

B.Sc. in Computer Science and Engineering Thesis

A Large-Scale Empirical Study of COVID-19 Contact Tracing Mobile App Reviews

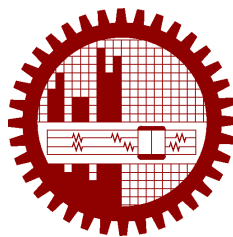
Submitted by

Sifat Ishmam Parisa
201505016

Md Awsaf Alam Anindya
201505114

Supervised by

Dr. Anindya Iqbal



Department of Computer Science and Engineering
Bangladesh University of Engineering and Technology

Dhaka, Bangladesh

February 20, 2021

CANDIDATES' DECLARATION

This is to certify that the work presented in this thesis, titled, “A Large-Scale Empirical Study of COVID-19 Contact Tracing Mobile App Reviews”, is the outcome of the investigation and research carried out by us under the supervision of Dr. Anindya Iqbal.

It is also declared that neither this thesis nor any part thereof has been submitted anywhere else for the award of any degree, diploma or other qualifications.

Sifat Ishmam Parisa
201505016

Md Awsaf Alam Anindya
201505114

CERTIFICATION

This thesis titled, “**A Large-Scale Empirical Study of COVID-19 Contact Tracing Mobile App Reviews**”, submitted by the group as mentioned below has been accepted as satisfactory in partial fulfillment of the requirements for the degree B.Sc. in Computer Science and Engineering in February 20, 2021.

Group Members:

Sifat Ishmam Parisa

Md Awsaf Alam Anindya

Supervisor:

Dr. Anindya Iqbal

Associate Professor

Department of Computer Science and Engineering

Bangladesh University of Engineering and Technology

ACKNOWLEDGEMENT

We are indebted to Dr. Gias Uddin, Assistant Professor, Data Intensive Software Systems, University of Calgary without his guidance and persistent help this thesis would not have been possible.

Dhaka
February 20, 2021

Sifat Ishmam Parisa
Md Awsaf Alam Anindya

Contents

<i>CANDIDATES' DECLARATION</i>	i
<i>CERTIFICATION</i>	ii
<i>ACKNOWLEDGEMENT</i>	iii
List of Figures	vi
List of Tables	vii
<i>ABSTRACT</i>	viii
1 Introduction	1
1.1 Research Questions	2
1.2 Outline	3
2 Related Works	4
2.1 Studies on COVID-19 Contact Tracing App	4
2.2 Software Engineering research related to COVID-19	5
2.3 Topic Modeling in Software Engineering Research	5
3 Background	7
3.1 Topic Modeling	7
4 Data Collection and Preprocessing	9
4.1 Data Collection	9
4.2 Preprocessing	9
4.3 LDA topic modeling	11
5 Empirical Study	15
5.1 Topics in COVID-19 contact tracing app reviews (RQ1)	15
5.1.1 Motivation	15
5.1.2 Approach	16
5.1.3 Results	16
5.2 Evolution of the popularity of topics over time(RQ2)	16

5.2.1	Motivation	16
5.2.2	Approach	18
5.2.3	Results	18
5.3	User complaints and appraisals across topics (RQ3)	18
5.3.1	Motivation	18
5.3.2	Approach	19
5.3.3	Results	20
5.4	Evolution of user complaints and appraisals across the continents (RQ4)	20
5.4.1	Motivation	20
5.4.2	Approach	21
5.4.3	Results	21
5.5	Discussions	21
5.5.1	Implications to the contact tracing app developers	22
5.5.2	Implications to the emergency response app developers	22
5.5.3	Implications to the users	22
6	Threats to Validity	23
6.1	External Validity	23
6.2	Internal Validity	23
6.3	Construct Validity	24
7	Sentiment Analysis	25
7.1	Related Work	25
7.2	Fine Tuning BERT	26
7.3	Sentiment Evaluation	26
8	Conclusion	27
	References	28
A	Example	34
B	Codes	38
B.1	Scraper code	38
B.2	Preprocessing Code	40
B.3	LDA training code	46
B.4	LDA topics visualization	52

