User Feedback Analysis for Business Intelligence: Semantics, Sentiment and Model Robustness

by

Chengfei Wang

A dissertation submitted to the Graduate Faculty of
Auburn University
in partial fulfillment of the
requirements for the Degree of
Doctor of Philosophy

Auburn, Alabama
Dec 10, 2022

Keywords: BERT, Text Mining, Network Analysis, Adversarial Example.

Approved by

Xiao Qin, Chair, Alumni Professor of Computer Science and Software Engineering
Ashish Gupta, Co-Chair, Globe Life Professor of Analytics
Wei-Shinn Ku, Professor of Computer Science and Software Engineering
Shubhra Karmaker, Assistant Professor of Computer Science and Software Engineering

Abstract

Business intelligence (BI) is a set of enterprise decision support tools designed to help managers, analysts, and executives rapidly make wise decisions. The growth of the business intelligence applications is fueled and challenged by the large amounts of data of customers arising from internet and the adoption of the technology of Artificial Intelligence for sophisticated data analysis. The corporate world shows particularly strong interests in user-feedback analysis by leveraging the state-of-the-art AI-based natural language processing technology. Automated systems supporting the design of software play a vital role in the field of BI. For example, in the mobile app market, a multi-billion-dollar industry, and online consumer feedback can provide good insights into product strengths and weaknesses. Therefore, in the first part of this dissertation, we put forward an AI-driven framework that supports application design. We utilize deep unsupervised learning models to build a framework for harnessing potential customer feedback information. The first component of the framework applies Bidirectional Encoder Representations from the transformers (BERT)-based topic modelling approach to identify topics and key themes that emerge from user reviews of mobile applications belonging to the health and fitness genre. Sentiment analytics integrates the accompanying ratings to reveal the market acceptance of various aspects of product design.

Asides from the semantic information, the emotional information extracted from the nature language data of customer is also valuable for marketing and branding activities. Therefore, in the second part of this dissertation study, we delve into the emotion analysis of public opinion in the scenario of telemedicine, in which we devise a novel emotional analytics framework. We investigate several emotion models and proposed a novel framework to solve the challenge of non-polarity emotion analysis. We use BERT to extend the traditional dictionary-based to detect the emotion of all words in the context even not included in the

dictionary. We compare our new method against the other baseline methods in semantic analysis task and applied the best performance method on user reviews of telemedicine applications and reveal the social acceptance of the telemedicine.

As the wide use of deep learning models in the business intelligence and its applications, the robustness of deep learning models becomes a challenging issue. When deep learning models are deployed in decision support, the reliability is a big concern especially when it comes to life-critical missions. As the last part of this dissertation research, we study the well-known problem of adversarial example in the realm of image data recognition. Evidence shows that small interpretation on input images before feeding fail the AI classifiers. We argue that the small-difference transformations commonly used are the blame and; therefore, we propose a new model-agnostic defense using a large-difference transformation. Specifically, we apply the novel primitive-based transformation that re-builds input images by primitives of colorful triangles. In terms of the distortions required to completely break the defenses, our experiments on the ImageNet subset demonstrate that significantly large distortions (0.12) are needed to break the defense compared to other state-of-the-art model-agnostic defenses (0.05-0.06) under strong attacks. This finding indicates that large difference transformations tend to improve the adversarial robustness, thereby suggesting a promising new direction towards solving the challenge of adversarial robustness.

Acknowledgments

I am deeply indebted to Dr. Qin and Dr.Gupta for all aspects of my research, career, and life matters. I will never forget every meetings with Dr. Qin and Dr. Gupta, from guide my research plan to improving my writing, all his efforts at every step of this research. Without the guidance, patience, knowledge, and endless support from Dr.Gupta and Dr. Qin, this dissertation research would have never been possible. I believe I have the best advisors in the world, and will always remember and benefit from what they had taught me.

I would also like to thank my committee members, Dr. Wei-Shinn Ku, Dr. Karmaker Santu, their advice and assistance have been very helpful, and their contribution is greatly appreciated. I would like to extend my sincere thanks to Dr. Uzma Raja, one of the most responsible university readers I have ever known, who reviewed this dissertation word by word and offered lots of detailed and valuable suggestions.

Last and most, I would like to express my heartfelt appreciation to my parents and my wife for all the love and support and for helping me be the best version of myself. Thank you, for everything I can imagine.

To my parents and my wife

Table of Contents

Abstract ii

Acknowledgments iv

List of Figures x

List of Tables xiv

1 Introduction 1

1.1 Design of Mobile Application Driven by User Feedback 1

1.2 Sentimental Analysis on Telemedicine App Reviews 4

1.3 Robustness of Deep Learning Model 7

1.4 Contributions of the Dissertation 9

1.4.1 Contributions for Mobile Application Design Driven by User Feedback 9

1.4.2 Contributions for Analysis on Telemedicine Application Reviews . . . 9

1.4.3 Contributions for the Robustness of Deep Learning Model 10

1.5 Organization of the Dissertation 10

2 Literature Review 12

2.1 Topic Modeling and Deep Learning Model 12

2.1.1 Topic Modeling Technique: LDA 13

2.1.2 Deep Learning Models 13

2.2 Analytic Systems for Customer Reviews from Mobile Apps 14

2.2.1 Feature extraction 14

2.2.2 Classification 18

2.2.3 Grouping 20

2.2.4 Ranking 21

2.2.5 Summarization 22

2.3 Methods of Sentimental Analysis 24
2.3.1 Emotional models 24
2.3.2 Methods of Sentimental Analysis 25
2.4 Adversarial Attack and Defense 26
2.4.1 Adversarial Attack 27
2.4.2 Adversarial Defense 29
3 Design Applications from User Feedback using Deep Unsupervised Learning . . 31
3.1 Research Method 31
3.1.1 Data Collection 32
3.1.2 Data Pre-processing 33
3.1.3 Text Analysis 34
3.1.4 Sentiment analysis 41
3.2 Experiment Results of Health Apps 42
3.2.1 Results of Lifestyle App Reviews 42
3.2.2 Results of Nutrition App Reviews 46
3.2.3 Results of Meditation App Reviews 47
3.2.4 Results of Workout App Reviews 48
3.2.5 Results of Prescription App Reviews 50
3.3 Further Discussions 52
3.4 Summary 57
4 Sentimental Analysis of Telemedicine Application Reviews 58
4.1 Research Method 58
4.1.1 Data Collection 59
4.1.2 Sentiment Measurement 59
4.1.3 Network Analysis 61
4.2 Experiment Results 64
4.2.1 Analysis of the Popularity 65

4.2.2 Analysis of by NRC emotion lexicon 65
4.2.3 Analysis of Other Linguistic Dimensions 75
4.2.4 Cluster of Successful Applications 80
4.2.5 Cluster of Applications with Fake Positive 82
4.2.6 Cluster of Unpopular Applications 85
4.3 Discussions 88
5 Improving Robustness via Large-difference Transformation 90
5.1 Motivation and Challenges 90
5.1.1 Optimization Algorithms 91
5.1.2 Strong Attacks 92
5.1.3 Large Transformations 92
5.2 Primitive-based Transformation as a Defender 92
5.3 The Strong Attack: Backward Pass Differentiable Approximation (BPDA) . 94
5.4 Experiments and Results 94
5.4.1 Experiment Setup 95
5.4.2 Baseline: Defense in a None-Attack Setting 95
5.4.3 White Box: Attack on Original Classifier 96
5.4.4 Grey Box: Attack on Protected Classifier 97
5.5 Transfer Experiment 101
5.6 Discussion and Summary 102
6 Conclusions and Future Research Directions 105
6.1 Main Contributions 105
6.1.1 Mobile Application Design Driven by User Feedback 106
6.1.2 Analysis on Telemedicine Application Reviews 106
6.1.3 Robustness of Deep Learning Model 107
6.2 Future Projects 107
6.2.1 Dynamic Monitoring 108

6.2.2 Fake Review Detection 108
6.2.3 Reconstruction with 3D Primitives 108
Bibliography 110

List of Figures

1.1 Adding slight perturbation to the original image (left) we easily get adversarial example (middle) mistakenly classified as guacamole. After primitive-based transformation that reconstruct the image by composing of colourful triangles (right), the perturbations are suppressed and the image is classified correctly again. 8

3.1 The framework of Design from User Feedback using Deep Unsupervised Learning 32

3.2 Clustering results visualized by UMAP topics 37

3.3 Average scores of clusters 38

3.4 Clustering result visualized by PyLDAvis 39

3.5 Connection between subcategory, cluster and themes I 53

3.6 Connection between subcategory, cluster and themes II 54

4.1 Profiling of telemedicine apps based on ratings, authenticity, sentiments 62

4.2 ANOVA analysis of popularity 65

4.3 Rank sum of trust 66

4.4 Statistics of trust 66

4.5 Rank sum of anger 67

4.6 Statistics of anger 67

4.7 Rank sum of anticipation 68

4.8 Statistics of anticipation 69

4.9 Rank sum of disgust 69

4.10 Statistics of disgust 70

4.11 Rank sum of fear 70

4.12 Statistics of fear 71

4.13 Rank sum of sadness 71

4.14 Statistics of sadness 72

4.15 Rank sum of joy 72

4.16 Statistics of joy 73

4.17 Rank sum of joy 73

4.18 Statistics of joy 74

4.19 Rank sum of vader scores 74

4.20 Statistics of vader scores 75

4.21 Rank sum of emotional tones 75

4.22 Statistics of emotional tones 76

4.23 Rank sum of authenticity 76

4.24 Statistic of authenticity 77

4.25 Rank sum of words per-sentence 77

4.26 Statistics of words per-sentence 78

4.27 Rank sum of big words 78

4.28 Statistics of big words 79

4.29 Clustering Result of Applications 79

4.30 The first cluster of Apps 80

4.31 Example App in Cluster 1 81

4.32 Ratings and Reviews of the representative app in Cluster 1 82

4.33 The second cluster of applications 83

4.34 Example application in the second largest cluster 84

4.35 The third cluster of Apps 86

4.36 Example application in the third largest cluster: HMH NOW 87

5.1 Reconstructed images with 10 (left), 100 (middle) and 1000 triangles (right) from scratch. As more triangles are added, the reconstructed image resembles more like the original reference (input image). 93

5.2 Original (a) and protected classifier (b) without any attack. 95

5.3 Attacks to original classifier without any defender (a), vanilla attack to the classifier protected by the defender (b) and strong attacks to the classifier protected by the defender (c) 96

5.4 Reconstructed distortion 97

5.5 distortion summary. 98

5.6 triangles summary. 99

5.7 Strong Attack 99

List of Tables

2.1 Summarizing analytic systems in literature. 15

3.1 Summarizing of collected applications 33

3.2 Lifestyle 43

3.3 Nutrition 45

3.4 Meditation 47

3.5 Workout 49

3.6 Prescription 51

3.7 WisCom, quality metrics comparison I 52

3.8 WisCom, quality metrics comparison II 55

4.1 Three example reviews from the application QuickDr 85

4.2 The only review from the application HMH NOW 88

5.1 Summary of accuracy under BPDA attack 100

Chapter 1

Introduction

In this dissertation research, I proposed a novel deep learning model based framework to analysis the user feedback for business intelligence. I will introduce the background knowledge of semantic and sentiment analysis approaches, deep learning models for nature language processing, and the robustness risk of the deep learning model itself. Most importantly, this Chapter highlights the motivations for the three research thrusts: semantic analysis of user feedback for health application design, sentiment analysis to understand public opinion on telemedicine applications, and the robustness issue of the deep models are used.

Specifically, this chapter is organized as follows. I first introduce the background of health and fitness mobile application industry and the challenge of application design leveraging user feedback in Section 1.1. Then I elaborate on the motivations of study the public opinions on telemedicine applications and basics of sentiment analysis approach in Section 1.2. Next I share the motivation for research on the robustness of deep learning model in Section 1.3. After that, Section 1.4 concludes the contributions of this dissertation research. Last, but not least, I show the organization of this dissertation in Section 1.5.

1.1 Design of Mobile Application Driven by User Feedback

The mobile phone application market is highly competitive and has grown into a multi-billion-dollar industry. The app marketplaces such as Google and Apple app store host millions of apps that fiercely compete with each other. The popularity of these apps follow a typical power law distribution: While more than 95% apps are downloaded by fewer than 1,000 devices, few apps receive over a million downloads [70]. A recent study [116] by Gartner

supports this trend as it predicts that less than 0.01% of consumer mobile apps would be considered a financial success in the future. Therefore, it is important for app developers to understand the characteristics of a successful mobile apps which could draw consumer's attention.

There are several reasons that contribute towards the wide adoption and usage of a mobile apps or their failure. For example, poor design, lack of functionality, poor aesthetics, etc. Consumer reviews posted on app distribution sites such as google store could also influence a prospective user's opinion about an app as they provide good insights into various strengths and weakness of apps. Unlike other digital products such as music and movies that are often sold as finished products, mobile apps as software products offer developers an opportunity to integrate customer feedback received on apps belonging to similar genre into various design stages of apps for improving future app functionalities. These app reviews also provide guidance for developing new apps with desirable features.

Recent studies have reported that later app releases, which typically have improved design features and functionalities, tend to have greater influence on an app's success in terms of higher sales performance than early releases [66]. Improvement in app features could benefit from deeper understanding of specific customer needs as analysis of these mobile app reviews could provide guidance for developers to improve various app features.

Mobile app reviews are great source for text mining due to the short length of each comment, limited scope and large quantity of data available. It also offers tremendous insights into a product. However, analysis of reviews text is still a challenging task for various reasons. First, information of bug/issue report, overall user experience or requests for new features are usually mixed in the data set of user comments [77]. Second, labels of the data are usually not available. Even a data set of apps from one category has been labeled, it do not share the same characteristics of another, therefore, can not be reused for analysis new category. It is also difficulty to processing and analysis comments manually

by developers due to the huge number. Given the reasons mentioned above, an automatic unsupervised framework able to extract detailed semantic information should be developed.

Design of a successful software product is a core research topic in quality assurance and various metrics have already been proposed to promote the software quality [41]. Although those general metrics of software quality doesn't provides specific know-how, it inspires the developer what type of information should be looked into when analyzing user feedback. To extract useful information from reviews two approaches are often used. One approach is applying topic models such as Latent Dirichlet Allocation [14] on review text and summarize the topic of each clustered comments. Another approach is applying pre-defined part-of-speech rules [58] on review text of each app, and counting the most frequent term to emerge app features that mentioned the most by users. Those two approaches are rely on the statistical assumption of words. Most competitive analysis, summarizing, classification tools of mobile app reviews in the literature are developed on those two approaches. But extracted words may hard to interpret its semantic meaning when taken out of context. The recent rising of deep unsupervised learning model lifts the nature language processing to its new level. Pre-trained model such as BERT [29] now are able to encode the words according to the context. It also assign semantically similar sentences to close vectors very well. The progress allow us to build a new system that more sensitive and better understand user feedback and eventually bring more design insights.

Therefore, this study has three major objectives. One is to review prior research related to mobile app design, including 1) the challenges and metrics of successful mobile app design and development, 2) how design insights are reflected by the pattern of user reviews on certain app or market 3) the strategies used to build analysis tools to help developers. The second objective is to compare and contrast popular and unpopular features among a certain category of apps and identify the determinants of successful mobile app design. We use the health and fitness category in Google Play Store as an example in this study, but the framework can be generalized to any category of mobile apps. The third objective is to

generalize this process to a new system that be able to frames app design guidelines when scanning reviews of certain apps, eventually help developer avoid risks and prioritize the development of most important function of similar apps. Those guidelines can include very advance strategies of marketing. For example, influencer marketing can be used to attract more users. Gamification strategy can be used to improve physical exercises of fitness app users. Currently, few study in the literacy has extended focus to marketing, and the complete guidelines of a success mobile app should go beyond merely the quality of itself.

1.2 Sentimental Analysis on Telemedicine App Reviews

Telemedicine had become a remarkably important healthcare tool amid the threatening COVID-19 pandemic. Healthcare and medical services are challenged by the COVID-19 in many aspects.

- One major challenge, for instance, is the relative shortage of medical and healthcare professionals caused by COVID-19 pandemic. The spiked number of infected cases was beyond the normal capacity of most medical facilities. COVID-19 created further shortage of medical services coupled with crashed medical systems when panicked patients rushed to hospitals in the early stage of the break-out [127]. This trend even occurred in those countries with high availability of medical facilities, advanced technologies and large pool of healthcare professionals. Many countries, such as the U.S., Germany, France, Peru, Brazil, Japan and Lebanon, have experienced that a growing number of medical systems are under pressure. For instance, it has been reported that one third of hospitals in the U.S. were running short of medical staffs [60]. Since abundant medical resources and staff members are diverted from their regular activities to test and treat COVID-19 cases, the shortage will not be eased any time soon.

- A second challenge include the risk of hospital transmission and limited access to medical facility due to the side-effect of some efforts to mitigate the spread of COVID-19, such as mandated lock downs or travel restrictions [92]. Medical facilities tend to cause further spread of virus, where frontline health workers required close personal exposure to patients are at high risk of infection [13]. Besides, people now avoid accessing healthcare providers due to the risk of exposure instead of rushing hospital earlier.

All those aforementioned reasons promote the adoption of telemedicine as a viable venue to address the medical needs during the pandemic.

During the course of the COVID-19 crisis, telemedicine innovations implemented before and after the pandemic demonstrated the genuine values response to disasters and public health emergencies. Telemedicine is defined as technologies and devices that are able to remotely deliver health-related services and medical care to patients [27]. Telemedicine systems collect rich information about a patient's health status to aid in deciding if there is a need for health workers to intervene. Historically, the advantages of telemedicine on rural settings are now leveraged by patients who self-quarantined at home with fears of exposure. Averting close contacts, the telemedicine reduces the risk of cross-contamination in medical facilities. Forward triage, the sorting of patients before their arrivals in an emergency department, is a traditional strategy for healthcare surge control. Telemedicine as a digital approach to forwarding triage allows patients to be efficiently screened [54], thereby conserving healthcare resources and alleviating the shortage of medical resources. After the pandemic, a raft of new telemedicine tools for remote COVID-19 diagnosis, monitoring and management have emerged [75] to address the crisis. The widespread of commercial wearable devices and smartphone-based platform can be used for continuous monitoring of individuals for contact tracing [37]. Equipped with special hardware such as inexpensive electrochemical

sensors, telemedicine systems may offer ample information on infection status, immune response, inflammatory markers and metabolic markers, bolstering the accuracy of COVID-19 diagnosis [75].

A variety of factors become impediment to the adoption of telemedicine in the battle against COVID-19. Since the telemedicine does not allow the same closeness for a physician to obtain details as revealed by physical medical examination, telemedicine may only be deployed as supporting tools. Video or phone calls as remote visits may not be sufficient to formulate correct diagnosis or even worse mislead the diagnosis. In such a case, the risk of diagnostic or therapeutic errors is high and may result in malpractice claims [97]. Consequently, the concerns on privacy, regulatory and insurance coverage increases, and research on the efficacy and quality of care delivered remotely are not enough. A comprehensive review on ethical, legal and social issues has be conducted by Bonnie Kaplan summarized 11 issues including quality of care, consent and autonomy, accessibility to the care and technology, legal and regulatory, clinician responsibilities, patient responsibilities, relationships, commercialization of healthcare, policy, information needs, and the evaluation or assessment [59].

After a thorough literature review, we observe that there is the lack of studies on the factor of social acceptance from patient-centric views. Although the opinions of patients on those issues of telemedicine are not always consistent with policymaker, the public sentiment on telemedicine is rarely explored. The widespread of mobile devices and applications today make telemedicine apps feasible to a wide range of users. The comments left by users in an app store website are a powerful source archiving user opinions. Therefore, we propose a sentiment analysis scheme focusing on telemedicine application reviews (see Chapter 4) to study the public sentiments on telemedicine systems. Sentiment analysis, namely opinion mining, is one of the most essential and technological underpinnings in the field of nature language processing. Sentiment analysis solution analyzes the mood, emotion or feeling of a text toward certain objects, such as product, services, organizations, individuals and events [62].

The sentiment analysis techniques have been applied to various fields, including products reviews, e-payment services [3], politics [6], chat systems [91] and social media to support decision makings, improve customer satisfaction, and detect cyberbullying. Sentiment analysis tasks vary in granularity (e.g., document level, sentence level), dimension (e.g., polarity, emotions), and target (e.g., given certain target, abstract by itself) [130]. Early developed methods heavily rely on manually crafted features, including rule-based (e.g., dictionary) and machine-learning-based (e.g., SVM) schemes [124]. In recent years, sentiment analysis techniques, anchored on deep learning models with machine learned feature, demonstrate high performance and accuracy [124].

1.3 Robustness of Deep Learning Model

The deep leaning models are widely used by many researchers and in many analytical frameworks, including frameworks that are proposed in this dissertation. Despite the remarkable success of machine learning models utilizing deep neural networks in solving a variety of problems including image classification [61, 109, 53], object detection [43, 103, 131], semantic segmentation [74, 21], visual concept discovery [119], it has also been shown that these models are highly vulnerable to small, carefully chosen modifications to the inputs [113, 45], known as adversarial examples. Researchers have demonstrated that the adversarial examples can be generated easily by using optimization methods to find perturbations that maximize the loss of the network [45, 19] and appears in a wide range of different model architectures, even in other machine learning algorithms ([93] and [94]). Besides, the change of input is usually imperceptible to a human. This phenomenon also raises legitimate concerns on the apparent generalization capability of deep neural networks, which is its key success factor. It also raises an increasingly important security issue as deep neural networks are being deployed in life-critical missions such as self-driving cars [20] and medical diagnostics [25] since such attacks can be easily found in reality [4, 38].

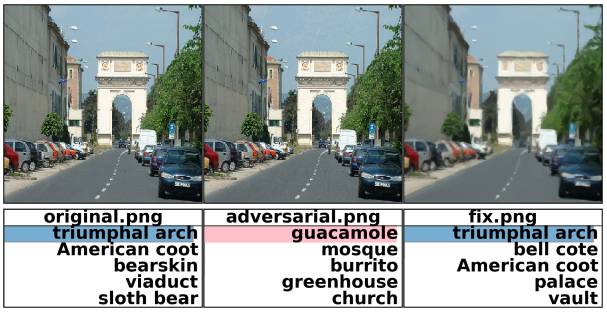

Figure 1.1: Adding slight perturbation to the original image (left) we easily get adversarial example (middle) mistakenly classified as guacamole. After primitive-based transformation that reconstruct the image by composing of colourful triangles (right), the perturbations are suppressed and the image is classified correctly again.

In response to the susceptibility of neural networks to adversarial examples, a lot of efforts have been made in detecting adversarial examples [85, 35, 123] and building robust systems, including model-specific defenses [76, 26, 69], model-agnostic defenses [32, 50, 122] and adversarial training [63, 107]. A model-agnostic defense usually uses a separated step that does not need to be adapted to a certain model and applies transformations to the input as a preprocess before being fed to the original model. Unlike model-specific defenses, the model-agnostic defenses do not make strong assumptions about the nature of the adversary therefore naturally support Kerckhoffs principle: the attacker should be allowed to alter itself to circumvent defense and security should not depend only on specific model. Unfortunately, despite lots of research mentioned above, no method completely solves the adversarial robustness issue. Work towards building strong defense requires estimations under strong attack, and recent research shows many of them are not robust under strong attacks such as BPDA [7].

We notice that all the previous model-agnostic defenses under strong attack method BPDA share one common feature. The difference between the input image and the image after transformations is small. It is easy for attackers to estimate the gradients and penetrate the defenses. The failure of small-difference transformations raises a legitimate question as to whether robustness of model-agnostic methods improves if we conduct a large difference transformation.

1.4 Contributions of the Dissertation

This section summarizes the main contributions of this dissertation research. Three technical underpinnings are embraced and the contributions of each are articulated in Section 1.1, Section 1.4.2, and Section 1.4.3, respectively.

1.4.1 Contributions for Mobile Application Design Driven by User Feedback

In this work, I proposed a novel framework to analysis the user feedback of health mobile applications, and converted the feedback to guidelines for mobile application design. The contributions of this work listed as follows:

1. A new analytic framework to utilize the wisdom of the crowd for mobile application design introduced.

2. A new model to understand the semantic meaning of review comments according to the context. The new model leveraged the cutting-edge AI achievements for rich and insightful understanding of the content.

3. Discovery of critical issues to the success of the applications in the market after instigation the topics discussed by the users.

1.4.2 Contributions for Analysis on Telemedicine Application Reviews

In this work, I proposed a network approach to analysis the public opinion on telemedicine applications via sentiment analysis, and contrast the different responses on different types of telemdicine applications. The contributions of this work listed as follows:

1. A new network analytic approach that utilize the deep learning model BERT and cluster applications according to the similarity measurement of reviews.

2. Discovery of the features of fake positive reviews after features of application clusters with different sentiment responses investigated. This will further be developed into a novel effective detection system.

3. Summarizing of the public opinions on telemedicine applications.

1.4.3 Contributions for the Robustness of Deep Learning Model

In this work, I investigated the robustness of deep learning model under the task of computer vision and how to mitigate the robustness issue of the deep learning model. The main contributions of this work listed as follows:

1. A new model-agnostic defense against strong adversarial attack based on a large-difference-transformation approach, which provides a new promising direction towards a complete solution of adversarial robustness.

2. Empirical experiments show that the primitive-based representation used in our transformation can achieve state-of-the-art robustness under BPDA attacks. The normalized l_2 distortion required to fully break the classifier on the ImageNet dataset is increased from 0.06 to 0.12.

1.5 Organization of the Dissertation

This dissertation is organized as follows. Chapter 2 comprehensively and extensively reviewed the previous related work for the three research components in this dissertation, including the semantic analysis approaches, sentiment analysis approaches and the current research on the robustness of deep learning models.

The Chapter 3 described the detail of the proposed semantic analytic framework, in which transformer-based neural network is used. I also discussed the result of applying this framework to the collected reviews from health and fitness applications, and concluded several key issues for application in different subcategories.

In Chapter 4, I described the framework of network approach which are use in study of the sentiment of the reviews from telemedicine applications, and the experimental result. I presented the difference of the response on the different types of applications and summarized the public opinion. A new fake positive review detection was also proposed for further work as the outcome of our discovery of the difference.

In Chapter 5, I dived into the important research issue of the robustness of deep learning model, which is the underpin of many approaches and framework include the ones described in Chapter 3 and Chapter 4 in this dissertation. My novel defense approach is expected to achieve the state-of-the-art result to mitigate the adversarial problem, which is demonstrated in the experiment in the chapter.

The last chapter (Chapter 6), I concluded this dissertation with major research contributions and discuss future important research directions from various perspectives that have not yet been fully addressed.

Chapter 2

Literature Review

As the value of mobile application industry grown into a multi-billion-dollar, the volume of customer reviews also snowball to an unprecedented level. Although the reviews of applications are the great source for text mining, the rich sentiment and semantic information embedded in the text data raise the difficulty for building an automated analytical system. The fake reviews in the data and the unrobustness of the deep learning model makes it even more challenging. In this chapter, I review various of previous studies that closely related to the dissertation.

Specifically, this chapter includes following topics. Section 2.1 I reviewed the previous topic modeling techniques used in the text mining, which is usually the first step of analysis. Then Section 2.2 I reviewed the previous the previous automated semantic analytical frameworks. After that, I reviewed the sentimental measurement research in Section 2.3. Finally, I reviewed the reported robustness issue of deep learning model and current research on this issue.

2.1 Topic Modeling and Deep Learning Model

Topic modeling is a task of finding the topics of the given documents belongs to. Although it is similar to the task of clustering, there are several subtle difference between two tasks. First, a document could belongs to multiple topics, therefore, topic modeling is a soft clustering problem. Second, the primary objective of topic modeling is identifying the hidden topics behind the collections of the documents, while the accuracy of the assignment of each document to the correct topic is only a secondary objective. Last, the process should also generate descriptive words representing the content of the topic, which are usually keyword

list. I reviewed the most popular topic model in Section 2.1.1 and then the most recent deep learning model in Section 2.1.2.

2.1.1 Topic Modeling Technique: LDA

For topic modeling, Latent Dirichlet allocation (LDA) is a popular generative statistic model used by many researchers [117, 135, 22, 51, 40]. LDA is a generative statistic model follows several assumptions. It assumes each document is a bag of words from one or multiple topics. A word can belong to different topics with different probabilities and also within a topic a word can be used with different probabilities. Eventually, each document belongs one or multiple topics with different probabilities. Those assumption allows LDA handle the ambiguous words very well. Mathematically, LDA assumes that in certain position of certain document, the choose of the topic and the choose of the word associated to the topic follows multinomial distribution. To simplify the calculation, it further assumes the Dirichlet distribution as the conjugated prior of the multinomial distributions. The estimation of the parameters of the distribution can be easily conducted by the Gibbs sampling, which makes LDA an efficient approach used by many researchers.

2.1.2 Deep Learning Models

Despite the successful performance and widely used by the researcher, LDA oversimplified the language model as bag of word. Besides, LDA also dose not consider the effect of context. As the recent development of deep learning model, pre-trained model such as BERT [29] now are able to encode the words according to the context and reached unprecedented performance in many NLP tasks. To overcome the structure limit of BERT model and reduce the time of finding similar pair of sentences, Sentence-BERT (SBERT) [101] was proposed by Reimers *et.al.* SBERT is a modification of the pretrained BERT network that

use siamese and triplet network structures to derive semantically meaningful sentence embeddings. The output of SBERT be compared using cosine-similarity. SBERT is used in our novel framework as a state-of-the-art sentence embeddings methods.

2.2 Analytic Systems for Customer Reviews from Mobile Apps

Texting mining has been extensively used for developing insights from customer reviews. while a plethora of research on customer reviews focuses on traditional text mining approaches, more recent research is utilizing deep learning based text analytic models. We performed extensive review of extant literature and noticed that a majority of studies could be classified into one or more of the five groupings based on their study objectives as summerized in Table 2.1. These are (1) feature extraction, (2) classification, (3) grouping, (4) ranking and, (5) summarization. We will, first, briefly discuss important research done in each of these five areas and subsequently describe deep learning based approaches as applied to understand customer reviews.

