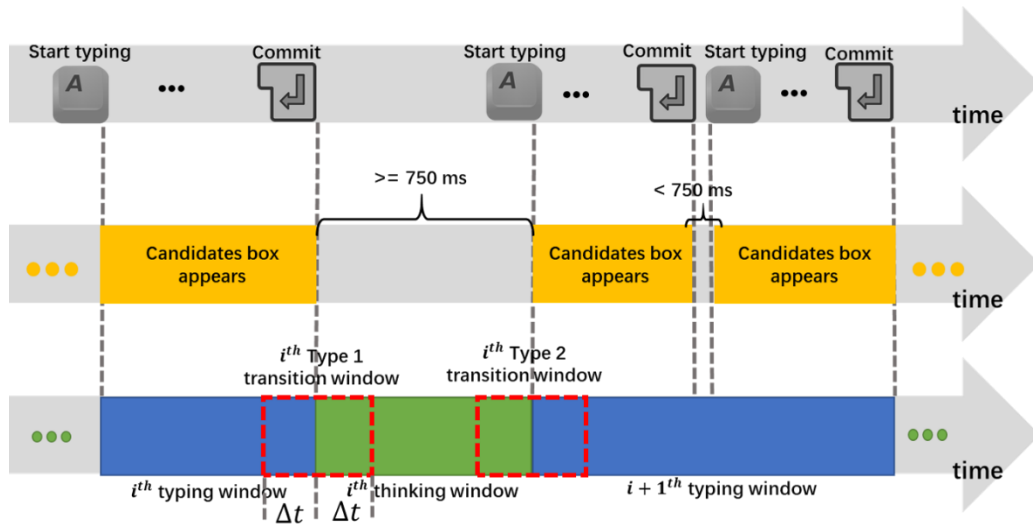


Figure 4-5 The three types of time windows and their correspondence with the appearance of the pop-up candidates box. Two adjacent typing-windows are merged if the time gap is less than 750 *ms*. The duration for each transition window is $2\Delta t$, where Δt is 250 *ms*.

According to this logic, three different temporal windows are defined to capture different behaviors. The *thinking window* is defined as a continuous time period that a subject is fixated on the screen without typing on the keyboard. The *type window* is, on the other hand, defined as the time period that a subject is formulating texts and inputting them by typing on the keyboard. Intuitively, we believe that the thinking windows are likely composed of the planning and reviewing phases of cognitive activity, and the translating phases are mainly included in the typing windows. Inspired by the previous works about the gaze and mouse coordination, which are introduced in Section 2.1, some human cognition elements manifest in the coordination between gaze locations and hands activities. For example, to click a button, a user needs to determine the button location first by eye gaze before moving the mouse toward the button. The spatial and time distance between the gaze and mouse encode the cognitive information of the user. Specific to inputting Chinese text using the Pinyin input method, a user needs first to determine the correct Chinese word/phrase inside the pop-up

candidates box by eye gaze before selecting it by hands. Also, for the non-touch typists, when they start typing, they need to move their gazes down to the keyboard before typing.

Therefore, we identify the third type of time window, namely *transition window*, to capture those behaviors. The transition window is defined as a short time period around the transition point between a thinking window and a typing window. For the transition window, we further define Type 1 as the transition from the typing window to the thinking window and Type 2 as the transition from the thinking window to the typing window. The relationship among these three types of time windows and their correspondence with the candidates box's appearance and typing activities are presented in Figure 4-5.

The appearance of the pop-up candidates box is utilized as an important hint to identify each type of time window from the data. The time period that the candidates box appears on the screen is regarded as one typing window, which starts when the subject type the first keystroke and ends when he/she commits to a word/phrase from a number of candidates. If the time between two adjacent typing windows is less than 750 *ms*, which was considered to be the minimum time needed to interpret five characters [95], we combine these two adjacent typing windows together as one continuous typing window. If the time between two adjacent typing windows is larger or equal to 750 *ms*, we consider it as one thinking window. To validate the appropriateness of the 750 *ms* as the minimum length for a thinking window, we observe the behaviors of subjects inside thinking windows with a duration shorter than 750 *ms*. We find that across all subjects, the average number of fixations inside these windows is 0.93, and only around 10% of time gaze are focused on the screen. Around 70% of the time, the subjects *glance* at the screen, and the gaze stays in the same place for a duration shorter

than the minimum duration of the fixation, which we set at 170 *ms*. One possible purpose of the glance is to confirm that the chosen characters have indeed been generated and appended to the previous text. Around 20% of the time, the subjects do not even look at the screen. Therefore, we consider that a thinking window that is shorter than 750 *ms* can be regarded as a part of the previous typing window.

After identifying typing windows and thinking windows, transition windows can be determined, which is the 500 *ms* period spanning a thinking window and the adjacent typing window. The 500 *ms* parameter was chosen as it has been shown that the gaze usually starts to move 500 *ms* before the mouse moves, and the gaze always leads the mouse [48]. Figure 4-5 presents an example showing the three types of time windows and their relationship to the presence of the candidates box. A whole writing process, thereby, can be considered as a sequence of transitions between typing windows and thinking windows. We believe that different cognitive activities are involved in the thinking and typing windows. Therefore, different behavior patterns should manifest in various types of windows.

4.2.2 Extracting Gaze-typing Features

The gaze-typing features are extracted both at the window-level and the session-level along the spatial and temporal dimensions. The overall process of feature extraction is shown in Figure 4-6. In the feature extraction process, a session is defined as the activity collected during the entire time it takes to compose a given essay. We first identify the time windows in different types from each session based on the typing activity and the candidates box's presence. A session, therefore, is composed of multiple thinking windows, typing windows, and transition windows.

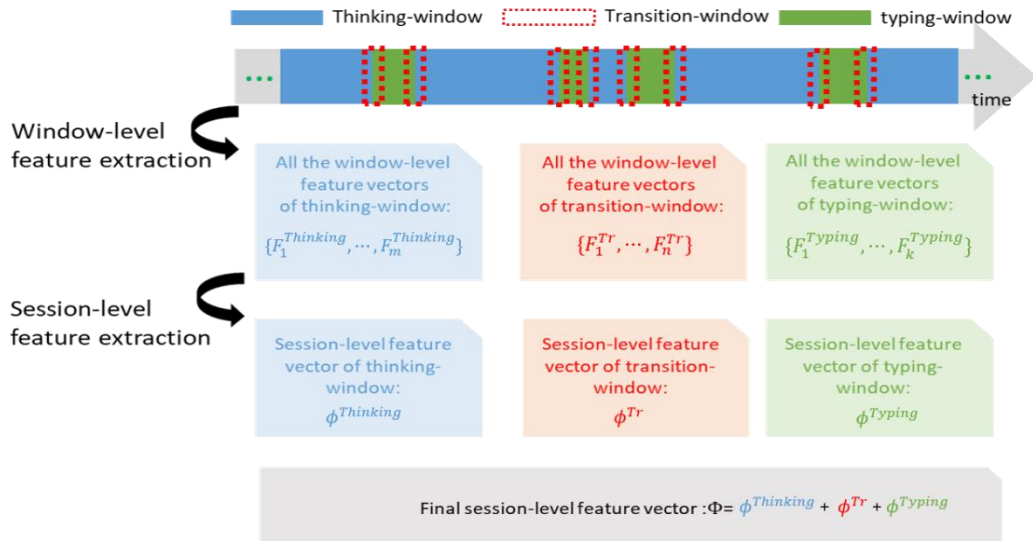


Figure 4-6 Overview of feature extraction. $F^{Thinking}$, F^{Typing} , F^{Tr} are window-level feature vectors and $\phi^{Thinking}$, ϕ^{Typing} , ϕ^{Tr} are session-level feature vectors for each window type. ϕ is the final overall session feature vector that aggregates all information across all window types and all individual windows.

For each type of window, we extract features to generate a window-level feature vector $F^{Thinking}$, F^{Typing} and F^{Tr} from different types of time windows. The feature vectors of the same type are then aggregated to form the session-level feature vectors with corresponding types: $\phi^{Thinking}$, ϕ^{Typing} and ϕ^{Tr} . The final overall session-level feature ϕ is generated by concatenating the session-level feature vectors for the three types of windows together.

The window-level features of the thinking window are first extracted. Based on the definition of the thinking window, there has been no keyboard activity for more than 750 ms. During this period, the subject is expected to either generating writing ideas for the following content or reviewing the already-generated content. Thus, most of the features extracted from the thinking window are used to describe the eye gaze movements.

Feature	Meaning	Formulation
$f_1^{Thinking}$	Duration	Duration of the time-window
$f_2^{Thinking}$	Number of fixations	Number of fixations in the time-window
$f_3^{Thinking}$	Fixation duration	Average duration of fixations in the time-window
$f_4^{Thinking}$	Time that gaze is off-screen	Percentage of time that the subject is looking away from the screen
$f_5^{Thinking}$	Horizontal spread of fixations	Width of the bounding box enclosing all fixations in the time window
$f_6^{Thinking}$	Vertical spread of fixations	Height of the bounding box enclosing all fixations in the time window
$f_7^{Thinking}$	Horizontal distance to the caret	Average horizontal distance of fixations to the caret position (positive to the right)
$f_8^{Thinking}$	Vertical distance to the caret	Average vertical distance of fixations to the caret position (positive downwards)

Table 4-6 Window-level features extracted from the thinking window

Table 4-6 lists the set of features extracted from each thinking window. Based on the definition, an eye gaze fixation is a period of time during which eye gaze is relatively stable within a specific location, and that comprehensive takes place during fixations [95], most of our features are defined by the fixations occurring within the window, especially the duration and the number of fixations. Also, the fixation position relative to the caret position is important, which can indicate whether a subject is in the reviewing phase. We also design features to model the size of the context reviewed through the fixations inside the window. Also, whether a user looks at the screen is important, as some users look away from the writing environment when they are thinking or formulating ideas.

We then extract window-level features for the typing window. The typing window is defined as a period of time that a subject is translating writing ideas into language and inputting by typing on the keyboard. In a typing window, it is also possible for a subject to review the already-generated content so that the

newly generated texts are coherent with previous texts. Thus, from the typing window, both eye fixation and typing features are extracted.

Feature	Meaning	Formulating
f_1^{Typing}	Duration	Duration of the time window
f_2^{Typing}	Number of fixations	Number of fixations captured in this time window
f_3^{Typing}	Fixation duration	Average duration of fixations captured in the time window
f_4^{Typing}	Time that gaze is off-screen	Percentage of time that the subject is looking away from the screen
f_5^{Typing}	Horizontal spread of fixations	Width of the bounding box enclosing all fixations in the time window
f_6^{Typing}	Vertical spread of fixations	Height of bounding box enclosing all fixations in the time window
f_7^{Typing}	Horizontal distance to the caret	Average horizontal distance of fixations to the caret position (positive to the right)
f_8^{Typing}	Vertical distance to the caret	Average vertical distance of fixations to the caret position (positive downwards)
f_9^{Typing}	Number of keypresses	Number of keypresses in the time window
f_{10}^{Typing}	Number of Characters	Number of Chinese characters actually generated in the time-window
f_{11}^{Typing}	Number of Deletes	Number of delete presses in the time window

Table 4-7 Window-level features extracted from the typing window

Table 4-7 presents all the features extracted from the typing window. f_2^{Typing} - f_8^{Typing} are the same features extracted from the thinking window, which model fixations and the relationship between eye gaze and the caret. f_9^{Typing} and f_{10}^{Typing} are two ways to measure the quantity of language is generated. f_9^{Typing} - f_{11}^{Typing} to some extent reflect the amount of effort that a subject dedicates during a typing window.

Lastly, we extract features from the transition window. By definition, the transition window is a short transition period between a thinking window and a

typing window. During this time, a subject either starts to type on the keyboard or has just finished typing and therefore enters the reviewing or thinking phase. Previous studies [49] show that gaze-hand patterns around the transition point are indicative of the cognitive state of the human being. We, therefore, follow their work in extracting similar features around the transition point.

Since half of the subjects in this experiment are non-touch typists, and they have to look down to the keyboard to locate keys. Such behavior results in many travels of the gaze between the screen, keyboard, and the candidates box. Even for the touch typists, though the times of gaze shifting between screen and keyboard are much less than non-touch typists, there are still many times of gaze shifting between content and the candidates box. This information is utilized to extract the following information:

In Type 1 transition-window, the time gap between the last keypress and the gaze beginning to leave the candidates box area is measured. If we cannot find such behavior, then the feature is set to 0.

In Type 2 transition-window, the time gap between the first keypress and the gaze beginning to shift downwards to the keyboard is measured. If the first keypress occurs earlier, then the time gap is measure in a positive value. If there is no such behavior happens during the time period, then the feature is set to 0.

Based on this information, we extract two features, as shown in Table 4-8.

Feature	Meaning	Formulation
f_1^{Tr}	Time to look away from the candidates box	For Type 1 transition-window: Time between the last keypress and gaze moving away from the candidates box area
f_2^{Tr}	Time to look toward the keyboard	For Type 2 transition-window: Time between first keypress and gaze moving toward the keyboard

Table 4-8 Window-level features extracted from the transition window

After extracting the window-level features, two kinds of session-level features are extracted based on statistical behaviors to model the average behaviors and the variation behaviors for each session. We believe that a subject's macro behavior can be modeled by these statistical session-level features.

As previously mentioned in Figure 4-5, a session is composed of m thinking windows, n transition windows and k typing windows in total. After extracting window-level features, we achieve m window-level feature vectors for the thinking windows $F_1^{Thinking}, F_2^{Thinking}, \dots, F_m^{Thinking}$, n feature vectors for the transition windows $F_1^{Tr}, F_2^{Tr}, \dots, F_n^{Tr}$ and k feature vectors for the typing windows $F_1^{Typing}, F_2^{Typing}, \dots, F_k^{Typing}$. $\phi^{Thinking}$ is the session-level feature vector extracted from $F_1^{Thinking}, F_2^{Thinking}, \dots, F_m^{Thinking}$ by computing the mean value and the standard deviation of each feature $f_i^{Thinking}$, where $i \in \{1, 2, \dots, 8\}$ among $F_1^{Thinking}, F_2^{Thinking}, \dots, F_m^{Thinking}$. ϕ^{Typing} and ϕ^{Tr} are constructed in the same way as $\phi^{Thinking}$. The final session-level feature vector ϕ is built by concatenating $\phi^{Thinking}, \phi^{Typing}$ and ϕ^{Tr} together.

4.2.3 Feature Selection

The purpose of feature selection is to select indicative features and exclude the redundant and irrelevant features from the entire feature set. Totally, we extract 42 features, where 16 features are extracted from the thinking windows, 22 features are extracted from the typing windows, and 4 features are extracted from the transition windows. Through the feature selection process, we want to find the optimal subset of features.

We adopt the wrapper method with the best-first search and a stopping function of 10 consecutive non-improving search nodes for feature selection on the leave-one-subject-out mechanism. Specifically, we select features on the data

from $N_{subject} - 1$ subjects based on the classification performance on data from the remaining subject. This process is repeated for $N_{subject}$ times, where $N_{subject} = 46$. A feature is deemed indicative if it is selected multiple times over several folds.

4.2.4 Evaluation of Age-group Detection

Feature	Formulation
$f_1^{Tr_mean}$	Average value of all the time differences between the last keypress and the time that the subject starts to move the gaze away from the candidates box area across the writing process
$f_1^{Tr_std}$	Standard deviation of all the time differences between the last keypress and the time that the subject starts to move the gaze away from the candidates box area across the writing process
$f_9^{Typing_mean}$	Average number of keypress in each typing-window across the writing process
$f_6^{Typing_mean}$	Average value of the vertical spread of fixation in each typing-window across the writing process
$f_8^{Typing_mean}$	Average value of the y-distance between fixations to the caret in each typing window across the writing process

Table 4-9 Selected indicative features of capturing differences among different age-groups

We evaluate our features on detecting the age group of a user while he/she is writing an article on the computer. Since a real-life situation requires a trained model to be able to perform prediction on unseen users, we use the leave-one-subject-out cross-validation for evaluation in our experiment. In other words, the model is built (with feature selection) and trained based on the data from $N_{subject} - 1$ subjects and evaluate on the remaining subject. This process is iterated for $N_{subject}$ times, where $N_{subject} = 46$. The average correctness (CCR) of the model is reported as the overall performance.

Performance	Precision	Recall	F-measure
Ground truth			
Children	0.83	0.81	0.82
College Students	0.66	0.83	0.74
Elderly	0.98	0.85	0.91

Table 4-10 Detailed performance of the age-group detection

Predicted as	Children	College Students	Elderly	Total
Ground truth				
Children	44	9	1	54
College Students	5	25	0	30
Elderly	4	4	46	54
Total	53	38	47	138

Table 4-11 Confusion matrix of the age-group detection

We first want to investigate the indicative features that can capture behavioral variations among children, college students, and elderly typists. After the feature selection step, the initial whole feature set, containing 42 features, is trimmed down to 5 indicative features, which is presented in Table 4-9. Using a support vector machine (SVM) with RBF kernel, we achieve an overall performance (CCR) of 83.3%, compared with a baseline of 39.1%, achieved by classifying every instance as majority class. Table 4-10 presents our model's detailed performance, and the confusion matrix is shown in Table 4-11.

Our feature selection results suggest that age-factors affect gaze-typing

behaviors while writing on the computer, and the transition window and the typing window capture more significant different behaviors across different age groups. Comparing subjects from different age-groups, it appears that college students and children move their gaze away from the candidates box earlier than elderly subjects (**children:** $f_2^{Tr_mean} = 76\text{ms}$, **college students:** $f_2^{Tr_mean} = 59\text{ms}$, **elderly:** $f_2^{Tr_mean} = 96\text{ms}$) and elderly subjects tend to review material just generated more often than other groups during the translating phase (**children:** $f_6^{Typing_mean} = 39\text{ px}$, $f_8^{Typing_mean} = -173\text{ px}$ **college students:** $f_6^{Typing_mean} = 36\text{ px}$, $f_8^{Typing_mean} = -186\text{ px}$, **elderly:** $f_6^{Typing_mean} = 32\text{ px}$, $f_8^{Typing_mean} = -147\text{ px}$). In addition, elderly subjects use fewer keypresses to generate the same number of Chinese characters (**children:** 53 keypresses, **college students:** 63 keypresses, **elderly:** 41 keypresses to move the caret 1000 px forward), mainly as college students and children prefer to input longer strings of pinyin equivalents compared with the elderly, who are more willing to input in shorter phrases or even character-by-character, resulting in more keystrokes in verifying and committing to the text.

It is interesting to notice that no indicative feature selected belongs to $F^{Thinking}$, which suggests the differences in typing behaviors and the gaze-hand transition behaviors among different age groups are more substantial than the difference in thinking behaviors while writing. According to the results presented in Table 4-10 and Table 4-11, it illustrates that based on the gaze-typing behaviors, we can successfully detect the age-groups of users. It also shows that among different age-groups, the elderly can be differentiated easier from the others compared to discriminating between children and college students. It can be explained that almost all the subjects in the elderly group are non-touch typists and share similar typing behaviors.

4.3 Investigating the Effect of Article Genres

In the last section, we investigate the effect of the age-factors on the cognitive process of writing based on gaze and typing behaviors. Two kinds of time windows: thinking window and typing window, are defined to divide the writing process into thinking and typing processes. We extract different gaze-typing features from both thinking process periods, typing process periods, and transition periods between the thinking and typing processes to model behaviors. Finally, results show that gaze-typing features extracted from the typing windows and transition windows are more indicative of determining the age group of the subject.

In this section, we want to explore the effect on the cognitive process of writing illustrated by the gaze and typing behaviors when a subject generates texts in different article genres. Similarly, different types of time windows are defined to isolate the various stages of the writing process. However, just dividing the writing processes into the thinking process and typing process is not enough this time, since the effect of article genres is more complicated than the effect of age-factors, which can be easily overshadowed by other factors, such as writing skills. Hence, the thinking window and typing window are further divided into sub-categories based on the gaze and typing activities so that we can capture the behaviors at a more detailed level.

4.3.1 Identifying Different Writing Processes

Figure 4-5 presents the relationship of different types of time windows with the writing activity and the appearance of the candidates box. Basically, a typing window is a consecutive time period that a subject is typing on the keyboard. On the other hand, a thinking window is a consecutive time period that there is no keyboard activity, and a transition window is a short period of time spanning two

adjacent typing window and thinking window. The detailed definitions and procedures of identifying each kind of time window can be found in Section 4.2.1. Therefore, a whole writing process can be treated as a series of transitions between thinking windows and typing window. We believe that different cognitive activities are involved in the thinking and typing windows. Therefore, different behavior patterns should manifest in different types of windows.

4.3.1.1 Types of Thinking Window

The thinking window is a period of time between two typing windows when there is no typing activity. It has two main functions: 1) to review the texts that just be generated and 2) to think about what to write next. We expect these two functions will generate different behavior patterns. For example, if a subject is in the reviewing phase, there is a higher possibility that he/she may be rereading the already-generated texts, with more scanning behavior, and if a subject is in the planning phase recalling some writing material, we expect fixations with longer duration. Therefore, to better capture the changing of the cognitive activities, we differentiate the thinking into three types: off-screen (*O*), reading (*R*), and fixating (*F*), based on the behavior patterns.