List of Figures

5.1	The absolute impact scores of different categories between February 2020 and September 2020	19
5.2	The relative impact scores of different categories between February 2020 and September 2020	19
5.3	The star rating of topics across different categories	20
5.4	The star rating of reviews across different continents	21

List of Tables

4.1	The contact tracing apps in the dataset	10
4.2	LDA Experiments Results	14
5.1	Distribution (frequency) of topics in our dataset per category. Each colored bar denotes a continent (Black = Europe, Green = North America, Magenta = Australasia, Blue = Asia, Red= Middle East, Yellow = Africa, Orange = Latin America)	17
7.1	Sentiments of reviews across continents	26
A.1	Example review for topics	37

ABSTRACT

Since the beginning of 2020, the novel coronavirus has begun to sweep across the globe. While the recent development of vaccines offers hope, the mutations of the coronavirus have enforced countries around the world to adopt and enforce strict safety requirements like wearing a mask. Given the prevalence of smartphones everywhere, many countries across continents also developed COVID-19 contact tracing apps that users can install to get a warning of potential contacts with infected people. Unlike regular apps that undergo detailed requirement analysis, carefully designed development, rigorous testing, contact tracing apps were deployed after rapid development. Therefore such apps may not reach expectations for all end users. Users share their opinions and experience of the usage of the apps in the app store. This thesis aims to understand the types of topics users discuss in the reviews of the COVID-19 contact tracing apps across the continents by analyzing the app reviews. This insight can be useful to learn about the specific problems the users face, the concerns they have about the apps, and the positive/negative experience they share about the design and usage of the apps. We collected all the reviews of 36 COVID-19 contact tracing apps developed by 34 countries across the globe. We group the app reviews into the following geographical regions: Asia, Europe, North America, Latin America, Africa, Middle East, and Australasia (Australia and NZ). The purpose is to ensure that we can capture region-specific diversity and specificity during the analysis of the app reviews. We run topic modeling on the app reviews of each region. We analyze the produced topics and their evolution over time by categorizing them into hierarchies and computing the ratings of reviews related to the topics. While privacy could be a concern with such apps, we only find the privacy-related topics in Australasia, North America, and Middle East. Topics related to usability and performance of the apps are prevalent across all regions. Users frequently complained about the lack of features and negative/hung user interfaces in the apps, as well as the negative impact of such apps on their mobile batteries. Despite the negative experiences, we also find that many users praised the apps because they helped them stay aware of the potential danger of getting infected. The finding of this study is expected to help the app developers utilize their resources in addressing the reported issues in a prioritized way.

Chapter 1

Introduction

The global health crisis of 2020-21 is the ongoing COVID-19 pandemic, caused by Severe Acute Respiratory Syndrome Coronavirus 2 (SARS-CoV-2) [1]. The worldwide outbreak compelled governments to adopt countermeasures such as lockdown, social-distancing, wearing masks, self-isolation, work-from-home, and contact tracing [2]. Conventional contact tracing is a manual procedure of approximating the risk score of contacts of the infected individuals conducted by health officials. However, with the high number of cases, the traditional contact tracing approach for COVID-19 is complicated, time-consuming, and not entirely effective. Hence, authorities and stakeholders have relied on data analytics and digital technologies to automate the contact tracing process. Consequently, several smartphone apps for contact tracing have been developed [3]. Although vaccination has started to be administered in developed countries, it has been predicted that it will take more than the whole 2021 to bring required number of people across the globe to achieve hard immunity. Hence, to combat the ongoing second wave and for a long time afterwards, the role of Contact Tracing Applications remains highly significant.