2.2.1 Feature extraction

A majority of research on feature extraction focuses on identifying one or more important features. There are typically product-based features(e.g. product type [73]), lexical features(e.g., part-of-speech [79]), linguistic features (e.g., review length [71]), social features (e.g. number of views [68]) or sentiment features (e.g., likes or dislikes [40]) Those features can be divided into the following major categories: (1) product features: the names and class of product or component mentioned by reviews. (2) lexical features(keywords): the frequency difference of terms in different classes of reviews, including some distinctive words (smoke words) in the reviews. (3) linguistic features: the styles and characteristics of the vocabulary and sentence used in reviews. (4) social features: the attention received on social media and social attributes of users profile. (5) sentiment features: the emotion express

No.	Year	Author	feature	classification	clustering	ranking	summerization	inference	Key Findings
1	2022	Goldberg et al.	√	√		√	√		Smoke words extracted from reviews of feature request, irritator, compliment categories are useful
2	2021	Verkijika et al.			√		√		Ease of use, usefulness, convenience are positive themes, customer support, recieved cost, lack of trust are negative themes
3	2021	Ha et al.	√				√		Form morphological matrix from extracted keywords provides new idea of innovation
4	2020	Liu et al.	√	√			√	√	(1) consumers are not sensitive to the price of elderly phones, (2) but sensitive to the price of other smartphones, (3) wide and thin phones are more competitive in size
5	2020	Zheng et al.		√	√		√		(1) use three filters, namely the sentiment filter, the component-symptom filter and the similarity filter, to select informative threads (2) identifies the threads related to product defects and provides detailed defect information including defect types, defective components and defect symptoms
6	2019	Liu et al.	√	√		√	√		(1) seeding words can be expanded into domain-specific sentiment lexicon (2) identify comparative text and competitive product from forum, and aspect comparison information from another pre-categorized source
7	2019	Chen et al.	√	√	√	√	√		Suggestions ranking high by both count & rating have a higher probability of improving the upgrade
8	2019	Dalpiaz et al.	√	√			√		High review volume may reduce false positive case of feature extraction, human analytical skills
9	2019	Shah et al.	√	√			√		Categorizes review sentences into feature evaluation, bug report and feature request
10	2018	Malik et al.	√		√	√			Similar app represented in similar feature tree and be compared
11	2018	Liu et al.	√	√					Essemble learning of bagging can improves performance of identify product complain thread
12	2018	Zhu et al.	√	√	√		√		5 students evaluation show the designed system very concise and usefull to retival sentences
13	2018	Marcacini et al.	√						(1) providing a unified representation of feature spaces between different domains through heterogeneous transductive networks (2) using a cross-domain transfer learning process to propagate label
14	2018	Shah et al.	√	√					Classified into Praise, Feature Evaluation, Bug Report, Fearture request, other. The simple CNN model has comparative result to Max Entropy but much slower
15	2017	Singh et al.	√	√				√	Devideded reviews into behavior, form, function, service. for android phones, fucntion and form are positively related to ratings and behavior and services are negatively, decision tree J48 have good performance
16	2017	Johann et al.	√	√					Unfiltered user reviews reached average precision of 24% a recall of 71%. Features extracted from user reviews are still noisy but catch majority features discussed by the users.
17	2017	Di Sorbo et al.		√		√			Summarize app reviews and generate an interactive, structured and condensed list of recommended software changes by classification of intention and topic rank
18	2017	Li et al.	√	√				√	A deep learning-based approach for understanding and predicting users' rating behaviors unifying aspect ratings and review contents show good performance in rating prediction
19	2016	Flory et al.	√		√	√			Helpfulness is defined as relavance to customer search
20	2016	Shah et al.	√		√	√	√		Extract feature and sentiments from user reviews
21	2015	Maalej et al.	√	√					Naïve Bayes better than decision tree and Max Entropy and keywords baseline method, sentiment score is important, 4 binary classification better than 1 multi-classification
22	2015	Gu et al.	√	√	√		√		SUR-Miner provides reliable results on review classification, aspect-opinion extraction, and sentiment analysis, with each average F1-scores of 0.75, 0.85 and 0.80. SUR-Miner more focus than AR-Miner
23	2014	Li et al.	√	√		√		√	Combine all the feature generate the best product portfolios support by amazon best seller result
24	2014	Chen et al.		√	√	√	√		Topic modeling by LDA shows better performance than ASUM
25	2014	Guzman et al.	√		√		√		The extracted features were coherent and relevant to requirements
26	2013	Zheng et al.	√	√	√		√		(1) the social features of reviewers improve classification results (2) classification affected by product type due to the different purchase habits of consumers (3) reviews are contingent on the inherent nature of products, such as search goods or experience goods, digital products or physical products
27	2013	Fu et al.	√		√		√		The top-3 complaints around the same issues: content attractiveness, stability, and cost

Table 2.1: Summarizing analytic systems in literature.

by the words, sentences or entire document in reviews. Correlation-based feature selection (CFS) is the method most used by many researchers to select important features [72, 71].

Feature extraction methods can be loosely divided into predefined rule based and corpus statistics based. Lexical rule-based method is the most common rule-based method. Among the 27 papers we reviewed, 13 of them [52, 24, 28, 106, 79, 80, 105, 58, 39, 104, 49, 68, 51] apply lexical rules to extract words and phrases. Lexical rule based methods first tag the text by part-of-speech labels via mature nature language toolkit, and then apply a library of predefined patterns to extract words of interest. Some research [49, 105]build additional tree or graph by lexical rules to preserve structural information extracted from the text. Other research combine lexical rule with predefined keywords dictionary to enhance the performance of specific mining tasks. For example, Liu *et al.* [73] build an aspect dictionary, a product name dictionary and a comparison keywords dictionary first, then use Jaccard coefficient to match text for comparative information mining.

Statistics-based extraction methods explicitly use statistics of the corpus linguistics to aggregate opinions from reviews. In those method, the text are treated as bag of words (BOW). For example, number of characters, sentences, exclamatory sentences, interrogative sentences, adjectives, verbs and other linguistic statistics of the text are engineered by Liu *et al.* [71] as features of the reviews. Intuitively, the most frequently mentioned term in certain category of reviews is usually the focus of the text of this category. Goldberg *et al.* proposed the CC score algorithm to extract smoke terms [44] from reviews as features. The CC score algorithm assesses the relevance of terms based on how often they occur in relevant versus irrelevant reviews.

In practice, many proposed text extraction process combined both rule-based and statistics-based method together for better performance. Lexical rules are often used together with frequent mining techniques such as item-set mining [104, 51] to extract words combination of frequent concurrence. The success of simple library of lexical patterns such as proposed by Johann *et al.* [58] shows that, although not explicitly, the lexical rule method itself implicitly

depends on corpus statistics. Chen *et al.* [24] applies TF-IDF criteria to check the relevance of terms extracted by lexical rules. A keywords dictionary with initial candidate words are later expanded by adding domain-specific words according to some statistics of the corpus. Liu *et al.* use pointwise mutual information (PMI) to measure the relavence of co-occurring words and therefore expand the seeding dictionary [72].

As the recent rising and widely using of neural network, many neural network word encoder can embed words into feature vectors and the coordinates of the vector in the space represents relative semantic relationship between the words. Li *et al.* [67] proposed bidirectional long short term memory network (Bi-LSTM) as the word encoder for later rating prediction task. Then the system use another neural network take both aspect ratings and vectors of each words can accurately predict the overall ratings.

The feature engineering of sentiment analysis is similar. The lexical rule based sentiment analysis tool such as SentiStength is widely used by researchers. Early researchers [40, 79] are using linear regression and customized keywords dictionary to map words into sentiments. Gu *et al.* [49] suggest to use Deeply Moving, an advance deep neural network model to get sentiment feature from text. Since the ratings come along with the reviews text are often available, some researchers take an aspect extraction approach to analysis the sentiment. The task becomes extraction of the aspect the review focuses on and the ratings are used as the measurement of sentiment. Marcacini *et al.* [80] proposed an innovative method that connects the labels of text between different domains through heterogeneous transductive networks, and then transfer the knowledge from knowing domain to new domain to extract the aspect of the text.

In some proposed system, feature extraction is not executed as the first step. Zhu *et al.* [135] conducted topic modeling first, and then encoded whether a review belongs to a cluster as a feature since the tip-ming system does not limits its interest in certain domian. In the case of Liu *et al.* [73], the extraction of name of the entity conducted simultaneously when conduct rule-based classification of reviews to comparative text or not, because this

task is specific tailed for comparative text mining and rule-based method has been proved effective. It depends on the task and goal of the information retrieval system.

2.2.2 Classification

The classification step can serves many purposes for the system. Classification could servers as a filter to remove non-informative or non-relevant reviews according to the design of the system. The task could be binary classification of whether the review contains comparative text [73, 72], whether it contains requirements for update [24], whether it discusses the certain topic [135, 30] or certain product [58], whether it is useful and informative [134, 22] and whether it is a product defect complaint [133, 71]. The filtering process may contain several round of filtering, and some filtering could simply based on similarity of document, which is measured by doc2vec and cosine similarity. After that, the system can continue process only the relevant message and reduce the noise of the result.

Classification could also servers as an initial step to identify the intention of each review. The task is classify the reviews eventually into several category such as feature request, bug report, feature evaluation [106, 110, 77, 49]. To answer a specific question, a certain category of review from this classification is very useful. For example, to answer which part of the apps are loved by user, the answer should be found in the feature evaluation category. Sometimes, the catalog of the products under review is available. Therefore the labels of category can be directly used for associated reviews and determine the scope of analysis [44, 68, 72].

Rule-based methods are basic technique to automatically categorize a user review. Maalej *et al.* [77] use a list of predefined keywords to classify reviews by their intentions. Liu *et al.* [73] use a dictionary of comparison keywords combined with lexical rules to classify whether the review contains comparative text. It determines a comparative text and extracts the name of entity by lexical rule at the same time. Chen *et al.* [24] build a dictionary of terms from app update record and use the keyword of the dictionary to classify whether the review is user requirements to update. Johann *et al.* [58] build a dictionary of terms

from app descriptions and use the keywords of the dictionary to classify whether the review discuss the app features. Zhu *et al.* use Fisher test to build a dictionary of tokens that have a significantly higher frequency in the reviews of the business than in any of its similar other bossiness. business [135]. Some rules of classification are based-on manually crafted scores. Dalpiaz *et al.* [28] define the feature performance score of a feature of the app by the percentage of the product of ratings and number of reviews that mentioning the feature in sum of all products. The scores then used to classify the reviews into one of the four category of SWOT matrix. Li *et al.* engineered a total score of importance that composed of scores of fitness, expertise, influence, rating of reviews and reviewers to classify whether the review is important or not. [68]

Classification is a classic supervised learning task, many machine learning models can be applied on classification of the reviews. Naive Bayes is a popular binary classification algorithm which is simple efficient and require relatively small training set. Decision tree is another popular classification that assumes all features have finite domains and a single target feature representing the classification(tree leaves). More advance decision tree models includes random forest, J48, GBDT, XGBoost and AdaBoost. KNN use the majority votes from k-nearest neighbours to determine the class. SVM search the linear hyper plane that maximize the margin of the boundary of two class in training set, and use the hyper plane to classify new input. Those methods are all used and compared in many research [133, 72, 110, 22]. The multinomial logistic regression, also known as Max Entropy is a popular multi-classification model that assumes a linear combination of the features and some review-specific parameters can be used to determine the probability of each particular review type. We notice lots of research [105, 77, 49, 106] report that Max Entropy has the best performance. Besides method mentioned above, Liu *et al.* and Zheng *et al.* [71, 134] use ensemble learning to improve the performance of classification further.

2.2.3 Grouping

After classification, similar reviews can be further grouped into the same sub-categories by cluster analysis. After converting reviews text into document vectors by TF-IDF, many classic clustering method such as k-means method can be applied. Flory *et al.* [39] proposed a new k-means method (DPSO-KM) based on discrete particle swarm optimization to speed up the clustering. Affinity propagation is another available clustering algorithm if the similarity between any two reviews are well-defined. Chen *et al.* [24] proposed similarity measurements between two reviews from three aspects. The author first combine the rating difference, the intersect-over-union of words and the cosine similarity of document vectors to measure the similarity of all pairs of any two reviews, and then use the affinity propagation to cluster reviews.

However, a document text may associated with multiple topics, and if our purpose is to discover the hidden topic among the reviews, each containing a set of associated keywords, then the topic modelling algorithm server the purpose better than the cluster analysis. For topic modeling, Latent Dirichlet allocation(LDA) is a popular generative statistic model used by many researchers [117, 135, 22, 51, 40]. Compared to other topic modeling algorithm such as Aspect and Sentiment Unification Model(ASUM), Chen *et al.* [22] shows that the LDA has better performance for all topics in terms of F-measure. Since LDA delivers the percentage of each review that allocated to all topics, the percentage can be used as features of each review. Therefore, Zhu *et al.* [135] suggested run LDA first and then combine the percentages features before the classification step. Zheng *et al.* [133] proposed a novel probabilistic graph model to discover the product defect types from reviews. Similar to LDA, this customized probabilistic graph model are based on Dirichlet distribution. It model three types of words (general, component and symptom) from certain number of product defect types and show better performance due to its refined design.

Besides on document-level, clustering is also applied on token-level. Users can use different words to refer to the same feature, therefore the extracted keywords from previous

steps, need to be grouped before further analysis. Unlike review text which consists of multiple sentences, the extracted keywords and phrases are often combinations of few several words. Therefore, researchers [49, 51, 104] suggest using item-set mining algorithms and support counts to efficiently discover important occurrence of words first due to short lenghth, and then refer Wordnet as synonym dictionary to group them by certain rule. For example, if any two collections of words share the same synonyms in the dictionary, and the word is also the more common word used in collections, then the two collections are grouped together.

The app recommendation system proposed by Malik *et al.* [79] takes a different approach. It groups extracted features from the same app together as a tree. Feature space of an app is often hierarchically structured, therefore the features from the same app can form a tree by formal concept analysis (FCA) of their relationship. Since the main task of the app recommendation system is comparing the similarity of apps, similarity of features are not compared directly. After assembled, features are compared as a whole in form of tree similarity rather than compared as individual features.

The ensemble learning also can gain better performance from the regroup of the features. Zheng *et al.* [134] use Independent Component Analysis (ICA) to transform the original feature space into two mutually independent projection spaces and then used for co-training of ensemble learning. The ensemble learning exhibits high generalization performance when the average error rate of component classifiers is low.

2.2.4 Ranking

In this step, the results are listed and sorted by relevance and importance. The simplest method is ranked each group of reviews by their volume. [28, 68] Other complicated scores that estimate the relevance or importance are developed for different tasks. Scores engineered by researchers are mostly derived from three sources. The first source is the linguistic statistics of the text. Flory *et al.* [39] extended the web-based similarity kernel (WSK), which measures similarity of two text based on co-existing term frequency between them, to

score the relevance between review and customer input. The competing app analysis tool proposed by Shah *et al.* ranks the extracted features from reviews by their support count and ranks the competing apps by their number of common features shared with the baseline app. Di Sorbo *et al.* [30] proposed a sentence score that based on both the ratio of normal and frequent words related to topics appearing in the sentence with respect of the number of total words in the sentence to measure the relevance. Goldberg *et al.* [44] first defined the number of reviews in the top N-ranked set that refer to true instances as precision, and then proposed the Tabu heuristic algorithm that maximize the precision by adding or removing smoke term in the list to score the relevance best.

The second source is the rating values provided by the reviewers. The user requirement mining system proposed by Chen *et al.* [24] assumed that the issue with high number of reviews (volume) has more impact on user and low average ratings are more urgent to address, therefore the scores are the ratings divided by the volume. AR-miner [22] use group scores base on similar design but also consider the time window and adapted it for time serials data.

Sentiment analysis results are the third source. Liu *et al.* [72] use Average Sentiment Orientation and Average Sentiment Score to rank the target product among the other competitive products. Malik *et al.* [79] construct tree by sentiment scores of the app features and use the weighted-tree similarity to rank and recommend similar apps.

2.2.5 Summarization

In the final step, the results are summarized in either text form or numerical form, with additional visualization and inferences drawn from the data. The simplest summarization in text form is the list of the extracted keyword in random or in the order from previous ranking step is due to the rich semantic information of the words [117, 44, 52, 40]. Chen *et al.* [24] use a detailed template to organize the extracted text and generate product upgrade suggestions. Dalpiaz *et al.* categorized the extracted keywords into the classic strengths,

weaknesses, opportunities, and threats matrix (SWOT) for competing analysis. [28] Some researcher use the technique from business and management. Ha *et al.* build quality function deployment matrix from extracted keywords and then use morphological analysis to provide innovated business idea. [52]

The numerical summarization includes binary label such as whether the review is useful or not [134], the value of product specification liked most by customer [68] or prediction of overall rating from user reviews [67]. Guzman *et al.* list the topic of high-level feature with their sentiment scores. [51] Liu *et al.* first use total number of reviews that one product is better than another product on each aspect to construct competitiveness index and then use mutual information between competitiveness in specific aspect and the overall to find the key contribution aspect. [73]

The numerical result often can be visualized in plots. Shah *et al.* use bar chart of sentiments to compare competing apps on features. [104] Shah *et al.* use the size of the icons to represent the frequency and coordinates to plot metrics of app features. [106] Gu *et al.* use the line plots with respect to time to show the aspect trend and dot with size plot to show popularity of the feature aspects. [49] Chen *et al.* use radar chart to present the score of each aspects of apps. [22] Fu *et al.* use ternary plot to show the scores on top three aspects of apps [40]

To understand the effect of those numerical result, regression methods are used. Li *et al.* use linear regression to study whether written review or aspect ratings affect the overall rating and helpfulness most. [67] Singh *et al.* use linear regression to explore effect of review category on ratings [110] Liu *et al.* uses the quantile regression model to explore the effect of product attributes on the corresponding key competitiveness of products. [73]

2.3 Methods of Sentimental Analysis

In this section, we first describe representational models of emotions in Section 2.3.1. Then, we elaborate on the approaches to sentiment analysis proposed in the literature in Section 2.3.2.

2.3.1 Emotional models

Emotions are states of belief that caused by psychological changes. To depict the emotional states, the research community proposed various computational models to understand insightful meanings of emotions. All the computational models of emotions are divided into two different classes: categorical models and dimensional models. A categorical model only selects one emotion out of a set of emotions that best capture emotional status. A dimensional model, in contrast, allows quantitative values on multidimensional scales.

The two typical categorical models are the Ekman's basic emotion classes [33] and the Graesser's Domain Depended model [46]. The Ekman's model contains six basic classes, namely anger, disgust, fear, joy, sadness, and surprise. Graesser, on the other hand, argued that different field requires different set of basic emotion classes. For example, a five-category set of boredom, confusion, joy, flow, and frustration is suitable for the field of instruction and education.

The categorical model has an impressive advantage: it is easy to understand the model with distinct labels of emotion. Nevertheless, mandatory selection of one label is a problematic conceptualization - such a selection may lead to non-optimal emotion detection. For instance, it is arduous for one to pick a label when feeling is neutral. Therefore, dimensional models were introduced to address the issue. Plutchik *et al.* [121] proposed a model in a so-called *activation-evaluation space*. Two numerical values of angle and range of a wheel represent activation and evaluation of each emotion. Mehrabina's model [83] uses three-dimensional representation of pleasure, arousal, and dominance - PAD - to locate each

emotion state. Dimensional models assume each emotion can be defined as a combination of component such as arousal and pleasant, thereby making the representation flexible.

2.3.2 Methods of Sentimental Analysis

Sentiment classification is the most common task undertaken during the course of sentimental analysis. Classification techniques can be categorized into two types of models based on how the features of the input are build. In early time, most input features of a proposed model are handcrafted and selected. Models with handcrafted features are further divided into lexicon-based methods [1] and early machine-learning-based models. Lexicon-based methods are, by nature, rule-based methods following customized dictionaries of word-emotion relationship. Therefore, Mohammad *et al.* [86] emphasized the importance of large and high quality of the dictionary of lexicons, and they constructed the Word-Emotion Association Lexicon - NRC EmoLex - from a crowdsourcing effort with carefully designed questions to prevent malicious data entries. The EmoLex obtained annotations at sense level associated with eight basic emotions, namely, anger, fear, anticipation, trust, surprise, sadness, joy, and disgust. Unfortunately, the EmoLex indicates only coarse categories of affection without providing intensities of each lexicon. Therefore, the NRC Emotion Intensity Lexicon [88] of 10,000 entries for the eight basic emotions with 0 to 1 real value scaling was proposed and implemented later as an extension. Another important and intriguing lexicon dictionary [87], NRC Valence, Arousal and Dominance (NRC-VAD), was devised catering for dimensional representation of emotions.

Instead of being manually crafted, recent models with machine-learned features surged and were widely applied and substituted early machine-learning models. The early machine-learning-based models includes supervised model such as SVM, Random Forest, Decision Tree [1], and unsupervised clustering [36], to name just a few. Those methods are anchored on manual engineering of features, which limit the methods' wide applications. Recent machine learning models are powered by deep neural network such as CNN [65], RecNN [9],

LSTM [128], GRU [132], Deep Belief Network [57], Attention Network [111, 125], Bidirectional RNN [120], and Capsule network [126, 31]. Among those deep learning models, transformer-based models like BERT are particularly promising and viable. The transformer-based models yield significant improvements over numerous NLP tasks with context-aware embeddings. Although deep neural network models show significant performance advantage on use cases of predictions, the models are inadequate for interpreting results due to the black-box nature, especially compared with rule-based methods. Some researchers [100] argued that the attention mechanism of BERT statistically are focused on emotional words, but rationale behind of a target text being associated with certain emotions is not well studied at this moment.

Network analysis applied on natural language processing, such as text networks, is a popular non-supervised approach to offering various perspectives to explore the sentiment of text. Text network can be created where individual words are nodes and the edges between the nodes model the co-occur of both words in documents. Since each word may associate with certain emotion according to emotion lexicon or other emotion detection models, a two-mode network of both emotions and text can be built. Although a plethora of network analysis schemes have been proposed in the past [8], a holistic and explainable framework of both sentiment and semantic and their interplay were in an infancy in the literature. Further, network analysis of public opinions on telemedicine has not been yet conducted at this stage.

2.4 Adversarial Attack and Defense

Researchers has found that the unrobustness of deep learning models. Deep learning models are highly vulnerable to small, carefully chosen modifications to the inputs, known as adversarial examples. Many studies has been conducted on both attack and defense.

2.4.1 Adversarial Attack

In the following part of the subsections, we reviews different types of generating adversarial examples in the literature, including white-box adversarial attacks and black-box adversarial attacks.

White-box Adversarial Attacks

Gradient-based methods are popular white-box attack methods in the literature. Gradient methods use derivatives in order to achieve goal described in equation (1). The Fast Gradient Sign method (FGSM) [45] was the first introduced method after the phenomenon of adversarial examples first described in image classification networks. As a gradient-based method, FGSM seek to minimize the necessary perturbation and maximize the misclassification of correctly classified inputs based on the gradient of the cost function with respect to that input value. FGSM modifies the input in the direction of the sign of the gradient, multiplied by a hyperparameter constant:

$$x' = x + \alpha \operatorname{sgn}\left(\nabla_x L(\theta, x, y)\right) \tag{2.1}$$

The basic FGSM can be further extended to a more powerful multi-step variant. The method projected gradient descent on the negative loss function, known as PGD [78], is essentially one of those variants. Those can be summarized as below:

$$x^{t+1} = \Pi_{x+\mathcal{S}}\left(x^t + \alpha \operatorname{sgn}\left(\nabla_x L(\theta, x, y)\right)\right) \tag{2.2}$$

The Jacobian Saliency Map Algorithm(JSMA [95]) method use the gradient of network itself rather than the cost function. The gradient of the network with respect to the given input, also known as the Jacobian of the network is calculated first and an adversarial

saliency map is derived. The adversary can perturb the input where the model are sensitive to according to the map.

The adversary could also use other optimization techniques to solve the equation (1) directly. DeepFool [89] formulate a closed-form solution under the linearity approximation of the model. It projects input onto a linearization of the decision boundary in each iteration, and calculates the perturbation necessary to cause a misclassification.

$$\mathbf{x}^{t+1} = \mathbf{x}^t - \epsilon \cdot \frac{c\left(\mathbf{x}^t\right)}{\|\nabla_{\mathbf{x}} c\left(\mathbf{x}^t\right)\|_2^2} \nabla_{\mathbf{x}} c\left(\mathbf{x}^t\right) \tag{2.3}$$

Carlini-Wagner's attack [19] use a similar but easy-to-optimize variant:

$$\min\left(|x' - x| + \epsilon \cdot h(x')\right) \tag{2.4}$$

The parameter ϵ trades off the perturbation norm and the objective function $h(\cdot)$. The objective function can be chosen anything decrease as the miss-classification metric increase, however, for L_2 attacks author suggest using $h(x') = \max\left(\max\left(f(x')_{i \neq c(x)}\right) - f(x')_{c(x)}, -\kappa\right)$ for best result. The original author also suggests using $tanh(\cdot)$ to automatically box constrains $0 < x' < 1$ and use Adam optimizer to solve (8).

Black-box Adversarial Attacks

Many methods has also been proposed to craft adversarial example under black-box settings. Since gradient information is not available on black-box setting, a nature extension is to estimate the gradient from other sources of information. Indeed, attackers can estimates gradients based on confidence provided by neural network. The method ZOO [23] and [12, 114] numerically estimate the gradients and then conduct gradient descent. The SPSA was introduced to improve effectiveness by [115] due to the difficulty of optimization of loss surface. The stochasticity from sampling perturbations of SPSA allows it converge well in

even noisy objectives. The NES [56] effectively generates adversarial examples with a limited number of queries.

Attackers can also recover gradients from label results returned by queries to the neural network, as shown by [94]. A hard label attack even only require access to the $\arg\max_i f(x)_i$ output. The Boundary Attack [17] is a hard label attacks that performs a descent along the decision boundary using a rejection sampling approach.

2.4.2 Adversarial Defense

A slightly narrower problem of adversarial robustness is detecting whether an input is adversarial or not [122]. Some methods [48] are based on the hypothesis that adversarial inputs do not come from the same distribution as normal inputs. [35] detected adversarial examples by looking at the Bayesian uncertainty estimates of the input images in dropout neural networks and by performing density estimation in the subspace of deep features learned by the model. Some methods view detection as a separate classification problem. MagNet [84] use detector network to detect adversarial examples with large perturbation and pushes adversarial examples with small perturbation towards the manifold of clean images.

However, detection methods fall short of resisting adversarial noise in all circumstances. [18] defeats all ten detecting methods with varying degrees of success. The limited complexity of decision boundary of the model in practice could also cause adversarial example, supported by [16]'s work that incorporating RBF kernel that has highly-nonlinear decision boundary improves robustness. If detecting method can reach enough complexity of decision boundary to tell different high-dimension distributions, the original machine learning model should also did and no adversarial examples in the first place.

Improve adversarial robustness through better training is probably the most intuitive response to the prevalence of adversarial examples. [64] incorporates the adversarial calculation as part of the cost function of the model during training on more complex architecture

and large datasets like ImageNet. [78] rephrases the adversarial training as a minimax optimization problem with respect to robustness, and found adversarial training constituted robust optimization when applied to first-order adversaries like FGSM. [55] thinks that adversarial examples as entirely different category of input during training and augment the output class labels with additional adversarial label. [129] seek to withstand adversarial noise by modifying the final architecture of the model and adding an upper bound to the rectified linear unit (ReLU). [26] instroduce Parseval networks, which accomplish robustness by constraining the network's Lipschitz constant via layer-wise regularization. DeepCloak [42] identifies which extracted features vary the most between adversarial and ordinary input and delete these features if they are unnecessary and victim to adversarial noise. [63] improved the robustness by train the network against transferability on an ensemble of adversarial images generated from the trained model itself and other pre-trained models. However, again, adversaries still can circumvent these improvements by utilizing crafting techniques that are not sensitive to model's gradient [94].

Since many adversarial crafting techniques take advantage of the gradient information to find efficient perturbation, protecting and fortifying this information provides potential defensive robustness. Adversarial training effectively smooths the gradient of the model in the neighborhoods, which often referred as "Gradient Masking". [96] transfer learning from one trained neural network to the second one utilizing the soft label output of the first one. The defensive distillations masks the gradient of the output around the training points and effective at preventing white-box JSMA attacks. However, defensive distillation and other gradient masking and penalizing methods are vulnerable to black-box attacks [94]. Besides, an surrogate model without masking can be crafted and adversarial examples crafted from the surrogate can be transfer back to the targeted model as attacks.

Despite numerous techniques discussed in this section, no method for complete robustness to adversarial noise has yet been discovered. Adversarial robustness remains an open and significant research problem.

Chapter 3

Design Applications from User Feedback using Deep Unsupervised Learning

It is demanding for developers of the health and fitness mobile applications to analysis the feedback and understand the requirements of customers. Excellent design of application should start from customer's need. In this chapter, I proposed an analytical framework to automated the analysis of the large volume of review text. New framework show better topic coherence compared to previous approach. The rich semantic information in the text are converted by context-aware model SBERT into vectors then with its dimension reduced by UMAP. The clustering results and selected keywords represent the content of each topic. Finally, business and design guidelines such as "meditation application has the best cost-effectiveness" and "the biggest issue of fit-bit application is the complain of connections". The detailed workflow of the framework and experiment results are described in this chapter. The chapter is organized as follows. Section 3.1 introduced the detailed workflow of the framework, including data collection, data pre-processing, text analysis, summarizing combined with review scores. After this workflow, reviews of each subcategories of health applications (lifestyle, nutrition, meditation, workout and prescription) are summarized as several topics, which are reported in the Section 3.2. Finally, I concluded the guidelines for each subcategory in Section 3.3 from analytical result of customer feedback.

3.1 Research Method

The whole workflow of our framework is illustrated in Figure 3.1, which includes examples of processed review text and major steps of data collection, pre-processing, embedding (vectorization), clustering, summarization (frequency word mining and theme deduction)

and sentiment analysis (tripartite graph). The framework performed topic modelling to discover underlying topics and themes from collected user reviews. We use the transformer (BERT)-based methodology suggested by Maarten Grootendorst [47], which takes the topic modelling as a clustering task. After clustering, we apply multiple frequency mining methods to the pre-process text and extracted words from each topic as their representations. Finally, we combine them with the ratings and visualize the results by tripartite graphs.