A thinking window is determined as Type *O*, if a subject does not look at the screen for more than 50% of the time window. This thinking window appears more frequently when the subject is a non-touch typist. Two possible scenarios during Type *O* thinking window may occur are when a subject is conceiving what to write next, or when a subject is recalling material. Alamargot et al. [6] show that when a subject is composing a text, long pauses are observed when he/she is contemplating *what to write next*, or when he/she is considering the best way to express ideas. During this period, attention may not necessarily stay focused on

the writing environment, which is referred to as *averting the gaze*. Therefore, Type *O* thinking windows exist in both the planning and translation phases.

Type *F* thinking windows are similar to Type *O* windows. During the window period, a subject focuses on the writing environment for a long time (long fixations) without rereading the already-generated texts (lack of reading saccades).

A thinking window of Type *R* happens when a subject spends the majority ($\geq 50\%$) of the time rereading previously written texts. According to previous studies [30, 62], rereading often occurs when a subject externalizes his/her ideas into text or reviews what he/she just writes. Thus Type *R* thinking windows appear in both the translation and reviewing phases.

4.3.1.2 Types of Typing Window

The primary function of a typing window is to generate the actual text which was formulated in the last thinking window. Based on the typing behavior, we define three different types of typing windows: windows with lower keystroke frequency (*L*), windows with uniform keystroke intervals (*U*), and windows with non-uniform keystroke intervals (*N*). These windows capture different types of typing behavior patterns, which may reflect different mental states of the subject.

Type *L* typing windows are usually shorter in duration, as the keystroke frequency is lower, containing fewer keypresses. We set the threshold to be not more than four keystrokes, including the final committing press that selects the character(s) to be generated. Considering that the average number of keypresses per typing window is 10.0, which is rough equals to 3-5 Chinese characters, these kinds of typing windows are fairly uncommon. When typing in Chinese, the usual practice is to generate the approximation of a sequence of Chinese characters in the same candidates box before committing. As shown in Table 4-12, many

phrases generated in Type *L* typing windows are functional phrases, especially auxiliary words, which are often used with a main verb to express tense, aspect, modality, voice, emphasis, etc., and may reflect the mental state of the subject.

Type	Percentage	Type	Percentage
Auxiliary	23%	Link verb	3%
Preposition	7%	Pronoun	1%
Conjunction	7%	Other	54%
Adverb	4%		

Table 4-12 Types of phrases generated in Type L typing window

Window	Type	Description
Thinking window	Off-screen (<i>O</i>)	The subject looks away from the screen for the majority ($\geq 50\%$) period of the window
	Reading (<i>R</i>)	The subject rereads the texts ahead of the caret
	Fixation (<i>F</i>)	The subject fixates on a place on the screen
Typing window	Less-press (<i>L</i>)	The subject presses fewer keys during the period of the window
	Pressing with uniformed keypress intervals (<i>U</i>)	The subject presses several keys and time intervals between every two keypresses are similar in length
	Pressing with non-uniformed keypress intervals (<i>N</i>)	The subject presses several keys, and there exists at least one time interval between two keys, whose length is significantly greater than others

Table 4-13 Different types of thinking window and typing window based on gaze and typing activities

Type *N* and Type *U* typing windows contain more than four keystrokes. The distinction between them is that Type *N* typing windows contain at least one interval between successive keypresses, which lasts significantly – at least three standard deviations (over all keypress intervals of the subject) – longer than the others. This distinction attempts to capture pauses in writing, which indicates cognitive processing [121]. Table 4-13 lists all the types of thinking windows and

typing windows with their descriptions.

4.3.2 Extracting Statistics-based Gaze-typing Features from

Time Windows

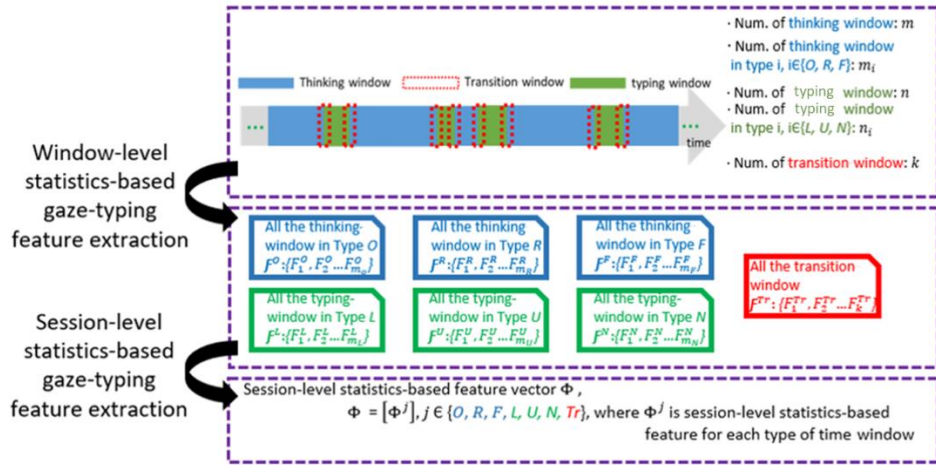


Figure 4-7 Overview of feature extraction of statistics-based gaze-typing features

Similar to what we do for the age-factors detection, statistics-based gaze-typing features both at window-level and session-level along the temporal and spatial dimensions are extracted. The process of extracting statistics-based gaze-typing features is shown in Figure 4-7. We define a session as the activity collected during the time it takes to compose a given article. Similarly, thinking windows and typing windows are extracted from the session using the appearance and disappearance of the candidates box as indicators. A session, therefore, consists of multiple thinking windows, typing windows, and transition windows. We further differentiate thinking windows and typing windows into different types based on the gaze and typing activities listed in Table 4-13. For each type of thinking window and typing window, different sets of features are extracted to generate a window-level feature vector F_i^j , where $j \in \{O, R, F, L, U, N, Tr\}$ indicates the

type of the feature vector and i indicates that the feature vector is extracted from the i^{th} time window in type j of the session. The feature vectors of the same type are then aggregated to form the session-level feature vector ϕ^j , where $j \in \{O, R, F, L, U, N, Tr\}$. Appending the session-level feature vectors for different types of thinking window and typing window together gives us the final overall session-level statistics-based gaze-typing feature vector ϕ , where $\phi = [\phi^j]$.

4.3.2.1 Extracting Statistics-based Gaze-typing Features from Thinking Window

After defining three different types of thinking windows based on the gaze behavior patterns during the window period, we can construct the window-level features to capture behavioral differences when generating articles in different genres.

Feature	Meaning	Formulation
f_1^O	Gaze off-screen duration	Sum of the duration when gaze is off-screen

Table 4-14 F^O : Features describing the behavior in Type O thinking window

Since there are no keyboard events during the thinking window, by definition, thinking window features are related to the eye gaze. In Type O thinking windows, we want to model behavior that characterizes a subject's formulating ideas for additional content while not focusing on the writing environment. However, since we cannot detect the gaze position reliably when the subject's gaze is off-screen, the only feature (f_1^O) that we can define is the duration while the subject's gaze is off the screen, as shown in Table 4-14. This feature gives us a sense of the length of the pause while the subject either recalls the material that

will be generated next or while he/she translates ideas into texts. We define a time period as being an off-screen gaze if 1) the eye tracker cannot capture any eye gaze inside the screen area and 2) the duration of the period is equal to or larger than 400 ms, which is the average duration of an eye blink [12].

In Type R thinking windows, a subject is mainly rereading already-generated texts. We thus design the first part of the feature set (f_1^R) to describe the text that is being reread by the subject. If the location of the text that is being read is close to the caret, it is likely that this text was just generated in the previous typing windows, and the subject is likely to be in a reviewing phase. However, if the location that is being read is 2 or 3 sentences away from the caret, then the subject may be translating his/her ideas into a sentence that integrates with the previous text.

Feature	Meaning	Formulation
f_1^R	Distance of the reread texts	Average distance between all the reread texts to the caret during the time window ($Mean(d_i)$)
f_2^R	Length of the reread texts	Total length of the reread texts ($Sum(l_i)$)
f_3^R	Rereading duration	Total duration spent in rereading already-generated texts
f_4^R	Number of fixations	Total number of fixations in the time window
f_5^R	Duration of fixations	Average duration of the fixations in the time window

Table 4-15 F^R : Features describing the behavior in Type R thinking window

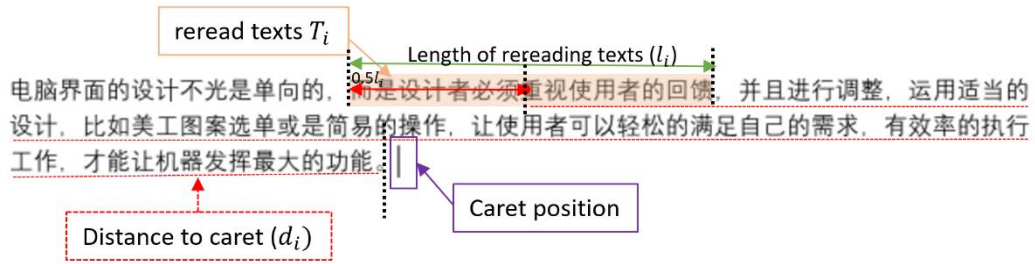


Figure 4-8 Illustration of the features that describe the reread texts

Another feature (f_2^R) measures the amount of texts reread by the subject. We define the distance between the reread texts and the caret as the number of pixels from the midpoint of the reread texts to the position of the caret along the text line. Figure 4-8 illustrates an example. The green line shows the reread texts, and the red dash line denotes the distance to the caret. We also extract the features ($f_4^R - f_5^R$) to describe the fixation, including the number of fixations and average duration of fixations. Table 4-15 lists this set of features.

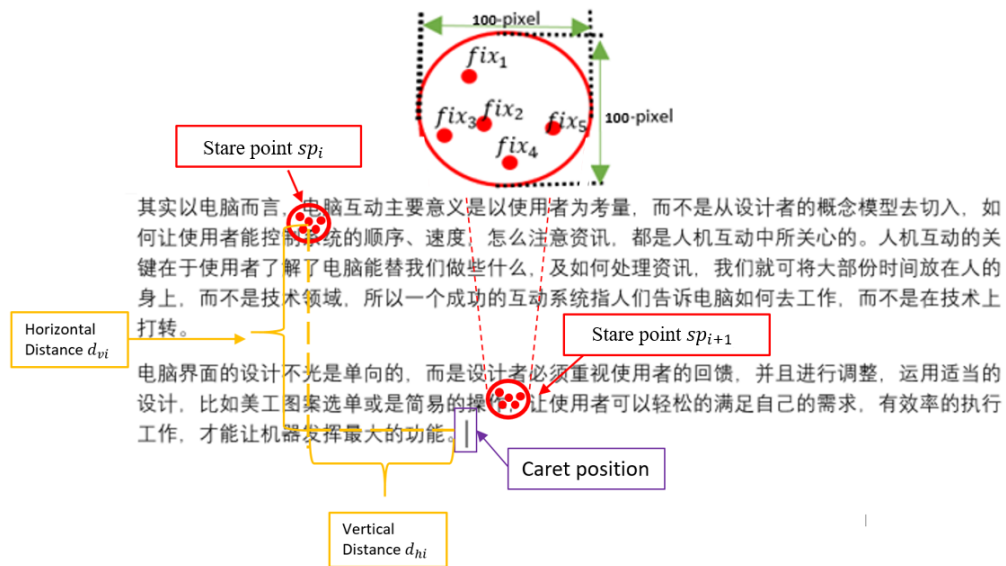


Figure 4-9 Illustration of the features that describe the staring point

For Type F thinking windows, we want to capture behavior patterns similar to Type O windows but model the act in which a subject fixates on the screen

without rereading already-generated texts. Besides features (f_1^F), which measures the duration of the pause, we also extract features (f_2^F) to describe the location of the fixation relative to the caret, as shown in Figure 4-9, and features ($f_4^F - f_5^F$) that describe the fixation.

We observe from our data that there are time periods during which the user seems to stare at a small area for an extended period of time. This behavior generates many fixations within a small area. This also appears to be correlated with thinking behavior on the part of the user, as they do not seem to correspond to reading behavior. We, therefore, define these *stare points* (sp) as areas with a radius of 50 *pixels* or less (two Chinese words take up 100 *pixels*) with several fixation points.

The distance between each stare point and the caret is measured from the center of the staring point to the center of the caret, and the duration of the i^{th} stare point (sp_i) t_i is defined as the total duration of all fixations in stare point sp_i . Table 4-16 shows all the features with meaning and formulation.

Feature	Meaning	Formulation
f_1^F	Horizontal distance to the caret	Average horizontal distance of stare points to the caret position ($Mean(d_{hi})$)
f_2^F	Vertical distance to the caret	Average vertical distance of stare points to the caret position ($Mean(d_{vi})$)
f_3^F	Total duration spent in staring and thinking	Total duration spent in staring at screen and thinking
f_4^F	Number of fixations	Total number of fixations in the time window
f_5^F	Duration of fixation	Average duration of fixations in the time window

Table 4-16 F^F : Features describing the behavior in Type F thinking window

4.3.2.2 Extracting Statistics-based Gaze-typing Features from Typing Window

The definition of the typing window is the period of time during which a

subject is typing on the keyboard, which we believe corresponds to the activity of translating ideas into language. As previously introduced, a typing window contains keystrokes processed by the system through a series of pop-up candidates boxes. Similar to the thinking window, we define different types of typing window based on typing patterns and design different groups of features to model the behavioral patterns.

Feature	Meaning	Formulation
f_1^L	Duration	Duration in which the pop-up candidates window is visible on the screen
f_2^L	Keypress interval	Average interval between every two keypresses

Table 4-17 F^L : Features describing the behavior in Type L typing window

Table 4-17 presents the features extracted from Type L typing windows. These windows contain a few keyboard presses, which we observe usually correspond to the generation of functional characters or phrases. The language modeling inside the keypress-to-character conversion mapping sorts commonly-seen characters or phrases to the top, which means that the user often only needs to type the first character instead of the complete phonetic mapping. For example, the phonetic mapping for "I" and "We" are "wo" (我) and "wo men" (我们), respectively. Since these words are so often used, the Chinese input software will generate these words as soon as the user types "w", without the following "o". Because they are so commonly used, these characters are usually generated proficiently and at high speed. Features ($f_1^L - f_2^L$) are designed to capture the impact of the different cognitive activities on the generation of these common terms.

Type U and Type N typing windows contain more keypresses. This allows us to extract more complex features to model behavior patterns. Wallot et al. [121]

illustrate that, compared with simple typing, generating texts create more complex keystroke activity, which manifests in two ways: 1) longer keypress intervals, reflecting longer pauses in writing, and 2) increased number of edition and deletions. Our features are designed to describe these two aspects of behaviors, as described in Table 4-18 and Table 4-19.

Feature	Meaning	Formulation
f_1^U	Number of keystrokes	Total number of keystrokes during the window period
f_2^U	Keypress interval	Average interval between every two keypresses
f_3^U	Recurrence	Total number of deletes and edits performed during the window period
f_4^U	Duration	Duration in which the pop-up candidates window is visible on the screen

Table 4-18 F^U : Features describing the behavior in Type U typing window

Feature	Meaning	Formulation
f_1^N	Number of keystrokes	Total number of keystrokes during the window period
f_2^N	Keypress interval	Average interval between every two keypresses
f_3^N	Pause duration	Total duration of intervals, in which the duration is 3-deviations away from the average
f_4^N	Recurrence	Total number of deletes and edits performed during the window period
f_5^N	Duration	Duration in which the pop-up candidates window is visible on the screen

Table 4-19 F^N : Features describing the behavior in Type N typing window

4.3.2.3 Extracting Statistics-based Gaze-typing Features from Transition

Window

Like what we do for the age-factors detection, we also extract statistics-based gaze-typing features from the two types of transition window, where the transition window in Type 1 captures the transition from the typing process to the thinking process and the Type 2 transition window captures the transition from the opposite direction. Features extracted from the transition window are utilized to model the gaze-hand transition, which is either a subject moves his/her gaze from the screen to the keyboard and starts to type, or he/she finishes selecting the intended word/phrase and moves his/her gaze away from the candidates box area, which is presented in Table 4-20.

Feature	Meaning	Formulation
f_1^{Tr}	Time is taken in looking away from the candidates box	For Type 1 transition windows: Time between the last keypress and gaze moving away from the candidates box area
f_2^{Tr}	Time is taken in looking towards the keyboard	For Type 2 transition windows: Time between first keypress and gaze moving toward the keyboard

Table 4-20 F^{Tr} : Window-level features extracted from the transition window

4.3.2.4 Building Session-level Statistics-based Gaze-typing Features

Session-level statistic-based gaze-typing features are used to model the overall gaze-typing behaviors in a session, which is the activity collected during the entire time of composing a given article. We believe these statistics-based level features can represent the macro behaviors of a subject. Therefore, we extract two types of session-level features based on statistics from the window-level features: the average behavior and the variation inside a session. For example, a session consists of m thinking windows, which include m_O Type O , m_R Type R and m_F Type F thinking windows, where $m = m_O + m_R + m_F$. There are also n typing windows, which includes n_L Type L , n_U Type U and n_N Type N typing windows, $n = n_L + n_U + n_N$. k transition windows are also extracted from the session.

A window-level feature vector is extracted from each time window based on its type as introduced before. We construct ϕ^j , a session-level statistics-based feature vector of type j , where $j \in \{O, R, F, L, U, N, Tr\}$, by computing the mean value and standard deviation for each feature from the window-level feature vector F^j across all the time windows in type j during the session. For instance, the session-level statistics-based feature vector ϕ^R would be calculated as $[\mu_1, \sigma_1, \dots, \mu_5, \sigma_5]$, where μ_i and σ_i are the mean and standard deviation of the i^{th} feature in the window-level feature vector F^R across all Type R thinking windows and $i \in [0, 5]$. Session-level statistics-based feature vectors for other types of windows can be extracted in the same way. The final overall session-level statistics-based feature vector ϕ is built by concatenating all types of session-level feature vectors together.

4.3.3 Extracting Sequence-based Gaze-typing Features from

Session

Our statistics-based features are used to model the gaze-typing behavior patterns inside each type of time window. In construct, the sequence-based features are designed to model the change of a subject's behaviors across the session, which we hypothesize can distinguish between writing genres. To build the sequence-based features, we first construct the behavior-transition sequence for each session, which captures the whole of the behavior transition exhibited by a subject across an entire session. We then extract *indicative* subsequences, or patterns, from this behavior-transition sequence. The details of the process are described in this section.

4.3.3.1 Modeling the Behavior Transition within a Session

Based on the definition, a session represents the activity during the entire process of composing an article, which can be represented as a sequence of transitions between thinking windows and typing windows. We also introduce how we categorize thinking windows and typing windows into six types that are designed to capture distinctive behaviors. Following this, we model the change in users' behaviors during the whole process of writing an article through the transition over the different types of time windows within a session.

For instance, the i^{th} session ($Sess_i$) contains m thinking windows and n typing windows. Since thinking windows and typing windows appear alternately, thus $|m - n| = 1$ or $m = n$. Given this, we generate a session-level behavior sequence $s_i = \{state_i\}_{i=m+n}$, where $state_i \in \{O, R, F, L, U, N\}$ corresponds to the type of i^{th} time window in $Sess_i$. The label of s_i is the genre of $Sess_i$, which can be *Reminiscent*, *Logical*, or *Creative*.

	Length of Sequence	State Ratio					
		O	R	F	L	U	N
Reminiscent	259	15.7%	11.4%	23.6%	13.8%	12.0%	23.4%
Logical	245	14.6%	12.4%	23.8%	13.5%	11.3%	24.4%
Creative	237	14.9%	11.8%	23.8%	13.2%	12.1%	24.2%

Table 4-21 Overview of behavior types for different genres of writing

Table 4-21 presents an overall of behavior information, including the average length of the behavior sequence, and the distribution of the various behavior types within the sequences, for each of the writing genres in our dataset.

4.3.3.2 Extracting Indicative Patterns from the Behavior Sequence

A pattern is a subsequence of behaviors, which can be regarded as a series of actions. For example, a pattern $F \Rightarrow U$ is commonly seen in our dataset, and it describes the situation whereby a subject stares at the screen for a while to think, followed by typing texts on the keyboard with uniform keypress intervals. However, this pattern is so frequently seen in all behavior sequences across different writing genres. In this sense, it is not indicative as its presence does not provide distinguishing information between the different genres of writing.

An *indicative* pattern is, therefore, a subsequence, which occurs differently across behavior sequences from different genres of writing. In order to judge the degree of *indicativeness*, we define a weighting scheme that assigns an appropriate weight to each pattern to imply the amount of genre information provided by that pattern. Inspired by the work from text categorization [24, 66, 93], our pattern weighting scheme comprises of three components: pattern frequency (pf), relevance frequency (rf) and trend distance weightings (td). The

weighting (w) can be computed as:

$$w = pf \times rf \times td \quad 4-1$$

Relevance frequency (rf) component is introduced first. Relevance frequency was initially proposed by Lan et al. [66] for text categorization. In the traditional text categorization problem, the rf factor is a supervised term, weighted with their indicativeness, which can be roughly interpreted as their power of discriminating the documents into positive and negative categories.

We map the original problem to our sequence classification task by viewing patterns and behavior sequences to terms and documents. We map each of the genres reminiscent, logical, and creative to positive and the two other genres to negative in turn. Given all behavior sequences with positive labels (S_+) and all sequences with negative labels (S_-), then the relevance frequency of pattern p can be computed as:

$$rf(p, S_+, S_-) = \log \left(2 + \frac{|\{s \in S_+ : p \in s\}|}{|\{s \in S_- : p \in s\}|} \right) \quad 4-2$$

Where $|\cdot|$ returns the number of elements in the set.