Contact tracing apps notify a person who has been in the proximity of Covid infected patients with the help of smartphones' built-in services such as Bluetooth interface and location service. These apps differ from regular apps in the following ways. Regular application development is expected to follow all standard software development steps, i.e., planning, requirement analysis and documentation, functionality design, programming, comprehensive testing, deployment, implementation, and support. On the contrary, contact tracing apps were developed as an emergency response tool to control the virus spread. The time constraint did not allow careful requirement planning and feature design. It is also highly likely that the design and testing phase did not take all requirements into account. As a result, using the apps was problematic for the majority of users, as found in their app reviews.

Users can submit their opinions regarding app usage experience by star ratings and text reviews in online application distribution platforms like Google Play Store and App Store. As

presented in empirical studies, [4], [5], [6], and [7], these reviews reveal user experience and requirements, bug reports, and feature requests. Every app development team can get an idea of their own strengths/weaknesses as well as that of similar other applications. Developers can use the data gathered from the reviews as guidelines for updating the application in subsequent releases [8]. It is likely that different development teams are supposed to address particular types of issues (e.g., front-end/UX, authentication/security, etc.) that need specialization to some extent. We have identified the topics of the reviews and then grouped the topics into high-level categories that would help the management assign issues easily to respective development groups. Therefore, we expect the outcome of this study would help the continuous improvement of the contract tracing applications and benefit the community. Moreover, the insights gained will be useful for any future emergency situation where such quick developments will be required. The apps being developed for nationwide vaccination management are examples of such emergency response application development that may benefit from the insight presented by this study.

To conduct this study, we collected a total of 455640 app reviews of 36 contact tracing apps released by governments/official authorities of 34¹ countries around 7 continents. Then we distribute the reviews to 7 corpora and apply topic modeling to each corpus independently. We label the topics discovered by the topic model and classify them into main categories. Then we systematically analyzed the identified topics in the collection of reviews. The contribution of this thesis is threefold:

- We have compiled a dataset of reviews of 35 COVID-19 contact tracing apps and divided them into seven corpus based on continents.
- We then divide the topics discussed into seven categories.
- Finally, we study the evolution of the topics across the continents.

1.1 Research Questions

Specifically, we explore the following research questions in this study:

1. What are the topics discussed in COVID-19 contact tracing app reviews across the continents?

We identify 31 topics in the app reviews. The topics can be grouped into seven categories - "Usability", "Reliability", "Performance", "Privacy and security", "Configuration", "Feature shortcoming", and "Version". Among the categories, Usability is the most popular one.

¹We collected data from 2 contact tracing apps of USA and Malaysia.

2. How does the popularity of the topics evolve over time?

We investigate the time-specific trends of the reviews and discover that user activity was at its highest level from April to July in 2020. During this period, the frequency of reviews under each category increased up to May and then gradually decreased.

3. How do the complaints and appraisals of the users vary across the topics?

Based on star reviews, we discover that 88% of reviews under Usability are user appraisals, while the reviews under the other categories are more about complaints than appraisals.

4. How do the complaints and appraisals of the users evolve across the continents?

We analyze the ratio of appraisal and complaint for each continent and observe that 88% of reviews are appreciative in Asia, higher than other continents. Europe, North America, Africa, and Latin America show relatively more appraisals than complaints. In the other continents, the ratio between appraisal and complaint differ marginally.

1.2 Outline

The thesis is organized as follows.

Chapter 1, the current chapter, is the introduction to our research work. This chapter briefly discusses the pandemic issues, concepts of contact tracing and finally summarizes the research questions.

Chapter 2 highlights the previous works in this field. We organize the related works into three fronts: studies on COVID-19 contact tracing apps, software engineering(SE) research related to COVID-19, topic modeling in SE research.

Chapter 3 provides an overview of LDA topic modeling with notations from the original paper. In chapter 4, we explain our methodology. We discuss the data scraping and preprocessing steps. We also show the experimental results of LDA topic modeling in this section.

Chapter 5 discusses the answers to the research questions and the implications of their findings. Chapter 6 is threats to validity followed by Sentiment Analysis in Chapter 7. Finally, Chapter 8 concludes the thesis.