3.1.1 Data Collection

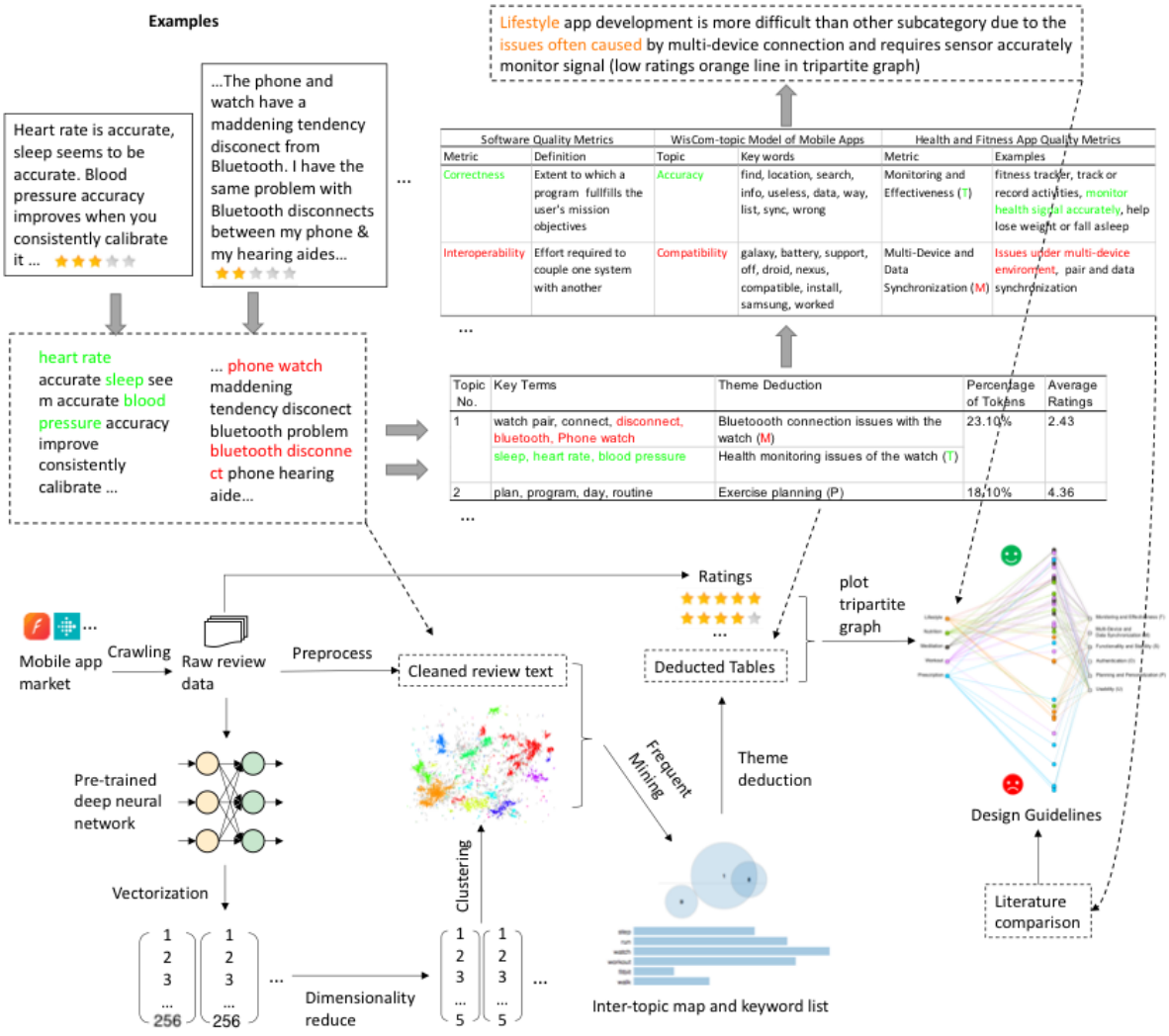

Figure 3.1: The framework of Design from User Feedback using Deep Unsupervised Learning

We developed custom web page crawlers in Python for collecting user review data from the Google app store (Figure 3.1 crawling). Our data acquisition efforts were restricted to apps belonging to the health and fitness category. We further divided health and fitness category into 5 sub-categories. For each sub-category, we first used 2 or 3 popular apps as seeds for search, which was later extended to more apps. Then for each app, we collect the 100 most relevant comments. It eventually extended to 410 apps belonging to the health and fitness category. A total of 410,000 reviews were collected over a wide range of time frames starting from each app's launch. The names of seed apps, number of apps and number of reviews of each sub-category are summarized in Table 3.1.

Sub-category	Apps as search seeds	Number of apps	Number of reviews	Minimal size of cluster
Lifestyle	Fitbit, VeryFitPro, Strava, Runkeeper	410	38592	100
Nutrition	Lifesum, YAZIO	288	35035	70
Meditation	Calm, BetterSleep	225	21161	50
Workout	Smartabase Athelete, Fiton, Peloton	334	28130	70
Prescription	Humana, Go365 for Humana,	95	8135	30

Table 3.1: Summarizing of collected applications

We utilized two different crawlers to scrape the app reviews efficiently. The first crawler uses Selenium library drive to control the web browser (Chrome) and stores a list of apps similar to the searching seeds into a JSON file. The browser driven by the Selenium-based crawler works by extracting information rendered by JavaScript. The second crawler uses an open-source python library (google play scraper) to connect with Google Play API directly and download the review data of apps listed by the first crawler. We also use a paid proxy service (MeshProxy) to increase the concurrency of the crawling process. The two-crawler approach provides a key advantage in terms of ease of use and efficient data collection.

3.1.2 Data Pre-processing

The performance of the deep unsupervised learning model we used (BERT) in this paper is not affected by the NLP pre-processing, however, the frequent mining of topic keywords

requires the review text be pre-processed. We applied several NLP pre-processing steps provided by python library NLTK on the original review (Figure 3.1 pre-process). We first tokenized each review text and converted all into lower cases. Second, we eliminated the tokens in the stop word list of the NLTK library. Stop words are ubiquitous words in all documents without any distinguishing feature and semantic meaning, such as that, the, who, etc. In the third step, we used the WordNet Lemmatizer of the NLTK to transform words into their basic form to merge the occurrence of the same word. Finally, we filtered out unqualified reviews which are less than 2 words or non-English. The reviews we collected from Google Play are already high quality and mostly English, therefore, simple rule that more than the half letters are non-Latin letters is enough to detect non-English reviews. Figure 3.1 illustrated examples of raw review before and after through a series of pre-processing phases mentioned above.

3.1.3 Text Analysis

We performed topic modelling to discover underlying topics and themes from user reviews of the apps. Specifically, we use the transformer (BERT)-based methodology suggested by Maarten Grootendorst [47], which takes the topic modelling as a clustering task. After clustering, we apply multiple frequency mining methods to extracted words from each of the topics as their representations.

Embeddings

To perform the embedding step, we use a language model pre-trained under the method of Sentence-BERT (SBERT [101]) to convert review texts into vector representations (Figure 3.1 vectorization). Bidirectional Encoder Representations from Transformer (BERT [29]) is an unsupervised deep learning framework. Compared to the traditional embedding techniques that take bag-of-word features of documents as input, this contextual-aware transformer-based framework overcomes the semantical indistinguishability of words in different context,

therefore shows strong robustness on noisy raw text and great performance in representing word semantics as vectors. SBERT is a variation of BERT and regarded as the state-of-the-art framework specialized on sentence embedding [102]. To our knowledge, no prior studies have used this state-of-the-art embedding approach to study review texts of health and fitness mobile apps. Specifically, we use all-MiniLM-L6-v2, a model pre-trained under the method of SBERT by Devlin *et al.* on English corpus, as our embedding model. It maps each document into a 386-dimension dense vectors which are semantically comparable.

We assume the review texts of the same topic are semantically similar, therefore, vector representations of the same topic are close to each other in the vector space. The state-of-the-art pre-trained embedding model servers as the upstream for the downstream tasks to cluster semantically similar documents. Since any advance unsupervised model can be used as the upstream model, the quality of topic modelling continuously grow as new state-of-the-art pre-trained models are developed. This SBERT-based topic modelling method was applied on user reviews of health and fitness apps to discover insights of user feedback. Such insights eventually will be used to guide the design of health and fitness mobile apps.

Clustering

The high dimensionality of data is a major challenge to clustering task, we applied the UMAP to reduce the dimensions of the learned representations from previous step (Figure 3.1 dimensionality reduce). As data increases in dimensionality, the difference between the distance to the nearest data point and the distance to the farthest data point decrease [2, 11], therefore, the spatial locality becomes ill-defined, known as the curse of the dimensionality. Compare to other dimensionality reducing methods such as PCA and t-SNE, UMAP is a state-of-the-art method that preservers well both local and global features of high-dimensional data projected into lower dimensions [82]. We tune the hyper-parameters number of neighbours n = 15 and target embedding dimension d = 5 recommended by [47].

We then use HDBSCAN algorithm [81] to clustered dimension-reduced embeddings, as suggested by [5]. The DBSCAN is a single linkage hierarchical clustering algorithm on the transformed space according to data point density. The HDBSCAN algorithm extends it to a soft-clustering approach allowing noise to be modelled as outliers, prevents multi-topic documents to be assigned to any single cluster. The minimal size of clusters is a hyper-parameter of HDBSCAN algorithm. However, no agreed formula exists to determine the optimal minimal size. Therefore, we qualitatively check whether the selected value generates a meaningful set of topics as well as quantitatively measure the distances among topics enough to separating different topics, which is suggested by [90]. We used a web-based visualization package, PyLDAvis [108] (Mabey 2018) to take both two approaches from the bigram list and the inter-topic distance map (Figure 3.4). After several running, we determine the optimal parameter (minimal number of members in each cluster) for minimal overlapping and meaningful topics for each subcategory of review texts, listed in the last column of Table 3.1. Figure 3.2 shows an example of clustering result from lifestyle apps.

Summarizing

We summarized a topic with a list of words that selected from the collection of reviews that assigned to that topic. The selected words as its representation should allow us to know what makes one topic different from another. The words in the list are either monograms or bigrams. [118] pointed out that bigrams, such as "New York", are better analysis units for topic modelling than single word tokens such as "New" and "York" since bigrams are able to better maintain the semantic value of words and capture context within the documents while balancing the dimensionality of the vocabulary constructed.

To generate a semantic meaningful representation, we used multiple frequency-based techniques to mine the keywords. We first used c-TF-IDF, which is a modified TF-IDF

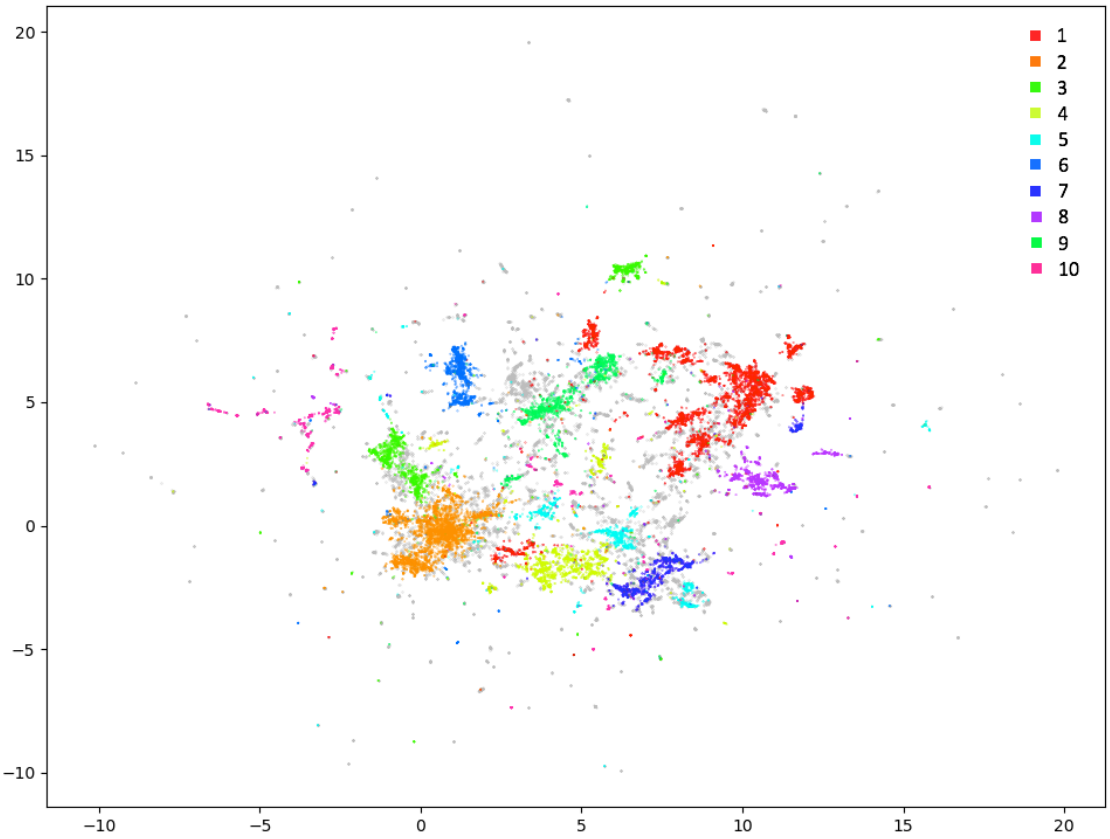

Figure 3.2: Clustering results visualized by UMAP topics

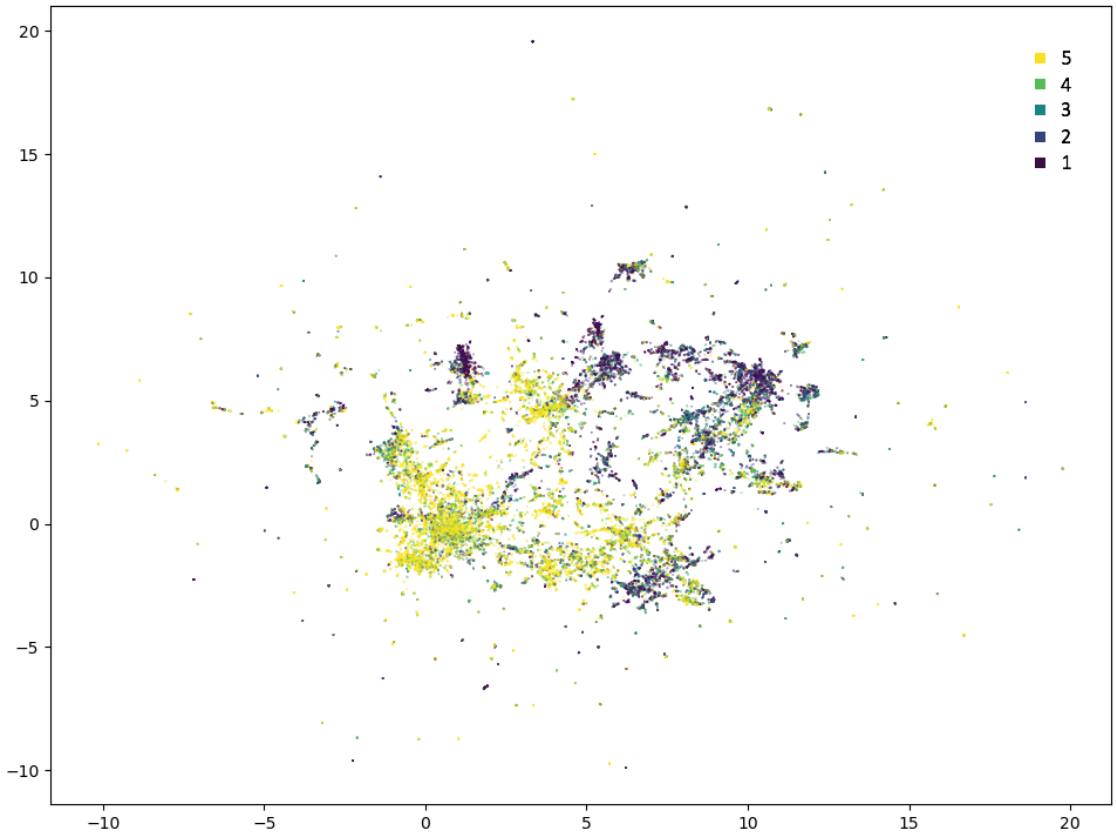

Figure 3.3: Average scores of clusters

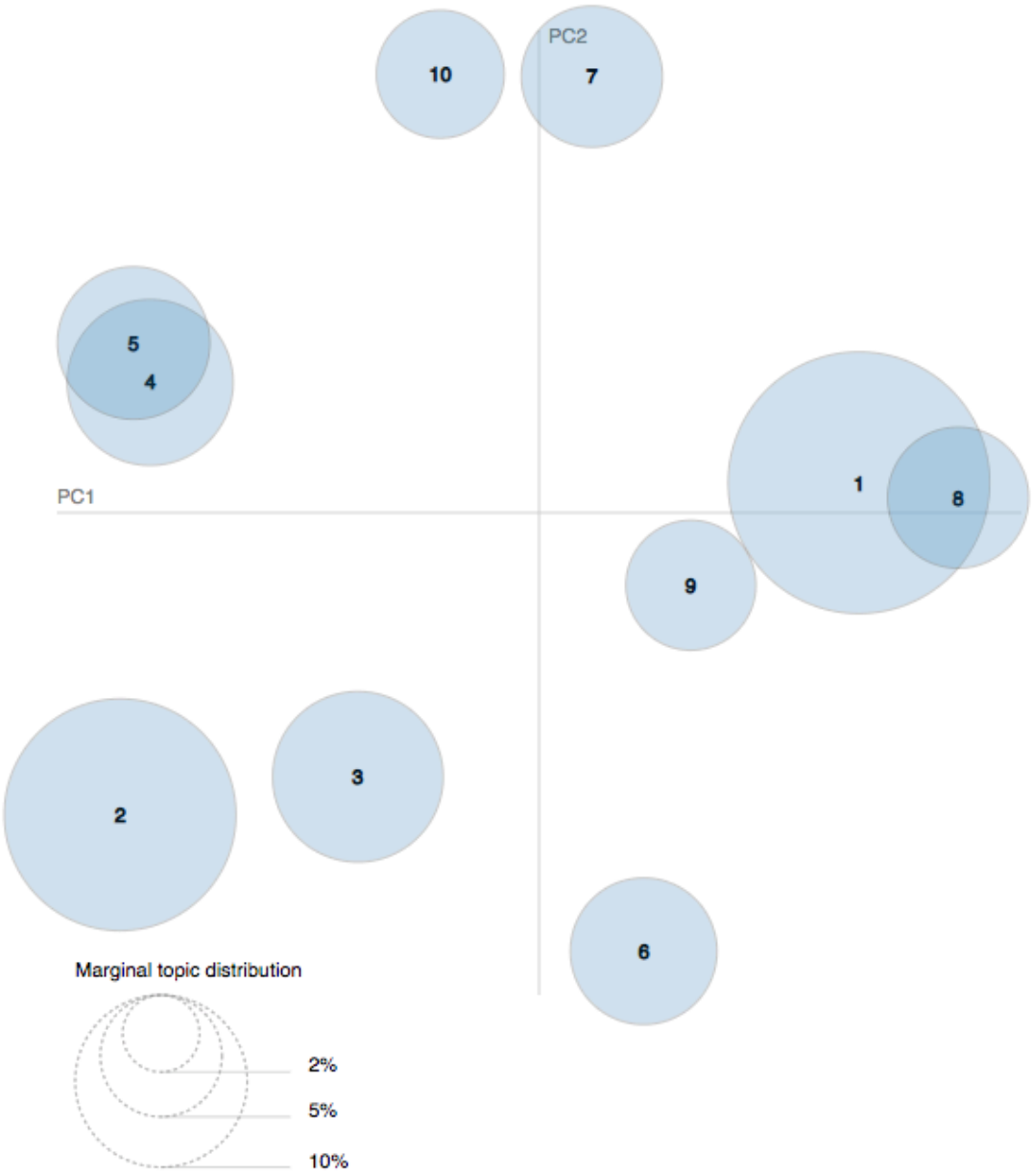

Figure 3.4: Clustering result visualized by PyLDAvis

approach formulated as following:

$$W_{t,c} = f_{t,c} \cdot \log\left(1 + \frac{A}{f_t}\right) \tag{3.1}$$

The c-TF-IDF value is the product of two major parts, term frequency and inverse cluster frequency. We treat all documents in a cluster as one large concatenate document, therefore, the term frequency f is defined as the frequency of term t in a cluster c. Similarly, the inverse cluster frequency is defined as the logarithm of the average number of words per cluster A divided by the frequency of term t across all the clusters. The one is added to the division within the logarithm to output positive value only.

We also used the relevance suggested from the visualization tool pyLDAvis to measure the importance of words, which is defined as following:

$$r(t, c \mid \lambda) = \lambda \log\left(\phi_{t,c}\right) + (1 - \lambda) \log\left(\frac{\phi_{t,c}}{\phi_t}\right) \tag{3.2}$$

The $\phi_{t,c}$ denote the probability of term t appears in the topic (cluster) c, while the ϕ_t denote the marginal probability of term t in the corpus. The relevance calculated as the weighted sum of two logarithms of the topic-specific probability $\phi_{t,c}$ and the *lift*. The lift is a ratio of a term's probability within a topic (cluster) $\phi_{t,c}$ to its marginal probability ϕ_t across the corpus. Although the lift is useful to find important topic-specific words, it is also very sensitive to the rare terms. Combine both two logarithms avoids noisy results. Unlike LDA, in this study the clustering result is available from previous steps, the probability ϕ can be simply estimated by frequency of terms.

The pyLDAvis visualization tool provides two major interactive functions. First it allows user to modify the parameters λ interactively and watch the change of the keywords list. The weight of two logarithms in relevance is determined by the parameter λ. Second, the plot on the left panel (Figure 3.4) shows inter-topic distance map of user review topics on fitness apps identified in this study. Each circle corresponds to one topic. The size of a circle

suggests the prevalence of that topic in the corpus with larger circle containing more word tokens. For instance, the circle of topic 1 in Figure 3.4 is the biggest and it is also the most prevalent topic in lifestyle app sub-category. The distance between two circles reflects the similarities of the topics such that shorter distance suggests a higher level of similarity. For example, as circles shown in Figure 3.4, Topic 1 (multiple device connection issues) is more similar to Topic 8 (data synchronization issues) than to Topic 9 (app crash).

In this summarizing step, we manually merged the results from the two methods and qualitatively checked the coherence of topics and extracted the themes within each topic. We collected and merged all the topic terms under three parameter values (λ =0.3, 0.6, 0.9), and then manually selected one term for every synonym sets. Two researchers then independently went through topic terms to understand these latent topics and deduce themes within each topic. The In preliminary experiments, we divided all the fitness and health apps into 5 subcategories, therefore, this process was also repeated for all 5 subcategories. The themes generated from this process were then summarized from Table 3.2 to 3.6 and were reported in the Results section.

3.1.4 Sentiment analysis

We combined the extracted topics and deducted themes with additional sentiment information, which is the rating scores comes along with the reviews. The authors of the app reviews also provide their ratings after each comment in scale of 5. In many sentiment researches, the user ratings are used as the ground truth for sentiment prediction task [34], therefore, the ratings can be used directly as a measure of sentiment. The sentiment information of the ratings provides better understanding of emotion embedded in the review text. For example, as Figure 3.3 show, we plotted vector representation of each review from lifestyle apps with different colours, which stands for different rating scores. We differentiated opinions in similar topics by comparing Figure 3.3 with Figure 3.2, which uses different colours for different clustering index. The Topic 4, 7 and 10 in Table 3.2 are all about the

similar app function of tracking pace, distance and steps. The Topic 4 and 7 are close to each other in Figure 3.2, while Topic 10 and Topic 7 are close in Figure 3.4. However, from the ratings we can easily tell the latter is complain the technical issue often occurred when use the function while the former is praising the usefulness of it. This provides valuable information to guide the design of mobile app. Developers are able to foresee the potential risk and prepared for the issue in advance. It also allows company allocated the resources wisely to work on the most cost-effective aspects of the app first and minimize the over damage to the business.

3.2 Experiment Results of Health Apps

After running clustering, we extracted in total 55 topics from the 5 sub-categories of the health and fitness app reviews from Google Play market, with 9 12 topics emerged from each sub-category. Table 3.2 to 3.6 show the theme deduction result of each sub-category, which will be discussed in the following paragraphs. Since some apps are multi-functional and belong to multiple sub-categories, it is common that the sub-categorization of the app reviews is non-exclusive. To avoid repetition, the overlapped topics will be discussed mostly in the most associated sub-categories. We also assigned a tag for each theme and then discussed similar themes as groups in the next section to obtain overall design insights.

3.2.1 Results of Lifestyle App Reviews

The lifestyle app subcategory covers mostly tracking and guiding apps on all aspects of a healthy lifestyle in general. The size of 10 emerged topics from lifestyle app reviews, measured as the percent of tokens, ranges from 5.50% to 32.10% percent. The first topic of lifestyle app review is about the smart watch. It is the largest topic and consisting of 4 themes. Themes focus on the issues of Bluetooth connections between phones and smart watches, health signal monitoring issues of watches, instability after software update and customization of smart watch faces. The second emerged topic consists of two themes focusing on guide of

Topic No.	Key Terms	Theme Deduction	Percentage of Tokens	Average Ratings
1	watch pair, connect, disconnect, bluetooth, Phone watch	Bluetoooth connection issues with the watch (M)	23.10%	2.43
	sleep, heart rate, blood pressure	Health monitoring issues of the watch (T)		
	notification, Update	Version stability (S)		
	watch face	Watch face customization issues (A)		
2	home workout, equipment, muscle group, great/good/best workout, full body, stretch, personal trainer, variety exercise, options	Guide to exercises (I)	18.10%	4.36
	plan, program, day, routine	Exercise plan (P)		
3	food item, nutrient, keto, carbs, gram, body fat, belly fat, sodium, eat, BMI, meal, recipe	Food and nutrition guide (I)	9.80%	3.68
	daily calorie, calorie macros, food track, kg/pound, measurement, lose weight	Weight monitoring and calorie counting (T)		
	barcode scanner	Convenient scanner (U)		
	scale, connect scale	Connection issues with the scale (M)		
4	runner, fasting, stats run, pace control, guided run, marathon, 5k	Guide for runner (I)	9.30%	4.00
	track pace	Track user's pace (T)		
	treadmill	Connection issues with the treadmill (M)		
	race	Exercise encouragement (E)		
5	find trails,	Information of trails (I)	7.80%	4.34
	live logging, calorie burn, step, walk, hike	Walk tracking (T)		
	daily walk	Walk planing (P)		
	motivated, goal, group ride	Exercise encouragement (E)		
	bike computer	Connection to the bike computer (M)		
	zwift, best cycling, easy	Cycling userability (U)		
6	cancel membership, collect coin, unsubscribe, auto renewal, charge free, redeem, try cancel, bank account, impossible cancel, refund policy, charge account, force pay, refund money, service use, premium package, free trial, customer service, purchase, email	Price and Customer Service (C)	7.20%	2.40
	full screen, click ad	Distruptive advertisement (U)		
7	mileage counter, distance, gps track, accurate	Location and distance tracking (T)	6.70%	2.50
	gps work, gps signal, signal lose, gps stop, update weather, weather location	Positioning stability (S)		
8	google fit, sync fitbit, mi fit, sync google, fit data, fit samsung, jyoupro, launcher, band	Google fit synchronization issues (M)	6.70%	3.20
	find fitbit, lose fitbit,	Locating lost device (U)		
	launcher	Launching issue (S)		
9	crash immediately, always crash, download open, feature compare, immediately open, say pending, start freeze, amount people, bug android, install reinstall, open already, won't open, load, try open	Application crash (S)	5.70%	3.21
10	pedometer, step counter, track step, sensitivity, accurate, simple/easy/basic pedometer	Pedometer tracking accuracy (T)	5.50%	3.54
	best/great pedometer	Pedometer usability (U)		

Table 3.2: Lifestyle

home workout as well as exercise planning. The third emerged topic mostly revolves around themes about healthy eating, such as food and nutrition guide, weight monitoring and calorie counting. Other themes are issues of the connection between smart scale and app and positive feedback on the convenient feature of barcode scanner, which can record food and provide related information by scanning. The fourth topic is about running exercises. The themes of fourth topic are the guide for runners, tracking user's running pace and encourage user by virtual racing, and issues of connection to the treadmill. The fifth topic is about walk and cycling. Similarly, the themes of the fifth topic are guiding information of trails, tracking and planning daily walk, useability of cycling app, encouragements such as goal setting, group riding. Another theme in this topic is the issue of connection to the bike computer. The sixth topic focuses on charge disputes and disruptive advertisement experience. The seventh topic focuses on positioning issues, including themes of inaccuracy tracking and unstable service. The eighth topic focuses on the issues of Google fit data synchronization, app launching issues and useful function of locating lost devices. The tenth topic focuses on pedometers, including themes of tracking their accuracy and usability.

The above analysis summarised in Table 3.2 first provides specific guidelines for developing of certain type of app. For example, a success running exercise app should at least consider factors such as providing training guide, pace tracking, encouragement by holding virtual race and connection to treadmills. At the same time, we noted that the multi-device environment is the major challenge for lifestyle app developing. The quality of connection is a significant issue for lifestyle apps, including connecting to smart watches, to smart scales, to treadmills and bike computers. We also noted that a healthy lifestyle app may covers broad aspects of life, such as sleep, eat and workout, which will be discussed more in the following paragraphs.