The relevance frequency formula gives higher weights to patterns that infrequently occur in S_+ class and more frequently in S_- class. However, there is a possibility that it will identify rare patterns, which occur only once or twice in the entire dataset. These patterns are not helpful for our purpose, as they may not be generalizable. We, therefore, include the pattern frequency factor to balance the indicativeness with generalizability. Pattern frequency (pf) measures how frequently a pattern p occurs in a behavior sequence. Since the length of the sequence may vary from session to session, the pattern frequency is normalized by the length of the sequence. Given a behavior sequence s , the pattern frequency

of a pattern p can be computed as:

$$pf(p, s) = \log(n_{p,s} / len(s)) \quad 4-3$$

Where $len(\cdot)$ returns the length of the sequence and $n_{p,s}$ is the number of occurrences of pattern p in the behavior-transition sequence s .

Lastly, trend distance (td) is introduced. The process of writing an article is dynamic, and as such, the writing behaviors may change during the writing process. For example, when a subject writes a reminiscent article, recall behavior may appear more frequently at the beginning than at the end of the writing. Figure 4-10 presents an example. We have behavior sequences s_1 and s_2 , belonging to different genres of writing, both of which contain 15 occurrences of Patterns p_1 and p_2 . On the surface, it appears that p_1 and p_2 are not significantly discriminative. However, when we consider the different stages of writing, it can be seen that p_1 and p_2 have very different appearance patterns - p_1 appears more frequently toward the beginning of s_1 , and more frequently towards the end of s_2 . These kinds of differences cannot be readily captured by tf and rf factors. Hence, we need a new factor to capture this difference.

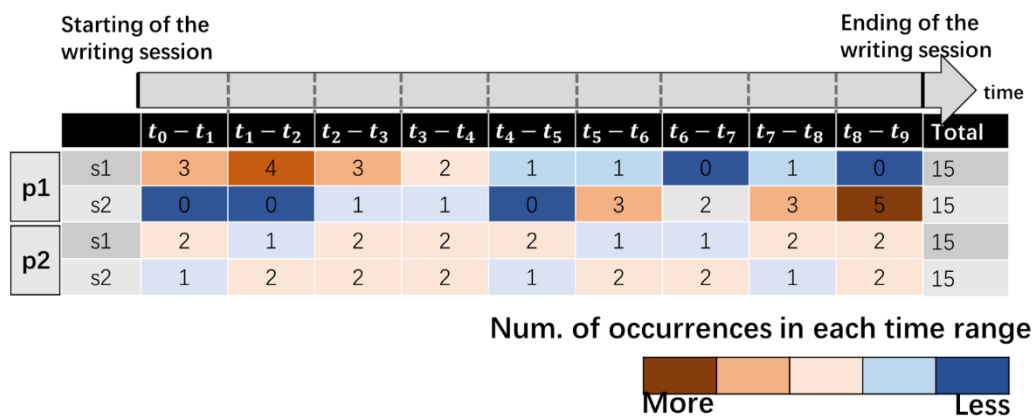


Figure 4-10 Examples of two patterns, which have the same total occurrence times but gave different trend distance weighting

Based on this analysis, we propose a new weighting factor, which we call the trend distance weighting, which takes into account the occurrences across the whole process of writing an article. We first assume that a writing process consists of π behavior subsequences. Figure 4-11 shows how we generate our behavior subsequences. We first divide the session into π stages of equal duration (red dotted box). The behavior subsequence s^{par_i} is then simply the sequence of time window types of the windows that appear in the partition. In our example, the i^{th} stage consists of 5 time windows with Types U, O, U, F, L . The behavior subsequence s^{par_i} is therefore $U \Rightarrow O \Rightarrow U \Rightarrow F \Rightarrow L$. Likewise, s^{par_i+1} is $O \Rightarrow U \Rightarrow O \Rightarrow N \Rightarrow R \Rightarrow N$.

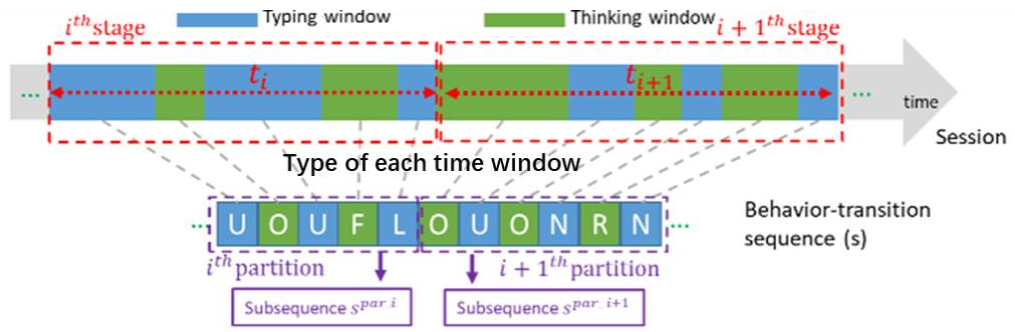


Figure 4-11 Generating behavior subsequences from session data

To compute the trend distance weighting, first, we count the number of occurrences of the pattern in each partition. Given a behavior-transition sequence s_i with π partitions, the number of occurrences of the pattern p in each partition can be expressed as a vector $N_{s_i}^p$ and $N_{s_i}^p = [f_{p,s^{part_1}}, f_{p,s^{part_2}}, \dots, f_{p,s^{part_\pi}}]$. Given all behavior-transition sequences with positive labels (S_+) and all sequences with negative labels (S_-), the trend distance weighting (td) of a pattern p can be computed as:

$td(p, S_+, S_-) = \text{Euclidean}(Q_{S_+}^p, Q_{S_-}^p)$, where

$$Q_{S_+}^p = \frac{\sum_{i=1}^{|S_+|} N_{s_i}^p / \|N_{s_i}^p\|}{|S_+|}$$

$$Q_{S_-}^p = \frac{\sum_{i=1}^{|S_-|} N_{s_i}^p / \|N_{s_i}^p\|}{|S_-|} \quad 4-4$$

4.3.3.3 Sequence-based Gaze-typing Features

So far, we have defined a weighting scheme to select indicative patterns, which represents some gaze-typing behaviors that may potentially distinguish writing activities based on the exhibited behaviors. When a subject writes an article in one of the reminiscent, logical, creative writing genres, he/she is more likely to show behaviors that are indicative of that genre. This means that extracted patterns that are indicative for a particular genre should occur more frequently in behavior sequences generated from writing sessions corresponding to the genre.

We use a bag-of-words model [122] to generate the sequence-based gaze-typing features from the behavior-transition sequences. We select the k highest-weighted patterns as our indicative patterns, or *words*, and represent each behavior-transition sequence by the occurrence frequencies of the word contained in a bag-of-words approach. If k patterns are selected, and each behavior-transition sequence contains π partitions, then the size of the sequence-based gaze-typing feature vector is $k \times \pi$ and the value of the i^{th} entry is the number of occurrences of the $(\lfloor (i-1)/\pi \rfloor + 1)^{th}$ pattern in the $(i - 3 \times \lfloor (i-1)/\pi \rfloor)^{th}$ partition of the sequence.

4.3.4 Evaluation of Detecting Article Genres

We evaluate our statistics-based and sequence-based gaze-typing features on the task of detecting the genre of an article that a subject is currently working on. In this section, we first analyze features to understand gaze and typing behaviors across different genres, and then we build our article genre detection model based on the analysis results. The detection model is evaluated on the datasets that we constructed in Section 4.1, and the performance will be reported at the end of the section.

4.3.4.1 Understanding Statistics-based Gaze-typing Features

Statistics-based gaze-typing features are extracted from different types of the time window. Since our objective is to build a user-independent model, we want our features to be effective at capturing behavior differences across different subjects. However, for different subjects, the range of a feature can be entirely different. For example, some subjects are used to generating a series of Chinese characters in one candidates box and then revising them by correcting typos. On the other, some subjects are used to typing phrase by phrase or even character by character. This means that the range of the features: the number of keystrokes (f_1^U and f_1^N) is completely different, and the raw features f_1^U and f_1^N are not generalizable across users. To solve this problem, we apply min-max normalization for all statistics-based gaze-typing features across different sessions of the same subject to mitigate the effect of user variation. After normalization, the ranges of all the features are within $[0,1]$. Since the scope of normalization is across all the sessions in different article-genres of the same subject, so the normalized features are still capable of capturing the differences between the

different article-genres and can be compared across different subjects and used in a user-independent fashion.

To better understand the gaze-typing behaviors, we analyze the window-level statistics-based gaze-typing features in different article-genre sessions by answering two questions: 1) whether there is a significant difference between different article-genre groups for each feature and 2) how they are different.

First, we group all the window-level features with the same type together, and then they are divided into three groups: reminiscent, logical, and creative, based on the genre of their corresponding article. A Kruskal Wallis H test [112] is then performed to test whether features in the three groups originate from the same distribution. In other words, if the test shows that a particular feature is significantly different, it means that feature can potentially capture the differences between writing articles in different genres. Kruskal Wallis H test is a non-parametric method, which is the extension of Mann-Whitney U test [86] to support multiple groups (more than 2) comparison. Compared with the one-way analysis of variance test, Kruskal Wallis H test does not need the population to be normally distributed, nor does it assume that standard deviations of the groups are all equal. To see out how these features are different across different article-genres, we apply the Dunn's test with Bonferroni correction [26], a non-parametric post hoc test, on the features shown significant by the Kruskal Wallis H test. Table 4-22 lists all the significant features by the Kruskal Wallis H test with their p-values with correction of Dunn's test for touch typists. For both p-values, if $p \leq 0.05$, then it will be considered as *significant* under such a test. The mean values of all significant features are also shown in the table for comparison across different groups.

Results show that rereading behaviors differ between the writing reminiscent and creative articles. When a subject composes an article in the creative genre,

he/she tends to spend more time in rereading already-generated texts with more fixations, but each fixation is shorter in duration compared with composing an article in the reminiscent genre. Intuitively, the results make sense. Rereading behaviors appear more frequently in the translating phase and reviewing phase. Compared with reminiscent writing, composing an article in the creative genre requires continually ensuring that the plot is reasonable. Therefore, it makes sense that they spend more time rereading the texts and reread longer chunks of text.

We also observe some *pause* behaviors when a subject stares at a position on the screen for a while during the typing period. During this time, the candidates box window remains on the screen, but there are few gaze movements and no keypresses. These pauses appear less often while composing reminiscent articles. One possible reason is that reminiscent writing is less complex compared with others. It is known that the frequency and duration of these pause behaviors are positively correlated with the complexity of the writing task [121].

Significant feature	P-value with correction of Dunn's test			P-value of Wallis H test	Normalized Mean		
	R vs.L	R vs.C	L vs.C		R	L	C
Length of the reread texts in Type R (f_2^R)	1.00	1.00	0.02	0.02	0.30	0.27	0.32
Reading duration in Type R (f_3^R)	0.31	0.01	0.01	0.01	0.13	0.14	0.17
Number of fixations in Type R (f_4^R)	0.07	0.01	1.00	0.01	0.10	0.12	0.14
Duration of fixations in Type R (f_5^R)	1.00	0.02	0.01	0.01	0.32	0.31	0.28
Total duration of staring and thinking in Type F (f_3^F)	0.01	0.01	1.00	0.01	0.12	0.13	0.13
Duration of typing in Type L (f_1^L)	0.01	1.00	0.01	0.01	0.10	0.12	0.09
Keypress interval in Type L (f_2^L)	0.01	0.92	0.01	0.01	0.10	0.11	0.09
Pause duration in Type N (f_3^N)	0.02	0.01	1.00	0.01	0.08	0.10	0.11
Duration of typing in Type N (f_5^N)	0.28	0.04	1.00	0.04	0.11	0.13	0.17
Time to look toward keyboard in Type 2 transition window (f_2^{Tr})	0.01	0.01	0.01	0.01	0.45	0.52	0.39

Table 4-22 Kruskal Wallis H test and Dunn's test results of significant statistics-based gaze-typing features for touch typists (R: Reminiscent, L: Logical, C: Creative)

Significant feature	P-value with correction of Dunn's test			P-value of Wallis H test	Normalized Mean		
	R vs. L	R vs. C	L vs.C		R	L	C
Duration of fixations in Type R (f_5^R)	<0.01	0.87	0.01	<0.01	0.28	0.33	0.30
Total duration of staring and thinking in Type F (f_3^F)	0.24	0.01	0.06	<0.01	0.19	0.20	0.16
Duration of fixations in Type F (f_5^F)	0.37	<0.01	0.10	<0.01	0.24	0.23	0.20
Duration of typing in Type L (f_1^L)	1.00	<0.01	<0.01	<0.01	0.14	0.13	0.12
Keypress interval in Type L (f_2^L)	1.00	<0.01	0.01	<0.01	0.15	0.14	0.16
Keypress interval in Type U (f_2^U)	1.00	<0.01	<0.01	<0.01	0.27	0.26	0.22
Duration of typing in Type U (f_4^U)	1.00	<0.01	<0.01	<0.01	0.28	0.28	0.23
Number of keystrokes in Type N (f_1^N)	<0.01	1.00	0.05	<0.01	0.18	0.16	0.18
Time to look toward keyboard in Type 2 transition window (f_2^{Tr})	<0.01	<0.01	<0.01	<0.01	0.48	0.44	0.37

Table 4-23 Kruskal Wallis H test and Dunn's test results of significant statistics-based gaze-typing features for non-touch typists (R: Reminiscent, L: Logical, C: Creative)

The Kruskal Wallis H test is also applied to the data from the non-touch typists in a similar fashion. In their cases, most of the significant features are extracted from the Type *F* thinking window and the Type *L* and *U* typing windows. Significant features from the thinking phase include: the total duration of staring and thinking (f_3^F) and fixation duration (f_5^F, f_5^R) of the Type *F* and *R* thinking windows. Significant features from the typing phase are the keypress intervals (f_2^L, f_2^U) in both Type *L* and *U* typing windows and the typing duration (f_1^L, f_4^U) in both Type *L* and *U* typing windows.

The results of the Dunn's test with Bonferroni correction on the significant features are shown in Table 4-23. Based on the results, we can find that composing an article in the creative genre has the most distinguishable typing behaviors, which are mainly shown in two aspects: keypress interval and the typing duration. For creative writing, both the keypress interval and the typing duration are the shortest in Type *L* and *U* typing windows. A similar phenomenon is also found by Wallot et al. [121] that keypress intervals are somewhat faster when the piece of writing is more complex.

We also observe that when a subject composes an article in the logical genre, she/he tends to have longer fixations when rereading the already-generated texts than in other genres. One of the possible explanations is that these longer fixations are indicative of more complex language processing. Henderson et al. [43] have observed that texts with a higher degree of logical complexity require greater attentional focus and more effort in language processing, as subjects attempt to connect the linkage between different parts of the text. This increases cognitive activity manifests in longer fixations.

The (f_2^{Tr}) feature of the Type 2 transition window shows that there is a significant difference in the writing behavior between every pair of genres for both

touch and non-touch typists. When subjects are composing in the creative genre, they exhibit the smallest normalized feature value of f_2^{Tr} , which means that a subject's gaze moves downward earliest when composing a creative article, compared to other genres. This phenomenon suggests that writing a creative article is a more cognitively complex task than the other two genres since a higher cognitive load induces people to move their gaze away from the target, scan more hastily and at a higher speed [49].

4.3.4.2 Understanding Sequence-based Gaze-typing Features

Sequence-based gaze-typing features are extracted from the behavior sequence of each session to capture the occurrence patterns of *indicative* patterns across the behavior-transition sequences. Indicative patterns are behavior subsequences, which differ across different writing genres. A weighting scheme was previously defined to determine whether a subsequence is an indicative pattern. Potential indicative patterns are all the possible subsequences with lengths ranging from [2,4] time window transitions. The reason we restrict the maximum length of the pattern to 4 is that based on the observation that most of the clauses are generated within four time windows.

In this section, we address two important questions: 1) whether the weighting scheme can help us to select patterns with discriminating power, and 2) what the selected indicative patterns are. Previously defined the means by which the indicativeness of a pattern can be quantified by the weighting (w), which can be computed through pf , rf , and td , where pf helps to avoid selecting a rare subsequence as a pattern, and rf and td determine the discriminating power of a pattern from different perspectives: rf measures the differences of the pattern's occurrences between the positive and negative groups and td measures the trend

difference between the positive and negative groups.

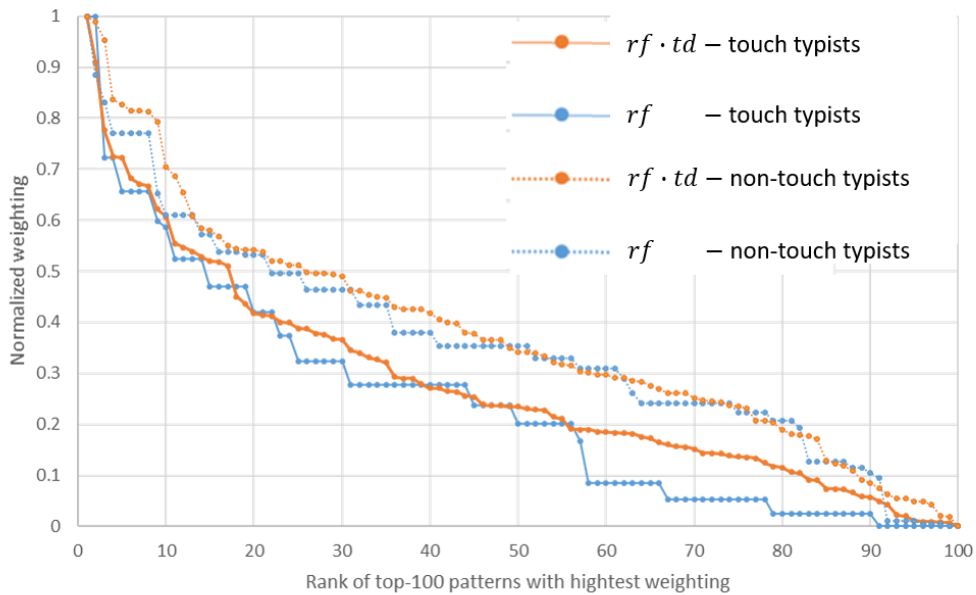


Figure 4-12 Top-100 normalized rf weights and Top-100 normalized $rf \cdot td$ weights for touch typists and non-touch typists

Figure 4-12 shows the top 100 largest rf weights and top 100 largest $rf \cdot td$ weightings in descending order for both touch and non-touch typists. For comparison, the weightings are normalized into $[0,1]$ range using min-max normalization. The value of the 1st largest weighting is mapped to 1 and the 100th largest weighting is mapped to 0. It is obvious that many patterns share the same rf weighting. Even when the value of the weight is at a high level, this phenomenon still occurs quite often. The reason for this is that our sequence classification problem gives us six different states, where the transition is strictly between one of O, R, F states and one of L, U, N states. This gives us a total of $3^3 = 27$ different kinds of transitions, which may not be complex enough to cover the different behaviors evidenced in our dataset. Figure 4-12 shows that many patterns, which appear to be quite dissimilar, do share the same rf weight. This suggests that the rf term may not be sufficient enough on its own to

quantify the discriminating power of the pattern. We, therefore, involve the td term for additional information.

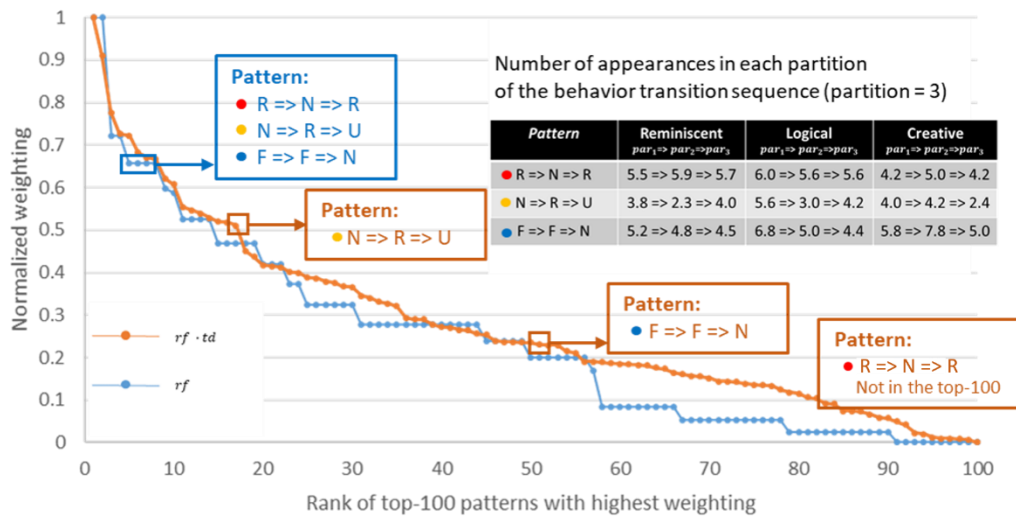


Figure 4-13 Examples of how the td term further distinguishes the discriminating power of patterns with the same td weight

As an example, Figure 4-13 compares three patterns: $R \Rightarrow N \Rightarrow R$, $N \Rightarrow R \Rightarrow U$ and $F \Rightarrow F \Rightarrow N$, based on their rf and $rf \cdot td$. Three patterns have the same rf weighting value of 0.65. However, the $rf \cdot td$ weightings of these 3 patterns are completely different:

- $N \Rightarrow R \Rightarrow U$ has the highest $rf \cdot td$ of 0.51. In reminiscent and logical writing, it decreases in frequency as the writer approaches the middle part of writing and then increases again as the writer approaches the conclusion of the writing period. However, in creative writing, this pattern slightly increases as the writer approaches the midpoint of the writing activity but decreases dramatically as the conclusion approaches.
- $F \Rightarrow F \Rightarrow N$ has a lower $rf \cdot td$ of 0.23. From the figure, even though there is some difference in the behavior of the pattern across different genres, the difference is less dramatic than $N \Rightarrow R \Rightarrow U$. For $R \Rightarrow N \Rightarrow R$, its $rf \cdot td$

does not make the top 100 list.