Chapter 2

Related Works

Due to the huge importance, the Contract Tracing apps have gained attention from researchers right from the beginning. They have conducted multiple studies from different perspectives. Also, other software engineering research works are carried out related to various aspects of Covid 19. In this section, we briefly discuss these prior works. Since this is a topic modeling based study, previous similar studies are also explored.

2.1 Studies on COVID-19 Contact Tracing App

The privacy concern of the users was highly pronounced since the inception of contact tracing apps. This was reflected in the academic research also as discuss below. The underlying protocols, evaluation methods, and deployment architecture were also formally studied by the researchers.

Vaudenay [9] provides a comprehensive study of the advantages and vulnerabilities of centralized and decentralized contact tracing systems. Redmiles [10] explores user tradeoffs for different contact tracing architectures. Functions, evaluation methods, and the cost assessment of digital and traditional contact tracing methods are discussed in [11]. Simko et al. [12] survey public opinion on privacy concerns regarding contact tracing apps. Sun et al. [13] developed a security and privacy tool to examine security issues, privacy leaks, embedded trackers, and malwares in contact tracing apps. Azad et al. [14] conduct a manual investigation on privacy, security, and permission requirements of contact tracing apps. Li et al. [15] review the deployment of centralized and decentralized architectures. Reichart et al. [16] present a survey on various automatic contact tracing approaches and classified them on privacy risk score. Tang [17] studies privacy issues for privacy-aware contact tracing solutions and proposes an interdisciplinary research agenda to potential solutions. Cho et al. [18] study the privacy issues of contact tracing apps in depth. They propose techniques to develop a contact tracing app aiming to incorporate

privacy enhancements for all users. Ahmed et al. [19] present the key attributes of contact tracing apps, i.e., architecture, data management, privacy, security, contact estimation, and attack vulnerability. They study 15 tracing apps based on different tracing protocols.

2.2 Software Engineering research related to COVID-19

Software engineering research related to COVID-19 is limited. We have identified three recently published/shared papers in software engineering that have focused on the pandemic. Two of the papers [20, 21] investigate the impact of productivity and wellbeing of developers during the pandemic, while the third paper [22] analyzes the bugs in COVID-related apps.

The recently published article by Ralph et al. [20] surveyed 2,225 developers from 53 different countries and asked them questions about how their productivity and wellbeing are affected by COVID-19. They found that in general productivity and wellbeing have declined and they are closely related. This means that lack of wellbeing can impact a loss in productivity. They also found heterogeneity in the needs of the developers from their organizations, i.e., different people expected and needed different types of support. They also find that the pandemic may have affected more severely women, parents, and people with disabilities.

The second paper, still unpublished but generously shared by the authors Bao et al. [21] in arXiv, focuses on understanding wellbeing and productivity of Baidu employees while working from home during the COVID-19 pandemic. Baidu is a large Chinese software company. The paper quantitatively analyzes 4,000 records (e.g., code commits, reviews, etc.) of 139 developers' activities of 138 working days. Out of the records, 1,103 records are submitted when the developers worked from home due to the COVID-19 pandemic. Contrary to Ralph et al. [20], they found that work from home has both positive and negative impacts on developers' productivity in terms of different metrics, such as the number of commits.

Rahman and Farhana [22] studied bugs in open source projects related to COVID-19. They identified 8 categories of bugs in 7 different kinds of projects.

2.3 Topic Modeling in Software Engineering Research

Our motivation to use topic modeling to understand COVID-19 contact tracing app reviews stems from existing research in software engineering that shows that topics generated from textual contents can be good approximation of the underlying themes [23, 24, 25]. Topic models are used recently to understand software logging [26] and previously for diverse other tasks, such as concept and feature location [27, 28], tracability linking (e.g., bug) [29, 30], to understand software and source code history evolution [31, 32, 33], to facilitate code search by categorizing

software [34], to understand the challenges of blockchain development [35], to understand the evolution of new languages [36], to refactor software code base [37], as well as to explain software defect [38] and various software maintenance tasks [39, 24]. In between studies also focused on the effective use and parameterization of topic modeling for software engineering tasks [40, 41]. In this paper, we adhere to the recommended parameterization of topic modeling. The Stack Overflow (SO) posts are subject to a number of recent papers to study various aspects of software development using topic modeling, such as what developers are discussing in general [42], or about a particular aspect, e.g., concurrency [43], big data [44], chatbot development [45]. We are aware of no previous research which specifically focused on understanding the COVID-19 contact tracing app reviews.