Topic No.	Key Terms	Theme Deduction	Percentage of Tokens	Average Ratings
1	workout apps, gym, workout plan/program, equipment, posture, resistance band, stretch, muscle, finess, workout without	Guide to exercise muscle (I)	24.40%	4.4
	custom workout, daily yoga, routine, plan	Exercise planning (P)		
	deep sleep, workout tracker, rem sleep	Sleep monitoring (T)		
2	reset password, fitbit, sync samsung, fitbit versa, update, email, login, closing, fit sync, won't sync, account, forget password, crash, log, connect internet	Login and synchronization issues (S)	18.50%	3.62
	much ad, video ad	Disruptive advertisement (U)		
	edit food	Inconvenient editing (P)		
	track	Food tracker (T)		
3	cancal, subscription, charge, pay, refund, free trial, money, payment, customer service, premium, scam, try cancel, unsubscribe, paypal, credit card, bank account, google pay, try contact	Price and Customer Service (C)	9.90%	1.87
4	calorie counter, food, track calorie, eat, calorie intake, calorie burn, weight, calculate calorie	Food tracker (T)	7.90%	4.15
	meal	Guide to meal plans (I)		
	easy	Easy-to-use (U)		
	item	Customize meal plan (P)		
5	lose weight, help lose, pound, exercise diet, buddy activity, calorie deficit	Weight monitoring (T)	6.90%	4.73
	diet plan, healthy eat	Diet plans (I)		
	challenge	Encourage users on diets (E)		
6	track step, walk, mile, count, accurate, pedometer, count step, gps, distance	Walk tracker (T)	6.30%	4.03
7	drink water, water intake, hydrate, coffee, beverage, tea	Water drinking tracker (T)	5.60%	4.32
	notification, reminder, octopus	Encourage user drink water (E)		
8	track weight, bmi, weight loss, measurement, progress, goal, body fat, date, idea weight, moving everage	Weight monitoring (T)	5.00%	4.15
	chart, line graph, simple	Result visulization (U)		
	backup restore	Data backup and restore (S)		
9	keto diet, recipe, food, meal, low carb, eat, ingredient, macro, find, cook, option, list, tasty	Diet recipe (I)	4.90%	4.26
10	intermittent fast, timer, stage, fast plan, fast tracker, start/end	Fasting tracker (T)	4.20%	3.92
11	workout, exercise level, quick exercise, body, hard, video, variety, muscle, air squat, exercise equipment, feminine, great range, non-stop, workplace	Video guide to exercise (I)	3.80%	4.75
	beginner chanllenge, easy follow, goal, feel good	Encourage user by challenge (E)		
	routine	Exercise planning (P)		
12	barcode scanner, easy use, food item, interface, database, search product, barcode find, barcode feature, search brand, store brand, look food, helpful	Convenient food tracker using barcode scanner (U)	2.80%	4.31

Table 3.3: Nutrition

3.2.2 Results of Nutrition App Reviews

Table 3.3 summarised 12 emerged topics from the nutrition app reviews. The size of the topics varies from 2.80% to 24.40% percent, which is comparable to the other sub-categories. Topic 1 is about guide and planning of muscle build exercises, and sleep monitoring, which are overlapped with workout and meditation app subcategories. Topic 2 includes themes of food tracker issues, such as account login, disruptive advertisement, inconvenient editing the food items. Topic 3 focuses on the same issue as the Topic 6 in lifestyle app sub-category, which is the charge disputes. Topic 4 focuses predominantly on 4 different themes of calorie tracking apps. The first theme is the useful function of calorie tracking. The second theme is the useful guide to make health meal. The third theme is the easy-use experience. The last theme is the customizability of meal plan. Topic 5 focuses on exercise diet apps, including themes of weight loss monitoring function, diet planning, and encouraging users to complete the diet challenges. Topic 6 is about walk exercise tracker which is an overlapped topic. Topic 7 is about water drinking tracker, including themes of encouraging user to drink water more and track the total intake of water. Topic 8 focuses on visualization of weight loss results. Other themes in the topic are weight monitoring and backup data restore. Topic 9 is the rich information of diet recipe provided by apps. Topic 10 is the tracker of intermittent fasting. Topic 11 focuses mainly on the guide to exercise in the intuitive form of video. Exercise planning and using challenges to encourage users are other two themes in this topic. Topic 12 is the convenience of tacking food by barcode scanner.

The topic overlap of nutrition and workout sub-categories observed above implies that developers may design nutrition and workout features jointly when targeting clients to lose weight. We also noted that rich information of health food recipe, convenient barcode scanner for food, easy-to-use tracking, intuitive visualization and planning are expected by the user of nutrition app. Last, we found the pricing, charging and customer service related complains are both issues in previous and this sub-category. In fact, this topic is a general negative

factor appearing every sub-categories and worth paying attention to, as we will also see in the other following results.

3.2.3 Results of Meditation App Reviews

Topic No.	Key Terms	Theme Deduction	Percentage of Tokens	Average Ratings
1	guided meditation, mindfulness, practice, help, session, time, recommend, teacher, music, voice	Guided meditation (I)	19.70%	4.45
	free	Paywall Complaints (C)		
2	workout, exercise, yoga, breathe, stretch, beginner, weight, video, fitness, practice, trainer	Guide to exercise (I)	12.80%	4.52
	easy	Easy-to-use (U)		
3	fall asleep, help sleep, relax, drift away	Help sleep and relax (T)	12.00%	4.37
	sleep sound, story, wake, music, listen peaceful, night story, michelle sanctuary	Various peaceful sounds (I)		
	time, phone, wake alarm, night alarm	Sleep schedule (P)		
4	cancel subscription, payment, charge, free trial, refund, premium, money, card, service, even, try cancel, want, scam, version, paypal, billing, sign, customer service, credit card, bank	Price and Customer Service (C)	6.00%	2.29
5	white noise, sound, music, binaural beat, sleep, fan, listen, option	Various binaural beats (I)	8.40%	4.45
6	deep sleep, night, fall asleep, detect snore, time, sleep monitor, insomnia, track, rem, chart, bedtime routine, microphone, accuracy, rem sleep, sleep pattern, wake, use	Sleep monitoring (T)	8.00%	4.37
	free	Paywall Complaints (C)		
7	relax, fall sleep, music, sound, help, asleep, calm, meditation, listen, soothe	Calm music (A)	6.70%	4.74
8	reduce anxiety, help, panic attack, mental health, anxious stress, therapy, calm, depression, counsel	Reduce the anxiety (T)	5.10%	4.76
	offline mode	Paywall Complaints for offline mode (C)		
9	keep crash, open, update, load, try, time, reinstall, login, improve stability, account easy, able login, black screen, front screen, crash continuously, download install, wifi, issue open	Crash and login issues (S, O)	4.90%	3.87
	ability personalize, great customizable	customizable options (P)		
	easy use, simple, maneuver	easy-to-use (U)		
10	rain sound, thunderstorm, drink water, plant, sleep, cute, night, roof, rain wind, distant thunder	Rain sound (A)	4.00%	4.56
11	mix sound, volume control, different sound, save combination, option, custom, quality, timer, preset, arrangement, baby monitor, bitrate, edit sound, custom, play save, great UI, sound control, selection	Sound customizable (P)	3.70%	4.57
12	ad pop, sound, loud, intrusive, postcard, remove ad, screen ad, annoy, full screen, commercial, obnoxious, ad play, unskippable, interruption ad, ad ridiculous, hate ad, ad problem, ad begin, ad close, ad problem, ad uninstalled, pop everytime	Disruptive ads (U)	3.60%	4.05

Table 3.4: Meditation

Table 3.4 summarized the 12 emerged topics from the meditation app reviews, whose size varies from 3.60% to 19.70% percent. Topic 1 consists of two themes focusing on guide to meditation through stillness or relax and complaints of the paywall. Topic 2 is about meditation through breathing or movement such as yoga and stretch exercises. The topic consists of two themes focusing on the guide to practice and the easy-to-use experience of

the app. Topic 3 focuses on sleep improvement. Two themes of the topic are various relaxing sound and sleep cycle scheduling. Topic 4 focuses on charge dispute issues. Topic 5 focuses on various binaural beats. Topic 6 focuses mostly on sleep monitoring. Another theme of this topic is also complaints of the paywall. Topic 7 focuses on positive experience of calm music. Topic 8 is about relieving anxiety. Complaints of offline mode purchase are also one theme of this topic. Topic 9 consists of three themes about the functionality. The first theme are issues of account login and crash of the app. The second theme is great customizability. The last theme is manoeuvrability of the app. Topic 10 focuses on positive experience of rain sound. Topic 11 is about the popular function that allows sound customizable by users. Topic 12 focuses on disruptive ads.

We observed that the richness of sound or music is critical to the success meditation apps and mentioned most often by users. Developing meditation apps is also less challenging and may has higher success rate, based on overall higher ratings of each topics comparing to other sub-categories. We also note that the app crash and account login are very common technique issues in meditation and other sub-categories, as we have already seen before and will see in the following.

3.2.4 Results of Workout App Reviews

Table 3.5 summarized the 12 emerged topics from sub-category of workout application reviews. Topic 1 focuses on the dance fitness, including themes of guide to dance fitness, using challenges to encourage user and easy-to-use experience. Topic 2 is the charge disputes. Topic 3 focuses on the gym exercises, consisting of themes of guide to gym exercises, track lift exercises, planning and easy-to-use experience. Topic 4 in about the application crash and account login issues. Topic 5 overlaps the nutrition subcategories, consisting of themes of recipe, weight tracker and convenient barcode food scanner. Topic 6 focuses on yoga practice, including themes of guide to practice yoga and motivated voice and music. Topic 7 focuses on plank and cross-fitness, including themes of fitness guide and complaints of

Topic No.	Key Terms	Theme Deduction	Percentage of Tokens	Average Ratings
1	workout, exercise, body, day, beginner, kira, follow, fun, minute, need, move, level, variety, keep, start, help, jessica, body groove, melissa, vicky justiz, jiilian, dance, kira strokes, great range	Guide to dance workout (I)	17.00%	4.63
	chanllenge	Encourage user by challenge (E)		
	easy use	easy-to-use (U)		
2	cancel subscription, charge, free, pay, free trial, refund, money, email, month, customer service, sign, try cancel, payment, account, day, premium, membership, credit card, scam, bank, bill, debit, trial charge, renewal, impossible cancel	Price and Customer Service (C)	15.40%	1.95
3	different stretch, workout, exercise, day, body, start, gym, maternity, exercise library, tutorial, find pregnant, body, muscle, level, recommend, variety	Guide to gym exercises (I)	10.60%	4.64
	really easy, easy use	Easy-to-use (U)		
	track lift	Track lift exercises (T)		
	routine	Exercise planning (P)		
4	work, login screen, log, try open, reset password, crash, use, video, email click, ca, fix, update, network error, issue account, even, download, phone, load, try sign, reinstall, problem, credential, verify email, glitch, login issue, crash, update video, confirm email, data cache, try uninstalling, email address	Login issue and crash (O, S)	8.80%	2.20
5	food item, meal plan, calorie burn, eat, recipe, diet, scan, list, nutrition, intake, nutrient, carbs, ingredient, food diary, vitamin, macros, intake, database	Guide to meal plans (I)	8.70%	3.92
	track calorie, count, weight	Weight tracker (T)		
	barcode, scanner,	Convinent barcode scanner (U)		
6	yoga practice, pose, meditation, jessamyn, beginner, instruction, session, relax, teacher, video, asana, class	Guide to yoga (I)	8.10%	4.33
	voice, music	Motivated voice and music (E)		
7	plank workout, trainer, fitness, exercise, wods, equipment, program, amaze trainer, biceps back, alive inside, crossfit, elbow, best trainer, home	Guide to plank and crossfit (I)	8.00%	3.74
	feature parity, phone memory, workout crash, connect network, fix, update, load, version, bug annoy	Crash and network issues (S)		
8	bike, watch, connect, fitbit, device, pair, monitor, cadence sensor, sync, google, data, heart rate, garmin, strava, fit, resistance band	Connection issue with the sensors (M)	6.20%	3.34
9	tv, cast tv, chromecast, workout, cast option, would, make, screen, stream, connect tv, hearing, playlists, google tv, smart tv, roku	Connection issue with the TV cast (M)	5.10%	3.70
	music, voice, spotify music, audio, sound, phone volume	Pleasant music (A)		
10	track, gps, mile, distance, track progress, run, tracker, update, walk, plan, start , weight, gps, drain battery, run track, distance run, lane, last set, phone gps, , keep	Track walk exercise (T)	4.60%	4.09
	fasting, intermittent fast, stop fast, body fast	Track intermittent fast (T)		
11	lose weight, belly fat, pound, diet, help lose, exercise, reduce, recommend, reduce weight, extremely satisfied, follow diet, healthy diet	Exercise diet (I)	4.10%	4.72
12	amaze, easy use, simple, content user, amazing, excellent, beginner, awesome, need, english version, chinese version, miss package, hindi miss, 531 program, discipline great, thanks, respond question, helpful	Easy-to-use (U)	3.40%	4.72

Table 3.5: Workout

application crash and network issues. Topic 8 focuses on connection issue with the device sensors, such as smart watch, bike computer and band. Topic 9 is about the issue of TV cast. Plenty of great music is another theme in this topic. Topic 10 overlaps the lifestyle application sub-category about walk exercise and intermittent fasting tracker. Topic 11 overlaps the nutrition application sub-category about guide to exercise diet. Topic 12 is the general positive user experience.

From the above results, we note the fitness guide and encouragement provided by the workout apps is the core of successful design. Workout apps not only should provide extensive training information for certain fitness program, but also encourage users continue exercise by various persuasive techniques, including motivated music, virtual races, challenges, achievement system or even gamification. We also noted many internet influencers and their channel names, such as Vicky Justiz, Kira Strokes are frequently mentioned by users. Developers can introduce fitness influencers into app design to enhance the coaching and encouragement features of the workout apps. As we see in the lifestyle subcategory, workout apps that requires multi-device connection also face similar technique issue and complaints from users.

3.2.5 Results of Prescription App Reviews

Table 3.6 summarized the 9 emerged topics of sizes ranging from 3.00% to 40.90% percent from sub-category of prescription app reviews. With the predominately largest size, topic 1 focuses on the appointment scheduling issues. Topic 2 focuses on complaints of login issues. Topic 3 focuses on the missing notification of notification from pharmacy. Topic 4 is about complaints of app crash and unavailable health document. Topic 5 is about complaints of incorrect claim information and data synchronization, login issues due to inactivity. Topic 6 focuses on account login issues due to various reasons. Topic 7 overlaps the nutrition app sub-category, consisting of similar themes of food tracker, guide to nutrition, complaints of the price of the produce sold, convenient barcode scanner and meal customization. Some

Topic No.	Key Terms	Theme Deduction	Percentage of Tokens	Average Ratings
1	doctor, health, appointment, insurance, need, care, information, medical, provider, test, card, schedule, appointment, workout, symptom, nurse, vaccination, hospital, qr, menopause, booster, caresource, gym, record, health care, record	Appointment scheduling issues (P)	40.90%	3.46
2	forgot password, log, fingerprint login, time, try log, sign, work, fingerprint, reset, change password, account, website login, reset username, code, wrong, every time, error, recognize device, account lock, get verification, reset username, sms, sso, try install, type password, authentication say, time log	Fingerprint login issues (O)	12.50%	1.45
3	refill prescription, pharmacy, pick order, medication, humana, rx, script, request refill, store, fill, prescription ready, call, reminder, delete, refill date, delivery, mile away, expire prescription, otc, cart, item	Pharmacy notification issues (P)	11.00%	2.67
4	open, upload, document, try, fix, crash, version, update available, need update, upload document, download, load, phone, screen, uninstalled, closing, prompt update, home address, update try, work fix, let update, throw, upload take, always shut, card information, cleaner, crash constantly, document like, download nothing, fail update, get open, hold hour, junk	Crash and document unavailable (S)	10.20%	1.77
5	claim, sync, samsung health, fitbit, health, information, fix, track, bcbs, connect, step count, step tracker, like error, pharmacy claim, sorry look, sync fitbit, access data, 3rd party, activity like, attempt say, benefit information, button, competent, detail enough, fsa claim, give permission, health connect, hire competent, count, issue, track, never connect	Claim information synchronization incorrect (M)	9.30%	2.04
	due inactivity,	Login issues (O)		
6	log, password, register, login, try, sign, account, cant, email, try reset, fix, enter, create, wrong, error, back signin, hard register, information correct, account try, able register, almost impossible, already register, create username, even create, almost impossible, extremely frustrating, login spin, message password	Login issues (O)	5.20%	1.47
7	meal, numi eat, calorie, weight loss, track, intake, log food, water	Food tracker (T)	4.78%	2.58
	nutrisystem food, journal, dietician, recipe, diet	Guide to food nutrition (I)		
	produce, snack, healthy	Price complaints of the produce sold (C)		
	scanner, scan	Convenient scanner (U)		
	item	Customize meal plan (P)		
8	email message, connect server, update, say, download, connect, login, new, server error, can not connect, new version, uninstalled, tell, message open, download enough, wait install, original version, server, can't connect	Connection issues with the server(S)	3.20%	1.41
9, 10	easy use, easy navigate, information easy, difficult accessibility, desktop site, navigate relevant, compass, convenient informative, abundance information, add echeck, amaze easy, fingertip great, everyting, user friendly, appreciate, convenient helpful, basic complaint, benefit record	Easy-to-use (U)	3.00%	4.62

Table 3.6: Prescription

prescription apps advertise other nutrition apps inside their apps. Topic 8 is about the complaints of issues of connecting server. Topic 9 focuses on easy navigating of the apps.

From above results, we note the prescription apps received overwhelmed negative feedback and significant lower ratings compared to other sub-categories. The complaints cover various techniques issues including account login, connecting to the server, data synchronization, crash, notification lost. The low quality may be caused by the fragmentation of developing, that is, each institution developed their own apps with limited budget and cause the low quality of those apps. A unified app severed as a third-party platform for all institutions may solve the problem. The easy navigation is the most expected from users.

3.3 Further Discussions

Software Quality Metrics		WisCom-topic Model of Mobile Apps		Health and Fitness App Quality Metrics	
Metric	Definition	Topic	Key words	Metric	Examples
Correctness	Extent to which a program fullfills the user's mission objectives	Accuracy	find, location, search, info, useless, data, way, list, sync, wrong	Monitoring and Effectiveness (T)	fitness tracker, track or record activities, monitor health signal, help lose weight or fall asleep
Interoperability	Effort required to couple one system with another	Compatibility	galaxy, battery, support, off, droid, nexus, compatible, install, samsung, worked	Multi-Device and Data Synchronization (M)	Issues under multi-device enviroment, pair and data synchronization
Reliability	Extend to which a program satisfies its specifications	Stability	closes, close, load, every, crashes, keeps, won, start, please, closing	Functionallity and Stability (S)	no crash or other severe errors, function properly after upgrade, version
Integrity	Extent to which access to software or data by unauthorized persons can be controlled	Connectivity	log, error, account, connect, login, connection, sign, let, slow, website	Authentication (O)	user account management, access user's account and profile data successfully
Flexibility	Effort required to modify an operational program	Picture	pictures, picture, pics, camera, save, wallpaper, see, photes, upload, pic	Planning and Personalization (P)	build exersise plan, schedule appointment with health care provider, customize meal plan
Usability	Effort required to learn, operate, prepare input, and interpret output of program	Spam	ads, notification, spam, bar, notifications, adds, annoying, many, pop, push	Usability (U)	simple and intuitive interface, frictionless user experience, easy to navigate and use, minimal interuption, no spam

Table 3.7: WisCom, quality metrics comparison I

We plotted the tripartite graph (Figure 3.5, 3.6) to visualize the relationships between theme groups and emerged topics from the 5 sub-categories. Five dots in the left column represent the 5 sub-categories. The dots in the middle column represent the emerged topics

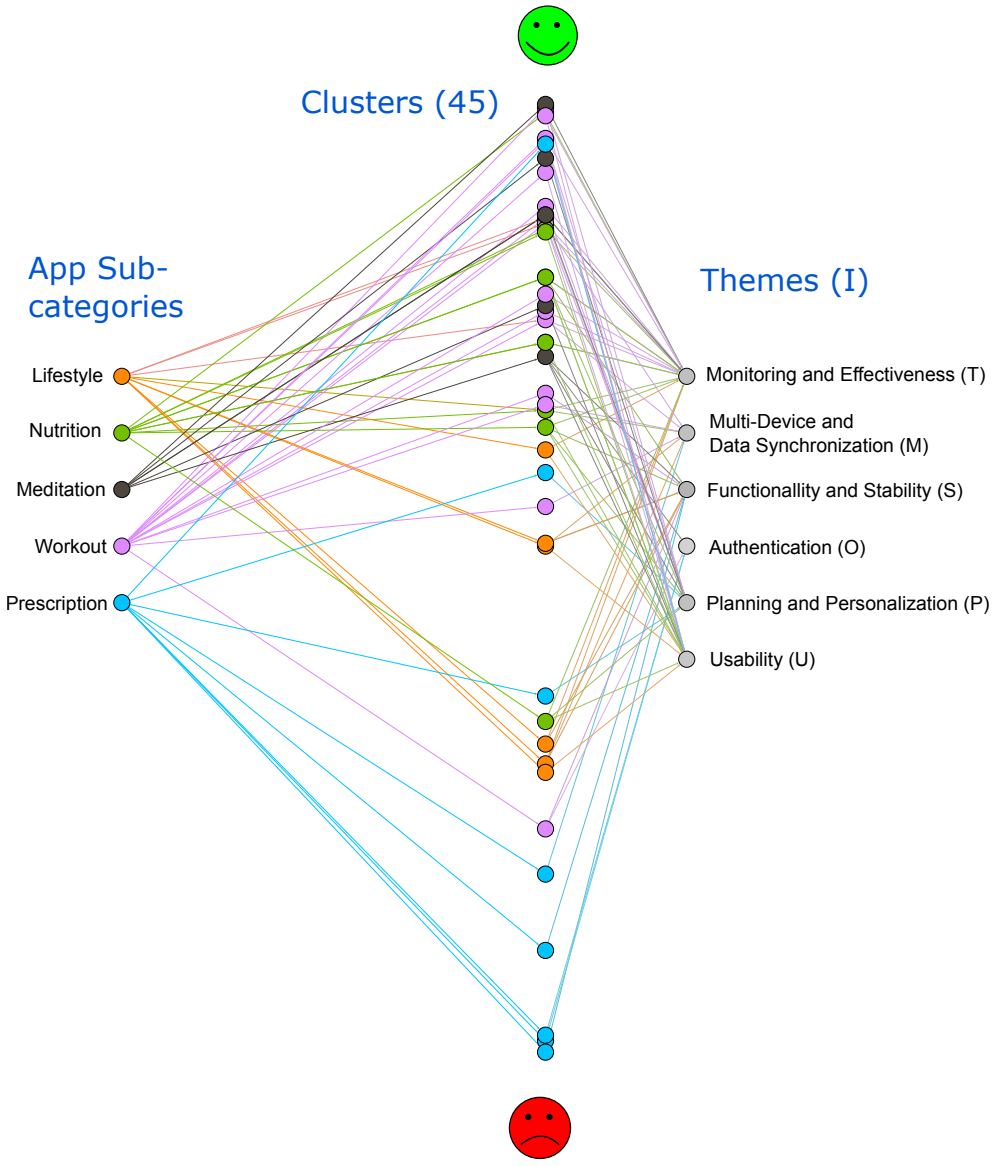

Figure 3.5: Connection between subcategory, cluster and themes I

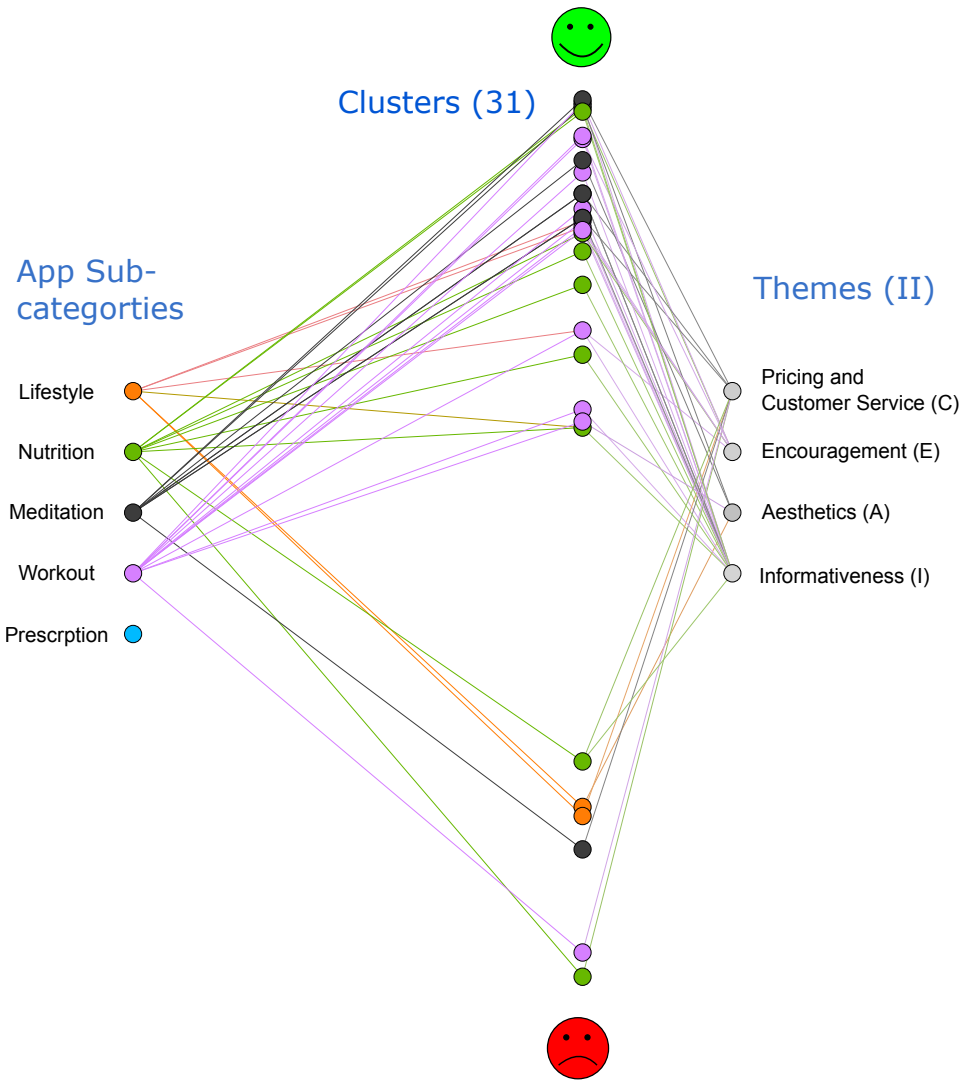

Figure 3.6: Connection between subcategory, cluster and themes II

WisCom-topic Model of Mobile Apps		Health and Fitness App Quality Metrics	
Topic	Key words	Metric	Examples
Cost	free, money, buy, pay, paid, refund, want, back, bought, waste	Pricing and Customer Service (C)	charging dispute, payment experience, price, paywall, membership, premium, in-app purchasing
Telephony	uninstall, want, need, send, message, delete, let, contacts, calls, off	Encouragement (E)	persuasive features including gamification and social encouragement to increase physical exercise.
Attractiveness	boring, bad, stupid, waste, dont, hard, make, way, graphics, controls	Aesthetics (A)	beautiful design of user interface, nice bgm
Media	video, sound, watch, videos, songs, audio, sounds, hear, record, anything	Informativeness (I)	informativeness and extensiveness of the content, variety of tutorial and guidance, influencer-drivern

Table 3.8: WisCom, quality metrics comparison II

from each sub-category, connected with the line of the same colour as sub-category dot. The y-coordinate values of topic dots in the middle are the average ratings of each topic, from 1 start to 5 starts. The dots on the right column are the theme groups. We first introduced the software quality metrics and their definitions proposed by Gaffney Jr *et al.* and aligned some of them (Table 3.7 Column 1) with the most related theme groups (Table 3.7 Column 3) from our results. We ignore the metrics such as portability or maintainability, since the users have no access to the source code. We then also introduced the WisCom topic modelling of mobile app reviews based on LDA proposed by Fu, Lin *et al.*[40] (Table 3.7 Column 2, Table 3.8 Column 1) and also aligned the WisCom-topics to theme groups by comparing their keywords.

We suggest developers prepare enough resources when build the mentioned functions and thoroughly test in all mentioned scenarios in the following discussion of metrics. The correctness metric means the extent to which a program fulfils the user, which is related to the themes of effectiveness of monitoring and tracking (T) in our context. While the tracking accuracy of sleep, food and weight received relatively positive feedback, the accuracy of

health signals, location, walking steps is often complained by the users. The interoperability metric means the effort required to couple one system with another. Tracking apps work under multiple device and sensor environment (M) face the challenge of connection and synchronization issues, such as connected to the smart watch, cadence sensor and sync up with google fit account. The reliability metrics means the extent to which a program satisfies its specifications. The related themes are functionality and stability (S) issues, especially critical errors that cause app crash. It often happens when user login account, connected to the network, restore the backup data, sync with the server and update to new versions. The integrity metrics means the extent to which access to software or data by unauthorized persons can be controlled. As already discussed on reliability, account login is one of the most complained issues. The flexibility metrics means the effort required to modify an operational program, which is related to the group of themes about planning and personalization (P), such as exercise planning. Except some text editing (Topic 2 in Nutrition apps) and appointment scheduling issues (Prescription apps), mobile apps perform very well on flexibility, which may be contributed by the mature UI testing technology today. Finally, the usability metric means the effort required to learn, operate, prepare input and interpret output of programs. The group of themes on usability mostly are positive feedback of easy-to-use. It also mentioned two convenient features, weight tracker result visualization and food barcode scanner. Although disruptive advertisements affect useability negatively, users seem more tolerate to them compared to other issues given the relatively higher ratings.

The WisCom-topic study argues that the ten common topics they found are almost emerged from apps of all categories. There are four theme groups are not covered by quality metrics and only related to WisCom-topics (Table 3.8). The first theme group is the pricing and customer service (C) issue, which is a common complaint in every sub-categorizes. To improve the payment processes, developers may consider use API provided by third-party fintech company such as Swipe. The developer may also consider adjust the price or even change monetization from purchase to advertisement, since users relatively tolerate it. The

theme group encouragement (E) is about persuasive techniques which are discussed in Section 4.4. Those techniques have significant positive effects on nutrition and workout apps. The aesthetics (A) is the third theme group and plays particularly important role in meditation apps. Beautiful smart watch face is also very popular feature but requires some efforts to allow users customize face themes smoothly. The fourth theme group is informativeness (I). Today, mobile apps are beyond tool and also server as media platforms, therefore, the rich and informative content is important as the examples of meditation, workout, nutrition apps. The four theme groups are new design principles that extended the design of focus from software quality to overall values brought to the customers.