	Touch typist		Non-touch typist	
	$pf \cdot rf$	$pf \cdot rf \cdot td$	$pf \cdot rf$	$pf \cdot rf \cdot td$
1	$F \Rightarrow F \Rightarrow F$	$O \Rightarrow O \Rightarrow O$	$L \Rightarrow F \Rightarrow F$	$L \Rightarrow F \Rightarrow F$
2	$F \Rightarrow F \Rightarrow N$	$L \Rightarrow N \Rightarrow R$	$N \Rightarrow N \Rightarrow R$	$N \Rightarrow N \Rightarrow R$
3	$N \Rightarrow F \Rightarrow F$	$N \Rightarrow O \Rightarrow O$	$N \Rightarrow R \Rightarrow N$	$N \Rightarrow F \Rightarrow N$
4	$F \Rightarrow N \Rightarrow F$	$N \Rightarrow N \Rightarrow R$	$F \Rightarrow N \Rightarrow R$	$F \Rightarrow N \Rightarrow R$
5	$F \Rightarrow F \Rightarrow U$	$O \Rightarrow O \Rightarrow N$	$F \Rightarrow U \Rightarrow N$	$F \Rightarrow U \Rightarrow N$

Table 4-24 Top-5 selected patterns for both touch typists and non-touch typists

Table 4-24 lists the top 5 selected indicative patterns for both touch and non-touch typists based on the $pf \cdot rf$ and $pf \cdot rf \cdot td$ weightings. We note that in most of the selected patterns, at least two of three states are the same (e.g., $F \Rightarrow F \Rightarrow N$ has two F states). This suggests that the indicative patterns describe a period of time during which the subject's state is relatively stable. For example, the pattern $F \Rightarrow F \Rightarrow F$ describes the behavior in which a subject stares at the screen for a while (presumably thinking) before typing. The top-ranked indicative patterns differ depending on the weighting terms used. In particular, for touch typists, the top 5 indicative patterns selected based on the $pf \cdot rf$ weighting contain more F states, whereas the $pf \cdot rf \cdot td$ weighting more highly weighs the O states. Compared to touch typists, the top 5 indicative patterns selected based on the $pf \cdot rf$ weighting and the $pf \cdot rf \cdot td$ weighting for non-touch typists are more similar to each other. One possible reason is that non-touch typists are less efficient when typing, and the process of hunting for the correct key on the keyboard dominates the behaviors across the entire process of writing the article.

4.3.4.3 Evaluating the Performance of Writing Genre Detection

Our writing genre detection method is evaluated on the datasets constructed by us in Section 4.1. In real-life applications, a method should be able to work for a never-seen-before new user. Therefore, we employ a leave-one-subject-out cross-validation mechanism for evaluation. Specifically, a supervised learning model will be built based on the statistics-based gaze-typing features and the sequence-based features. The model will be trained on all but one of the subjects and evaluated on the remaining subject. The process will be iterated for N_s times, where N_s equals the total number of subjects. Since we build separate models for touch and non-touch typists, our approach's overall performance is calculated as the weighted average of the performance achieved over the touch and non-touch groups.

We first investigate the proper parameter values for our approach. The parameter (n_{par}) determines the number of partitions that a behavior sequence will be segmented into, which will be used to compute the td term. Physically, it also represents the number of writing stages, so it is not reasonable to have an overlarge or over small n_{par} . The parameter n_{select} denotes the number of indicative patterns that will be considered, sorted by weight. A too-small n_{select} may omit some useful patterns, but an over-large n_{select} will select some non-indicative patterns, which may dilute the impact of the truly indicative features.

In this experiment, we explore the impact of different value combinations of n_{par} and n_{select} on the performance. Linear support vector machine (SVM) models are built based on the concatenation of the statistic-based features with the sequence-based features, which are generated by different values of n_{par} and n_{select} . Figure 4-14 summarizes the results.

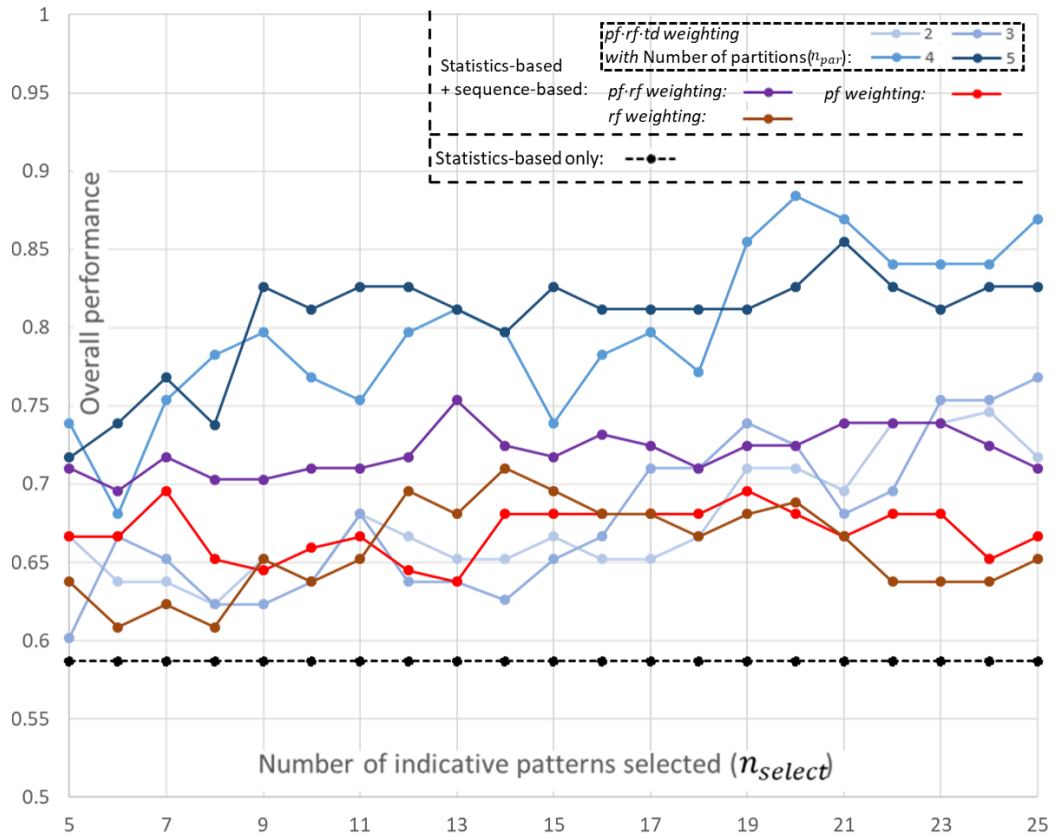


Figure 4-14 Overall performance trends of writing genre detection approach with a different number of partitions n_{par} and number of indicative patterns selected n_{select}

	predicted as	Reminiscent	Logical	Creative
Ground truth				
Reminiscent		18	1	2
Logical		0	24	1
Creative		1	2	22

Table 4-25 Confusion matrix of the article-category detection for touch typists

	predicted as	Reminiscent	Logical	Creative
Ground truth				
Reminiscent		22	1	2
Logical		4	16	1
Creative		1	0	20

Table 4-26 Confusion matrix of the article-category detection for non-touch typists

Compared with the overall baseline of 36.2%, which is achieved by predicting every instance as the majority class, the best performance of our approach ($n_{par} = 4, n_{select} = 20$) can achieve 88.4% accuracy, which is quite promising. Table 4-25 and Table 4-26 show the detailed confusion matrixes.

We notice from the figure that when $n_{par} = 4$ or 5 , our approach always yields the best performance. It makes sense since usually most articles can be divided into three parts: introduction, body, and conclusion, and the body part has around 2-3 times the length as the length of the introduction and the conclusion parts. We note that when $n_{par} = 4$ or 5 , the max performance is achieved when n_{select} is around 20, which also meets our intuition that selecting too many patterns will worsen the overall performance since non-indicative patterns may be included.

Figure 4-14 also presents the performance trend of linear SVM models built on the statistics-based features and sequence-based by using the $pf \cdot rf$ weighting scheme with different n_{select} values, pf weighting scheme, rf weighting scheme and the performance of only using statistics-based features. It is clear that with reasonable values of n_{par} , the overall performance of the $pf \cdot rf \cdot td$ weighting scheme is always better than the others.

We also evaluate the performance of our approach without differentiating between touch typists and non-touch typists. We construct a new dataset by combining data from all subjects and training linear SVM models on the dataset with different values of n_{par} and n_{select} . Based on the previous results, potential values of n_{par} are 4 and 5. Figure 4-15 shows the performance trends. The best performance is around 77%, which is attained when $n_{par} = 5$ and $n_{select} = 15$. According to the figure, we acknowledge that the performance of the $pf \cdot rf \cdot td$ weighting scheme is better than $pf \cdot rf$, which is consistent with the results of

training separate models for touch typists and non-touch typists. It can also be seen that the combined model performs worse than training separate models for different levels of typing ability. A possible reason is that gaze-typing behaviors differ so much between touch typists and non-touch typists, and these inconsistent behaviors may confuse the model.

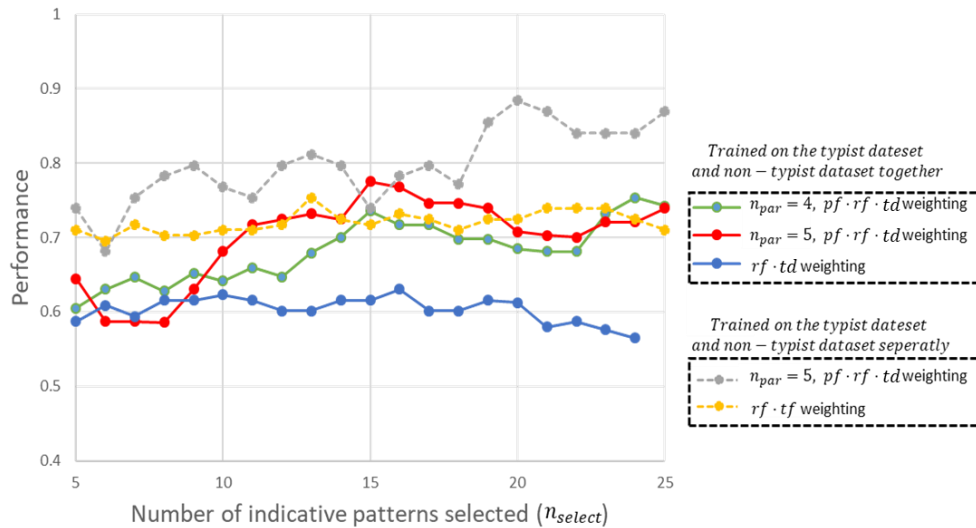


Figure 4-15 Performance trends of writing genre detection approach trained on the touch typists dataset and non-touch typist dataset together

Finally, we evaluate the performance of our approach across different age groups to ascertain the effect of the age factor. As shown in Table 4-1, the college student group are all touch typists and all but one subject in the child group are non-touch typists. As are around 33% of subjects in the elderly-age group, we therefore further divide the elderly-age group into the touch typist elderly-age group and non-touch typist elderly-age group. For each specific age group, we then construct a linear SVM model on the concatenation of statistics-based and sequence-based features with $pf \cdot rf \cdot td$ weighting scheme, where $n_{par} = 4$ and $n_{select} = 20$, which achieves the best performance in the previous evaluation.

Table 4-27 presents the results of the evaluation by age group. It can be seen

that the performances for the child group and college student group are close to the best performance achieved by differentiating the touch typists and the non-touch typists (Figure 4-14). However, for the elders, the performance drops more significantly compared to the two other groups, approaching the performance we achieved in Figure 4-15 when there was no differentiation between the touch typists and the non-touch typists. However, when the elders are broken down into touch typists and non-typists, the performance improves significantly, even outperforming the best performance previously achieved. These observations suggest that (1) the typing skill has a bigger effect on writing genre detection than the age factor, and (2) the age factor may provide additional information that can contribute additionally to the performance of writing genre detection after the dominant factor (typing skill) is accounted for.

Article-category detection for					
	Children	College students	Elders	Touch typists in elderly-age group	Non-touch typists in elderly-age group
CCR	87.0%	83.3%	77.8%	91.2%	88.9%

Table 4-27 Article-category detection for different age groups

4.4 Summary

In this chapter, we focus on exploring the cognitive process of writing based on gaze-typing behaviors. In this study, we focus on investigating how the age-factors and writing genres affect the cognitive process of writing by analyzing gaze and typing behaviors when subjects are generating their own texts. Since there is no published dataset available, which satisfies requirements, we construct our datasets by collecting data from 46 subjects (18 in child age group, 10 in college

students group, and 18 in the elder age group), 138 articles (46 in reminiscent, 46 in logical, and 46 in creative articles).

Based on the result of the age-group detection experiment, our statistics-based gaze-typing features can successfully determine the age group of the subject during the process of writing with a high accuracy, which is 83.3%. It indicates that the age-factors do affect the writing process. According to the result of the feature selection for age-group detection, we know that the age-factors' effect mainly reflects in the typing phase and transition phase (from thinking to typing). It is because that the typing skills and the capacities of the working memory are different among these three age groups.

For the effect of the writing genres, we successfully utilize statistics-based gaze-typing features and sequence-based features to determine an article's genre based on gaze and typing behaviors during writing. It indicates that the writing genres influence the writing cognitive process and can be inferred by gaze and typing behaviors. We find that when a subject is composing a complex article, which involves more idea-generating phases and text-organizing phases, he/she will reread already-generated texts more frequently. The purpose of rereading already-generated texts could be providing hints of what to write next or helping organize the current generating sentence. They can be differentiated by the length of the rereading texts since organizing the sentence needs to reread longer length to ensure the correctness both logically and semantically. Another important finding is that unlike the copy-type tasks, when the subject generates their own texts, keypress intervals are not consistent, and pauses exist throughout the task. This is most likely because the process of composition requires subjects to convert their ideas into text in addition to inputting the text via the keyboard. The observation supports this hypothesis that longer pauses are observed in the logical

and creative writing, requiring the subject to imagine and visualize a scenario and express it coherently in textual language with logical and semantic correctness. These requirements presumably require more cognitive effort than reminiscent writing, in which subjects are simply asked to recall an event. We also notice that non-touch typists always shift their gaze away from the screen and towards the keyboard before they start typing earlier for logical and creative writing. This is consistent with previous work [49] on a different domain (mathematics calculation), which shows that when a subject is in a high cognitive load state, they are more likely to move their gaze away from the target earlier at a higher speed.

The results in Figure 4-15 show that the best performance of writing genre detection is achieved by combining statistics-based and sequence-based gaze-typing features. It implies that sequential gaze-typing behaviors can model the writing process. The transition between different kinds of behaviors also appears to capture the information of writing's cognitive process, especially with certain behaviors that frequently appear during a particular process of the activity.

We also discover that the same behavior may have different causes. One example is that the non-touch typists need to look at the keyboard while typing, their eye gaze movements exhibit many saccades with greater variation along the y-axis, and the eye gaze cannot be captured for large amounts of time. However, for touch typists, saccades with more significant variation along the y-axis are generally related to rereading the previously generated texts, and periods of time when the subject's gaze is off-screen are often associated with deep thought and planning what to write next. These behaviors, though superficially the same, have very different causes, which argues for the need to train separate models based on the typing proficiency of the subject.

In conclusion, our results indicate that people from different age groups

compose articles in different genres, the writing cognitive processes are different, which can be inferred by gaze-typing behaviors. The effects of age-factors are mainly reflected in the typing behaviors, and the effects of article genres are mostly on the rereading behavior, pauses during typing, and transitions between different behaviors. In a nutshell, our results are promising and provide a more in-depth understanding of human behavior in writing.

5 Inferring Users' Cognitive Process of Summarization Based on Gaze and Typing Behaviors

Chapter 4 introduces the way of investigating the writing cognitive process by analyzing the gaze-typing behaviors when users are generating their own texts. We design the gaze-typing features extracted from different writing processes such as planning the writing ideas, rereading, converting writing ideas into texts, typing texts into computers, and reviewing already-generated texts to model behaviors. Based on our gaze-typing features, we show that the age-factors and different writing genres affect the writing cognitive process, and the differences in the writing cognitive process can be captured successfully by our gaze-typing features. In this chapter, we take a step further by investigating the cognitive process of summarizing writing based on the gaze-typing features extracted from multimodalities.

Compared with writing (generating own texts on a computer), summarizing a document is a complex text that needs a person to multitask between reading comprehension and writing based on understanding. Unlike the reading and writing behaviors that have been investigated substantially, the exploration of summarization is at the very early stage. Known that the cognitive load during reading and writing are dependent upon the level of comprehension or difficulty of the article, it suggests that it should be possible that the different difficulty levels of summarization task may also affect the cognitive process of the person and can be analyzed based on the gaze-typing features, similar as what we do to analyze the writing cognitive process. Another important goal of this study is to examine whether multimodal features can improve the supervised learning performance and explore the possible reasons for our multimodal features' superior

performance.

In this study, we first construct our summarizing task dataset by recruiting 20 subjects to accomplish three different summarizing writings in different difficulty levels. The details of experiment settings and designs are introduced in Section 5.1. In Section 5.2, we are going to introduce the procedures of extracting multimodal features from the gaze, keyboard, time modalities to capture the differences of cognitive processes of summarizing texts in different difficulty levels, followed by the evaluation of our multimodal features in the aspect of the performance of recognizing different difficulty levels in Section 5.3. In Section 5.4, we will compare the performances between the multimodal features with the features extracted from every single modality, followed by the summary in Section 5.5.

5.1 Constructing Summarizing Task Datasets

5.1.1 Experiment Settings

The purpose of this research is to examine the cognitive process of summary writing, in particular, to explore the variations in cognitive processes when summary writing on the basis of gaze and typing behaviors at different difficulty levels. Therefore, as our research focuses on writing in the Chinese language, in our experiment, 20 experimental subjects (*Ages 25 - 50, Mean = 35.4, STD = 7.5*) were recruited, and all of them were native Chinese speakers with at least a high school education. A pre-experiment survey was conducted prior to the experiment and showed that all the recruited subjects were familiar with typing on the keyboard using the Chinese Pinyin input method, which mapped Latin phonetic symbols to Chinese Characters. In addition, every subject had reading and writing habits and spent at least half an hour reading every week in the

previous year and wrote at least once a week.

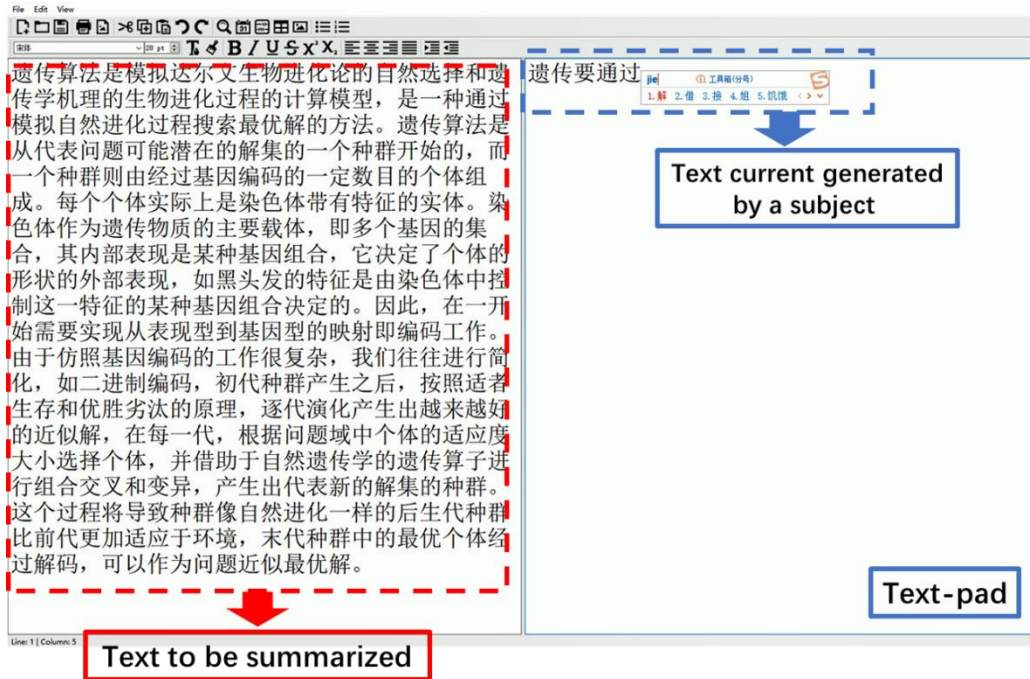


Figure 5-1 Experimental interface for summarizing task

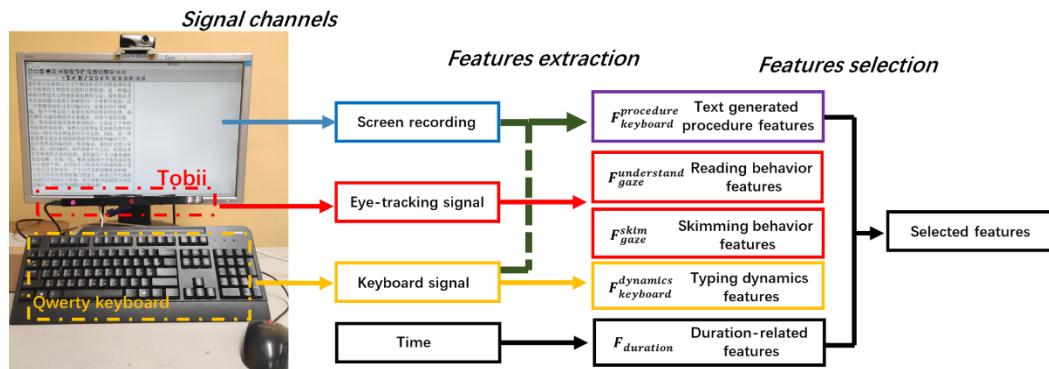


Figure 5-2 Experimental environment and the overall framework of the multimodal approach

As seen on the left side of Figure 5-2, the experiment was conducted in a standard office room. The configuration consisted of a 22" LCD display at 1680×1050 resolution, and a Tobii EyeX eye tracker was installed to the bottom of the display, a full-size QWERTY keyboard, and a regular optical mouse. A subject was positioned about 60 cm away from the display in the experiment,

and the height of the display and chair had been changed to accommodate the preference of the subject.