Chapter 3

Background

3.1 Topic Modeling

Topic modeling is a statistical approach for discovering abstract topics from a collection of discrete data. One of the most popular tools for topic modeling is Latent Dirichlet Allocation (LDA) [46]. LDA is a generative probabilistic model for text documents. Blei et al. [46] uses the following notations:

- Word is the basic unit in the corpus. Word representation is a one-hot vector. For the vocabulary size of $|V|$, words are indexed by $\{1, 2, \dots, V\}$.
- Document is a sequence of words. A document of N words is represented by,
 $\mathbf{w}=(w_1, w_2, \dots, w_N)$
- A corpus of M documents is represented by $\{\mathbf{w}_1, \mathbf{w}_2, \dots, \mathbf{w}_M\}$

LDA considers document as mixture of topics and topic as mixture of words. LDA assumes the following process for document generation:

1. Select $N \sim \text{Poisson}(\xi)$
2. Select $\theta \sim \text{Dir}(\alpha)$
3. For each word, w_n in the document,
 - Select topic $z_n \sim \text{Multinomial}(\theta)$
 - Conditioning on z_n multinomial probability, $p(w_n|z_n, \beta)$, a word w_n is selected.

α and β are parameters which are approximated. β is a $k \times V$ matrix that denotes the probability of a word given a topic, e.g., $\beta_{ij} = p(w_j = 1 | z_i = 1)$. k is the dimensionality of the Dirichlet

variable θ , therefore, the number of topics.

Given a document, the posterior distribution of LDA model parameters and hidden variables can reveal latent semantics in the corpus to classify documents of different topics. However, the posterior distribution is computationally intractable, so inference algorithms such as variational approximation, Markov Chain, Monte Carlo, etc. are used. Hoffman et al. [47] proposed an online variational inference algorithm for LDA. It is based on online stochastic optimization, does not need to store the collection of documents, and converges faster than batch algorithms on large datasets. We use the open-source NLP framework Gensim [48] for topic modeling. The core estimation for LDA in Gensim is the online variational inference algorithm.

Chapter 4

Data Collection and Preprocessing

4.1 Data Collection

We download app review data from Google Play Store through an open-source scraping tool `google-play-scraper`¹. It retrieves the reviews with the additional information, i.e., a unique review ID, user name, star rating, timestamp, count of thumbs up of the review, and app version corresponding to the review. Our initial dataset contains 4,55,640 reviews in total from 36 apps. The app names with the corresponding country names are shown in Table 4.1

4.2 Preprocessing

The raw data is very noisy, so we perform the following preprocessing steps.

- First, we discard the URL and emoticons from the reviews. We use an open-source tool called `demoji`² to detect and remove emoticons.
- Then we use `Googletrans`³, a python google translate API, to detect a particular review's source language. We only considered reviews that are in English.
- English words do not have the same letter three consecutive times. We modify words containing the same letter continuously for more than three times to two occurrences of that letter. This step, for example, converts the word "goooooood" to "good".
- We inspect the most frequent numerical terms in the reviews and select the relevant words to include and discard all other numbers. An example of a selected numerical term is