3.4 Summary

We are now face a highly competitive and lucrative mobile app market ever growing. This paper examined the emerged topics from health and fitness app reviews using AI-based approach. We deducted the themes of sub-category for specific app design guidelines. We also discussed quality metrics related themes and provided all error-prone scenarios which should be thoroughly tested to assure the quality of apps. The discussion of new theme groups extends the quality metrics with new customer values. All above guidelines from deep understanding of the nature and characteristics of customer feedbacks lay the foundations for the design of next popular app. The whole framework can quickly be applied to any new category of apps and improves the chance of successful design.

Chapter 4

Sentimental Analysis of Telemedicine Application Reviews

In this chapter, we focus on exploring the emotions of telemedicine app reviews since the rating scores from user which indicate polarity of the text already available. Unlike polarity which is one dimensional measurement, no standard measurement and model of emotions is generally accepted by all. Some argued that all complicated emotions can be represented by few essential ones, while other claimed that emotions are not isolated and also depends on the semantical context. Therefore, we introduce a network approach which offers a promising methodology to capture the inter-plays of emotion-emotion and emotion-semantics. Our research objectives are answer three questions: (1) What is the emotional pattern of telemedicine app user and why (2) How emotions related to each other and semantic context and (3) is the representation reducible to essentials. Those insights provide direction for the future design and development of telemedicine apps, and eventually improve the social acceptance of the telemedicine. The chapter is organized as follows. First, the research methods were explained in the Section 4.1. Then the experiment result was described in the Section 4.2. After that, the final conclusion were discussed in Section 4.3.

4.1 Research Method

In this section we first discuss the collection of dataset in 4.1.1 and then introduce the approaches we used to measure the sentiments in 4.1.2, including lexicon-based method, similarity search-based method and measure-as-sentence method.

4.1.1 Data Collection

A customized web script based on library - Google-Play-Scraper - has been crafted to collecting user reviews from the Google app store. The glean data encompasses both meta information of 400 mobile applications, such as application ID, published date, price as well as 100 most relevant reviews of each application, such as reviewer ID, ratings, review content in text. All acquired data focus on telemedicine category. The entire dataset has 400,000 reviews in total over a wide range of time starting from launch of each application. The various attributes of the data, which summarized in table x, will benefit further studies conducted in the foreseeable future. Since the library connects to the APIs of the backend of Google website and it rarely changes, the web script collects data both efficiently and reliably. To avoid being detected as a web spider, we also use a paid proxy service - MeshProxy - to increase the concurrency of crawling and curb the risk of IP blockings.

4.1.2 Sentiment Measurement

The measurements of emotion are conducted in three levels: application-level, reviewer-level, and word-level. The application-level results are simply aggregated from the review/reviewer level; therefore, we mainly discuss the review/reviewer and word levels. Since the total number of reviews compared to the selected 100 reviews are large, two or multiple reviews belongs to one reviewer rarely happened. For this reason, we do not differentiate reviews and reviewers in this part of the dissertation study. A simple approach treats the whole review as an emotion assemble of each emotion associated with the work token in the document. This bag-of-words approach, however, ignores the function and weight of each word in the context of sentences. Recent transformer-based models such as RoBERTa are capable of allocating different attention weights on sentences according to context, thereby showing proved state-of-the-art performance. Meanwhile, simple majority votes from each token by its emotion performed badly on emotion-prediction benchmarks of whole sentences or documents. We will dive into the three approaches at both levels.

Lexicon-based Measurement

Mohammad *et al.* built a high-capacity and high-quality dictionary of emotion lexicons by crowd-sourcing. Each Tucker participated in the study was asked a synonyms multiple-choices question to ensure he understand the target word. The results of the questionnaires from the participating Tuckers were also evaluated statically how they agreed with others. To control the quality, only the results from non-outlier were collected to build the NRC dictionary. Although various emotion database of documents and sentences are proposed by many research, word-level emotion databases are not well studied. The NRC lexicons, recommended by a growing number of researchers as a gold standard, are applied in a diversity of applications including the Linguistic Inquiry and Word Count toolkit or *LIWC*. In this section, we use the binary tagged NRC lexicon EmoLex, real value scored NRC EIL, and the dimensional NRC LDA dictionary to gauge word-level emotions.

Similarity Search

It is worth mentioning that the lexicon-based approach is limited by the capacity of its dictionary. For any word not included in the dictionary, the word is, more often than not, scored as neutral by default. To solve this challenging issue, an alternative way is to locate the word's nearest neighbour in the dictionary as its surrogate, followed by weighing a score by similarity measures such as cosine similarity. A raft of embedding learning models are able to convert words into vectors that semantically close word being assigned with geometrically close representations. A similarity search is the process of finding the nearest neighbour in a given set. Nevertheless, since the number of dictionary consists of 10000 words approximately, a brute-force searching is not practical on large volume of documents in the big data era. The FAISS library is a fast similarity search library speeding up the nearest-neighbour finding procedure. The library adopts clustering of data points in advance and searches from nearest clusters in each iteration. In this study, we deploy the FAISS library to optimize the look-up process of nearest lexicon for words that are not residing

in the dictionary. Dividing the lexicon dictionary into test and validate sets, we carryout a benchmark test to evaluate the consistency between the similarity search and lexicon-based approaches.

Measure as a Sentences

Another approach to overcoming the limit of lexicon-based techniques is direct detection by BERT-based emotion classification neural networks such as emotion RoBERTa. In this part of the dissertation study, we treat the each token as a single-word sentence, and we apply the popular fine-tuned English emotion model RoBERTa from HuggingFace to the single-word sentence. Since the RoBERTa model is usually fine-tuned and tested on non-single-word sentences for emotion classification tasks, the RoBERTa model's validation ought to be compared against and cross-verified with the other emotion detection methods.

4.1.3 Network Analysis

In this research, the network analytic approaches are at the core of our framework to analyze public sentiment responses to the telemedicine applications. The overall workflow of the framework is illustrated in Figure 4.1, where the framework consists of three major phases.

- **Phase I.** The phase I orchestrates a high-level clustering analysis on review collections from each telemedicine application, and the first research question of what are the heterogeneous communities in the network of mobile apps is addressed in this initial phase. The major functionality performed in this phase are semantic similarity measurement between review sets of applications, semantic network built based on similarity, and modularity analysis of the network to obtain heterogeneous communities.

- **Phase II.** The second phase is a middle-level study at the sub-network of each community of the telemedicine applications. We apply multiple network characteristics

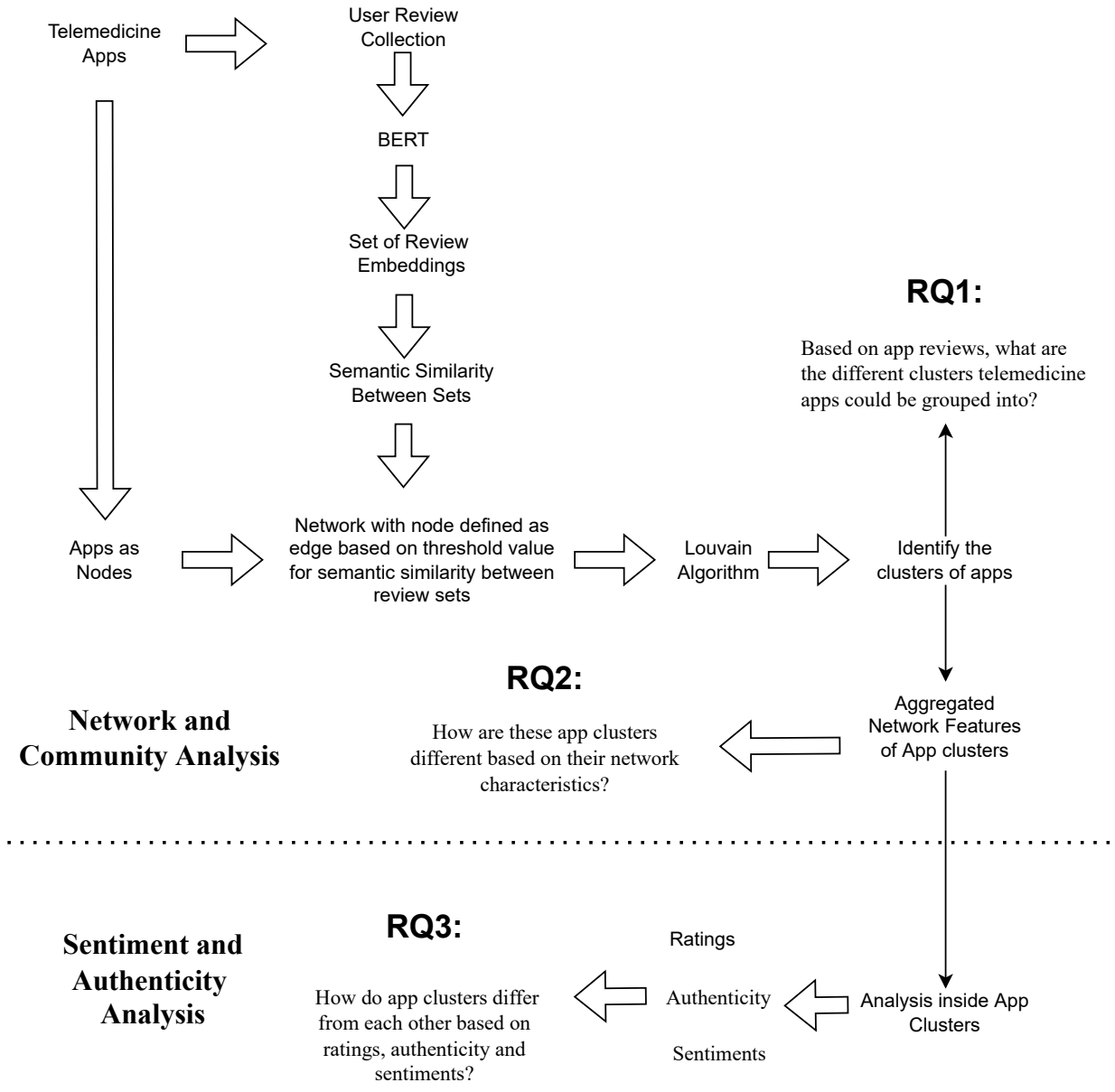

Figure 4.1: Profiling of telemedicine apps based on ratings, authenticity, sentiments

such as clustering coefficient, bridging centrality, average shortest path and others to identify the most critical node in the sub-network. The critical nodes are representative application case study to be investigated in the foreseeable future.

- **Phase III.** Finally, the third phase is focused on conducting an inter-group analysis to gain word-level sentiment and semantic information, and contrast differences among different clusters of applications. This phase, of course, provides insights into how public opinions differ on various types of telemedicine applications.

Semantic Similarity Between Applications

In the first phase of this research, a similarity network of the applications is built. As Equation 4.1 demonstrated, the similarity of any pair of application A and B was measured from the similarity of their reviews. An SBERT embedding of review r_i is calculated and the result is $S(r_i)$. The overall similarity between the two applications is the average cosine similarity of all pairs of review embedding $S(r_i)$ from application A ($r_i \in A$) and embedding $S(r_j)$ from application B ($S(r_j) \in B$).

$$\frac{1}{N_A N_B} \sum_{r_i \in A} \sum_{r_j \in B} \frac{S(r_i) \cdot S(r_j)}{|S(r_i)| \cdot |S(r_j)|} \tag{4.1}$$

Since the SBERT limits the length of text no longer than 512 tokens, it is not possible to concatenate multiple reviews as one large single document for each applications, instead, a set of reviews are used to represent the application.

Clustering and Modularity of App Network

After the semantic network being created, the Louvain method [15] was applied to the network using Gephi software [10] to detect the communities in the network. Gaphi is a popular network visualization tool and Louvain algorithm is an unsupervised clustering

method. The algorithm uses modularity score [15] as the objective function which measures the density of edges inside the groups with respect to edges between the groups.

4.2 Experiment Results

The clustering results of the app networks are plotted in Fig 4.29, where each node indicates an application and reviews associated with the application. The connection between any two applications entails the average similarity between the user reviews of the two applications. Similarity measures range anywhere from 0 to 1.0, and only connections with higher similarity readings are considered as edges among nodes in our network. Since the collection of reviews of telemedicine applications covers multiple topics of the user experience, it is expected that average similarity between review sets are less various compared to the similarity between two individual reviews. In fact, the highest similarity of the app networks is only 0.613. After the trial-and-error experiment, we determine the threshold as 0.254, and the modularity score of clustered apps achieved is 0.129 - a higher modularity score implies more distinguishable groups.

The network is divided into three major groups, including 95.8% of the 166 applications, and the rest nodes - apps - are out-liners. The rest seven apps are not linked to any other apps; each of these applications are assigned as a single point cluster. Due to the small portion (4.2%) of applications belonging to the category of single point clusters, we simply ignore those type of applications in the subsequent analysis because we put a laser focus on the three large clusters.

In the following sections, let us discuss the results of the first largest cluster of applications (40.36%) in Section 4.2.4, the second largest cluster (33.73%) in Section 4.2.5, and the third largest cluster (21.69%) in Section 4.2.6.

4.2.1 Analysis of the Popularity

The statistics analysis of the popularity are listed in the Figure 4.2. The first row of scores shows that the average scores from reviewers between Cluster 1 and Cluster 2 are significantly different. Since the average score of Cluster 1 is 4.29 and the average score of Cluster 3 is only 3.42, clearly the applications of Cluster 1 are successful and the applications of Cluster 3 are not. Interestingly, Cluster 2 has average scores of 3.93, which is between Cluster 1 and Cluster 3. However, although the scores seems supporting the claim that applications of Cluster 2 at least better than Cluster 3 with medium score, other doesn't. When loose the P value restriction to 0.01, the number of installs, number of ratings and number of review of applications on average per application between Cluster 1 and Cluster 3 are significantly different, and the values of Cluster 2 are the worst among all, particularly different from Cluster 1. This shows the applications of Cluster 2 are clearly overrated. This claim is further investigated by the other analysis in the following sections.

	Mean of Clusters Original/Rank Order			P-value of Non-parametric ANOVA (i.e., Kruskal -Wallis Test)	P-value of Pairwise Comparison of Clusters		
Category	1	2	3		1 vs. 2	1 vs. 3	2 vs. 3
Score	4.2910	3.9275	3.4248	**<0.0001*** **	**0.0883+**	**<0.0001*** **	0.2382 ns
# Real Installs	1280556	27533	348417	**<0.0001*** **	**<0.0001*** **	**0.0013** **	**0.0013** **
# Ratings	33538	247	7173	**0.0002*** **	**0.0071** **	**0.0015** **	0.9534 ns.
# Reviews	2790	3	2066	**<0.0001*** **	**<0.0001*** **	**0.0033** **	**<0.0001*** **

Figure 4.2: ANOVA analysis of popularity

4.2.2 Analysis of by NRC emotion lexicon

We use lexicon based approach analysed the emotions of reviews towards applications in each clusters. Since we do not know whether the distribution of average percentage of trust

lexicon is normal distribution or not, we take rank-sum test on them. The rank-sums test also named as Wilcoxon, or Mann–Whitney test, which is applied to non-normal distribution.

Figure 4.3 and Figure 4.4 show rank-sum test of the emotion trust. Result shows the highest value is cluster 2 and lowest in cluster 3. The differences are significant for all pairs of cluster. If the reviews of applications in Cluster 2 are fake positive, then the trust emotion is overrated.

Wilcoxon Scores (Rank Sums) for Variable mean_trust
Classified by Variable ClusterID

ClusterID	N	Sum of Scores	Expected Under H0	Std Dev Under H0	Mean Score
1	67	5417.50	5360.0	286.189193	80.858209
2	56	5767.50	4480.0	276.843634	102.991071
3	36	1535.00	2880.0	242.563651	42.638889

Average scores were used for ties.

Kruskal-Wallis Test

Chi-Square	DF	Pr > ChiSq
37.8191	2	<.0001

Figure 4.3: Rank sum of trust

Pairwise Two-Sided Multiple Comparison Analysis
Dwass, Steel, Critchlow-Fligner Method
Variable: mean_trust

ClusterID	Wilcoxon Z	DSCF Value	Pr > DSCF
1 vs. 2	-3.6456	5.1557	0.0008
1 vs. 3	5.3645	7.5865	<.0001
2 vs. 3	4.6011	6.5070	<.0001

Figure 4.4: Statistics of trust

The second emotion is anger. The Figure 4.5 and Figure 4.6 listed the rank-sum test result. Highest is cluster 1 and lowest in cluster 2. The mean anger of cluster 1 is significantly higher than that of cluster 2 with p-value less than 0.0001. The mean anger of cluster 3 is marginally significantly higher than cluster 2 with p-value less than 0.1. Clusters 1 and 3 are not significantly different in anger.

Wilcoxon Scores (Rank Sums) for Variable mean_anger
Classified by Variable ClusterID

ClusterID	N	Sum of Scores	Expected Under H0	Std Dev Under H0	Mean Score
1	67	6547.0	5360.0	280.351585	97.716418
2	56	3314.0	4480.0	271.196654	59.178571
3	36	2859.0	2880.0	237.615905	79.416667

Average scores were used for ties.

Kruskal-Wallis Test

Chi-Square	DF	Pr > ChiSq
22.3534	2	<.0001

Figure 4.5: Rank sum of anger

Pairwise Two-Sided Multiple Comparison Analysis
Dwass, Steel, Critchlow-Fligner Method
Variable: mean_anger

ClusterID	Wilcoxon Z	DSCF Value	Pr > DSCF
1 vs. 2	4.7993	6.7872	<.0001
1 vs. 3	1.8140	2.5654	0.1649
2 vs. 3	-2.0728	2.9314	0.0955

Figure 4.6: Statistics of anger

This is a surprising result that the successful applications are actually with higher anger emotion. Perhaps because the successful applications with more engagement are throughoutly tested contains more complaints from customer. This also surprised the positive review fakers.

For the rank sum test of the emotion anticipation listed in Figure 4.7 and Figure 4.8, the result is similar to the result of trust. Highest is cluster 2 and lowest in cluster 3. The differences are significant for all pairs.

Wilcoxon Scores (Rank Sums) for Variable mean_anticipation Classified by Variable ClusterID

ClusterID	N	Sum of Scores	Expected Under H0	Std Dev Under H0	Mean Score
1	67	5328.00	5360.0	286.189407	79.522388
2	56	5546.50	4480.0	276.843841	99.044643
3	36	1845.50	2880.0	242.563833	51.263889

Average scores were used for ties.

Kruskal-Wallis Test

Chi-Square	DF	Pr > ChiSq
23.6917	2	<.0001

Figure 4.7: Rank sum of anticipation

The results of emotion disgust are listed in Figure 4.9 and Figure 4.10. Highest is cluster 1 and lowest in cluster 2. The mean disgust of cluster 2 is significantly lower than that of cluster 1 and cluster 3 with p-value less than 0.01. Clusters 1 and 3 are not significantly different. This result is very similar to the previous result of emotion anger.

The result of emotion fear listed in the Figure 4.11 and Figure 4.12. Highest is cluster 1 and lowest in cluster 3. The mean fear of cluster 1 is marginally significantly higher than

Pairwise Two-Sided Multiple Comparison Analysis			
Dwass, Steel, Critchlow-Fligner Method			
Variable: mean_anticipation			
ClusterID	Wilcoxon Z	DSCF Value	Pr > DSCF
1 vs. 2	-3.2075	4.5361	0.0038
1 vs. 3	4.1455	5.8626	0.0001
2 vs. 3	3.5081	4.9613	0.0013

Figure 4.8: Statistics of anticipation

Wilcoxon Scores (Rank Sums) for Variable mean_disgust Classified by Variable ClusterID					
ClusterID	N	Sum of Scores	Expected Under H0	Std Dev Under H0	Mean Score
1	67	6703.50	5360.0	281.012673	100.052239
2	56	3052.50	4480.0	271.836154	54.508929
3	36	2964.00	2880.0	238.176219	82.333333
Average scores were used for ties.					

Kruskal-Wallis Test		
Chi-Square	DF	Pr > ChiSq
31.1858	2	<.0001

Figure 4.9: Rank sum of disgust

Pairwise Two-Sided Multiple Comparison Analysis			
Dwass, Steel, Critchlow-Fligner Method			
Variable: mean_disgust			
ClusterID	Wilcoxon Z	DSCF Value	Pr > DSCF
1 vs. 2	5.5586	7.8610	<.0001
1 vs. 3	1.8803	2.6591	0.1444
2 vs. 3	-3.0507	4.3143	0.0065

Figure 4.10: Statistics of disgust

that of cluster 2 and is significantly higher than that of cluster 3. Clusters 2 and 3 are not significantly different. The emotion of fear is not interpretable.

Wilcoxon Scores (Rank Sums) for Variable mean_fear Classified by Variable ClusterID					
ClusterID	N	Sum of Scores	Expected Under H0	Std Dev Under H0	Mean Score
1	67	6224.00	5360.0	284.560104	92.895522
2	56	4060.50	4480.0	275.267743	72.508929
3	36	2435.50	2880.0	241.182894	67.652778
Average scores were used for ties.					

Kruskal-Wallis Test		
Chi-Square	DF	Pr > ChiSq
9.4663	2	0.0088

Figure 4.11: Rank sum of fear

The sadness emotions are listed in Figure 4.13 and Figure 4.14. The highest is cluster 1 and the lowest is cluster 2. The mean sadness of cluster 2 is marginally significantly lower than that of clusters 1 and 3. Clusters 1 and 3 are not significantly different. This result is also very similar to the anger emotion.

Pairwise Two-Sided Multiple Comparison Analysis			
Dwass, Steel, Critchlow-Fligner Method			
Variable: mean_fear			
ClusterID	Wilcoxon Z	DSCF Value	Pr > DSCF
1 vs. 2	2.1903	3.0976	0.0728
1 vs. 3	3.0133	4.2615	0.0073
2 vs. 3	0.0784	0.1109	0.9966

Figure 4.12: Statistics of fear

Wilcoxon Scores (Rank Sums) for Variable mean_sadness Classified by Variable ClusterID					
ClusterID	N	Sum of Scores	Expected Under H0	Std Dev Under H0	Mean Score
1	67	6368.0	5360.0	284.215209	95.044776
2	56	3373.0	4480.0	274.934111	60.232143
3	36	2979.0	2880.0	240.890574	82.750000
Average scores were used for ties.					

Kruskal-Wallis Test		
Chi-Square	DF	Pr > ChiSq
17.9109	2	0.0001

Figure 4.13: Rank sum of sadness

Pairwise Two-Sided Multiple Comparison Analysis			
Dwass, Steel, Critchlow-Fligner Method			
Variable: mean_sadness			
ClusterID	Wilcoxon Z	DSCF Value	Pr > DSCF
1 vs. 2	4.2136	5.9589	<.0001
1 vs. 3	1.2812	1.8118	0.4058
2 vs. 3	-2.3497	3.3230	0.0492

Figure 4.14: Statistics of sadness

The result of joy emotion listed in the Figure 4.15 and Figure 4.16. The highest is cluster 2 and the lowest is cluster 3. The differences are significant for all pairs. This is a overrated example of Cluster 2 similar to trust.

Wilcoxon Scores (Rank Sums) for Variable mean_joy Classified by Variable ClusterID					
ClusterID	N	Sum of Scores	Expected Under H0	Std Dev Under H0	Mean Score
1	67	5143.50	5360.0	285.898835	76.768657
2	56	5788.50	4480.0	276.562757	103.366071
3	36	1788.00	2880.0	242.317554	49.666667
Average scores were used for ties.					

Kruskal-Wallis Test		
Chi-Square	DF	Pr > ChiSq
30.5431	2	<.0001

Figure 4.15: Rank sum of joy

The result of surprise emotion listed in Figure 4.17 and Figure 4.18. The highest is cluster 2 and the lowest is cluster 3. The mean surprise of cluster 3 is significantly lower than that of clusters 1 and 2. The mean surprise of cluster 2 is marginally significantly

Pairwise Two-Sided Multiple Comparison Analysis			
Dwass, Steel, Critchlow-Fligner Method			
Variable: mean_joy			
ClusterID	Wilcoxon Z	DSCF Value	Pr > DSCF
1 vs. 2	-4.0038	5.6623	0.0002
1 vs. 3	3.9554	5.5938	0.0002
2 vs. 3	4.2106	5.9546	<.0001

Figure 4.16: Statistics of joy

higher than cluster 1. From this statistics, surprise emotion is also overrated emotion in the Cluster 2.

Wilcoxon Scores (Rank Sums) for Variable mean_surprise Classified by Variable ClusterID					
ClusterID	N	Sum of Scores	Expected Under H0	Std Dev Under H0	Mean Score
1	67	5542.0	5360.0	284.871422	82.716418
2	56	5283.0	4480.0	275.568895	94.339286
3	36	1895.0	2880.0	241.446756	52.638889
Average scores were used for ties.					

Kruskal-Wallis Test		
Chi-Square	DF	Pr > ChiSq
18.6115	2	<.0001

Figure 4.17: Rank sum of joy

Finally, we check the vader compound scores of the reviews from applications. The vader compound scores is a measurement of overall positive or negative. From the result listed in the Figure 4.19 and Figure 4.20, we could notice that the highest is cluster 2 and the lowest is cluster 1. The mean vader of cluster 3 is significantly lower than the other two

Pairwise Two-Sided Multiple Comparison Analysis			
Dwass, Steel, Critchlow-Fligner Method			
Variable: mean_surprise			
ClusterID	Wilcoxon Z	DSCF Value	Pr > DSCF
1 vs. 2	-2.2521	3.1850	0.0628
1 vs. 3	4.3274	6.1199	<.0001
2 vs. 3	2.9571	4.1819	0.0087

Figure 4.18: Statistics of joy

clusters. Clusters 1 and 2 are not significantly different in vader. This results reveals the limitations of simple over-all positive/negative measurement, which can be easily faked by paid reviews.

Wilcoxon Scores (Rank Sums) for Variable mean_vader Classified by Variable ClusterID					
ClusterID	N	Sum of Scores	Expected Under H0	Std Dev Under H0	Mean Score
1	67	5945.0	5360.0	286.681314	88.731343
2	56	5747.0	4480.0	277.319685	102.625000
3	36	1028.0	2880.0	242.980755	28.555556
Average scores were used for ties.					

Kruskal-Wallis Test		
Chi-Square	DF	Pr > ChiSq
60.8724	2	<.0001

Figure 4.19: Rank sum of vader scores

Pairwise Two-Sided Multiple Comparison Analysis			
Dwass, Steel, Critchlow-Fligner Method			
Variable: mean_vader			
ClusterID	Wilcoxon Z	DSCF Value	Pr > DSCF
1 vs. 2	-1.8029	2.5497	0.1686
1 vs. 3	6.5015	9.1945	<.0001
2 vs. 3	7.2963	10.3186	<.0001

Figure 4.20: Statistics of vader scores

4.2.3 Analysis of Other Linguistic Dimensions

After check the basic emotions, we now check the other statistics in other linguistic dimensions provided by LIWC-22. We first check the emotional tone listed in Figure 4.21 and Figure 4.22. The highest is cluster 2 and the lowest is cluster 3. The differences are significant for all pairs. So, cluster 2 has the most positive tone while cluster 3 has the most negative tone. This result is consistent with the vader compound scores.

Wilcoxon Scores (Rank Sums) for Variable mean_Tone Classified by Variable ClusterID					
ClusterID	N	Sum of Scores	Expected Under H0	Std Dev Under H0	Mean Score
1	67	5276.0	5360.0	286.606844	78.746269
2	56	6358.0	4480.0	277.247646	113.535714
3	36	1086.0	2880.0	242.917637	30.166667
Average scores were used for ties.					

Kruskal-Wallis Test		
Chi-Square	DF	Pr > ChiSq
71.9654	2	<.0001

Figure 4.21: Rank sum of emotional tones

Pairwise Two-Sided Multiple Comparison Analysis			
Dwass, Steel, Critchlow-Fligner Method			
Variable: mean_Tone			
ClusterID	Wilcoxon Z	DSCF Value	Pr > DSCF
1 vs. 2	-5.0785	7.1821	<.0001
1 vs. 3	6.3324	8.9554	<.0001
2 vs. 3	7.0371	9.9520	<.0001

Figure 4.22: Statistics of emotional tones

The next dimension is the authenticity (honesty vs. evading) in the language. The results are listed in the Figure 4.23 and Figure 4.24. Texts scoring high on authenticity use more I words, present-tense verbs, and relativity words that contain time. Cluster 2 has significantly lower authenticity than the other two clusters. This shows the great usefulness of authenticity measurement.

Wilcoxon Scores (Rank Sums) for Variable mean_Authentic Classified by Variable ClusterID					
ClusterID	N	Sum of Scores	Expected Under H0	Std Dev Under H0	Mean Score
1	67	6926.00	5360.0	286.561681	103.373134
2	56	2118.50	4480.0	277.203958	37.830357
3	36	3675.50	2880.0	242.879359	102.097222
Average scores were used for ties.					

Kruskal-Wallis Test		
Chi-Square	DF	Pr > ChiSq
72.5913	2	<.0001

Figure 4.23: Rank sum of authenticity

Pairwise Two-Sided Multiple Comparison Analysis			
Dwass, Steel, Critchlow-Fligner Method			
Variable: mean_Authentic			
ClusterID	Wilcoxon Z	DSCF Value	Pr > DSCF
1 vs. 2	8.0530	11.3887	<.0001
1 vs. 3	-0.1280	0.1810	0.9910
2 vs. 3	-6.2272	8.8066	<.0001

Figure 4.24: Statistic of authenticity

Words per-sentence is surprisingly a good indicator of positive faker. The results of words per-sentence are listed in the Figure 4.25 and Figure 4.26. The value of Cluster 1 is the highest while cluster 2 is the lowest. This simple measurement shows that the real positive reviews on the successful applications are significantly higher than the unsuccessful ones.