The screen was equally divided into two areas of the same size during the experiment. On the left side of the screen, the piece of article that the subjects were expected to be summarized was shown, and the area is called the *reading area*. The *writing area* was on the right side of the screen, where the subjects wrote the summaries. The purpose of this side-by-side configuration is to emulate the split-view mode, which is frequently used in everyday life in many computer setups. The experimental interface is illustrated in Figure 5-1.

Articles to be summarized were in Chinese, and the font was set to 28 *pt* DengXian. Two data collection systems were running in the background during the experiment: one of them captured the eye gaze positions at 60 *Hz* and tracked the keyboard presses at 100 *Hz*, and the other one captured the screen video at 10 *Hz* simultaneously.

5.1.2 Experiment Design

Three articles in around 400 words length at various difficulty levels were to be summarized by the participants of our experiment. Two of them were selected from local newspaper reports, which were acquainted by all subjects. A general introduction of a machine learning algorithm, which none of the participants were familiar with, was selected as the third article. Below are the specifics of each piece of the article:

- **Easy:** A report on the bike-sharing economy in the local newspaper. The author first illustrates the present growth of the bike-sharing economy in the article and then discusses the price change in the future.
- **Medium:** An article from the local newspaper about the local price increase of

agricultural goods, which poses various opposing opinions with claims.

- **Hard:** An overview article on genetic algorithms, describing its fundamental principles, and how DNA selection is related.

Compared to the easy-difficulty article, the medium-difficulty article contained more argument shifts, and the topic of the hard-difficulty article was unrelated to the subjects. Both variables made the articles harder to comprehend and summarize. Each subject was expected to complete a summary of half the original articles' length for each article, which was around 150-200 words. There was no limit on the time spent on writing. It commonly took about 30 minutes for each subject to complete each summary.

Two warm-up pieces of summary tasks were undertaken by the participants before the formal experiment, which was intended to make the subjects familiar with the experimental interface and input devices. With the help of the warm-up tasks, we are able to normalize behavior features across subjects that will be completely introduced in Section 5.3.1. Articles with different difficulty levels were presented to the subjects in random order during the experiment. There was a 20-minute break after each summary writing to prevent subjects from being exhausted. The eye tracker was recalibrated after each break. We also required each subject to rank the difficulty of each summarization task from Easy to Hard to validate each summary task's relative difficulty as perceived by the subjects was consistent with our assumptions.

Finally, our summarizing task dataset consists of 55 instances, including 19 instances labeled as *Easy*, 19 instances labeled as *Medium*, and 17 instances labeled as *Hard* after excluding one subject, who did not complete the experiment, and 4 additional instances due to the equipment fault. Based on the self-reports, the difficulty of the task, as perceived by each subject, is consistent

with the assumption made in the experiment.

5.2 Extracting Multimodal Features from Input Signals

In this study, we are going to investigate how gaze and typing behaviors are affected based on the cognitive changes when summarizing articles of various degrees of difficulty. Multimodal features that incorporate eye gaze movements, keypresses, and text generation are studied. In this section, we will mainly introduce the feature extraction and selection procedures for the features. Figure 5-2 presents the overall framework of our method.

5.2.1 Pre-processing of Input Signals

Our experiment setup provides us three channels for detecting user behaviors, which include the eye-tracker, the keyboard, and the screen recording. The format of the eye-tracker outputs is a series of three-dimension tuples in the form of $\langle x, y, t \rangle$, which indicates that at time t , the subject is fixating on the screen position at (x, y) . For each period that the eye-tracker fails to capture any subject's eye gaze on the screen, if its duration is more than 400 *ms*, then it will be considered that the subject does not look at the screen, and the signals during that period will be removed. Else it will be considered as a blink [12], and the linear interpolation is utilized to approximate the eye gaze locations. To eliminate the impose noise from the eye-tracking signal, a two-phase heuristic filter [107] is then exploited. Fixations are intervals of time during which a subject holds his/her gaze at the same positions to understand the visual information, which can be detected by using the dispersion threshold identification (I-DT) algorithm [101] with the dispersion of 35 *px* and minimum stay time of 170 *ms*.

Apply these two algorithms returns us a series of fixations in the form of $\langle t_{fix}, dur_{fix}, x_{fix}, y_{fix} \rangle$, where t_{fix} is the time when a subject starts to fixate on

the location (x_{fix}, y_{fix}) and dur_{fix} , stands for the duration of the fixation. A saccade is defined as a spontaneous and continuous gaze movement from one fixation position to another. Thus, a saccade appears between every two adjacent fixations. Based on the definition, a saccade can be represented as $\langle t_{sac}, dur_{sac}, dist_{sac}, x_{sac_1}, y_{sac_1}, x_{sac_2}, y_{sac_2} \rangle$, where t_{sac} is the moment that the gaze starts to move from the on-screen location (x_{sac_1}, y_{sac_1}) to (x_{sac_2}, y_{sac_2}) and dur_{sac} and $dist_{sac}$ stand for the duration and the Euclidean distance of the saccade, respectively. Then the eye gaze movement can be measured by the scanpath [87] – a sequence of fixations and saccades represented as $Sp = \{fix_0, sac_0, fix_1, sac_1, \dots, fix_n, sac_n\}$.

The format of the keyboard signal is in the form of $\langle t_{key}, key_name \rangle$, which encodes that the key_name key is stroked at the time t_{key} . From the screen recording, the caret position at each timestamp $\langle t_{cur}, x_{cur}, y_{cur} \rangle$ can be detected, and the position of each text deletion, text insertion, and text appending can be achieved through the further process by combining the keyboard events and text caret positions together.

Mouse movement signal is also collected during the experiment in the form of $\langle t_{mouse}, x_{mouse}, y_{mouse} \rangle$, which encodes that mouse cursor stays at (x_{mouse}, y_{mouse}) at t_{mouse} . However, through the data visualization, we find that mouse is seldom used by subjects during the summarizing task. The only cases that a subject uses the mouse is to occasionally select a target word/phrase from a candidate box or relocate the text caret. Because the amount of meaningful mouse movement data is so rare, the mouse movement signal is not used in this study.

5.2.2 Extracting Features from Eye-tracking Signal

Previous studies [30, 126, 127] indicate that summary writing requires many

processes, including reading the text to be summarized, understanding it by grasping key information, translating the summarizing/writing ideas into sentences, and typing the sentences on the keyboard. However, there is considerable variance in a subject's behaviors, especially the behaviors of eye gaze when a subject is in the different phases. Therefore, the entire summary process is segmented into many phases so that the behaviors of the subject within each phase are relatively constant.

We first segment the scanpath into two types: Sp_{read} and Sp_{write} , where sub-scanpaths in Sp_{read} type occur within the reading area and sub-scanpaths in Sp_{write} are within the writing area. In practice, subjects do not keep focusing on the screen over the whole duration of the experiment, as their attention can be diverted by external influences. We, thus, ignore sub-scanpaths that are shorter than 500 *ms* in length.

In this study, all the summarizations are written in Chinese by using the Pinyin input method. The method of typing in Chinese in Pinyin, as introduced in Section 4.1, is very different from that of typing in English. Since it is an indirect text generation approach for the Pinyin input method and a subject needs to input the phonetic equivalent in alphabetic symbols to produce a Chinese word/phrase, which will then be translated to the actual text. Since the phonetic mapping is always one-to-many, in a pop-up word selection window (Figure 5-3), the system presents all possible candidates, which are changed dynamically as the keyboard is pressed. A subject then chooses his/her intended text from the presented candidates.



Figure 5-3 An example of the pop-up candidates words selection window

Known about how the texts are generated helps us to categorize Sp_{write} into two types: Sp_{type} and Sp_{reread} . Sp_{type} is a type of sub-scanpath in which the gaze of the subject stays within the pop-up word selection window region and Sp_{reread} is a kind of sub-scanpath outside the words selection window, suggesting that the subject rereads his/her previously generated text. Rereading is an important writing behavior during which the subject reviews generated texts for planning new ideas for the subsequent generation [30].

Feature	Meaning	Formulation
f_1	Sub-scanpath duration	Total duration of fixations and saccades inside the sub-scanpath
f_2	Number of fixations	Number of fixations inside the sub-scanpath
$f_{3,4,5}$	Fixation duration	Mean(f_3), standard deviation(f_4) and max(f_5) of the fixation duration
$f_{6,7,8}$	Saccade distance	Mean(f_6), standard deviation(f_7) and max(f_8) of the saccade distance
f_9	Number of switching-line saccades	Number of switching-line saccades inside the sub-scanpath

Table 5-1 Features of clustering Sp_{read} sub-scanpaths

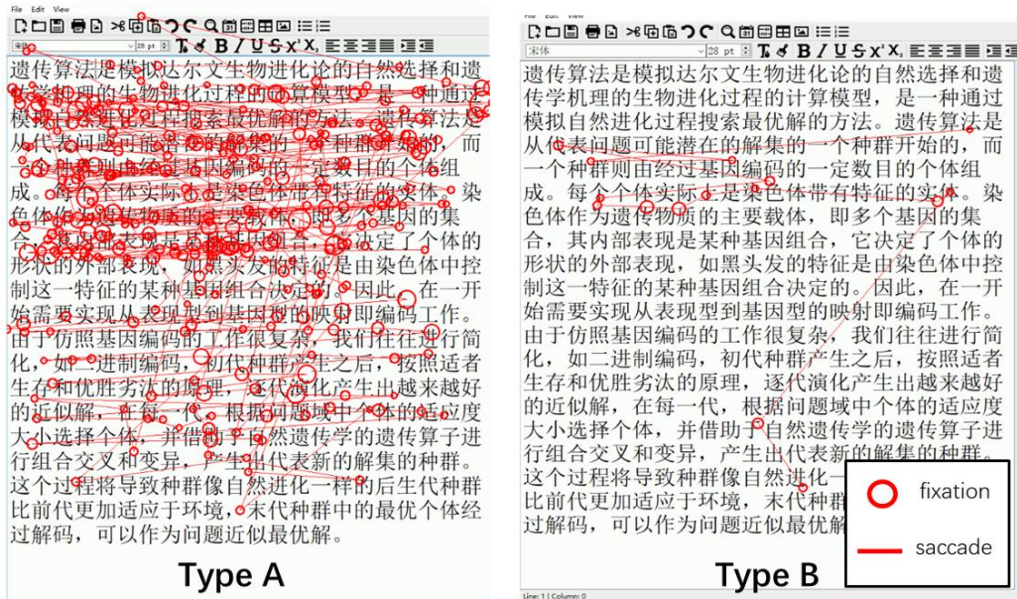


Figure 5-4 Examples of two types of Sp_{read} sub-scanpath

Based on the previous research, Sp_{read} behaviors suggest that when writing a summary, a subject is reading the article to grasp the main ideas in order to generate the writing ideas or reference the detailed material in the given article during writing. Compared to Sp_{write} , sub-scanpaths in Sp_{read} are much more diverse and complicated. Therefore, in order to group and categorize sub-scanpaths of the same kind, we apply the k -means algorithm [75] to cluster all the Sp_{read} sub-scanpaths based on the features in Table 5-1. The silhouettes measurement [99] is utilized to determine the optimal number of clusters, which is 2 based on the result.

Category	Sub-category	Usage
Within the reading area (Sp_{read})	Reading the text to be summarized ($Sp_{understand}$)	Understanding the basic idea of the text by a careful reading
	Skimming the text to be summarized (Sp_{skim})	Selecting and extracting relevant information for the current writing goal
Within the writing area (Sp_{write})	Within the pop-up words selection window (Sp_{type})	Selecting the final text from the phonetic equivalent
	Rereading previously generated text (Sp_{reread})	Reviewing the generated text and inspiring new writing ideas

Table 5-2 Summary of sub-scanpath in different categories

An example of each cluster is illustrated in Figure 5-4, where red circles stand for fixations and the length of each circle radius is proportional to the fixation duration. Red line segments represent saccades. It is evident that the sub-scanpaths in different clusters are distinct from each other. Type A Sp_{read} sub-scanpaths have a longer duration, have more fixations than Type B sub-scanpaths. Most of the saccades in Type A Sp_{read} sub-scanpaths are shorter in the distance than the saccades in Type B Sp_{read} sub-scanpaths. For Type A Sp_{read} sub-scanpaths, we also find that most of the saccades are horizontal, aligning with the direction of the text-lines, and the saccade length is around 1-2 times the width of a Chinese word. This indicates that Type A Sp_{read} sub-scanpaths are generated when a subject is reading through the texts.

On the other hand, Type B Sp_{read} sub-scanpaths contain relatively longer saccades, which always appear at the beginning and the end of the sub-scanpaths traversing vertically across multiple text-lines. Prior work [15] illustrates that Type B sub-scanpaths suggest skimming behavior. That is a subject focuses on selecting and extracting relevant information to his/her current writing goal from the texts

without paying too much attention to the irrelevant material [82]. We, thus, represent Type B sub-scanpaths as Sp_{skim} . Table 5-2 presents an overview of all categories of Sp_{read} with their descriptions.

Feature	Meaning	Formulation
$g_1 - g_4$	FSs distance, speed	Mean and standard deviation of all FSs distances and speeds
$g_5 - g_8$	BSs distance, speed	Mean and standard deviation of all BSs distances and speeds
$g_9 - g_{12}$	Distance and speed of all saccades	Mean and standard deviation of all saccades distances and speeds
g_{13}, g_{14}	Fixations after FS	Mean and standard deviation of all fixation durations after FS
g_{15}, g_{16}	Fixations after BS	Mean and standard deviation of all fixation durations after BS
g_{17}, g_{18}	Duration of all fixations	Mean and standard deviation of all fixation durations

* **FS stands for forward saccade; BS stands for backward saccade**

Table 5-3 $F_{gaze}^{understand}$: Features extracted from $Sp_{understand}$

For each category i of sub-scanpaths, we extract specific eye-gaze features F_{gaze}^i from all members of instances. For a sub-scanpath in $Sp_{understand}$ type, we extract the feature vector $F_{gaze}^{understand}$. According to the primary cognitive activity that is demonstrated by this sub-scanpath, we determine two kinds of saccades: *Forward Saccades (FS)* and *Backward Saccades (BS)*. FS is identified as a horizontal saccade following the direction of the text, and BS, also recognized as the regressive saccade, is opposite to the direction of FS. Previous studies have found that as the level of text increases in complexity, the length of FSs, and the duration of the fixations after the FSs also increase [94]. Table 5-3 shows the details of features in $F_{gaze}^{understand}$ with their formulations.

As illustrated in Figure 5-4 (Type B), a sub-scanpath in Sp_{skim} consists of

```

function GetSaccadeType(Sac)                                % Sac: a saccade in  $Sp_{skim}$ 
{
    if length of saccade  $\geq$  200 px
        type  $\leftarrow$  "Searching (SS)"
    else                                     % anti-clockwise, 0° is horizontal to the right
        if direction of Sac is in  $(-30^\circ, 30^\circ)$  or  $(150^\circ, 210^\circ)$ 
            type  $\leftarrow$  "Reading (RS)"
            if direction of Sac is in  $(-30^\circ, 30^\circ)$ 
                sub_type  $\leftarrow$  "ForwardReading(FRS)"
            else
                sub_type  $\leftarrow$  "BackwardReading(BRS)"
        else
            type  $\leftarrow$  "Undefined"
    return type, sub_type
}

```

Algorithm 5-1 Determine type saccade in Sp_{skim}

the *searching* saccades and *reading* saccades, where searching saccades are identified by long-distance saccades spanning across several text-lines, but reading

saccades are short-distance saccades along the text-line direction. When a subject refers to the text, searching saccades are used to locate the part of the text, and reading saccades are used to determine that this part of the text is useful. A rule-based approach to determine different kinds of saccades is presented in Algorithm 5-1.

Feature	Meaning	Formulation
$g_{19} - g_{34}$	SSs, RSs, FRs, BRs distance, speed	Mean, standard deviation of each kind of saccade distance and speed
$g_{35} - g_{38}$	Number of SSs, RSs, FRs, BRs	Number of each kind of saccades
$g_{39} - g_{42}$	All Saccades distance, speed	Mean, standard deviation of all saccades distances and speeds
$g_{43} - g_{50}$	Fixations after SSs, RSs, FRs, BRs	Mean, standard deviation of each kind of fixation duration
g_{51}, g_{52}	All fixations	Mean, standard deviation of all fixations durations
$g_{53} - g_{56}$	Cumulative distance of SSs along one direction	Cumulative distance of SSs along with <i>x-positive</i> , <i>x-negative</i> , <i>y-positive</i> , and <i>y-negative</i> directions
$g_{57} - g_{64}$	Distance of SSs along one direction	Mean and standard deviation of distances of SSs along with <i>x-positive</i> , <i>x-negative</i> , <i>y-positive</i> , and <i>y-negative</i> directions

***SS stands for searching saccade; RS stands for reading saccade; FR and BR stands for forward and backward reading saccade**

Table 5-4 F_{gaze}^{skim} : Features extracted from Sp_{skim}

Features F_{gaze}^{skim} extracted from Sp_{skim} sub-scanpaths are intended to describe various kinds of fixations and saccades. Features F_{gaze}^{skim} are listed in Table 5-4, where the *x-positive* direction is horizontal to the right, and the *y-positive* direction is vertical to the up.

According to the definition, Sp_{write} can be further classified into Sp_{type}

and Sp_{reread} . Because that Sp_{type} is highly correlated to the typing skill rather than the cognitive process of summarization, we only extract features from Sp_{reread} . Since Sp_{reread} is generated when rereading the previously generated text, our extracted features are the same as that of features extracted from $Sp_{understand}$ and indicated by F_{gaze}^{reread} , denoted from g_{65} to g_{82} . In total, 82 features are extracted from the eye-tracking signal to modal eye-gaze behaviors through different summary phases.

5.2.3 Extracting Features from Keyboard Signal

Prior study has illustrated that keystroke dynamics provide useful information for the cognitive state inference [115]. Therefore, we extract features related to keystroke dynamics from the keyboard signal for analysis.

First, we define typing phases, identified as periods during that the interval between every two keypresses is no more than 1.2 seconds [21]. During these time periods, we extract keyboard dynamics features $F_{keyboard}^{dynamic}$ from the keyboard signal, as presented in Table 5-5.

Feature	Meaning	Formulation
k_1, k_2	Number of keypresses	Mean and standard deviation of number of keypresses in each typing phase
k_3, k_4	Inter-key intervals	Mean and standard deviation of all inter-key intervals
k_5, k_6	Number of deletions	Mean and standard deviation of number of times the backspace key is pressed in each typing phase

Table 5-5 $F_{keyboard}^{dynamic}$: Features extracted from keyboard signal

The screen recording is also collected during the experiment to give us another channel of information. The information includes the text caret position at

each timestamp and the timestamps that the pop-up candidates box appears on the screen. Combining keyboard events, caret position, and the appearance of the pop-up words selection window allows us to reconstruct *insert*, *append*, and *delete* text generation operations, as well as the location of the operation and the number of words generated or deleted by that operation. We believe that the text generation procedure can provide indicative information. For example, if a subject performs on a difficult summary, it is possible that there would be relatively more frequent *insert* and *delete* operations. Table 5-6 presents the text generating features ($F_{keyboard}^{procedure}$) with their meanings and formulations. Totally, we extract 15 features from the keyboard signal to model the typing dynamics and the text generation procedure.

Feature	Meaning	Formulation
$k_7 - k_9$	Operation frequency	Number of insert, append and delete operations divided by task duration
k_{10}, k_{11}	Number of words generated	Mean and standard deviation of number of words generated in each typing phase
k_{12}, k_{13}	Number of words deleted	Mean and standard deviation of number of words deleted in each typing phase
k_{14}	Number of words generated since rereading action	Mean of number of words generated between every two rereading behaviors
k_{15}	Number of words generated since the last reading of the text to be summarized	Mean of words generated between every two understanding behaviors

Table 5-6 $F_{keyboard}^{procedure}$: Features extracted from text generation procedure

5.2.4 Extracting Duration-related Features

Finally, we extract duration-related features from the eye-tracking signal and

keyboard signal, which are designed to measure the time spent on each phase. We believe that the time can reflect the effort that a subject spent on each phase compared to others.