Wilcoxon Scores (Rank Sums) for Variable mean_WPS Classified by Variable ClusterID					
ClusterID	N	Sum of Scores	Expected Under H0	Std Dev Under H0	Mean Score
1	67	7653.0	5360.0	286.678319	114.223881
2	56	2709.0	4480.0	277.316787	48.375000
3	36	2358.0	2880.0	242.978216	65.500000
Average scores were used for ties.					

Kruskal-Wallis Test		
Chi-Square	DF	Pr > ChiSq
67.0076	2	<.0001

Figure 4.25: Rank sum of words per-sentence

Pairwise Two-Sided Multiple Comparison Analysis			
Dwass, Steel, Critchlow-Fligner Method			
Variable: mean_WPS			
ClusterID	Wilcoxon Z	DSCF Value	Pr > DSCF
1 vs. 2	7.7070	10.8993	<.0001
1 vs. 3	5.3637	7.5855	<.0001
2 vs. 3	-2.0282	2.8682	0.1055

Figure 4.26: Statistics of words per-sentence

In the last check, we also list the result of using big words in Figure 4.27 and Figure 4.27. The cluster 2 has the highest number of big words used. This can be explained by the fact that words can be easily replaced by a bot of dictionary. Therefore, the number of big words is not a good indicator.

Wilcoxon Scores (Rank Sums) for Variable mean_BigWords Classified by Variable ClusterID					
ClusterID	N	Sum of Scores	Expected Under H0	Std Dev Under H0	Mean Score
1	67	5334.00	5360.0	286.674681	79.611940
2	56	5257.50	4480.0	277.313268	93.883929
3	36	2128.50	2880.0	242.975133	59.125000
Average scores were used for ties.					

Kruskal-Wallis Test		
Chi-Square	DF	Pr > ChiSq
12.4971	2	0.0019

Figure 4.27: Rank sum of big words

Pairwise Two-Sided Multiple Comparison Analysis			
Dwass, Steel, Critchlow-Fligner Method			
Variable: mean_BigWords			
ClusterID	Wilcoxon Z	DSCF Value	Pr > DSCF
1 vs. 2	-2.2601	3.1962	0.0616
1 vs. 3	2.8980	4.0984	0.0105
2 vs. 3	2.6604	3.7624	0.0213

Figure 4.28: Statistics of big words

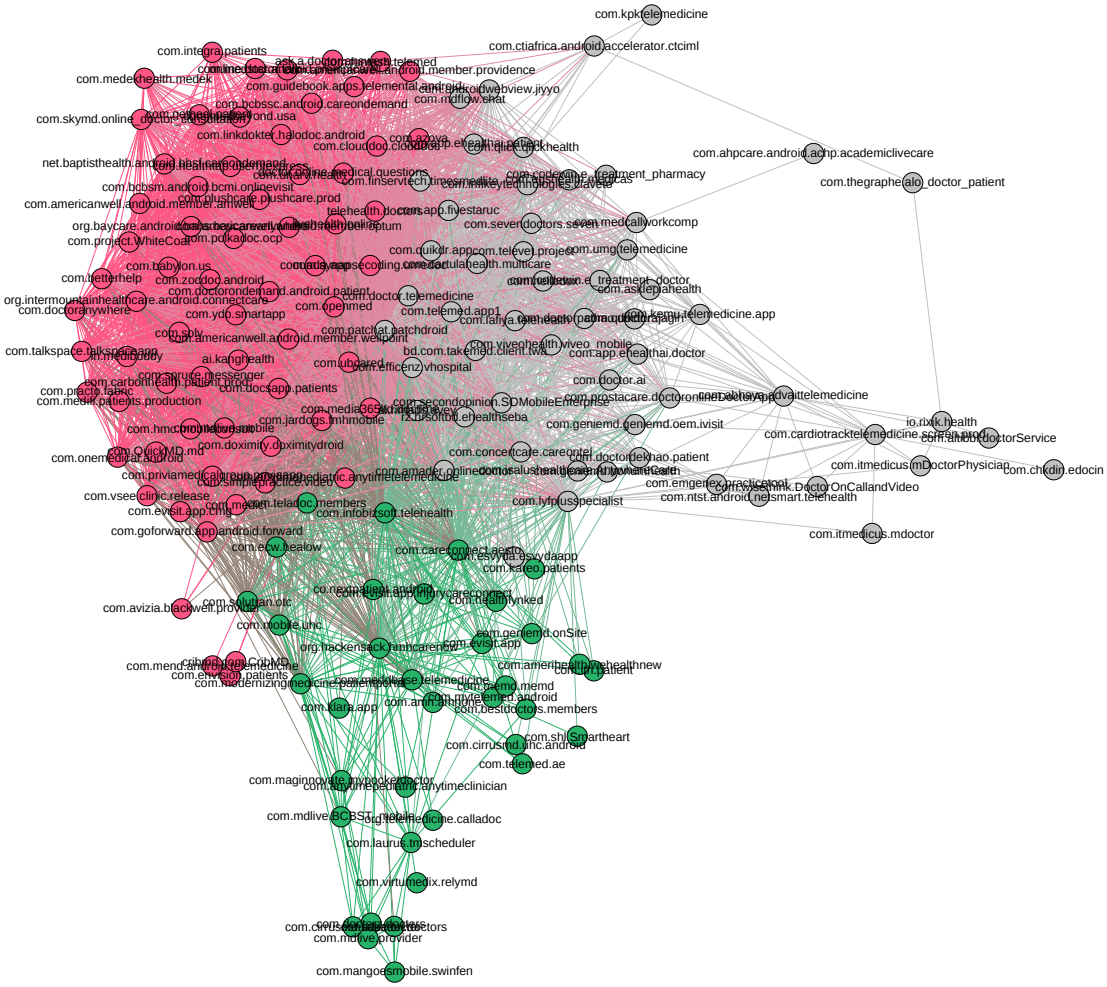

Figure 4.29: Clustering Result of Applications

4.2.4 Cluster of Successful Applications

In this section, we describe the general network characteristics of the cluster successful applications and then illustrate a representative example application in the cluster. Figure 4.30 reveals the first application cluster, which is centered around the applications of OpenMed, UBCared, and to name just a few. The edge of each connected application indicates the similarity between each other, and the similarity is derived from the collection of application reviews. With the same procedure, we only incorporate edges indicating higher similarity among applications. Through the trial-and-error test, we obtain only 10+ applications with more than two edges, including OpenMed, UBCared, PlusCare Prod, and SmartApp in the case of threshold being 0.35. These applications are at the center of Figure 4.30, indicating that these applications are the most representative ones.

Figure 4.30: The first cluster of Apps

Figure 4.31 unveils one of the most representative application: OpenMed: Doctors Near Me & Onl. This application has approximately 1.42K reviews with a rate score of 4.8 (total rate score is in a 1-5 scale), and more than 10K downloads. This application dedicates to help users to make an online doctor appointment requests for a huge number united healthcare providers in the United States.

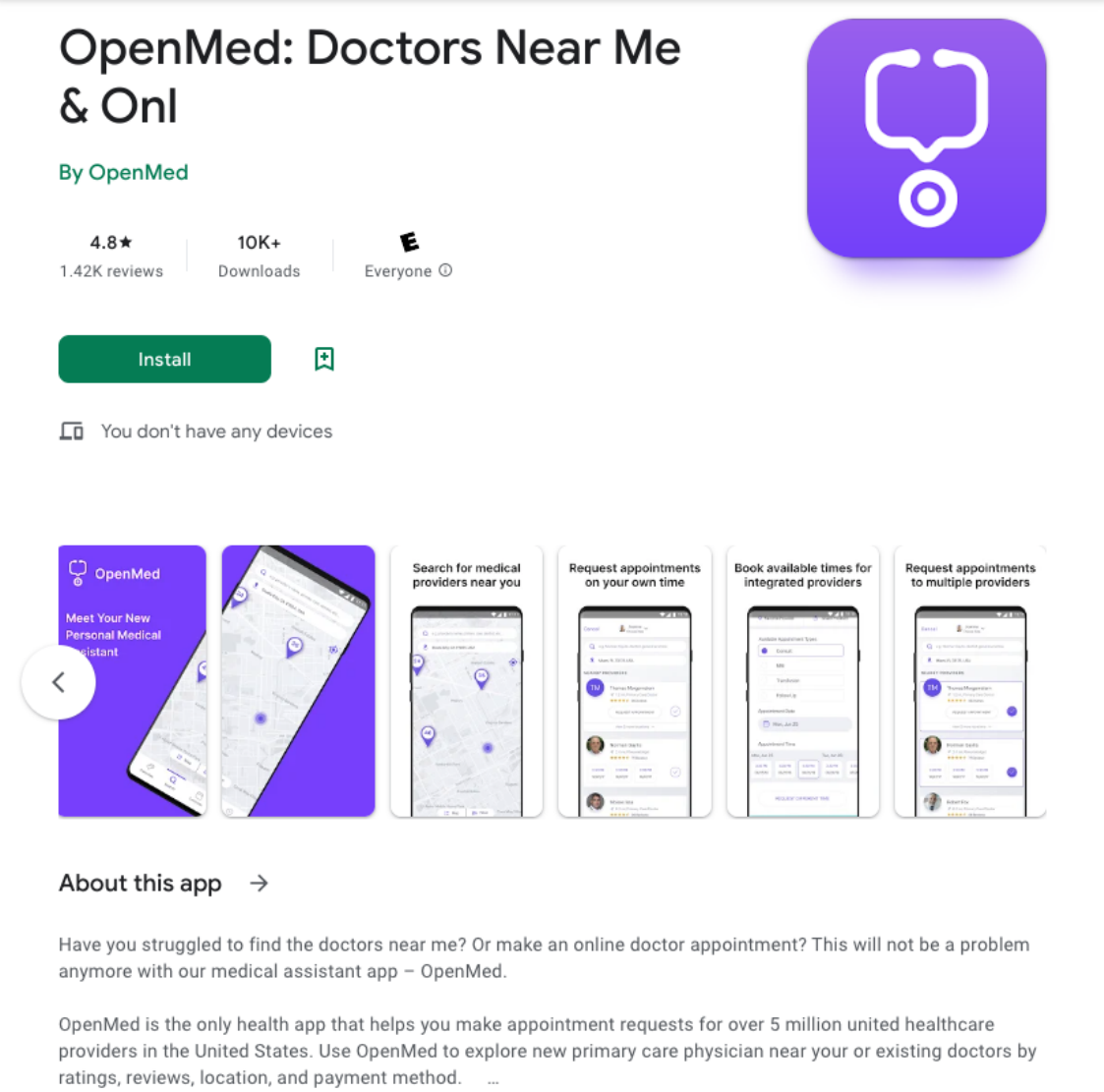

Figure 4.31: Example App in Cluster 1

Figure 4.32 is a screenshot of the reviews and ratings of the application OpenMed, which boasts a 4.8 rating with a vast majority of 5 stars. Two reviews are selected on the front page as these reviews are rated the most helpful ones by more than 5 people; We observed

that these reviews not only show affirmative feedback such as convince and usefulness, but also provides additional details on various issues, especially the issues of unreasonable and non-straightforward design of user interface. A list of sample user reviews are given in Figure 4.32

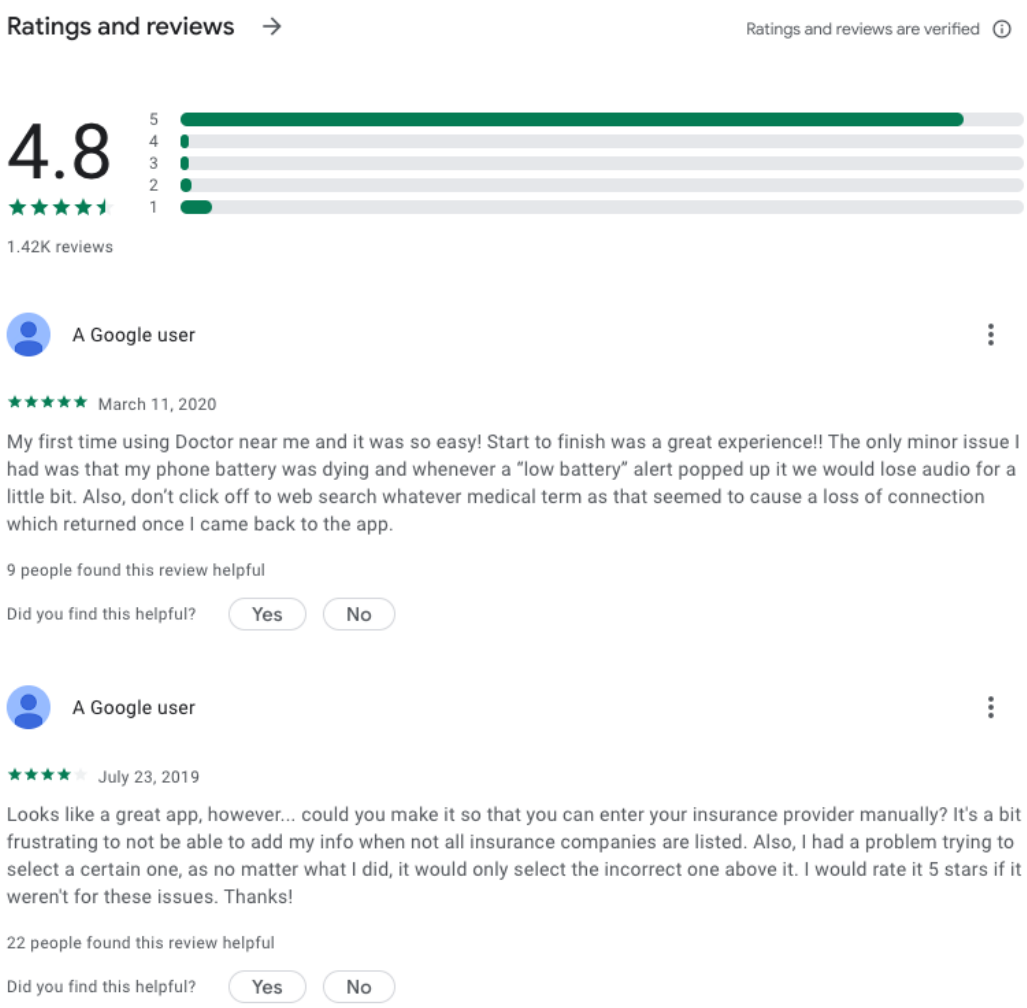

Figure 4.32: Ratings and Reviews of the representative app in Cluster 1

4.2.5 Cluster of Applications with Fake Positive

In this section, we describe the general network characteristics of the cluster successful applications and then illustrate a representative example application in the cluster of applications with fake positive. Figure 4.33 plots a cluster of applications with fake positive

reviews. Compared with the first application cluster, these applications are less popular with limited number of reviews and downloads. More specifically, most of the applications in this cluster have merely one review on the front page, where reviews tend to be short and affirmative content. Due to these traits, we gather the applications within a cluster. After the aforementioned trial-and-error procedure, the cluster is centered around applications such as QuickDr, HelloDox, and TakeMed. Only 10+ applications are jointed with multiple edges under the threshold of 0.35. The edges of jointed dots - representing applications - imply the resemblance among the applications.

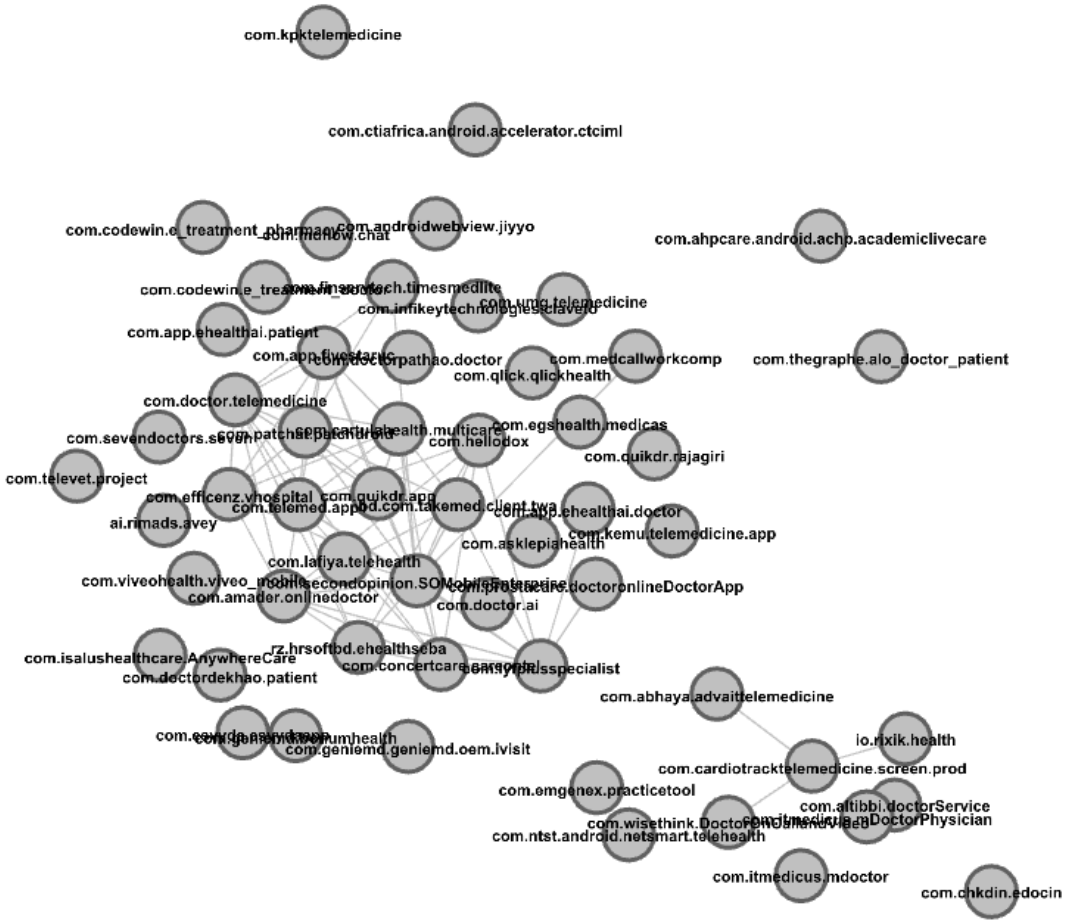

Figure 4.33: The second cluster of applications

Among the representative telemedicine applications, Figure 4.34 is an example application in the second largest cluster. The application, QuickDr - Consult Doctor Online, aims

at helping users connect with qualified doctors online in India. While this application only has been downloaded 5000+ times, and no reviews are selected on the front page of Google Play.

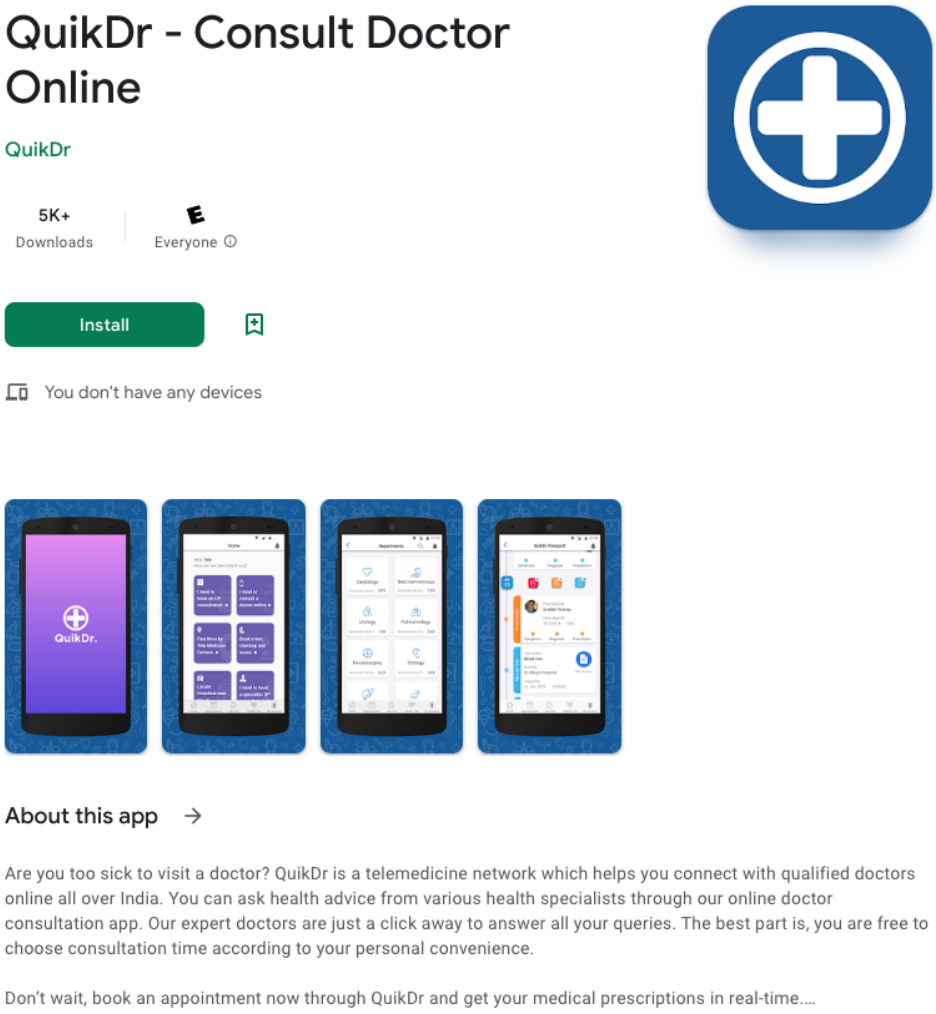

Figure 4.34: Example application in the second largest cluster

The three example reviews in Table 4.1 are fetched from the background from the application QuickDr. Two of them are extremely positive evaluations but with limited and broad descriptions. The last one is a negative review about user's subjective experience. Though the second review is rated as helpful by over 10 users, the content is subjective and broad. Google may consider it is rated by trolls and decide not to put this review on the

Content	score	thumbs up count	at
Excellent service!! Very prompt and very efficient. Doctor's medicines and tests all in one. One app for everything. Would definitely recommend other's to download it.	5	2	6/11/2020 10:38:55 PM
Superb app! I had a mild fever and was very much worried. I took a video consultation from this app and the doctor treated me well. He prescribed the best medicines and cleared all my doubts. also the support service was really great! Highly recommended.	5	11	2/13/2021 1:34:11 AM
Consultation fees is too high. I think they are targeting only rich persons	2	1	3/14/2022 6:12:50 AM

Table 4.1: Three example reviews from the application QuickDr

front page. The other two reviews are rated by less than 5 users, which may not represent the perspective of majority users, and Google hide these reviews as well.

4.2.6 Cluster of Unpopular Applications

In this section, we describe the general network characteristics of the cluster of unpopular telemedicine applications and then illustrate a representative example application in the cluster. Figure 4.35 is the cluster of unpopular telemedicine applications. Though they are still gathered as a cluster, they are not connected as intimately as the other two clusters(cluster of successful telemedicine applications and clusters of telemedicine applications with fake positive). The reason why the third cluster lacks of densely connection is that these applications are less downloaded and rated by users. Through the same processing procedures, the cluster is centered around the application of HMH NOW and has sparse links(under the threshold of 0.35).

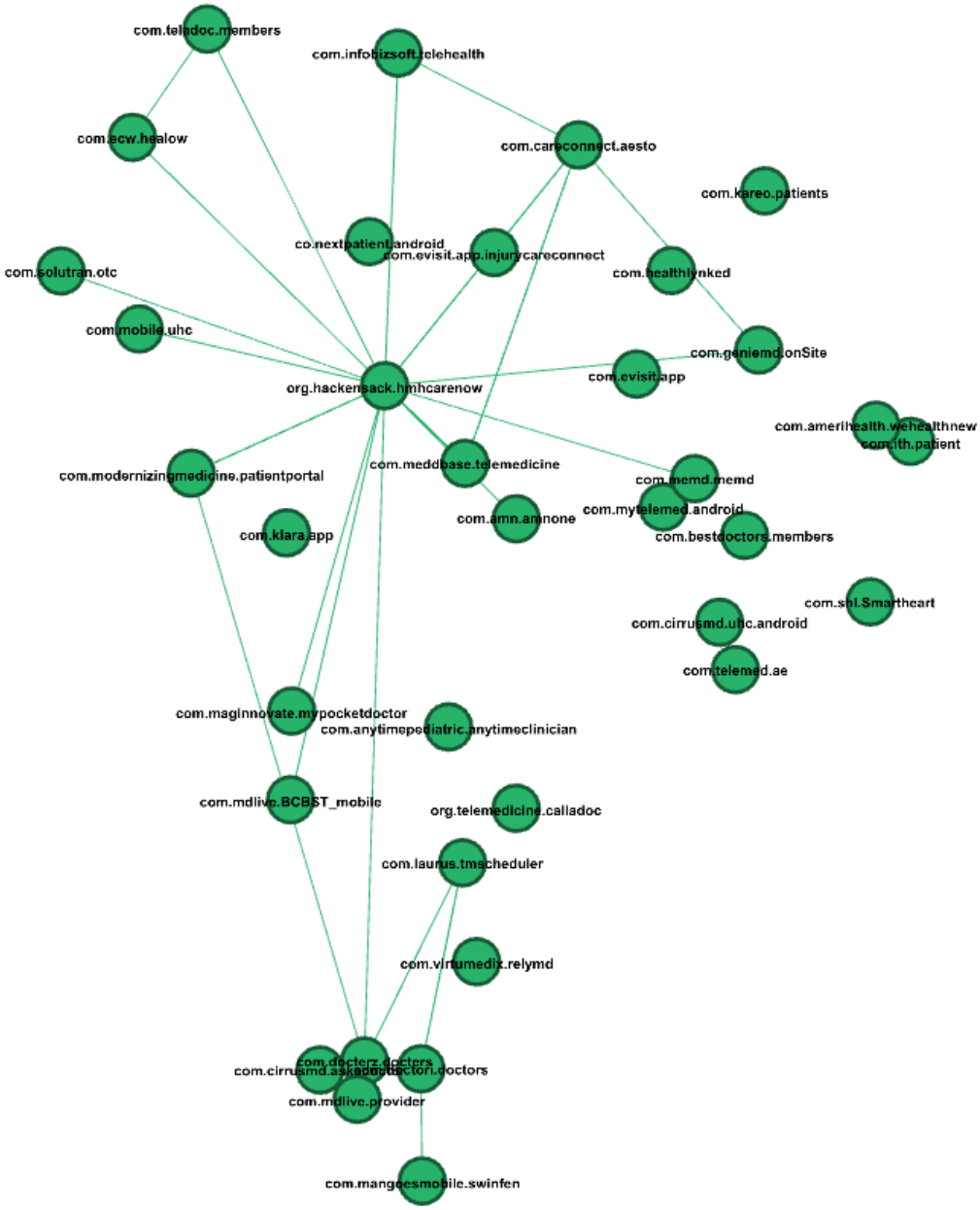

Figure 4.35: The third cluster of Apps

Figure 4.36 is a representative application in the third cluster. As we can see from this figure, it has only about 1000 downloads and no reviews or star rates are listed on the front page. Even this application still targets on providing online services of doctors.

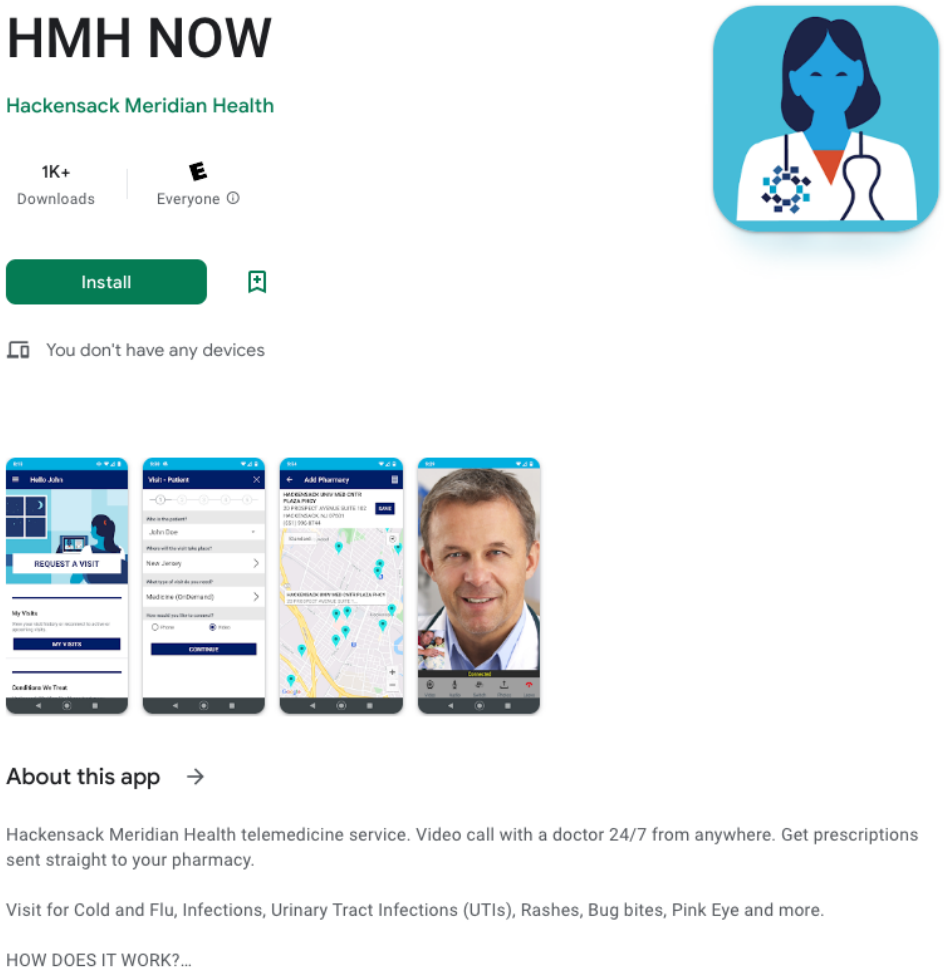

Figure 4.36: Example application in the third largest cluster: HMH NOW

Table 4.2 is the only review that we can fetch from the background. Though this negative review complaining about a specific user situation, due to its lack of user amount, it is still considered as useless by Google Play and hided from the front page.