After processing the eye-tracking signals and keyboard signals, we identify the different behaviors, including understanding or skimming the original article, rereading the already-generated texts, and writing by typing on the keyboard. As shown in Table 5-7, we establish duration-related features ($F_{duration}$), which measure the total cumulative time spent on each behavior.

Feature	Meaning	Formulation
d_1	Duration of completing the summary task (dur_{task})	$\frac{dur_{task}}{(n_{sum} + n_{gen})}$
d_2, d_3	Duration of understanding phases ($dur_{understand}$)	$\frac{dur_{understand}}{n_{sum}}$, $\frac{dur_{understand}}{dur_{task}}$
d_4, d_5	Duration of typing phases (dur_{type})	$\frac{dur_{type}}{n_{gen}}$, $\frac{dur_{type}}{dur_{task}}$
d_6, d_7	Duration of rereading phases (dur_{reread})	$\frac{dur_{reread}}{n_{gen}}$, $\frac{dur_{reread}}{dur_{task}}$

*** n_{sum} stands for number of words in the text to be summarized; n_{gen} stands for number of words are generated**

Table 5-7 $F_{duration}$: Duration-related features

5.3 Evaluation of Multimodal Features

In this section, our multimodal features' effectiveness is evaluated to discriminate different summary task difficulty levels. First, the indicative features are selected. Then based on the selected indicative features, a machine learning model is developed, and the performance of the model is evaluated by using the leave-one-subject-out cross-validation in the correct classification rate (CCR).

5.3.1 Feature Normalization and Feature Selection

As stated in the last section, we totally extract 104 potential multimodal features from different input channels, including 82 features extracted from the eye-tracking signal, 15 features extracted from the keyboard signal, and 7 features are associated with time duration. All these potential features can be categorized into six groups to model the gaze and typing behaviors in different summary writing activity phases.

It is recognized that because of the factors of age [4], proficiency in typing [89], etc., features, which are used to model gaze and typing behaviors may vary significantly between different subjects. Furthermore, the influence of these factors on gaze and typing behaviors is likely even more severe than the impact of the summary writing's cognitive process. Therefore, to eliminate the variations among subjects, we need to normalize the extracted features. We first extract 104 potential multimodal features from the two pieces of summary writings written in the warm-up sessions, and then the mean value for each feature is determined to be the *baseline* for each of that particular feature for each subject. The features extracted from the formal experimental sessions of generating the three summarizations in different difficulty levels are then normalized by the *baseline* corresponding to that subject, respectively:

$$f' = \frac{f - \textit{baseline}}{\textit{baseline}} \quad 5-1$$

Where f' is the normalized feature, and f is its original feature value.

Our feature extracting process totally gives us 104 potential multimodal features. It is possible that some features are duplicated to each other since they contain similar information, and it is also possible that some features are irrelevant

to the goal. To eliminate the influence of the duplicated and irrelevant information, the feature selection process is applied to select a good subset of features for our classifier.

In this study, the wrapper method with 10-fold cross-validation is applied to select an effective subset of features. A prior study has stated that, compared with other filter methods, the wrapper method is always able to produce a decent performance [105].

The support vector machine (SVM) with the RBF kernel is utilized as the classifier in this study. The final subset of features selected by the wrapper is present in Table 5-8, where group means the signal channel that the feature extracted from.

Group	Feature
Skimming	Mean speed of all searching saccades
	Mean distance of all searching saccades along with the <i>x-positive</i> direction
	Mean distance of all forward reading saccades
Text generation	Mean number of words generated in each typing phase
Duration-related	Duration of all understanding phases divided by the number of words in the text to be summarized
	Duration of all typing phases divided by task duration

Table 5-8 Indicative features selected from potential features

5.3.2 Performance of Difficulty Level Detection

After feature normalization and feature selection, we adopt SVM with RBF kernel as the classifier, which is constructed based on the selected features. We use the leave-one-subject-out cross-validation to evaluate the CCR performance of our model. Specifically, we split our dataset into $N_s - 1$ and 1 subject and the classifier is trained on the data collected from $N_s - 1$ subjects and evaluated on

the data collected from the left-out subject. The model's overall performance is the average CCR across N_s time, where $N_s = 20$ in this study.

Class \ Performance	Precision	Recall	F-measure
Easy	0.85	0.90	0.87
Medium	0.94	0.90	0.92
Hard	0.94	0.94	0.94

Table 5-9 Classification performance for difficulty level detection by using multimodal model

Ground truth \ Predicted as	Easy	Medium	Hard
Easy	17	1	1
Medium	2	17	0
Hard	1	0	16

Table 5-10 Confusion matrix for difficulty level detection by using multimodal model

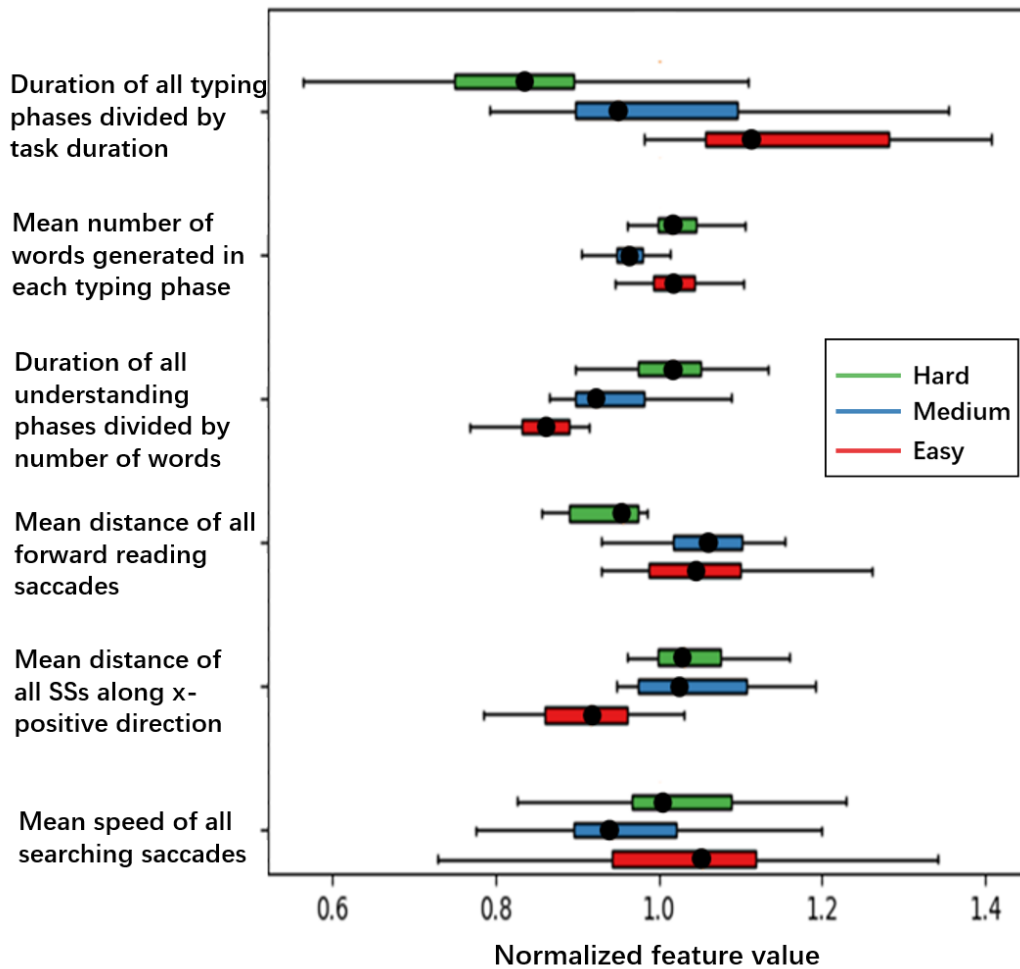


Figure 5-5 Box plot of selected features

The result shows that our model (SVM with RBF kernel), constructed based on the selected multimodal features listed in Table 5-8 yields an overall CCR performance of 91.0%, which is around 55% improvement of the baseline performance by classifying every instance into the majority class (easy class). The detailed precision, recall, F-measure for each class is shown in Table 5-9, and Table 5-10 presents the confusion matrix. Based on the results, it can be concluded that our multimodal approach can successfully discriminate the level of difficulty of the text that the subject is summarizing.

In order to investigate how these selected features can help us discriminate

the different difficulty levels of the text that the subject is summarizing, we plot the distribution of each selected indicative feature for different difficulty levels, which is shown in Figure 5-5. Inside the figure, each bar stands for a normalized value distribution and the black dot inside the bar marks the median. The range from the first quartile to the third quartile is covered by the box area, and the ranges from the minimum to the first quartile and from the third quartile to the maximum are covered by the left and right whiskers, respectively. It is obvious that most of the selected features have different distributions among different difficulty levels.

Based on the distributions shown in Figure 5-5, *Duration of all typing phases divided by task duration* and *Duration of all understanding phases divided by the number of word* are the features show the greatest difference among different difficulty levels. Both features measure the ratio of time that a subject spends on the understanding and typing phases. It shows a clear trend that with the increase of the difficulty, a subject will spend more time on understanding than typing. The explanation of this phenomenon is obvious that it does not take much effort for a subject to understand the flow of the text when the piece to be summarized is clear, particularly when the material is familiar to him/her. Hence, the ratio of time spent on typing gets increased.

In addition, since the text's logic is explicit for the easy summary, a subject does not need to read the texts back and forth to search for useful information. This is evidenced by the fact that when summarizing the easy text, the subjects exhibit the smallest value of *Mean distance of all SSs along x – positive direction*. However, if the content of the text is unfamiliar to the subjects when they read the text, it is hard for them to predict the next word, which causes the length of the reading saccade to be decreased [74], which can be shown

by *Duration of all understanding phases divided by number of words*.

Compared with the easy task and the hard task, the article in the medium task contains more logical changes due to the different arguments presented. Fully understand the article requires a subject to figure out the relationships among arguments, resulting in a high cognitive load. High cognitive load leads to a decrease in saccade speed [108] and a decrease in the number of words generated in each typing phase.

Another interesting point we find is that most of the selective features extracted from eye-tracking modality are in the skimming group, which can be shown in Table 5-8. One possible reason is that skimming behavior (Type B in Table 5-4) occurs more frequently than reading behavior (Type A in Table 5-4). To summarize, there are on average 27.8 occurrences of skimming behavior, a total of 154.8 seconds, compared with 1.2 occurrences of reading behavior with a total of 55.5 seconds.

5.4 Exploring the Benefits of Multimodal Features

To verify that the model constructed on the multimodal features truly outperforms the models built based on a single modality, we follow the same procedure introduced in Section 5.2 and build six different models based on the selected features extracted from the eye-tracking signal, the keyboard signal, the duration-related information, and combinations of the above. Selected features for every single modality are listed in Table 5-11. Same evaluation mechanism: leaving-one-subject-out cross-validation is applied to achieve the CCR performance of each model.

No.	Feature Selected from Each Modality		
	Eye-tracking	Keyboard	Duration-related
1	Mean distance of all SSs along the x-positive direction	Number of keypresses	Duration of all understanding phases divided by n_{sum}
2	Mean speed of all SSs	Number of words generated since the last rereading	Duration of all typing phases divided by task duration
3	Mean distance of all FRSs	Number of words generated since the last reading of the text to be summarized	Duration of completing the summary task divided by n_{gen}
4	Std. duration of all fixations after FRS	Mean number of words generated in each typing phase	Duration of all typing phases divided by n_{gen}
5	Std. duration of all fixations after BR		

*** n_{sum} stands for number of words in the text to be summarized; n_{gen} stands for number of words are generated; SS stands for searching saccade; RS stands for reading saccade; FRS and BRS stands for forward and backward reading saccade**

Table 5-11 Features selected from each modality

Figure 5-6 presents all the performances achieved by the models constructed on different combinations of the modalities or the contribution from each modality (or a combination thereof), and the results indicate that the multimodal approach achieves the best performance. Even for the eye-tracking modality, which contains 82 features taking over about 80% of the total potential features, the multimodal approach still exceeds the model built on the eye-tracking modality around 15%.

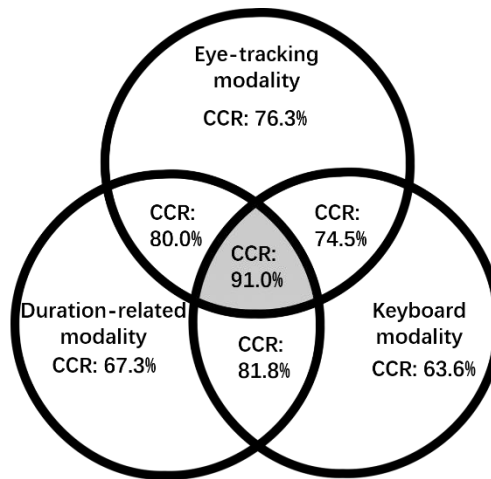


Figure 5-6 Performances (CCRs) contributed by different modalities

	Average of pair-wise R^2	Performance in CCR
Multimodal	0.04	91.0%
Eye-tracking modality	0.08	76.3%
Keyboard modality	0.31	63.6%
Duration-related modality	0.27	67.3%
Eye-tracking + keyboard modalities	0.17	74.5%
Eye-tracking + duration-related modalities	0.14	80.0%
Keyboard + duration-related modalities	0.21	72.7%

Table 5-12 Average of pair-wise R^2 for different modalities

To further investigate the reason that the multimodal approach shows the best performance, we define feature value sequence $VSeq_i$ for the i^{th} selected features: $VSeq_i = \{f_i^0, f_i^1, \dots, f_i^n\}$, where f_i^k stands for the value of the i^{th} selected feature for the k^{th} instance for a particular class and $n = 20$. For the selected features from all modalities, from only eye-tracking modality, from only keyboard modality, from only duration-related modality, and from their combinations, we generate feature value sequences separately. Then the average of the pair-wise square of the correlation coefficient (R^2) among feature value

sequences in their own group is computed and reported in Table 5-12. The value of R^2 is used to measure the relationship between two sequences, and the larger the value is, the greater the correlation exists. According to the results in Table 5-12, we notice that the multimodal approach shows the lowest average pair-wise R^2 , which indicates that compared to the models built based on the single modality, features used by the multimodal approach are less correlated with each other. It means that the multimodal approach captures a larger spectrum of possible behaviors. This provides a possible reason to explain why the multimodal approach outperforms the others.

5.5 Summary

In this chapter, we focus on investigating the cognitive process of summary writing. In this study, we propose a multimodal approach that analyzes the cognitive process of summary writing to recognize different difficulty levels of the summary task. To validate our approach, we construct our own dataset by conducting human experiments of performing summary writings with different difficulties over multiple iterations. During the experiments, multiple signal channels are collected, including the eye-tracking signal, the keyboard signal, and the screen recording. Combining and analyzing the information across multiple channels of signal, we extract multimodal features, including features to describe different types of fixations and saccades for both understanding and skimming behaviors, features to model the keyboard dynamics, features to model the text generation procedure, and features to capture the duration-related information. Evaluated on the dataset constructed by us, our multimodal approach achieves more than 90% accuracy, which is approximately 55% performance improvement above the baseline, which indicates that our multimodal features are able to

capture behavioral differences when summarizing the text in different difficulty levels.

In this study, we also investigate one of the potential reasons that the multimodal approach outperforms other models that use only a single modality or parts of full modalities. It is because that multimodal features can capture a wider range of possible behaviors, providing more independent information to the machine learning model. We hope these findings could help the community better understand human behaviors to fulfill the gap of understanding the cognitive process of summary writing and provide an example of how to design the multimodal features to capture more behavioral information.

6 Discussion

6.1 Performance evaluation for subjective experiment

This thesis aims to investigate the human state of users when they are interacting with computers. Precisely, we infer whether a user is stressed or not during the interaction with a computer and the user's writing/summarizing cognitive process based on gaze and mouse behaviors. For both stress detection and analysis of the writing/summarizing cognitive process, experiments are composed of multiple sessions in different settings. For example, when we investigate the effect of stress, two kinds of sessions, stressed sessions and non-stress sessions, are conducted. When analyzing the impact of the writing genre, three different sessions are completed by each subject during which he/she writes articles in different genres. Then features are designed to capture the differences in gaze and hand behaviors across different kinds of sessions. The final performance of our extracted features is evaluated as a classification problem, which is the correct classification rate of determining the kinds of sessions based on the extracted features.

The above evaluation process is effective based on an essential assumption that subjects' states are aligned with our expectations, which means that subjects become stressed in stress sessions and calm in non-stress sessions. To verify the alignment between a subject's state and our expectation, each subject needs to self-report after each session. Although the self-reporting approach is widely adopted in many previous studies, there also exist some concerns, which may affect the effectiveness of the evaluation process.

The biggest concern is the reliability of the self-reporting, specifically, whether a subject is able to perceive if he/she is under stress and the degree of the

stress level. In our case, after applying the efficient stress inducement methods (time-pressure and background noise), mental stress can usually be induced. Therefore, we believe subjects can tell when they are under stress in our experiment. Most of the subjects report that they are more concentrated on the tasks when they are in stressed sessions, and some of them even feel their heartbeat speeding up and sweating. However, the *degree* of stress, is not reliable if we only base on self-reporting, as it is hard for a subject to tell the difference between stress levels 3 and 4 on a 5-point scale, for example. To solve that problem, we would need to involve other equipment or methods in obtaining the ground truth.

One possible way is to involve physiological signals, such as galvanic skin response, heartbeat rate, and blood pressure signal. Nowadays, these signal detectors are integrated and embedded into a single wristband/smartwatch. Because of the small size of the equipment, presumably it will not affect the subjects' behaviors. Utilizing physiological signals alone cannot predict the stress level with a hundred percent accuracy. Still, we can achieve the change of the stress level by comparing physiological signals within a session. We can also check the reliability of the self-report by comparing physiological signals across different kinds of sessions for a specific subject. Another benefit of involving the physiological signals is that we can do long-term mental stress detection, also known as continuous stress detection. Self-reporting is not feasible for the long-term mental stress detection problem because it is hard to decide when to do the self-reports and the frequency of self-reports.

6.2 User-independent model vs. user-dependent model

In this thesis, all the models constructed are user-independent models, which means models are generated to capture the indicative behaviors that exist among

the majority of subjects. The benefit of a well-trained user-independent model is that it can work efficiently for a subject whose data is not even learned by the model. To construct a user-independent model, we need to eliminate the effect of individual differences so that a model can discover the common behaviors that exist across all the subjects. For example, in the writing experiment, we find that college students have better typing skills than elders, which reflects in the different typing speeds and the different number of words generated in each candidate box. Therefore, feature normalization across subjects is highly needed. Otherwise, it may confuse the machine learning model and impacts the performance. To build a better user-independent model, we need to collect data from different subjects, and the more subjects involved, the more robust a model we can achieve. But building a user-independent model does not require collecting too much data from the same subject.

On the other hand, a user-dependent model is trained on the data collected from a single subject and tested and used by that specific subject. Hence, a user-dependent model can capture the specific indicative behaviors, which only exist for that particular subject. The benefit of a well-trained user-dependent model is that it can always yield better performance than a user-independent model used for a specific subject. It is because that a user-dependent model is trained only based on the data collected from that particular subject, and it won't be confused by others' different behaviors. Therefore, to construct a user-dependent model, we need to collect a large amount of data from the same subject.

To deploy our models, such as the stress detection model in the real world. By considering the characteristics of a user-independent model and a user-dependent model. We can first deploy the user-independent model since it can work in a decent performance for all the users. When a user is using the model, we

can keep collecting his/her data till we achieve enough data to build a user-dependent model only for that user, which can work in the best performance.

6.3 Influences of various user environments in real life

The data from our study was collected in a lab or office environment, meaning that it is very homogeneous across subjects. However, in real use contexts, the environments may be quite different from ours, and furthermore, it may vary from subject to subject. This section mainly discusses the influence of the environment on the data and the subsequent performance of the model.

For both stress detection and writing/summarizing cognitive analysis, our experimental environments use the same hardware devices and UI settings (font size, line spacing, etc...). Therefore, the performance of models will be decreased if they are used in a dramatically different environment without re-training by involving the data collected in the new environment. But we believe the performance will not drop a lot based on the following reasons.

First, most extracted features are not significantly affected by the environment settings. For example, features in Table 3-4 and Table 3-5 are designed to model the MGAttraction signals, which are more related to gaze and mouse relative movement. Although the environment setting may affect the gaze and mouse movement individually, their relative movement (gaze-mouse coordination) is more related to the user's behaviors, which is not affected too much by the environment [48]. Also, for the writing/summarizing cognitive process analysis, most of the features extracted from a thinking window are used to describe the behaviors of different kinds of fixations and saccades, and features extracted from a typing window are used to describe typing dynamics and keypress activities. Those behaviors are related to computer-use habits, which are also less

affected by the environment.