Content	score	thumbs up count	at
This app doesn't work. I tried resetting my password and it did nothing. Now the app crashes. Terrible.	1	0	2022-01-04 08:15:54

Table 4.2: The only review from the application HMH NOW

4.3 Discussions

The experimental results in this section show the great potential of combining the network analytic approach and deep learning model embedding from SBERT. The similarity-based connections between applications illustrate the capability of handling complicated relationship understanding as a unsupervised learning problem. The deep learning model SBERT show strong practical usefulness in the framework.

The sentiment analysis of the basic emotions from reviews reveals the following patterns in the telemedicine application:

- The successful telemedicine applications also contain more negative emotions especially anger, disgust and sadness.
- The positive emotion such as trust, anticipation, joy, surprise are easy to fake by unpopular telemedicine application, but those positive review faker hard to fake reviews with complicated and multiple emotions.
- The overall positiveness indicator such as vader score and emotional tone are very limited to evaluate the telemedicine applications.
- Some simple indicator, such as authenticity and words per sentence are not easy to fake, while some other indicator such as the usage of big words is easy to fake.

The first pattern discovered is the most surprising finding in the research. This suggest the representation of emotions should be a multi-dimensional distribution, which can better handle the emotion complexity especially for users of telemedicine applications. It also

supports the existence of inter-play between different emotions. As more user engagement with the telemedicine application, more complaints will be generated on website as feedback, therefore, the positive and negative emotions may correlated for users of telemedicine applications. This study also tells us that the profile of unpopular telemedcine applications. The unpopular telemedcine applications can easily fake trust, anticipation, joy, surprise emotions by buying comments say "I trust this telemedicine app", "This telemedicine app is exactly what I expected" or "It surprises me with joy", however, they can not fake complicate emotions, which are the true experience of telemedicine users. This allows even simple indicator such as words per sentence help us identify the fake positive reviews.

Chapter 5

Improving Robustness via Large-difference Transformation

In this paper, I proposed a new defense method using primitive-based transformation which makes a large difference between the input image and the image after transformations. Based on an original input image, the primitive-based transformation generates a new image composed of colourful triangles, which are the basic 2D drawing primitives. The colours and the coordinates of the triangles are optimized in such a way that the final picture is similar to the original image . This optimization problem is solved by the hill-climbing algorithm due to its simplicity. Since the new images are reconstructed in completely different ways, the difference is significantly larger than normal adversarial perturbations. I hypothesize that it not only captures the main visual features due to the similarity optimization , but also suppresses the adversarial noise due to large difference. Thus, the adversarial example can be purified before being fed into the classifier. I later show experimentally that our method improves robustness of the classifier in terms of significantly large distortion required to be broken under strong attack compared to other state-of-the-art defense method. This chapter is organized as follows. Motivation and challenge are explained in the Section 5.1. The details of the proposed defense approach is described in Section 5.2. The strong attack algorithm (BPDA) used in the benchmark was explained in Section 5.3. The experiment results were demonstrated in the Section 5.4 and Section 5.5. Finally, the results and buisness implications were discuss in Section 5.6.

5.1 Motivation and Challenges

I describe the motivation and major challenge in this section. In Subsection 5.1.1, the basic question was formulated as a optimization problem. In Subsection 5.1.2, the challenge

of strong attacks was explained. To mitigate the challenge of strong attacks, I proposed the strategy which was described in the Subsection 5.1.3.

5.1.1 Optimization Algorithms

Consider a neural network classifier $c(\cdot)$ that attempts to predict a true label y given an input $\mathbf{x}$. The classifier defined as $c(\mathbf{x}) = \arg\max_i f(\mathbf{x})_i$. An *adversarial example* $\mathbf{x}'$ is a slight perturbation of the nature input $\mathbf{x}$, such that $c(\mathbf{x}') \neq c(\mathbf{x}) = y$. In image classification task, the adversarial example can be made visually indistinguishable from the original image for human, $|\mathbf{x}' - \mathbf{x}| < \epsilon$. Distance metric $|\cdot|$ often use l_p-norms. Search adversarial example is an optimization problem maximize a differential loss function $L(\cdot)$ and minimize the perturbation. The loss function measures the difference between correct label and predicted label on an adversarial input.

$$\min |\mathbf{x}' - \mathbf{x}| + \max L\left(c(\mathbf{x}') - c(\mathbf{x})\right) \tag{5.1}$$

In this paper, we discuss a *defense* which is a transform $g(\cdot)$ that attempts to fix the adversarial example x' and make the same correct prediction as on a nature input example x. As a protection, the $g(\cdot)$ usually are non-differentiable to prevent from break via reverse engineering.

$$c\left(g(\mathbf{x}')\right) = c(\mathbf{x}) = y \tag{5.2}$$

Attackers are assumed to have different levels of knowledge of machine learning model in different settings. In a *gray-box* setting, attackers have the completed information of $c(\cdot)$ but not of $g(\cdot)$, especially the gradient. While in a *white-box* setting, attackers have complete gradient information of the whole model or defense simply does not exits. Gray-box setting typically happened when a defense mechanism introduced and used when strictly evaluate

the effectiveness. To have meaningful and practical result, we emphasis that the general defense mechanism should be held in as transparent scenario as possible.

5.1.2 Strong Attacks

Approximation of the inaccessible gradients is a common practice that facilitates the attackers to bypass the defense, since gradients are often used by Equation 5.1 to generate adversarial example. It downgrades the black/gray box to nearly white box scenario. Specifically, besides neural network $c(\cdot)$ and the loss function $L(\cdot)$ of it are usually differentiable, transform can also be estimated approximately $\nabla g(\cdot) \approx 1$ if the transform difference is small $g(\mathbf{x}) \approx \mathbf{x}$ as in many previous works. This is vulnerable to white-box attacks because the entire gradient is now accessible despite the fact that $g(\cdot)$ is non-differentiable.

$$\begin{aligned}\nabla L\left(c\left(g(\mathbf{x}')\right)-y\right) =& \nabla L(m)|_{m=c(g(\mathbf{x}'))-y} \nabla c(n)|_{n=g(\mathbf{x}')} \nabla g(\mathbf{x}') \\ \approx& \nabla L(m)|_{m=c(g(\mathbf{x}'))-y} \nabla c(n)|_{n=g(\mathbf{x}')}\end{aligned} \tag{5.3}$$

5.1.3 Large Transformations

We propose here a new defensive mechanism called *large transform*. Ideally, it is an image transform made significant change, $|g(\mathbf{x}) - \mathbf{x}| > \delta$ and also holds (2). The gradient of $g(\mathbf{x})$ should be much larger than 1 if the δ is significant and $|\Delta\mathbf{x}|$ is small. We are interested in large transformation because it can disrupt (3) therefore improves the resistant against the attacks based-on it.

$$|\nabla g(\mathbf{x})| > \frac{\delta}{|\Delta\mathbf{x}|} + 1 \gg 1 \tag{5.4}$$

5.2 Primitive-based Transformation as a Defender

The goal of primitive-based transform is to filter out the adversarial noise by reconstruction of the original image using basic geometric shapes, therefore serving as a defense

mechanism. The reconstructed image preserves the major visual features of the objects in general, which can be identified easily by human, while significantly different from the original image in pixel-by-pixel level. Intuitively, the transformation can suppress the adversarial example, if the difference of the image before and after transformation is larger than the adversarial perturbation.

In this paper, we use Michael Fogleman's implementation of such transformation, namely "Primitive". It has the effect composing a new image that close to the original input image with certain number of colourful drawing primitives. We choose triangles as drawing primitives here. As described in the pseudo-code Algorithm 1, the algorithm sets the original image as target and tries to find the single most optimal triangle that can be drawn to minimize the error between the target and the new drawn image in each step. Staring from a blank canvas, it repeatedly adds one triangle at a time until total number of triangles added reach the setting value.

Figure 5.1: Reconstructed images with 10 (left), 100 (middle) and 1000 triangles (right) from scratch. As more triangles are added, the reconstructed image resembles more like the original reference (input image).

The algorithm takes hill climbing approach to find a proper triangle in each step. A triangle is generated randomly first and then scored by computing root-mean-square error between target and new drawn image after that triangle added. Then we mutate the triangle by tweaking a vertex and score it again. If the mutation improved the score (decrease in error), we keep it. Otherwise we rollback to the previous state. Since hill climbing is prone to getting stuck in local minima, the algorithm could conduct this many times with several

different starting triangles. Multiple random triangles are generated and the best one is picked before hill climbing.

We perform 16 times of random start and use 100 to 2000 triangles in experiment. Parallelism on multiple threads in practical speeds up the process. It also reduces running time to re-draw and re-calculate error only on updated area that described by scan lines instead of the whole canvas. For simplicity we briefly describe the method in the pseudo-code. The full details can be found from the source code used in experiments.

5.3 The Strong Attack: Backward Pass Differentiable Approximation (BPDA)

To test the defense effectively, we need to focus on the strongest attack possible. The main tool in our gray-box experiment is BPDA. BPDA is a powerful gradient estimation approach that has already broken many defense mechanism since it can back-propagation through non-differentiable transform. Pseudo-code (Algorithm 2) described BPDA implementation of the formulations discussed in Section 5.1.2. The classifier can be unfolded as $c(\mathbf{x}) = \arg\max_i f(\mathbf{x})_i$, where $f(\mathbf{x})$ is the logits output of neural network. The forward direction calculation $f(g(\mathbf{x}))$ combines with the defense $g(\mathbf{x})$, while the backward direction only need to calculate the deviates of $f(\cdot)$ and ignored $g(\mathbf{x})$, since the deviates of $g(\mathbf{x})$ approximately equals 1. Therefore, the whole algorithm is similar to PGD except the only difference is that we compute the transformed input(in this case the transform is “Primitive") forward. We set the number of iterations to 30 in our experiments.

5.4 Experiments and Results

Our proposed method uses the primitive-based transform as a defender to protect a DNN classifier from white-box and grey-box attacks. Since no method of complete robustness to adversarial attacks has been discovered yet and all defenses eventually been broken by attackers, our experiments focus on comparing the largest distortion of adversarial examples to benchmark the adversarial robustness improvement.

5.4.1 Experiment Setup

We choose a subset of ImageNet as our test set and pre-trained DNN classifier Inception v3 as the original classifier. Due to huge computational cost in image reconstruction step, primitive-based transformation is a time-consuming process, therefore, we benchmark the robustness on the subset of ImageNet, which is constructed by randomly picking 1000 images from the ILSVRC 2012 validate set of it.

5.4.2 Baseline: Defense in a None-Attack Setting

We first evaluate the effect of defense after being deployed as a baseline for late comparison. The classifier is not under any attacks.

Experiment The schemes of the original and protected classifiers without any attack are given in Figure 5.2. We collect the accuracy of (a) original Inception v3 classifier and (b) protected classifier after adding only the primitive-based transformation as defense on the subset of ILSVRC 2012 validate set. We also collect the average L_2 distances between transformed images and the input images over difference number of triangles in (b).

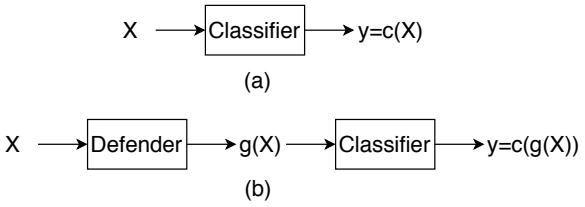

Figure 5.2: Original (a) and protected classifier (b) without any attack.

Results The accuracy of Inception v3 test in (a) is 76.5% , which is in line with the performance reported by [112]. We also know that increasing the size of test set does not affect the distortion of cases that have already been successfully attacked. Since the accuracy of the subset is in line with experiment on the full set, the size of the subset is sufficient for our research that mainly focusing on the distortion comparison.

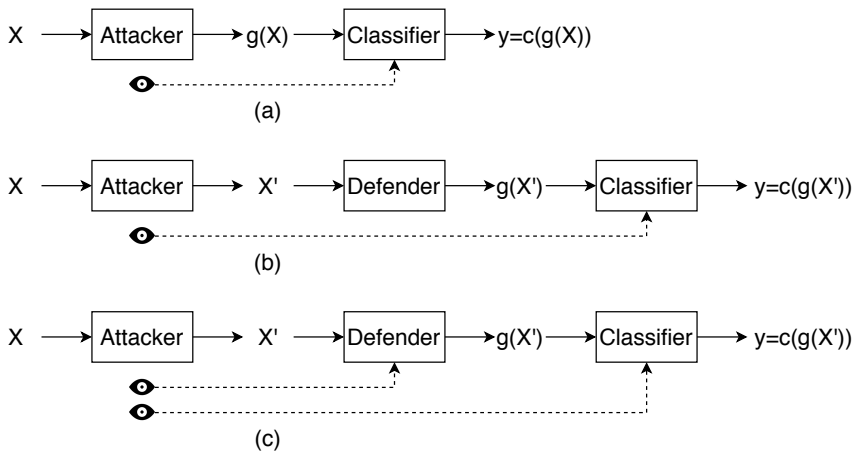

Figure 5.3: Attacks to original classifier without any defender (a), vanilla attack to the classifier protected by the defender (b) and strong attacks to the classifier protected by the defender (c)

The deployment of our primitive-based transformation decreases accuracy of (b) from 76.5% to 55.3% even without any attack. This indicates that the pre-trained classifier is not fully adapted to the transformed images. The large transformation of input made them out of the distribution of the previous training set. Introduce of the transformation reduces the overall performance. However, the classifier are still able to maintain the accuracy of 55.3%. This implies that the primitive-based transformation reserved the main visual features of the input images. The two schemes are also plotted in the Figure 5.6 as the original (no attack) and the protected (no attack) curve.

The threat models of adversary attacks usually use L_2 distance of 0.05–0.06 as the boundary of distortion. The distortions of transformed images (reconstructed) using different numbers of triangles at least 12.0 are significantly larger compared to that boundary (Figure 5.4). The primitive-based transformation caused a very large distortion by reconstruct the image with triangles.

5.4.3 White Box: Attack on Original Classifier

To study the performance of the original classifier under L_2 attack, we conduct experiment on white box setting. Note that since the transformation used by the defense

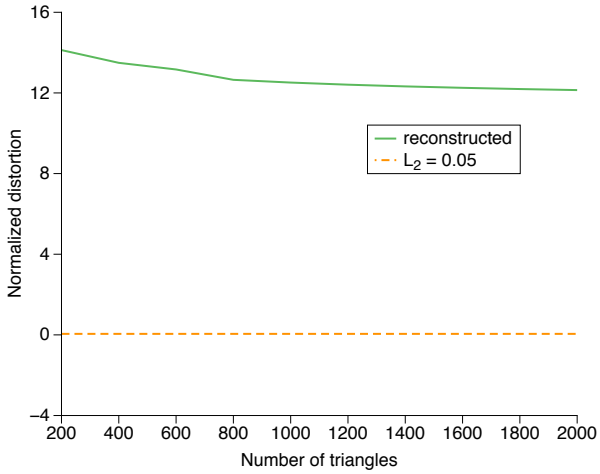

Figure 5.4: Reconstructed distortion

is non-differential, and the attacker can not directly access the gradients, by definition we consider all experiments of the protected classifier under attack as grey box setting in this paper.

Experiment We use the PGD as the attacking method with maximum 30 iterations. As shown by Figure 5.3 (a), attackers targeting original classifier under white-box setting can access all information of the classifier.

Results As plotted in Figure 5.5, the accuracy of the original classifier without any protection quickly drops from 76.5% to 5.6% within a very short L_2 distortion 0.01 and eventually drops to 3.7% within 0.02. As we expected, this shows vulnerability of original classifier under white-box setting.

5.4.4 Grey Box: Attack on Protected Classifier

After introducing the primitive-based transformation as defense, the testing scenario becomes gray box setting. We test the performance of the protected classifiers both follow the same scheme in white box and scheme unique in gray box. The former scheme dose not fully customized to the gray box scenario, and the attack is not as strong as the one in

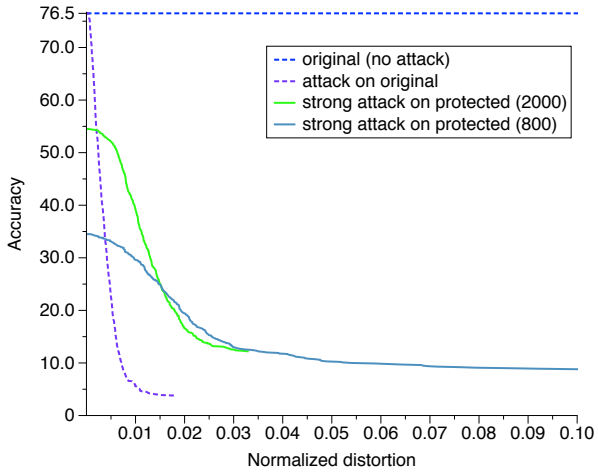

Figure 5.5: distortion summary.

latter scheme, which is customized and provides strong attack to convincingly benchmark the robustness of the defense.

Experiment The vanilla attack scheme using the same attack as Figure 5.3 (a) in white box setting is described in Figure 5.3 (b). Figure 5.3 (c) describes a strong attacks scheme that take fully advantage of both information of the defense and the classifier. Following the vanilla attack and the strong attack scheme, we collect (1) the accuracy and (2) the largest distortions of them when fully broken over different number of triangles, because the number of triangles used by the primitive-based transformation affects the overall performance of the experiments.

Results The relationship of accuracy and triangles on different schemes are plotted in Figure 5.6. Overall the accuracy increases along with the increasing number of triangles. Intuitively, the more triangles used, the more detail of the image can be presented. However, we find that the accuracy plateaued as the number of triangles increases. We observe limited accuracy increase when the number of triangles reaches 2000.

Over the number of triangles from 200 to 2000, the results of the L_2 distortions of all protected classifiers (strong attack) are plotted in Figure 5.7. Again, we find that classifiers

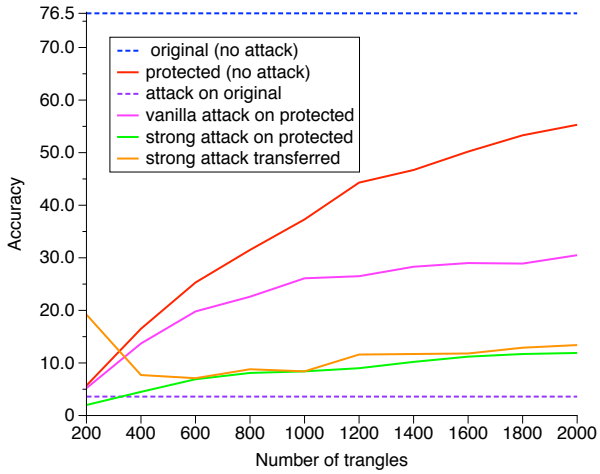

Figure 5.6: triangles summary.

with larger number of triangle has higher accuracy almost at all distortions. Specifically, the accuracy of the most classifiers, with number of triangles larger than 600, decreases slowly within 0.01 distortion, and then rapidly decrease when distortions are larger than 0.01. The accuracy eventually plateaued when decrease to 12%. This result shows that the distortions of 0.05–0.06 are not enough to reach 100% successfully break the protected classifier even for strong attack BPDA. The primitive-based transformation requires significant larger distortion to be broken and demonstrates improvement of robustness.

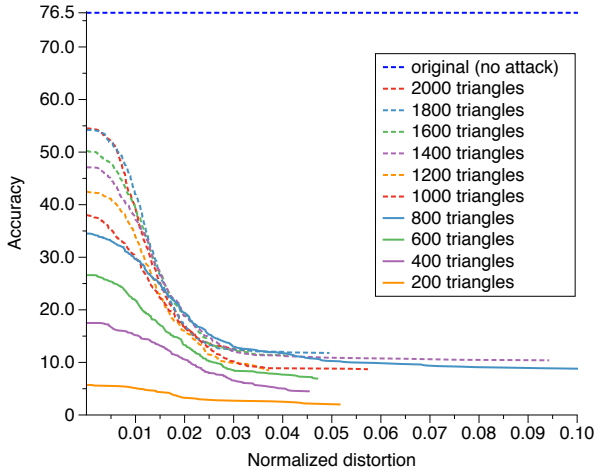

Figure 5.7: Strong Attack

Comparing curves of the vanilla attack on protected classifier, with 30.5% accuracy at 2000 triangles, and attack on original classifier with 3.7% accuracy, we can conclude that the primitive-based transformation does have defense effect, since the only difference between them is the defense. However, strict evaluation of robustness requires experiments under strong attack, and the result of vanilla attack only partially suggest improvement of the robustness of protected classifiers. Accuracy of strong attack on protected significantly lower than the vanilla attack due to the effectiveness of the BPDA.

We run BPDA in all strong attack cases. BPDA as a strong grey-box attack method, take advantages of both the defense and classifier. To compare the performance conveniently, we take the 2000 triangles and 800 triangles curve from Figure 5.7 and re-plotted in Figure 5.5. The accuracy of the protected classifier (2000 triangles) decrease from 54.5% to 12.2% and from 34.5% to 8.2% (800 triangles) as the distance of the distortion grows. But unlike the original classifier, the protected requires large distortions to decrease the accuracy. Within 0.01 distortion, the former slowly decreases to 39.3% and the latter to 29.7% while the original classifier decreases sharply to 5.6% at the same distance. This suggests substantial improvement of robustness of our protected classifiers.

The table 1 summarized the best distortion of our protected classifier (800 triangles) compared to other state-of-the-art defense methods under the strong attack scheme. Pixel deflection [98] and input transformation [50] defense are completely defeated under BPDA attack [7]. The accuracy of both drops to zero within l_2 normalized distortion 0.05 and 0.06 respectively, however, the accuracy of our method is still 8.2% at very large distortion 0.12. Larger distortion is required to completely defeat it.

Defense	Dataset	Distortion	Accuracy
Prakash *et al.*	ImageNet	$l_2(\epsilon = 0.05)$	0.0%
Guo *et al.*	ImageNet	$l_2(\epsilon = 0.06)$	0.0%
Ours (800)	ImageNet*	$l_2(\epsilon = 0.12)$	8.2%

Table 5.1: Summary of accuracy under BPDA attack

We also notice that the maximum distortions of the adversarial examples are still much smaller than the distortions of images reconstructed by the primitive-based transformation (Figure 5.4), that supports our assumption that the large transformation can suppress the adversarial examples noise. However, it also indicates that the adversarial robustness of the model is still not completely be solved since the defense can be broken when the distortion of adversarial example is still relatively short compared to the distortion of reconstructed image.

5.5 Transfer Experiment

In this paper we also study the transferability of adversarial examples. Under gray box setting, if the attackers do not know the specific parameter used in the defense, it may reduce the effectiveness of the attack because the parameters used in attack method is not in line with the one used in defense method.

Experiment We assume the number of triangles used in the transformation is 1000, but the attacker is not aware of that. The attacker enumerates the number of triangles from 200 to 2000 in each BPDA attack, and then feeds the adversarial examples generated in each setting into the protected classifier. We collect the accuracy over different number of triangles used in attack against the defense use the same number of triangles.

Results The accuracy is plotted in the curve of strong attack transferred in Figure 5.6 with respect to different number of triangles in attack. The curve of the strong attack transferred is very close to the normal strong attack on protected classifier. The adversarial examples can be transferred to the protected classifier using 1000 triangles in defense, except when the number is smaller than 600. This suggest that as long as the number of triangles is large enough, knowing the number of triangles used by the protected classifier in defense is not necessary. Hiding the parameters does not help improving the effect of the defense.

5.6 Discussion and Summary

In this paper we present a novel defense method using primitive-based transformation. Comprehensive experiments are conducted under various conditions. The experimental performance of the proposed method indicates the improvement of the robustness of the model. Compared to other similar state-of-art transformation methods, the distortion of adversarial examples required to break our defense method is significantly larger, which supports the claim of robustness improvement. The large distortions of reconstructed images by transformation can explain why the transformation successfully suppresses the adversarial noise. Our new framework serves a proof-of-concept that the large transformations can enhance the defense and is a promising new direction towards complete adversarial robustness.

Note that we still can not claim our defense method as a complete solution of adversarial robustness since the distortion of adversarial examples required to break the defense is relatively small compared to the distortion caused by our transformation. This adds to the accumulating evidence that complete robustness under strong attack is still challenging. In our method, the reconstructed images eventually approximate the original input despite the larger distortions, therefore, approximation assumption of attack methods BPDA still holds to some extent. Therefore, we emphasize that the large difference transformation is particularly crucial for the future work in this direction towards robustness. Additional studies that extend the large transformation are needed. Ultimately, if the defense could encrypt the features of input images, and new classifiers are re-trained on encrypted features, then the attack will not be able to approximate the gradient at all and the gray box attack will degenerate to less effective black-box attacks, which is guaranteed by cryptography. In the future work, we will work on this worth-exploring extension.

Algorithm 1 Primitive algorithm

Input: Image *target*
Parameter: Number of triangles T
Output:Image *result* with the same size to *target*
Define:State object contains a *triangle* shape and its RGBA *color*, current *image*, and *buffer*, the next image after adding the *triangle* to the *image*

new state object s_{best}
for each step in total T **do**
  $e_{best} \leftarrow$ INFINITY
  new state object s
  ▷ randomly get best initial state
  for each attempt in total ATTEMPTS **do**
    $s \leftarrow s_{best}$
    $e \leftarrow$ **call** mutate(s)
    if the first iteration **or** $e < e_{best}$ **then**
      $e_{best}, s_{best} \leftarrow e, s$
    end if
  end for
  ▷ randomly climb to better state
  $n_{fail} \leftarrow 0$
  while $n_{fail} <$ MAXFAIL **do**
    $s \leftarrow s_{best}$
    $s_{backup} \leftarrow s$
    $e \leftarrow$ **call** mutate(s)
    if $e > e_{best}$ **then**
      $n_{fail} \leftarrow n_{fail} + 1$
      $s_{best} \leftarrow s_{backup}$
    else
      $e_{best}, s_{best} \leftarrow e, s$
    end if
  end while
  $image[s_{best}] \leftarrow buffer[s_{best}]$
end for
return $image[s_{best}]$ as *result*

function mutate(s)
$triangle[s] \leftarrow$ a new triangle that $<$ SIZE $\times$ SIZE
repeat
  randomly move one vertex of the $triangle[s]$ from NORM distribution
until all angles of $triangle[s] >$ MINDEG
$color[s] \leftarrow$ average color in total equals $|image[s] - target|$ over $triangle[s]$ area
$buffer[s] \leftarrow triangle[s]$ of $color[s] + image[s]$
return root mean square error of $buffer[s]$ and *target*
endfunction

Algorithm 2 BPDA algorithm

Input: Image *original*, adversarial attack target *label*
Parameter: λ, R_{learn}
Required: Defense method $g(\cdot)$, logits of the classifier $f(\cdot)$
Output: Image *adversarial*

adversarial $\leftarrow$ *original*
for each integration in total ITERATIONS **do**
    regularizer $\leftarrow$calculate normalized L_2 loss from *adversarial* and *original*

    $\triangleright$ forward: *logits* equals $f(g(adversarial))$
    $x \leftarrow g(adversarial)$
    $logits \leftarrow f(x)$
    $softmax \leftarrow$ cross entropy with *logits* on *label*
    $loss \leftarrow softmax + \lambda \times regularizer$

    $\triangleright$ backward: stops on x at $f(x)$
    gradient $\leftarrow$derivatives of *loss* w.r.t. x
    $adversarial \leftarrow adversarial - R_{learn} \times gradient$
    adversarial $\leftarrow$ clip into range from 0 to 1
end for
return *adversarial*

Chapter 6

Conclusions and Future Research Directions

In this dissertation, I proposed a novel semantic analytical framework that uses deep unsupervised learning to mining the review text from users of mobile applications. The framework reaches better context-aware understanding and the clustering results demonstrated strong topic coherence. The topic modeling result of the user feedback is further converted into practical business guidelines, thereby helping the development of company greatly. Furthermore, I developed a novel sentiment analytic workflow by combining the network analytical approach with the SBERT embedding technique. The nodes are clustered into a group according to the modularity of the network and; then, the hidden sentiment patterns are exposed. The public opinions on telemedicine application shows complicated patterns. The experimental results imply insightful indicators of the user reviews of telemedicine applications. Last but not least, I studied the issue of the robustness of the deep learning models - the technological underpinnings of a wide range of modeling frameworks. After setting up the rigorous experiments, I devised a novel defense mechanism to mitigate the issue of adversarial examples. In the following part of this chapter, I tabulate my contributions of the three studies of this dissertation in Section 6.1, and then I discuss the future work in Section 6.2.

6.1 Main Contributions

The main contributions of this dissertation covers three inspiring projects. First, in Section 6.1.1, I present the contribution of the proposed technique that converts user feedback for health and fitness applications into design guidelines. Then in Section 6.1.2, I summarize the contribution of the research project in which the public sentiment on the telemedicine applications is analyzed. Last, I shed light on the contribution of my devleoped solution

that mitigates the robustness issue of deep neural network, namely adversarial examples in Section 6.1.3

6.1.1 Mobile Application Design Driven by User Feedback

In this work, I proposed a novel framework to analyze the user feedback of health mobile applications, and I converted the feedback to guidelines for mobile application design. Several contributions of this work are listed as follows.

Contribution 1: Utilized the Wisdom of the Crowd The new proposed framework can understand the complicate semantic and sentiment information embedded in the large volume of text by topic modeling combining rating scores associated. Instead of conducting surveys and questionnaires, the automated framework allows direct accesses to the customer feedback: the efficiency is greatly improved.

Contribution 2: Context-aware Understandings A framework based on deep learning models is beyond the simple bag-of-word model and understand the precise meaning according to the context. The new model leveraged the cutting-edge AI achievements for rich and insightful understandings of the content.

Contribution 3: Identified Issues Critical to Market Success Practical guidelines are concluded for five subcategories of health and fitness application market. For instance, the tracking applications work under multiple device and sensor environment frequently face customer complaints about connections, while the meditation application is relatively easy to succeed in the market.

6.1.2 Analysis on Telemedicine Application Reviews

In this work, I investigated a network approach combined with SBERT embedding to analyze the public opinions on telemedicine applications. After detecting network modularity, I contrasted the different responses to telemedicine applications from different network communities. The contributions of this part of the study are listed as follows:

Contribution 1: Novel Network Analytic Approach A new similarity-based network approach utilizes the output of deep learning model SBERT while converting the clustering task as a network modularity detection problem. The similarity-based network enables users to tune the threshold of connection, thereby optimizing data clustering performance.

Contribution 2: Discover Different Emotion Patterns I exploited emotion patterns and the other linguistic features among popular, unpopular and fake positive applications. The findings will further be expanded into a novel effective detection system of fake positive comments.

6.1.3 Robustness of Deep Learning Model

In this work, I investigated the robustness of deep learning model under the task of computer vision and how to mitigate the robustness issue of the deep learning model. The main contributions of this research are listed as follows:

Contribution 1: A Novel Model-agnostic Defense A new model-agnostic defense against strong adversarial attacks was proposed anchored on a large-difference-transformation approach, which provides a new promising direction towards a complete solution of adversarial robustness.

Contribution 2: Mitigating the Robustness Challenge The empirical experiments show that the primitive-based representation used in our transformation achieves state-of-the-art robustness under the BPDA attacks. The normalized l_2 distortion required to fully break the classifier on the ImageNet dataset is increased from 0.06 to 0.12.

6.2 Future Projects

This section elaborates on the further development extended from this dissertation research. In Section 6.2.1 I describe a solid plan to extend the feedback analytic framework by incorporating a monitoring framework. In Section 6.2.2 I present a future study focused on a novel way of building a new fake review detection system based on current findings.

The future direction, articulated in Section 6.2.3, is an extension of the image reconstruction from 2D to 3D for keeping attacks at bay.

6.2.1 Dynamic Monitoring

My developed framework - serving as an analytic tool - does not process any time related information. The current analytic framework only provides general information of all applications in a given category, which is not specific enough for individual applications within a certain time frame. An ideal system should also track the trends of its products and those from competitors. Using the previous topics and combining with the other features, I plan to add a classifier as a filter to differentiate feature requests, bug reports, praises and the like. I expect such a monitoring system will offer invaluable services to the business entities.

6.2.2 Fake Review Detection

A fake positive review detector can be built from the important features and the other linguistic patterns. Since fake positive reviews often lack diversity of the emotions, the distribution of emotions is generally centred on joy, surprise, anticipation, and trust. I intend to construct a framework to determine the possibility of fake reviews by checking the distribution of a review collection of target applications. Similarly, I also plan to build another framework to quantify the quality of reviews, aiming to prevent spam and post farming.

6.2.3 Reconstruction with 3D Primitives

As differentiable rendering libraries like PyTorch3D [99] become available, not only the current 2D primitives can be extended into 3D space, but also the raw hill climbing method can be replaced by an efficient gradient descent method. I will propose a new

way of reconstructing images with basic 3D objects - 3D primitives - in a hope to suppress adversarial attacks and to bolster the robustness of deep learning models.

Bibliography

[1] A. Abdi, S. M. Shamsuddin, S. Hasan, and J. Piran. Machine learning-based multi-documents sentiment-oriented summarization using linguistic treatment. *Expert Systems with Applications*, 109:66–85, 2018.

[2] G. Aggarwal, M. Bawa, P. Ganesan, H. Garcia-Molina, K. Kenthapadi, R. Motwani, U. Srivastava, D. Thomas, and Y. Xu. Two can keep a secret: A distributed architecture for secure database services. *CIDR 2005*, 2005.

[3] D. A. Al-Qudah, A.-Z. Ala'M, P. A. Castillo-Valdivieso, and H. Faris. Sentiment analysis for e-payment service providers using evolutionary extreme gradient boosting. *IEEE Access*, 8:189930–189944, 2020.

[4] M. A. Alcorn, Q. Li, Z. Gong, C. Wang, L. Mai, W.-S. Ku, and A. Nguyen. Strike (with) a pose: Neural networks are easily fooled by strange poses of familiar objects. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 4845–4854, 2019.

[5] M. Allaoui, M. L. Kherfi, and A. Cheriet. Considerably improving clustering algorithms using umap dimensionality reduction technique: a comparative study. In *International Conference on Image and Signal Processing*, pages 317–325. Springer, 2020.

[6] M. Z. Ansari, M. Aziz, M. Siddiqui, H. Mehra, and K. Singh. Analysis of political sentiment orientations on twitter. *Procedia Computer Science*, 167:1821–1828, 2020.

[7] A. Athalye, N. Carlini, and D. Wagner. Obfuscated gradients give a false sense of security: Circumventing defenses to adversarial examples. In *Proceedings of the 35th International Conference on Machine Learning, ICML 2018*, July 2018.

[8] C. A. Bail. Combining natural language processing and network analysis to examine how advocacy organizations stimulate conversation on social media. *Proceedings of the National Academy of Sciences*, 113(42):11823–11828, 2016.

[9] R. Baly, H. Hajj, N. Habash, K. B. Shaban, and W. El-Hajj. A sentiment treebank and morphologically enriched recursive deep models for effective sentiment analysis in arabic. *ACM Transactions on Asian and Low-Resource Language Information Processing (TALLIP)*, 16(4):1–21, 2017.

[10] M. Bastian, S. Heymann, and M. Jacomy. Gephi: an open source software for exploring and manipulating networks. In *Proceedings of the international AAAI conference on web and social media*, volume 3, pages 361–362, 2009.

[11] K. Beyer, J. Goldstein, R. Ramakrishnan, and U. Shaft. When is "nearest neighbor" meaningful? In *International conference on database theory*, pages 217–235. Springer, 1999.

[12] A. N. Bhagoji, W. He, B. Li, and D. Song. Practical black-box attacks on deep neural networks using efficient query mechanisms. In *European Conference on Computer Vision*, pages 158–174. Springer, 2018.

[13] J. R. Black, C. Bailey, J. Przewrocka, K. K. Dijkstra, and C. Swanton. Covid-19: the case for health-care worker screening to prevent hospital transmission. *The Lancet*, 395(10234):1418–1420, 2020.

[14] D. M. Blei, A. Y. Ng, and M. I. Jordan. Latent dirichlet allocation. *Journal of machine Learning research*, 3(Jan):993–1022, 2003.

[15] V. D. Blondel, J.-L. Guillaume, R. Lambiotte, and E. Lefebvre. Fast unfolding of communities in large networks. *Journal of statistical mechanics: theory and experiment*, 2008(10):P10008, 2008.

[16] J. Bradshaw, A. G. d. G. Matthews, and Z. Ghahramani. Adversarial examples, uncertainty, and transfer testing robustness in gaussian process hybrid deep networks. *arXiv preprint arXiv:1707.02476*, 2017.

[17] W. Brendel, J. Rauber, and M. Bethge. Decision-based adversarial attacks: Reliable attacks against black-box machine learning models. In *International Conference on Learning Representations*, 2018.

[18] N. Carlini and D. Wagner. Adversarial examples are not easily detected: Bypassing ten detection methods. In *Proceedings of the 10th ACM Workshop on Artificial Intelligence and Security*, pages 3–14, 2017.

[19] N. Carlini and D. Wagner. Towards evaluating the robustness of neural networks. In *2017 ieee symposium on security and privacy (sp)*, pages 39–57. IEEE, 2017.

[20] C. Chen, A. Seff, A. Kornhauser, and J. Xiao. Deepdriving: Learning affordance for direct perception in autonomous driving. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 2722–2730, 2015.

[21] L.-C. Chen, G. Papandreou, I. Kokkinos, K. Murphy, and A. L. Yuille. Deeplab: Semantic image segmentation with deep convolutional nets, atrous convolution, and fully connected crfs. *IEEE transactions on pattern analysis and machine intelligence*, 40(4):834–848, 2017.

[22] N. Chen, J. Lin, S. C. Hoi, X. Xiao, and B. Zhang. Ar-miner: mining informative reviews for developers from mobile app marketplace. In *Proceedings of the 36th international conference on software engineering*, pages 767–778, 2014.

[23] P.-Y. Chen, H. Zhang, Y. Sharma, J. Yi, and C.-J. Hsieh. Zoo: Zeroth order optimization based black-box attacks to deep neural networks without training substitute models. In *Proceedings of the 10th ACM Workshop on Artificial Intelligence and Security*, pages 15–26, 2017.

[24] R. Chen, Q. Wang, and W. Xu. Mining user requirements to facilitate mobile app quality upgrades with big data. *Electronic Commerce Research and Applications*, 38:100889, 2019.

[25] T. Ching, D. S. Himmelstein, B. K. Beaulieu-Jones, A. A. Kalinin, B. T. Do, G. P. Way, E. Ferrero, P.-M. Agapow, M. Zietz, M. M. Hoffman, et al. Opportunities and obstacles for deep learning in biology and medicine. *Journal of The Royal Society Interface*, 15(141):20170387, 2018.

[26] M. Cisse, P. Bojanowski, E. Grave, Y. Dauphin, and N. Usunier. Parseval networks: Improving robustness to adversarial examples. In *Proceedings of the 34th International Conference on Machine Learning-Volume 70*, pages 854–863. JMLR.org, 2017.

[27] M. Colucci. Communication technologies through an etymological lens: looking for a classification, reflections about health, medicine and care. *Medicine, Health Care and Philosophy*, 18(4):601–606, 2015.

[28] F. Dalpiaz and M. Parente. Re-swot: from user feedback to requirements via competitor analysis. In *International working conference on requirements engineering: foundation for software quality*, pages 55–70. Springer, 2019.

[29] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*, 2018.

[30] A. Di Sorbo, S. Panichella, C. V. Alexandru, C. A. Visaggio, and G. Canfora. Surf: summarizer of user reviews feedback. In *2017 IEEE/ACM 39th International Conference on Software Engineering Companion (ICSE-C)*, pages 55–58. IEEE, 2017.

[31] C. Du, H. Sun, J. Wang, Q. Qi, J. Liao, C. Wang, and B. Ma. Investigating capsule network and semantic feature on hyperplanes for text classification. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 456–465, 2019.

[32] G. K. Dziugaite, Z. Ghahramani, and D. M. Roy. A study of the effect of JPG compression on adversarial images. *CoRR*, abs/1608.00853, 2016.

[33] P. Ekman. An argument for basic emotions. *Cognition & emotion*, 6(3-4):169–200, 1992.

[34] X. Fang and J. Zhan. Sentiment analysis using product review data. *Journal of Big Data*, 2(1):1–14, 2015.

[35] R. Feinman, R. R. Curtin, S. Shintre, and A. B. Gardner. Detecting adversarial samples from artifacts. *arXiv preprint arXiv:1703.00410*, 2017.

[36] M. Fernández-Gavilanes, T. Álvarez-López, J. Juncal-Martínez, E. Costa-Montenegro, and F. J. González-Castaño. Unsupervised method for sentiment analysis in online texts. *Expert Systems with Applications*, 58:57–75, 2016.

[37] L. Ferretti, C. Wymant, M. Kendall, L. Zhao, A. Nurtay, L. Abeler-Dörner, M. Parker, D. Bonsall, and C. Fraser. Quantifying sars-cov-2 transmission suggests epidemic control with digital contact tracing. *Science*, 368(6491):eabb6936, 2020.

[38] S. G. Finlayson, J. D. Bowers, J. Ito, J. L. Zittrain, A. L. Beam, and I. S. Kohane. Adversarial attacks on medical machine learning. *Science*, 363(6433):1287–1289, 2019.

[39] L. Flory, K.-M. Osei-Bryson, and M. Thomas. A new web personalization decision-support artifact for utility-sensitive customer review analysis. *Decision Support Systems*, 94:85–96, 2017.

[40] B. Fu, J. Lin, L. Li, C. Faloutsos, J. Hong, and N. Sadeh. Why people hate your app: Making sense of user feedback in a mobile app store. In *Proceedings of the 19th ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 1276–1284, 2013.

[41] J. E. Gaffney Jr. Metrics in software quality assurance. In *Proceedings of the ACM'81 conference*, pages 126–130, 1981.

[42] J. Gao, B. Wang, and Y. Qi. Deepcloak: Masking DNN models for robustness against adversarial samples. *CoRR*, abs/1702.06763, 2017.

[43] R. Girshick. Fast r-cnn. In *Proceedings of the IEEE international conference on computer vision*, pages 1440–1448, 2015.

[44] D. M. Goldberg and A. S. Abrahams. Sourcing product innovation intelligence from online reviews. *Decision Support Systems*, page 113751, 2022.

[45] I. Goodfellow, J. Shlens, and C. Szegedy. Explaining and harnessing adversarial examples. In *International Conference on Learning Representations*, 2015.

[46] A. Graesser. Mind and body: Dialogue and posture for affect detection in learning environments. 2007.

[47] M. Grootendorst. Bertopic: Neural topic modeling with a class-based tf-idf procedure. *arXiv preprint arXiv:2203.05794*, 2022.

[48] K. Grosse, P. Manoharan, N. Papernot, M. Backes, and P. McDaniel. On the (statistical) detection of adversarial examples. *CoRR*, abs/17, 2017.

[49] X. Gu and S. Kim. " what parts of your apps are loved by users?"(t). In *2015 30th IEEE/ACM International Conference on Automated Software Engineering (ASE)*, pages 760–770. IEEE, 2015.

[50] C. Guo, M. Rana, M. Cisse, and L. van der Maaten. Countering adversarial images using input transformations. In *International Conference on Learning Representations*, 2018.

[51] E. Guzman and W. Maalej. How do users like this feature? a fine grained sentiment analysis of app reviews. In *2014 IEEE 22nd international requirements engineering conference (RE)*, pages 153–162. Ieee, 2014.

[52] S. Ha and Y. Geum. Identifying new innovative services using m&a data: An integrated approach of data-driven morphological analysis. *Technological Forecasting and Social Change*, 174:121197, 2022.

[53] K. He, X. Zhang, S. Ren, and J. Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778, 2016.

[54] J. E. Hollander and B. G. Carr. Virtually perfect? telemedicine for covid-19. *New England Journal of Medicine*, 382(18):1679–1681, 2020.

[55] H. Hosseini, Y. Chen, S. Kannan, B. Zhang, and R. Poovendran. Blocking transferability of adversarial examples in black-box learning systems. *arXiv preprint arXiv:1703.04318*, 2017.

[56] A. Ilyas, L. Engstrom, A. Athalye, and J. Lin. Black-box adversarial attacks with limited queries and information. In *Proceedings of the 35th International Conference on Machine Learning, ICML 2018*, July 2018.

[57] Y. Jin, H. Zhang, and D. Du. Improving deep belief networks via delta rule for sentiment classification. In *2016 IEEE 28th international conference on tools with artificial intelligence (ICTAI)*, pages 410–414. IEEE, 2016.

[58] T. Johann, C. Stanik, W. Maalej, et al. Safe: A simple approach for feature extraction from app descriptions and app reviews. In *2017 IEEE 25th international requirements engineering conference (RE)*, pages 21–30. IEEE, 2017.

[59] B. Kaplan. Revisiting health information technology ethical, legal, and social issues and evaluation: telehealth/telemedicine and covid-19. *International journal of medical informatics*, 143:104239, 2020.

[60] D. Khullar. America is running out of nurses. `https://www.newyorker.com/science/medical-dispatch/america-is-running-out-of-nurses`. Accessed: 2022-09-20.

[61] A. Krizhevsky, I. Sutskever, and G. E. Hinton. Imagenet classification with deep convolutional neural networks. In *Advances in neural information processing systems*, pages 1097–1105, 2012.

[62] A. Kulkarni and A. Shivananda. *Natural language processing recipes*. Springer, 2019.

[63] A. Kurakin, D. Boneh, F. Tramèr, I. Goodfellow, N. Papernot, and P. McDaniel. Ensemble adversarial training: Attacks and defenses. In *2018 International Conference on Learning Representations*, 2018.

[64] A. Kurakin, I. J. Goodfellow, and S. Bengio. Adversarial machine learning at scale. In *2017 International Conference on Learning Representations*, 2017.

[65] G. Lee, J. Jeong, S. Seo, C. Kim, and P. Kang. Sentiment classification with word localization based on weakly supervised learning with a convolutional neural network. *Knowledge-Based Systems*, 152:70–82, 2018.

[66] G. Lee and T. S. Raghu. Determinants of mobile apps' success: Evidence from the app store market. *Journal of Management Information Systems*, 31(2):133–170, 2014.

[67] Q. Li, D. D. Zeng, D. J. Xu, R. Liu, and R. Yao. Understanding and predicting users' rating behavior: A cognitive perspective. *INFORMS Journal on Computing*, 32(4):996–1011, 2020.

[68] Y.-M. Li, H.-M. Chen, J.-H. Liou, and L.-F. Lin. Creating social intelligence for product portfolio design. *Decision Support Systems*, 66:123–134, 2014.

[69] F. Liao, M. Liang, Y. Dong, T. Pang, X. Hu, and J. Zhu. Defense against adversarial attacks using high-level representation guided denoiser. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 1778–1787, 2018.

[70] X. Liu, W. Ai, H. Li, J. Tang, G. Huang, F. Feng, and Q. Mei. Deriving user preferences of mobile apps from their management activities. *ACM Transactions on Information Systems (TOIS)*, 35(4):1–32, 2017.

[71] Y. Liu, C. Jiang, and H. Zhao. Using contextual features and multi-view ensemble learning in product defect identification from online discussion forums. *Decision Support Systems*, 105:1–12, 2018.

[72] Y. Liu, C. Jiang, and H. Zhao. Assessing product competitive advantages from the perspective of customers by mining user-generated content on social media. *Decision Support Systems*, 123:113079, 2019.

[73] Z. Liu, C.-X. Qin, and Y.-J. Zhang. Mining product competitiveness by fusing multi-source online information. *Decision Support Systems*, 143:113477, 2021.

[74] J. Long, E. Shelhamer, and T. Darrell. Fully convolutional networks for semantic segmentation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 3431–3440, 2015.

[75] H. Lukas, C. Xu, Y. Yu, and W. Gao. Emerging telemedicine tools for remote covid-19 diagnosis, monitoring, and management. *ACS nano*, 14(12):16180–16193, 2020.

[76] Y. Luo, X. Boix, G. Roig, T. Poggio, and Q. Zhao. Foveation-based mechanisms alleviate adversarial examples. 11 2016.

[77] W. Maalej and H. Nabil. Bug report, feature request, or simply praise? on automatically classifying app reviews. In *2015 IEEE 23rd international requirements engineering conference (RE)*, pages 116–125. IEEE, 2015.

[78] A. Madry, A. Makelov, L. Schmidt, D. Tsipras, and A. Vladu. Towards deep learning models resistant to adversarial attacks. In *2018 International Conference on Learning Representations*, 2018.

[79] H. Malik, E. M. Shakshuki, and W.-S. Yoo. Comparing mobile apps by identifying 'hot'features. *Future Generation Computer Systems*, 107:659–669, 2020.

[80] R. M. Marcacini, R. G. Rossi, I. P. Matsuno, and S. O. Rezende. Cross-domain aspect extraction for sentiment analysis: A transductive learning approach. *Decision Support Systems*, 114:70–80, 2018.

[81] L. McInnes, J. Healy, and S. Astels. hdbscan: Hierarchical density based clustering. *J. Open Source Softw.*, 2(11):205, 2017.

[82] L. McInnes, J. Healy, and J. Melville. Umap: Uniform manifold approximation and projection for dimension reduction. *arXiv preprint arXiv:1802.03426*, 2018.

[83] A. Mehrabian. Pleasure-arousal-dominance: A general framework for describing and measuring individual differences in temperament. *Current Psychology*, 14(4):261–292, 1996.

[84] D. Meng and H. Chen. Magnet: a two-pronged defense against adversarial examples. In *Proceedings of the 2017 ACM SIGSAC Conference on Computer and Communications Security*, pages 135–147, 2017.

[85] J. H. Metzen, T. Genewein, V. Fischer, and B. Bischoff. On detecting adversarial perturbations. In *2017 International Conference on Learning Representations*, 2017.

[86] S. Mohammad and P. Turney. Emotions evoked by common words and phrases: Using Mechanical Turk to create an emotion lexicon. In *Proceedings of the NAACL HLT 2010 Workshop on Computational Approaches to Analysis and Generation of Emotion in Text*, pages 26–34, Los Angeles, CA, June 2010. Association for Computational Linguistics.

[87] S. M. Mohammad. Obtaining reliable human ratings of valence, arousal, and dominance for 20,000 english words. In *Proceedings of The Annual Conference of the Association for Computational Linguistics (ACL)*, Melbourne, Australia, 2018.

[88] S. M. Mohammad. Word affect intensities. In *Proceedings of the 11th Edition of the Language Resources and Evaluation Conference (LREC-2018)*, Miyazaki, Japan, 2018.

[89] S.-M. Moosavi-Dezfooli, A. Fawzi, and P. Frossard. Deepfool: a simple and accurate method to fool deep neural networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 2574–2582, 2016.

[90] M. J. Mortenson and R. Vidgen. A computational literature review of the technology acceptance model. *International Journal of Information Management*, 36(6):1248–1259, 2016.

[91] S. Murnion, W. J. Buchanan, A. Smales, and G. Russell. Machine learning and semantic analysis of in-game chat for cyberbullying. *Computers & Security*, 76:197–213, 2018.

[92] A. Núñez, S. Sreeganga, and A. Ramaprasad. Access to healthcare during covid-19. *International journal of environmental research and public health*, 18(6):2980, 2021.

[93] N. Papernot, P. McDaniel, and I. Goodfellow. Transferability in machine learning: from phenomena to black-box attacks using adversarial samples. *arXiv preprint arXiv:1605.07277*, 2016.

[94] N. Papernot, P. McDaniel, I. Goodfellow, S. Jha, Z. B. Celik, and A. Swami. Practical black-box attacks against machine learning. In *Proceedings of the 2017 ACM on Asia conference on computer and communications security*, pages 506–519, 2017.

[95] N. Papernot, P. McDaniel, S. Jha, M. Fredrikson, Z. B. Celik, and A. Swami. The limitations of deep learning in adversarial settings. In *2016 IEEE European symposium on security and privacy (EuroS&P)*, pages 372–387. IEEE, 2016.

[96] N. Papernot, P. McDaniel, X. Wu, S. Jha, and A. Swami. Distillation as a defense to adversarial perturbations against deep neural networks. In *2016 IEEE Symposium on Security and Privacy (SP)*, pages 582–597. IEEE, 2016.

[97] G. Perrone, S. Zerbo, C. Bilotta, G. Malta, and A. Argo. Telemedicine during covid-19 pandemic: Advantage or critical issue? *Medico-Legal Journal*, 88(2):76–77, 2020.

[98] A. Prakash, N. Moran, S. Garber, A. DiLillo, and J. Storer. Deflecting adversarial attacks with pixel deflection. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 8571–8580, 2018.

[99] N. Ravi, J. Reizenstein, D. Novotny, T. Gordon, W.-Y. Lo, J. Johnson, and G. Gkioxari. Accelerating 3d deep learning with pytorch3d. *arXiv:2007.08501*, 2020.

[100] E. Razova, S. Vychegzhanin, and E. Kotelnikov. Does bert look at sentiment lexicon? In *International Conference on Analysis of Images, Social Networks and Texts*, pages 55–67. Springer, 2022.

[101] N. Reimers and I. Gurevych. Sentence-bert: Sentence embeddings using siamese bert-networks. *arXiv preprint arXiv:1908.10084*, 2019.

[102] N. Reimers and I. Gurevych. Making monolingual sentence embeddings multilingual using knowledge distillation. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 4512–4525, Online, Nov. 2020. Association for Computational Linguistics.

[103] S. Ren, K. He, R. Girshick, and J. Sun. Faster r-cnn: Towards real-time object detection with region proposal networks. In *Advances in neural information processing systems*, pages 91–99, 2015.

[104] F. A. Shah, Y. Sabanin, and D. Pfahl. Feature-based evaluation of competing apps. In *Proceedings of the International Workshop on App Market Analytics*, pages 15–21, 2016.

[105] F. A. Shah, K. Sirts, and D. Pfahl. Simple app review classification with only lexical features. In *ICSOFT*, pages 146–153, 2018.

[106] F. A. Shah, K. Sirts, and D. Pfahl. Using app reviews for competitive analysis: tool support. In *Proceedings of the 3rd ACM SIGSOFT International Workshop on App Market Analytics*, pages 40–46, 2019.

[107] U. Shaham, Y. Yamada, and S. Negahban. Understanding adversarial training: Increasing local stability of supervised models through robust optimization. *Neurocomputing*, 307:195–204, 2018.

[108] C. Sievert and K. Shirley. Ldavis: A method for visualizing and interpreting topics. In *Proceedings of the workshop on interactive language learning, visualization, and interfaces*, pages 63–70, 2014.

[109] K. Simonyan and A. Zisserman. Very deep convolutional networks for large-scale image recognition. In *International Conference on Learning Representations*, 2015.

[110] A. Singh and C. S. Tucker. A machine learning approach to product review disambiguation based on function, form and behavior classification. *Decision Support Systems*, 97:81–91, 2017.

[111] K. Song, T. Yao, Q. Ling, and T. Mei. Boosting image sentiment analysis with visual attention. *Neurocomputing*, 312:218–228, 2018.

[112] C. Szegedy, V. Vanhoucke, S. Ioffe, J. Shlens, and Z. Wojna. Rethinking the inception architecture for computer vision. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 2818–2826, 2016.

[113] C. Szegedy, W. Zaremba, I. Sutskever, J. Bruna, D. Erhan, I. Goodfellow, and R. Fergus. Intriguing properties of neural networks. In *International Conference on Learning Representations*, 2014.

[114] C.-C. Tu, P. Ting, P.-Y. Chen, S. Liu, H. Zhang, J. Yi, C.-J. Hsieh, and S.-M. Cheng. Autozoom: Autoencoder-based zeroth order optimization method for attacking black-box neural networks. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 33, pages 742–749, 2019.

[115] J. Uesato, B. O'Donoghue, P. Kohli, and A. van den Oord. Adversarial risk and the dangers of evaluating against weak attacks. In J. Dy and A. Krause, editors, *Proceedings of the 35th International Conference on Machine Learning*, volume 80

of *Proceedings of Machine Learning Research*, pages 5025–5034, Stockholmsmässan, Stockholm Sweden, 10–15 Jul 2018. PMLR.

[116] R. Van der Meulen and J. Rivera. Gartner says less than 0.01 percent of consumer mobile apps will be considered a financial success by their developers through 2018. *Retrieved March*, 22:2014, 2014.

[117] S. F. Verkijika and B. N. Neneh. Standing up for or against: A text-mining study on the recommendation of mobile payment apps. *Journal of Retailing and Consumer Services*, 63:102743, 2021.

[118] H. M. Wallach. Topic modeling: beyond bag-of-words. In *Proceedings of the 23rd international conference on Machine learning*, pages 977–984, 2006.

[119] J. Wang, Z. Zhang, C. Xie, Y. Zhou, V. Premachandran, J. Zhu, L. Xie, and A. Yuille. Visual concepts and compositional voting. *Annals of Mathematical Sciences and Applications*, 3, 11 2017.

[120] X. Wang, Y. Li, and P. Xu. A hybrid blstm-c neural network proposed for chinese text classification. In *2018 Sixth International Conference on Advanced Cloud and Big Data (CBD)*, pages 311–315. IEEE, 2018.

[121] C. Whissell. The dictionary of affect in language, emotion: Theory, research and experience. the measurement of emotions, r. plutchik and h. kellerman, eds., vol. 4, 1989.

[122] C. Xie, J. Wang, Z. Zhang, Z. Ren, and A. Yuille. Mitigating adversarial effects through randomization. In *2018 International Conference on Learning Representations*, 2018.

[123] W. Xu, D. Evans, and Y. Qi. Feature Squeezing: Detecting Adversarial Examples in Deep Neural Networks. In *Proceedings of the 2018 Network and Distributed Systems Security Symposium (NDSS)*, 2018.

[124] A. Yadav and D. K. Vishwakarma. Sentiment analysis using deep learning architectures: a review. *Artificial Intelligence Review*, 53(6):4335–4385, 2020.

[125] M. Yang, Q. Qu, X. Chen, C. Guo, Y. Shen, and K. Lei. Feature-enhanced attention network for target-dependent sentiment classification. *Neurocomputing*, 307:91–97, 2018.

[126] M. Yang, W. Zhao, L. Chen, Q. Qu, Z. Zhao, and Y. Shen. Investigating the transferring capability of capsule networks for text classification. *Neural Networks*, 118:247–261, 2019.

[127] G. Yin, H. Song, J. Wang, S. Nicholas, and E. Maitland. The covid-19 run on medical resources in wuhan china: causes, consequences and lessons. In *Healthcare*, volume 9, page 1362. MDPI, 2021.

[128] S. Yoo, J. Song, and O. Jeong. Social media contents based sentiment analysis and prediction system. *Expert Systems with Applications*, 105:102–111, 2018.

[129] V. Zantedeschi, M.-I. Nicolae, and A. Rawat. Efficient defenses against adversarial attacks. In *Proceedings of the 10th ACM Workshop on Artificial Intelligence and Security*, pages 39–49, 2017.

[130] L. Zhang, S. Wang, and B. Liu. Deep learning for sentiment analysis: A survey. *Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery*, 8(4):e1253, 2018.

[131] Z. Zhang, S. Qiao, C. Xie, W. Shen, B. Wang, and A. L. Yuille. Single-shot object detection with enriched semantics. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 5813–5821, 2018.

[132] Z. Zhang, L. Wang, Y. Zou, and C. Gan. The optimally designed dynamic memory networks for targeted sentiment classification. *Neurocomputing*, 309:36–45, 2018.

[133] L. Zheng, Z. He, and S. He. A novel probabilistic graphic model to detect product defects from social media data. *Decision Support Systems*, 137:113369, 2020.

[134] X. Zheng, S. Zhu, and Z. Lin. Capturing the essence of word-of-mouth for social commerce: Assessing the quality of online e-commerce reviews by a semi-supervised approach. *Decision Support Systems*, 56:211–222, 2013.

[135] D. Zhu, T. Lappas, and J. Zhang. Unsupervised tip-mining from customer reviews. *Decision Support Systems*, 107:116–124, 2018.