Second, all the experiment settings for stress detection and writing/summarizing analysis are designed to be pursued in line with the actual situation. All the experiments are carried out in a conventional office environment with standard input devices, which are frequently used by common users. Also, during each task, we do not impose too many constraints to imitate actual usage as much as possible. For example, for the web searching task, subjects are allowed to visit any websites and view the information in any format (texts, images, videos). For the writing task, subjects are asked to write on Microsoft Word with the default view layout, and for the summarizing task, we choose to use the split-view mode to evenly put the texts to be summarized and the writing pad side by side. Both configurations are frequently used in everyday life to make sure there is no huge difference between our experiments settings and the real-life setups.

6.4 Other useful signal modalities under the same settings

This thesis aims to infer the human state based on gaze and hand behaviors. The input signal modalities utilized by us are the eye-tracker signal, mouse movement signal, keyboard dynamics, and keyboard activities signal. In section 3.3.2, we estimate eye gaze locations based on the webcam video instead of relying on the eye tracker signal. Hence, for the final setting, the input signal modalities contain webcam video signal, mouse movement signal, keyboard dynamics and activities signal. In this section, we are going to discuss what other information can be extracted to infer the human state besides gaze and mouse behaviors and their benefits and limitations.

First, facial expression signals can be extracted from the webcam video signal,

and facial expressions are highly related to the user's emotion. By using computer vision techniques, facial action units [113] can be extracted, which can be used as features to detect the user's emotion. Facial action units are the actions (raiser, lower, tightener, stretcher ...) of the brow, lip, lid, cheek, nose, and chin. The benefits of using facial expressions are easy to be extracted and interpreted. User-dependent models trained based on facial expression features can achieve a good performance. However, sometimes facial expressions can vary across different users, even when they are in the same mental state. Therefore, the performance of user-independent models built based on facial expressions tends to not be very satisfactory [113].

Second, the eye blink rate signal can be extracted from the webcam video. According to the previous works [68, 80], blinks occur during reading or speaking and reflect changes of attention and changes in thought process, which is negatively related to the amount of attention needed by a task. Therefore, eye blink rate can be utilized to infer the mental state, such as the cognitive load and user's attention level. However, using eye blink rate to infer the mental state in a real-life application is not simple. Because the eye blink duration is so short, around 100 ms to 400 ms, therefore, the speed of webcam video should be at least 30 frame-per-second (fps), which is 3 images for each blink. Also, blinking is an individualist behavior, which may lead to worse performance of the user-independent model.

Also, pupil dilation is another important modality that can potentially be extracted from the webcam video. Pupil dilation can be considered as pupil diameter. Usually, pupil diameter will be changed based on the light intensity. If a person is in a low light environment, his/her pupil diameter will become larger to absorb more light. Besides the light intensity, the pupil diameter is also affected

by the cognitive load. Previous works [45, 56] illustrate that the pupil diameter can be considered as a strong indicator of the cognitive load that pupil diameter becomes larger with the increment of the cognitive load. The limitations of the pupil diameter signal modality are that it requires that webcam video is in high resolution, and it will be influenced heavily by the light intensity of the environment.

According to the finding discovered in section 5.4, a multimodality model built based on features extracted from different uncorrelated signal modalities always outperforms than a single modality model. Therefore, it is highly valuable that we can build a multimodal model in the future by considering all the signal modalities mentioned above so that it can show the best performance in a real-world scenario.

6.5 Implications of the findings beyond this work

The main contribution of this thesis can be categorized into two parts. First, we propose a UI-agnostic stress detection method, which can successfully detect mental stress without relying on any special devices. Second, we explore the writing/summarizing cognitive process based on gaze and typing behaviors, including investigating the effect of the age factor, the writing genre, and the difficulty level. The implications of each part of the contribution will be discussed in this section.

The implication of the first part of the contribution is straightforward, which is to detect stress while a user is interacting with a computer. Since our proposed method does not rely on any special devices, our stress detection method can be easily implemented into the existing applications to monitor stress levels in real-time. As mentioned in the introduction, our stress detection method is beneficial

for e-learning applications to avoid the mental stress caused by the high cognitive load that affects the learning outcome. Also, the idea of our stress detection method can be applied to the driving scenario, where the gaze is used to view the road situation, hands are used to control the steering wheel, and the gaze-hand coordination can be utilized to infer the mental state of a driver.

The implications of the second part of the contribution mainly reflect in that we demonstrate an example of how to analyze a user's cognitive process based on users' behaviors. Basically, we first need to divide the whole process into several phases, and users show different behaviors in each kind of phase. Both statistics-based and sequence-based features are extracted. Statistics-based features contain both features to describe the overall behaviors for the whole process and the specific behaviors for each kind of phase. Sequence-based features are used to capture the transition information over different kinds of phases. The best performance is achieved by combining statistic-based and sequence-based features together. Moreover, we find that using the sequence-based features alone achieves much better performance than the statistics-based features achieve on their own. It illustrates that variation of user behaviors is a powerful indicator of users' cognitive and mental state. We believe that variation-based behavior features can be extended to other applications, such as stress detection and behavior-based continuous authentication.

The result shown in Figure 4-15 demonstrates that training separate models indeed achieves better performance than training a single model to cover both touch and non-touch typists, and the impact of typing skill is far greater than the impact of age, at least on our task of writing genre detection. A model trained with data from both touch and non-touch typists is easily confused by behaviors with the same patterns and causes. This implies that combining data from different

groups to increase the size of training data is not always helpful, as it risks conflating data with different root causes. Human-computer interaction studies often involve data from different groups, particularly different subject populations. Our results suggest that one should be very careful with managing the training data based on the understanding of user behaviors, a point which is seldom mentioned in previous work. We hope that our findings can benefit the human-computer interaction community and lead to better behavior-based models.

7 Conclusion and Future Work

7.1 Conclusion

Understanding the affective and cognitive state is essential in HCI study and draw huge attention recently. It provides essential knowledge for us to design and develop intelligent systems in different emerging areas. This thesis focuses on inferring users' affective state, specifically mental stress, based on gaze and typing behaviors and understanding users' cognitive process of writing and summarizing based on gaze and typing behaviors. Our study can also be considered as an excellent example of inferring the affective and cognitive state based on the gaze and hands behaviors during daily computer interaction.

We investigate the mental stress inference in three steps. First, we study mental stress detection in the static UI environment. We successfully detect stress based on the representative movement patterns extracted from the gaze and mouse transition sequences and gaze-mouse coordination features modeling the relative movements of gaze and mouse in the spatial, time, and speed domains. It proves that mental stress can be inferred effectively via the gaze and mouse behaviors. However, for stress detection in a dynamic UI environment, all the above features can no longer be extracted easily. Therefore, we propose a coordinate system named MGAttraction to measure the gaze and mouse attraction reflected by their relative movements in a translation- and rotation-invariant manner. A UI agnostic stress detection method is constructed based on the MGAttraction coordinate system. Both segment-level features and session-level features are extracted to model the specific gaze and mouse behaviors inside each segment and the overall behaviors across all the segments. Finally, to improve our stress detection method's generalizability, we estimate the gaze on-screen locations based on the facial

landmarks detected from the webcam video, which are used to infer mental stress instead of using the eye-tracker. Combining with the pupil movement features proposed by us to model the pupil movement behaviors, the performance of the webcam-based stress detection method is close to the performance of using the eye-tracker, which is a kind of special equipment to detect the gaze on-screen locations.

To understand the cognitive process of writing, we first show that the age-factors and writing genres affect the cognitive process of writing. The effects of the age-factors mainly reflect on the typing behaviors and the behaviors that gaze travels between the screen and keyboard when they are about to type on the keyboard. All these effects indicate typing skills and working memory capacities are different among different age groups. For the effects of the writing genres, unlike the age-factors, that the gaze behaviors are affected, especially for the rereading behavior. When a subject composes a complicated article, the rereading behavior appears more frequently. Also, when composing a complicated article, there exists more pauses with a long duration when they are typing on the computer. Both statistics-based and sequence-based features are extracted to model the gaze-typing behaviors and how gaze-typing behaviors are changed in a period of time.

Compared to writing, summarizing is a multitask of reading and writing. Therefore, to investigate the cognitive process of summarizing in different difficulty levels, we first divide the whole summarizing period into reading phases, including understanding and referencing the text and writing phases. Then multimodal features are extracted to model the gaze-typing behaviors in different phases. We find that when the text to be summarized is difficult, a subject will spend more time understanding the text with shorter reading saccades. When referring to the text content, longer skimming saccades appear since they need to

cover more information into one sentence for a complicated text.

This thesis concludes with the limitations of current work and potential future work.

7.2 Limitations and Future work

The first limitation of our study is that the size of our datasets is relatively small. For both stress detection datasets and summarizing task dataset, there are around 20 subjects are recruited. Although the size of each dataset is reasonable for proving the feasibility of our hand-crafted features and being used to analyze how behaviors are different across different groups, it is not sufficient for some data-driven approaches such as deep learning algorithms so that they cannot simply be applied to our dataset. Therefore, it is highly needed to increase the size of our datasets. Because data-driven approaches can automatically extract features based on the data, which is completely different from extracting hand-craft features based on background knowledge and the understanding of the question, it would be interesting to compare these two kinds of learning approaches in HCI problems.

Besides collecting more data, data augmentation is another possible way to enlarge our dataset. Data augmentation is the process of transforming existing data to generate new data for training with the aim of improving the performance of classifiers. However, most of the data in our study are in the time-series format. How to transform them without destroying their features in the time domain is still needed to be further explored.

In addition, our study can be improved by further diversifying our tasks in each experiment. Besides the web searching task, we can also involve online shopping and video gaming tasks for stress detection. Since both are daily

computer interaction tasks and contain lots of gaze and mouse movements. Also, the users' affective state is essential to be analyzed for these two tasks. For the writing and the summarizing tasks, more genres of articles can be required subjects to complete, and more articles with different difficulty levels can be involved to further validate our findings. Another interesting direction that we can further investigate the cognitive process of writing is to study whether and how the pressure, such as time pressure affects the cognitive process of writing shown by gaze and mouse behaviors. Since writing under pressure to meet strict deadlines is a common scenario in the real world. Therefore, such behaviors are worthwhile to be investigated.

Also, exploring the effects of different kinds of stress inducements on user behaviors is highly valuable and interesting. In the experiments of stress detection, time limit pressure and background noise are used to induce stress. Although both methods can induce stress efficiently, it is unclear whether they induce the same type of stress. The previous study [31] shows that mental stress can be divided into two categories: high cognitive load and environmental pressure. Different kinds of mental stress may have different effects on human behaviors. For example, some kinds of stress are beneficial and motivated to make users more concentrated on their tasks. However, some types of stress make users overwhelmed and affect working efficiency. Therefore, we can explore users' gaze and hand behaviors under different types of stress inducement methods separately for future work. Also, it is interesting that we can investigate the relationship between working efficiency and mental stress.

Finally, in our study, the states of subjects are accessed through the post-experiment questionnaires. The benefit of utilizing the post-experiment questionnaire is that it will not interfere with the experiment's process. But the

drawback is that we can only have an overview state for an entire period of the experiment session. In other words, we can only have an overall label for the entire session. While for the study of stress detection, it is hard to guarantee that a subject is in the stress state for the entire session, which may affect the further fine-grained analysis's performance. Therefore, in the future, besides only relying on the post-experiment questionnaire, we can also involve the real-time physiological detectors, such as E4 wristband [83], a kind of unobtrusive equipment, which will not disturb subjects too much, to access the states of subjects in the real-time. With the help of the real-time physiological signals, we can identify the time points when a subject becomes stressed and backs to normal exactly. Based on these kinds of information, we can further investigate what kinds of gaze and mouse behaviors can be used as signs of subjects will become stressed shortly so that different stress reduction approaches can be applied in advance.

References

- [1] Abouelenien, M., Burzo, M. and Mihalcea, R. 2016. Human acute stress detection via integration of physiological signals and thermal imaging. *ACM International Conference Proceeding Series* (2016).
- [2] Adams, P., Rabbi, M., Rahman, T., Matthews, M., Volda, A., Gay, G., Choudhury, T. and Volda, S. 2014. Towards personal stress informatics: Comparing minimally invasive techniques for measuring daily stress in the wild. *Proceedings - PERVASIVEHEALTH 2014: 8th International Conference on Pervasive Computing Technologies for Healthcare* (2014).
- [3] Aigrain, J., Dubuisson, S., Detyniecki, M. and Chetouani, M. 2015. Person-specific behavioural features for automatic stress detection. *2015 11th IEEE International Conference and Workshops on Automatic Face and Gesture Recognition, FG 2015* (2015).
- [4] Al-Showarah, S., AL-Jawad, N. and Sellahewa, H. 2014. Effects of user age on smartphone and tablet use, measured with an eye-tracker via fixation duration, scan-path duration, and saccades proportion. *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)* (2014).
- [5] Alamargot, D. and Chanquoy, L. 2001. *Through the Models of Writing*. Springer Netherlands.
- [6] Alamargot, D., Chesnet, D., Dansac, C. and Ros, C. 2006. Eye and Pen: A new device for studying reading during writing. *Behavior Research Methods*. (2006). DOI:<https://doi.org/10.3758/BF03192780>.
- [7] Alberdi, A., Aztiria, A. and Basarab, A. 2016. Towards an automatic early stress recognition system for office environments based on multimodal measurements: A review. *Journal of Biomedical Informatics*. 59, (Feb. 2016), 49–75. DOI:<https://doi.org/10.1016/j.jbi.2015.11.007>.
- [8] Bahreini, K., Nadolski, R. and Westera, W. 2016. Towards multimodal emotion recognition in e-learning environments. *Interactive Learning Environments*. 24, 3 (Apr. 2016), 590–605.

DOI:<https://doi.org/10.1080/10494820.2014.908927>.

[9] Baltrusaitis, T., Zadeh, A., Lim, Y.C. and Morency, L.P. 2018. OpenFace 2.0: Facial behavior analysis toolkit. *Proceedings - 13th IEEE International Conference on Automatic Face and Gesture Recognition, FG 2018* (2018).

[10] Bannon, L. 2011. Reimagining HCI. *interactions*. 18, 4 (Jul. 2011), 50. DOI:<https://doi.org/10.1145/1978822.1978833>.

[11] Barreto, A., Zhai, J. and Adjouadi, M. 2007. Non-intrusive physiological monitoring for automated stress detection in human-computer interaction. *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)* (2007).

[12] Bartoshuk, L.M. and Schiffman, H.R. 1977. Sensation and Perception: An Integrated Approach. *The American Journal of Psychology*. (1977). DOI:<https://doi.org/10.2307/1421748>.

[13] Bartsch, L.M., Loaiza, V.M. and Oberauer, K. 2019. Does limited working memory capacity underlie age differences in associative long-term memory? *Psychology and Aging*. (2019). DOI:<https://doi.org/10.1037/pag0000317>.

[14] Bereiter, C. and Scardamalia, M. 2013. *The psychology of written composition*.

[15] Biedert, R., Hees, J., Dengel, A. and Buscher, G. 2012. A robust realtime reading-skimming classifier. *Eye Tracking Research and Applications Symposium (ETRA)* (2012).

[16] Bieg, H.J., Chuang, L.L., Fleming, R.W., Reiterer, H. and Bülthoff, H.H. 2010. Eye and pointer coordination in search and selection tasks. *Eye Tracking Research and Applications Symposium (ETRA)* (2010).

[17] Brown, A.L. and Day, J.D. 1983. Macrorules for summarizing texts: the development of expertise. *Journal of Verbal Learning and Verbal Behavior*. (1983). DOI:[https://doi.org/10.1016/S0022-5371\(83\)80002-4](https://doi.org/10.1016/S0022-5371(83)80002-4).

[18] Butsch, R.L.C. 1932. Eye movements and the eye-hand span in typewriting.

Journal of Educational Psychology. (1932).
DOI:<https://doi.org/10.1037/h0073463>.

[19] Carneiro, D., Castillo, J.C., Novais, P., Fernández-Caballero, A. and Neves, J. 2012. Multimodal behavioral analysis for non-invasive stress detection. *Expert Systems with Applications.* (2012).
DOI:<https://doi.org/10.1016/j.eswa.2012.05.065>.

[20] Chen, S., Epps, J., Ruiz, N. and Chen, F. 2011. Eye activity as a measure of human mental effort in HCI. *International Conference on Intelligent User Interfaces, Proceedings IUI* (2011).

[21] Chukharev-Hudilainen, E. 2014. Pauses in spontaneous written communication: A keystroke logging study. *Journal of Writing Research.* (2014).
DOI:<https://doi.org/10.17239/jowr-2014.06.01.3>.

[22] Ciman, M., Wac, K. and Gaggi, O. 2015. ISensestress: Assessing stress through human-smartphone interaction analysis. *Proceedings of the 2015 9th International Conference on Pervasive Computing Technologies for Healthcare, PervasiveHealth 2015* (2015).

[23] Crawford, J.D., Medendorp, W.P. and Marotta, J.J. 2004. Spatial transformations for eye-hand coordination. *Journal of Neurophysiology.*

[24] Debole, F. and Sebastiani, F. 2003. Supervised term weighting for automated text categorization. *Proceedings of the ACM Symposium on Applied Computing* (2003).

[25] Deng, S., Chang, J., Kirkby, J.A. and Zhang, J.J. 2016. Gaze–mouse coordinated movements and dependency with coordination demands in tracing. *Behaviour and Information Technology.* (2016).
DOI:<https://doi.org/10.1080/0144929X.2016.1181209>.

[26] Dinno, A. 2015. Nonparametric pairwise multiple comparisons in independent groups using Dunn’s test. *Stata Journal.* (2015).
DOI:<https://doi.org/10.1177/1536867x1501500117>.

[27] Du, J., Huang, J., An, Y. and Xu, W. 2018. The Relationship between stress

and negative emotion: The Mediating role of rumination. *Clinical Research and Trials*. 4, 1 (2018). DOI:<https://doi.org/10.15761/CRT.1000208>.

[28] Edla, D.R., Mangalorekar, K., Dhavalikar, G. and Dodia, S. 2018. Classification of EEG data for human mental state analysis using Random Forest Classifier. *Procedia Computer Science* (2018).

[29] Feit, A.M., Weir, D. and Oulasvirta, A. 2016. How we type: Movement strategies and performance in everyday typing. *Conference on Human Factors in Computing Systems - Proceedings* (2016).

[30] Flower, L. and Hayes, J.R. 1981. A Cognitive Process Theory of Writing. *College Composition and Communication*. (1981). DOI:<https://doi.org/10.2307/356600>.

[31] Gaillard, A.W.K. and Wientjes, C.J.E. 1994. Mental load and work stress as two types of energy mobilization. *Work and Stress*. (1994). DOI:<https://doi.org/10.1080/02678379408259986>.

[32] Galy, E., Cariou, M. and Mélan, C. 2012. What is the relationship between mental workload factors and cognitive load types? *International Journal of Psychophysiology*. (2012). DOI:<https://doi.org/10.1016/j.ijpsycho.2011.09.023>.

[33] Gjoreski, M., Gjoreski, H., Luštrek, M. and Gams, M. 2016. Continuous stress detection using a wrist device - in laboratory and real life. *UbiComp 2016 Adjunct - Proceedings of the 2016 ACM International Joint Conference on Pervasive and Ubiquitous Computing* (2016).

[34] Gjoreski, M., Luštrek, M., Gams, M. and Gjoreski, H. 2017. Monitoring stress with a wrist device using context. *Journal of Biomedical Informatics*. (2017). DOI:<https://doi.org/10.1016/j.jbi.2017.08.006>.

[35] Gordon-Salant, S. and Cole, S.S. 2016. Effects of age and working memory capacity on speech recognition performance in noise among listeners with normal hearing. *Ear and Hearing*. (2016). DOI:<https://doi.org/10.1097/AUD.0000000000000316>.

[36] Greene, N.R., Naveh-Benjamin, M. and Cowan, N. 2020. Adult age

differences in working memory capacity: Spared central storage but deficits in ability to maximize peripheral storage. *Psychology and Aging*. (2020). DOI:<https://doi.org/10.1037/pag0000476>.

[37] Gregersen, A. 2014. Cognition. *The Routledge Companion to Video Game Studies*.

[38] Gudgeon, A.C. and Howell, D.C. 1994. Statistical Methods for Psychology. *The Statistician*. (1994). DOI:<https://doi.org/10.2307/2348956>.

[39] Haak, M., Bos, S., Panic, S. and Rothkrantz, L.J.M. 2009. Detecting Stress Using Eye Blinks And Brain Activity From EEG Signals. *Game-On 2009*. (2009).

[40] Han, H., Byun, K. and Kang, H.G. 2018. A deep learning-based stress detection algorithm with speech signal. *AVSU 2018 - Proceedings of the 2018 Workshop on Audio-Visual Scene Understanding for Immersive Multimedia, Co-located with MM 2018* (2018).

[41] Hansen, J.H.L. and Patil, S. 2007. Speech under stress: Analysis, modeling and recognition. *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*. (2007). DOI:https://doi.org/10.1007/978-3-540-74200-5_6.

[42] Healey, J.A. and Picard, R.W. 2005. Detecting stress during real-world driving tasks using physiological sensors. *IEEE Transactions on Intelligent Transportation Systems*. (2005). DOI:<https://doi.org/10.1109/TITS.2005.848368>.

[43] Henderson, J.M., Choi, W., Luke, S.G. and Desai, R.H. 2015. Neural correlates of fixation duration in natural reading: Evidence from fixation-related fMRI. *NeuroImage*. (2015). DOI:<https://doi.org/10.1016/j.neuroimage.2015.06.072>.

[44] Hernandez, J., Paredes, P., Roseway, A. and Czerwinski, M. 2014. Under pressure: Sensing stress of computer users. *Conference on Human Factors in Computing Systems - Proceedings* (2014).

[45] Hess, E.H. and Polt, J.M. 1964. Pupil size in relation to mental activity during simple problem-solving. *Science*. (1964).

DOI:<https://doi.org/10.1126/science.143.3611.1190>.

[46] Homer, B.D., Plass, J.L. and Blake, L. 2008. The effects of video on cognitive load and social presence in multimedia-learning. *Computers in Human Behavior*. (2008). DOI:<https://doi.org/10.1016/j.chb.2007.02.009>.

[47] Hovsepian, K., Al'absi, M., Ertin, E., Kamarck, T., Nakajima, M. and Kumar, S. 2015. CStress: Towards a gold standard for continuous stress assessment in the mobile environment. *UbiComp 2015 - Proceedings of the 2015 ACM International Joint Conference on Pervasive and Ubiquitous Computing* (2015).

[48] Huang, J., White, R.W. and Buscher, G. 2012. User see, user point: Gaze and cursor alignment in Web search. *Conference on Human Factors in Computing Systems - Proceedings* (2012).

[49] Huang, M.X., Li, J., Ngai, G. and Va Leong, H. 2016. StressClick: Sensing stress from gaze-click patterns. *MM 2016 - Proceedings of the 2016 ACM Multimedia Conference* (2016).

[50] Inhoff, A.W. and Gordon, A.M. 1997. Eye Movements and Eye-Hand Coordination During Typing. *Current Directions in Psychological Science*. 6, 6 (Dec. 1997), 153–157. DOI:<https://doi.org/10.1111/1467-8721.ep10772929>.

[51] Jääskeläinen, R. 2010. Think-aloud protocol. *Handbook of translation studies*. 371–374.

[52] Jiang, X., Li, Y., Jokinen, J.P.P., Hirvola, V.B., Oulasvirta, A. and Ren, X. 2020. How We Type: Eye and Finger Movement Strategies in Mobile Typing. (2020).

[53] Johansson, R., Wengelin, Å., Johansson, V. and Holmqvist, K. 2010. Looking at the keyboard or the monitor: Relationship with text production processes. *Reading and Writing*. (2010). DOI:<https://doi.org/10.1007/s11145-009-9189-3>.

[54] Joshi, A., Parmar, V., Ganu, A., Mathur, G. and Chand, A. 2004. Keylekh: A keyboard for text entry in Indic scripts. *Conference on Human Factors in Computing Systems - Proceedings* (2004).

- [55] Jyotsna, C. and Amudha, J. 2018. Eye Gaze as an Indicator for Stress Level Analysis in Students. *2018 International Conference on Advances in Computing, Communications and Informatics, ICACCI 2018* (2018).
- [56] Kahneman, D. and Beatty, J. 1966. Pupil Diameter and Load on Memory. *Science*. 154, 3756 (Dec. 1966), 1583–1585. DOI:<https://doi.org/10.1126/science.154.3756.1583>.
- [57] Kapoor, A. and Picard, R.W. 2005. Multimodal affect recognition in learning environments. *Proceedings of the 13th ACM International Conference on Multimedia, MM 2005* (2005).
- [58] Keck, C. 2006. The use of paraphrase in summary writing: A comparison of L1 and L2 writers. *Journal of Second Language Writing*. (2006). DOI:<https://doi.org/10.1016/j.jslw.2006.09.006>.
- [59] Kellogg, R.T. 1996. A Model of working memory in writing. *The science of writing. Theories, methods, individual differences and applications*.
- [60] Keys, A. 1957. The Stress of Life. *American Journal of Public Health and the Nations Health*. (1957). DOI:<https://doi.org/10.2105/ajph.47.5.624>.
- [61] Kirkland, M.R. and Saunders, M.A.P. 1991. Maximizing Student Performance in Summary Writing: Managing Cognitive Load. *TESOL Quarterly*. (1991). DOI:<https://doi.org/10.2307/3587030>.
- [62] Klein, P.D. 1999. Reopening Inquiry into Cognitive Processes in Writing-To-Learn. *Educational Psychology Review*. (1999). DOI:<https://doi.org/10.1023/A:1021913217147>.
- [63] Koldijk, S., Neerincx, M.A. and Kraaij, W. 2018. Detecting Work Stress in Offices by Combining Unobtrusive Sensors. *IEEE Transactions on Affective Computing*. (2018). DOI:<https://doi.org/10.1109/TAFFC.2016.2610975>.
- [64] Koldijk, S., Sappelli, M., Verberne, S., Neerincx, M.A. and Kraaij, W. 2014. The Swell knowledge work dataset for stress and user modeling research. *ICMI 2014 - Proceedings of the 2014 International Conference on Multimodal Interaction* (2014).

- [65] Lambert, W.W. and Lazarus, R.S. 1970. Psychological Stress and the Coping Process. *The American Journal of Psychology*. (1970). DOI:<https://doi.org/10.2307/1420698>.
- [66] Lan, M., Tan, C.L. and Low, H.B. 2006. Proposing a new term weighting scheme for text categorization. *Proceedings of the National Conference on Artificial Intelligence* (2006).
- [67] Lazarus, R.S. 1993. From psychological stress to the emotions: A history of changing outlooks. *Annual Review of Psychology*. (1993). DOI:<https://doi.org/10.1146/annurev.psych.44.1.1>.
- [68] Ledger, H. 2013. The effect cognitive load has on eye blinking. *The Plymouth Student Scientist*. (2013).
- [69] Li, J. 2014. Examining genre effects on test takers' summary writing performance. *Assessing Writing*. (2014). DOI:<https://doi.org/10.1016/j.asw.2014.08.003>.
- [70] Li, J. 2014. The role of reading and writing in summarization as an integrated task. *Language Testing in Asia*. (2014). DOI:<https://doi.org/10.1186/2229-0443-4-3>.
- [71] Likens, A., Allen, L.K. and McNameara, D. 2017. Keystroke Dynamics Predict Essay Quality. *CogSci* (2017).
- [72] Lin, H., Jia, J., Guo, Q., Xue, Y., Huang, J., Cai, L. and Feng, L. 2014. Psychological stress detection from cross-media microblog data using Deep Sparse Neural Network. *Proceedings - IEEE International Conference on Multimedia and Expo* (2014).
- [73] Lin, H., Jia, J., Guo, Q., Xue, Y., Li, Q., Huang, J., Cai, L. and Feng, L. 2014. User-level psychological stress detection from social media using deep neural network. *MM 2014 - Proceedings of the 2014 ACM Conference on Multimedia* (2014).
- [74] Liu, Y., Guo, S., Yu, L. and Reichle, E.D. 2018. Word predictability affects saccade length in Chinese reading: An evaluation of the dynamic-adjustment

model. *Psychonomic Bulletin and Review*. (2018).
DOI:<https://doi.org/10.3758/s13423-017-1357-x>.

[75] Lloyd, S.P. 1982. Least Squares Quantization in PCM. *IEEE Transactions on Information Theory*. (1982). DOI:<https://doi.org/10.1109/TIT.1982.1056489>.

[76] Logan, G.D. 1983. Time, Information, and the Various Spans in Typewriting. *Cognitive Aspects of Skilled Typewriting*.

[77] Lu, H., Rabbi, M., Chittaranjan, G.T., Frauendorfer, D., Mast, M.S., Campbell, A.T., Gatica-Perez, D. and Choudhury, T. 2012. StressSense: Detecting stress in unconstrained acoustic environments using smartphones. *UbiComp'12 - Proceedings of the 2012 ACM Conference on Ubiquitous Computing* (2012).

[78] Lundberg, U., Kadefors, R., Melin, B., Palmerud, G., Hassmén, P., Engström, M. and Elfsberg Dohns, I. 1994. Psychophysiological stress and emg activity of the trapezius muscle. *International Journal of Behavioral Medicine*. (1994). DOI:https://doi.org/10.1207/s15327558ijbm0104_5.

[79] Lyu, Y., Luo, X., Zhou, J., Yu, C., Miao, C., Wang, T., Shi, Y. and Kameyama, K.I. 2015. Measuring photoplethysmogram-based stress-induced vascular response index to assess cognitive load and stress. *Conference on Human Factors in Computing Systems - Proceedings* (2015).

[80] Magliacano, A., Fiorenza, S., Estraneo, A. and Trojano, L. 2020. Eye blink rate increases as a function of cognitive load during an auditory oddball paradigm. *Neuroscience Letters*. (2020). DOI:<https://doi.org/10.1016/j.neulet.2020.135293>.

[81] Marcos-Ramiro, A., Pizarro-Perez, D., Marron-Romera, M. and Gatica-Perez, D. 2014. Automatic blinking detection towards stress discovery. *ICMI 2014 - Proceedings of the 2014 International Conference on Multimodal Interaction* (2014).

[82] Masson, M.E. 1982. Cognitive processes in skimming stories. *Journal of Experimental Psychology: Learning, Memory, and Cognition*. (1982). DOI:<https://doi.org/10.1037/0278-7393.8.5.400>.

[83] McCarthy, C., Pradhan, N., Redpath, C. and Adler, A. 2016. Validation of the

Empatica E4 wristband. *2016 IEEE EMBS International Student Conference: Expanding the Boundaries of Biomedical Engineering and Healthcare, ISC 2016 - Proceedings* (2016).

[84] McGuigan, F.J. and Andreassi, J.L. 1981. Psychophysiology -- Human Behavior and Physiological Response. *The American Journal of Psychology*. (1981). DOI:<https://doi.org/10.2307/1422751>.

[85] Meena, Y.K., Cecotti, H., Wong-Lin, K. and Prasad, G. 2016. A novel multimodal gaze-controlled Hindi virtual keyboard for disabled users. *2016 IEEE International Conference on Systems, Man, and Cybernetics (SMC)* (Oct. 2016), 003688–003693.

[86] Nachar, N. 2008. The Mann-Whitney U: A Test for Assessing Whether Two Independent Samples Come from the Same Distribution. *Tutorials in Quantitative Methods for Psychology*. (2008). DOI:<https://doi.org/10.20982/tqmp.04.1.p013>.

[87] Noton, D. and Stark, L. 1971. Scanpaths in saccadic eye movements while viewing and recognizing patterns. *Vision Research*. (1971). DOI:[https://doi.org/10.1016/0042-6989\(71\)90213-6](https://doi.org/10.1016/0042-6989(71)90213-6).

[88] Papoutsaki, A., Daskalova, N., Sangkloy, P., Huang, J., Laskey, J. and Hays, J. 2016. WebGazer: Scalable webcam eye tracking using user interactions. *IJCAI International Joint Conference on Artificial Intelligence* (2016).

[89] Papoutsaki, A., Gokaslan, A., Tompkin, J., He, Y. and Huang, J. 2018. The eye of the typer: A benchmark and analysis of gaze behavior during typing. *Eye Tracking Research and Applications Symposium (ETRA)* (2018).

[90] Paredes, P.E., Ordoñez, F., Ju, W. and Landay, J.A. 2018. Fast & furious: Detecting Stress with a car steering wheel. *Conference on Human Factors in Computing Systems - Proceedings* (2018).

[91] Rabkin, J.G. and Struening, E.L. 1976. Life events, stress, and illness. *Science*. (1976). DOI:<https://doi.org/10.1126/science.790570>.

[92] Rähkä, K.J. and Sharmin, S. 2014. Gaze-contingent scrolling and reading patterns. *Proceedings of the NordiCHI 2014: The 8th Nordic Conference on*

Human-Computer Interaction: Fun, Fast, Foundational (2014).

[93] Ramos, J. 2003. Using TF-IDF to Determine Word Relevance in Document Queries. *Urologic Clinics of North America*. (2003).

[94] Rayner, K., Chace, K.H., Slattery, T.J. and Ashby, J. 2006. Eye movements as reflections of comprehension processes in reading. *Scientific Studies of Reading*.

[95] Rayner, K., Smith, T.J., Malcolm, G.L. and Henderson, J.M. 2009. Eye movements and visual encoding during scene perception. *Psychological Science*. (2009). DOI:<https://doi.org/10.1111/j.1467-9280.2008.02243.x>.

[96] Ren, P., Barreto, A., Gao, Y. and Adjouadi, M. 2013. Affective assessment by digital processing of the pupil diameter. *IEEE Transactions on Affective Computing*. (2013). DOI:<https://doi.org/10.1109/T-AFFC.2012.25>.

[97] Rodden, K., Fu, X., Aula, A. and Spiro, I. 2008. Eye-mouse coordination patterns on web search results pages. *Conference on Human Factors in Computing Systems - Proceedings* (2008).

[98] Rodrigues, M., Gonçalves, S., Carneiro, D., Novais, P. and Fdez-Riverola, F. 2013. Keystrokes and clicks: Measuring stress on E-learning students. *Advances in Intelligent Systems and Computing* (2013).

[99] Rousseeuw, P.J. 1987. Silhouettes: A graphical aid to the interpretation and validation of cluster analysis. *Journal of Computational and Applied Mathematics*. (1987). DOI:[https://doi.org/10.1016/0377-0427\(87\)90125-7](https://doi.org/10.1016/0377-0427(87)90125-7).

[100] Salvador, S. and Chan, P. 2007. Toward accurate dynamic time warping in linear time and space. *Intelligent Data Analysis*. (2007). DOI:<https://doi.org/10.3233/ida-2007-11508>.

[101] Salvucci, D.D. and Goldberg, J.H. 2000. Identifying fixations and saccades in eye-tracking protocols. *Proceedings of the Eye Tracking Research and Applications Symposium 2000* (2000).

[102] Samura, T. and Nishimura, H. 2009. Keystroke timing analysis for individual identification in Japanese free text typing. *ICCAS-SICE 2009 - ICROS-*

SICE International Joint Conference 2009, Proceedings (2009).

[103] De Santos Sierra, A., Sánchez Ávila, C., Guerra Casanova, J. and Bailador Del Pozo, G. 2011. A stress-detection system based on physiological signals and fuzzy logic. *IEEE Transactions on Industrial Electronics*. (2011). DOI:<https://doi.org/10.1109/TIE.2010.2103538>.

[104] Schnall, P.L., Landsbergis, P.A. and Baker, D. 1994. Job strain and cardiovascular disease. *Annual Review of Public Health*.

[105] Sen, T. and Megaw, T. 1984. The Effects of Task Variables and Prolonged Performance on Saccadic Eye Movement Parameters. *Advances in Psychology*. (1984). DOI:[https://doi.org/10.1016/S0166-4115\(08\)61824-5](https://doi.org/10.1016/S0166-4115(08)61824-5).

[106] Spüler, M., Walter, C., Rosenstiel, W., Gerjets, P., Moeller, K. and Klein, E. 2016. EEG-based prediction of cognitive workload induced by arithmetic: a step towards online adaptation in numerical learning. *ZDM - Mathematics Education*. (2016). DOI:<https://doi.org/10.1007/s11858-015-0754-8>.

[107] Stampe, D.M. 1993. Heuristic filtering and reliable calibration methods for video-based pupil-tracking systems. *Behavior Research Methods, Instruments, & Computers*. (1993). DOI:<https://doi.org/10.3758/BF03204486>.

[108] Di Stasi, L.L., Renner, R., Staehr, P., Helmert, J.R., Velichkovsky, B.M., Cañas, J.J., Catena, A. and Pannasch, S. 2010. Saccadic peak velocity sensitivity to variations in mental workload. *Aviation Space and Environmental Medicine*. (2010). DOI:<https://doi.org/10.3357/ASEM.2579.2010>.

[109] Sun, D., Paredes, P. and Canny, J. 2014. MouStress: Detecting stress from mouse motion. *Conference on Human Factors in Computing Systems - Proceedings (2014)*.

[110] Sun, F.-T.T., Kuo, C., Cheng, H.-T.T., Buthpitiya, S., Collins, P. and Griss, M. 2012. Activity-aware mental stress detection using physiological sensors. *Lecture Notes of the Institute for Computer Sciences, Social-Informatics and Telecommunications Engineering, LNICST (2012)*, 211–230.

[111] Torrance, M., Johansson, R., Johansson, V. and Wengelin, Å. 2016.

Reading during the composition of multi-sentence texts: an eye-movement study. *Psychological Research*. (2016). DOI:<https://doi.org/10.1007/s00426-015-0683-8>.

[112] Vargha, A. and Delaney, H.D. 1998. The Kruskal-Wallis Test and Stochastic Homogeneity. *Journal of Educational and Behavioral Statistics*. (1998). DOI:<https://doi.org/10.3102/10769986023002170>.

[113] Viegas, C., Maxion, R., Lau, S.-H.H., Hauptmann, A., Maxion, R. and Hauptmann, A. 2018. Distinction of stress and non-stress tasks using facial action units. *Proceedings of the International Conference on Multimodal Interaction Adjunct - ICMI '18* (New York, New York, USA, 2018), 1–6.

[114] Visser, B., De Looze, M.P., De Graaff, M.P. and Van Dieën, J.H. 2004. Effect of precision demands and mental pressure on muscle activation and hand forces in computer mouse tasks. *Ergonomics*. (2004). DOI:<https://doi.org/10.1080/00140130310001617967>.

[115] Vizer, L.M. 2009. Detecting cognitive and physical stress through typing behavior. *Conference on Human Factors in Computing Systems - Proceedings* (2009).

[116] Vizer, L.M., Zhou, L. and Sears, A. 2009. Automated stress detection using keystroke and linguistic features: An exploratory study. *International Journal of Human Computer Studies*. (2009). DOI:<https://doi.org/10.1016/j.ijhcs.2009.07.005>.

[117] Vu, K.-P.L. 2005. *Handbook of Human Factors in Web Design*.

[118] van Waes, L., Leijten, M. and Quinlan, T. 2010. Reading during sentence composing and error correction: A multilevel analysis of the influences of task complexity. *Reading and Writing*. (2010). DOI:<https://doi.org/10.1007/s11145-009-9190-x>.

[119] Wagner, J., Kim, J. and André, E. 2005. From physiological signals to emotions: Implementing and comparing selected methods for feature extraction and classification. *IEEE International Conference on Multimedia and Expo, ICME 2005* (2005).

- [120] Wahlström, J., Hagberg, M., Johnson, P.W., Svensson, J. and Rempel, D. 2002. Influence of time pressure and verbal provocation on physiological and psychological reactions during work with a computer mouse. *European Journal of Applied Physiology*. (2002). DOI:<https://doi.org/10.1007/s00421-002-0611-7>.
- [121] Wallot, S. and Grabowski, J. 2013. Typewriting Dynamics: What Distinguishes Simple From Complex Writing Tasks? *Ecological Psychology*. 25, 3 (Jul. 2013), 267–280. DOI:<https://doi.org/10.1080/10407413.2013.810512>.
- [122] Wang, J., Liu, P., She, M.F.H., Nahavandi, S. and Kouzani, A. 2013. Bag-of-words representation for biomedical time series classification. *Biomedical Signal Processing and Control*. (2013). DOI:<https://doi.org/10.1016/j.bspc.2013.06.004>.
- [123] Weill-Tessier, P., Turner, J. and Gellersen, H. 2016. How do you look at what you touch? A study of touch interaction and gaze correlation on tablets. *Eye Tracking Research and Applications Symposium (ETRA)* (2016).
- [124] Xu, P., Sugano, Y. and Bulling, A. 2016. Spatio-temporal modeling and prediction of visual attention in graphical user interfaces. *Conference on Human Factors in Computing Systems - Proceedings* (2016).
- [125] Yamauchi, T. 2013. Mouse trajectories and state anxiety: Feature selection with random forest. *Proceedings - 2013 Humaine Association Conference on Affective Computing and Intelligent Interaction, ACII 2013* (2013).
- [126] Yang, H.C. 2014. Toward a model of strategies and summary writing performance. *Language Assessment Quarterly*. (2014). DOI:<https://doi.org/10.1080/15434303.2014.957381>.
- [127] Yang, L. and Shi, L. 2003. Exploring six MBA students' summary writing by introspection. *Journal of English for Academic Purposes*. (2003). DOI:[https://doi.org/10.1016/S1475-1585\(03\)00016-X](https://doi.org/10.1016/S1475-1585(03)00016-X).
- [128] Yi, K., Guo, Y., Jiang, W., Wang, Z. and Sun, L. 2020. A dataset for exploring gaze behaviors in text summarization. *MMSys 2020 - Proceedings of the 2020 Multimedia Systems Conference* (2020).

- [129] Yu, G. 2008. Reading to summarize in English and Chinese: A tale of two languages? *Language Testing*. (2008). DOI:<https://doi.org/10.1177/0265532208094275>.
- [130] Zhai, J. and Barreto, A.B. 2005. Instrumentation for automatic monitoring of affective state in human computer interaction. *Proceedings of the Eighteenth International Florida Artificial Intelligence Research Society Conference, FLAIRS 2005 - Recent Advances in Artificial Intelligence* (2005).
- [131] Zhao, X., Song, Z., Guo, J., Zhao, Y. and Zheng, F. 2012. Real-time hand gesture detection and recognition by random forest. *Communications in Computer and Information Science* (2012).
- [132] Zheng, Y., Xie, L., Liu, Z., Sun, M., Zhang, Y. and Ru, L. 2011. Why press backspace? Understanding user input behaviors in Chinese Pinyin input method. *ACL-HLT 2011 - Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies* (2011).
- [133] 2013. The Human-computer interaction handbook: fundamentals, evolving technologies, and emerging applications. *Choice Reviews Online*. (2013). DOI:<https://doi.org/10.5860/choice.50-3307>.