¹<https://github.com/JoMingyu/google-play-scraper>

²<https://github.com/bsolomon1124/demoji>

³<https://github.com/ssut/py-googletrans>

Table 4.1: The contact tracing apps in the dataset

Continent	App Name	Country
Africa	E7mi	Tunisia
	Wiqaytna	Morocco
	COVID Alert South Africa	South Africa
	GH COVID-19 Tracker	Ghana
Asia	MyTrace	Malaysia
	TraceTogether	Singapore
	Care Protect (pedulilindungi)	Indonesia
	Aarogya Setu	India
	GERAK MALAYSIA	Malaysia
	Corona Tracer BD	Bangladesh
Australasia	NZ COVID Tracer	New Zealand
	COVIDSafe	Australia
Europe	VirusSafe	Bulgaria
	Corona-Warn-App	Germany
	Rakning C-19	Iceland
	Smittestop	Denmark
	NHS COVID-19	UK
	VírusRadar	Hungary
	STOP COVID - ProteGO Safe	Poland
	Apturi Covid Latvia – SPKC	Latvia
	Immuni	Italy
	Stopp Corona	Austria
	COVID Tracker Ireland	Ireland
	Smittestopp	Norway
	StopCovid France	France
Latin America	CoronApp - Colombia	Colombia
	Coronavirus UY	Uruguay
	CUIDAR COVID-19 ARGENTINA	Argentina
	Coronavírus - SUS	Brasil
Middle East	EHTERAZ	Qatar
	BeAware Bahrain	Bahrain
	AMAN - Aman.jo JORDAN APP COVID-19	Jordan
	Hamagen	Israel
North America	COVIDWISE	Virginia - USA
	Stop Covid - let's fight this together	Georgia - USA
	COVID Alert	Canada

”109” representing the helpline number of Qatar’s contact tracing app.

- Smartphone model name with number is often found in the reviews. We search for such device identity-specific terms and collapse them into the term ”phone”. Hence, when a review contains the words ”Samsung A40”, this step transforms the words to ”phone”.
- We use a list of stopwords containing common English words that contribute nothing to

the topics. Our stopwords list comprises the default list of stopwords in Natural Language Toolkit, NLTK [49]. The list also contains specific stopwords such as the name of an application, country name, country code, etc. that we use independently for each application. Terms frequently used in application reviews are also included in the stopwords list. Examples of such terms are "app", "please", "fix", etc.

- Stemming reduces inflectional and derivationally related forms of a word to a common base form. We use snowball stemmer by NLTK. For example, stemming maps the words "permission", "permissions", and "permissible" to "permiss".
- We also use the Parts of Speech (POS) tagger by NLTK for conducting experiments on the selection of POS. We apply POS-tagging on entire reviews (with context) and word by word (without context). We consider both tags for the experiments.

Users from a variety of backgrounds comment on the play store. So the reviews can be informative and helpful, short and without context, mistakenly spelled, or words from the native language used in English, therefore misleading. We examined some translations performed by the translation tool we used and noticed inconsistencies. So we discarded reviews that are not detected as English by the tool. However, this tool fails to detect the source language correctly for some reviews.

4.3 LDA topic modeling

We group the app reviews into seven corpora according to the continents. Then, we apply LDA to each corpus individually. We experiment by varying the number of topics, parts of speech selection, and dictionary filtering. The number of topics is a model parameter that plays a vital role in performance improvement. The coherence score [50] is an appropriate metric for evaluating the interpretability of the discovered topics. We use Gensim module CoherenceModel that implements the four stage topic coherence pipeline proposed by Röder et al. [51] to compute the topic coherence for a specific value of the number of topics. We experiment for different values to maximize the coherence value. We run the LDA training ten times for different random seeds and store the best coherence score and the mean of coherence scores of 10 runs. Furthermore, we manually check the top 15 reviews of a topic based on the probability of being assigned to that topic. Relying on the coherence value and manual examination, we finalize the number of topics for each corpus. As mentioned earlier, our translation tool struggles with language detection and translation, resulting in some reviews, not in English detected as English review. Those reviews fall under the same topic after LDA model training and are discarded. Table 4.2 displays the results of our experiment.

Topics Number	POS	Best Score	Mean Score
Africa			
5	All	0.523	0.436
6	All	0.501	0.449
7	All	0.478	0.439
8	All	0.492	0.462
9	All	0.531	0.460
Asia			
15	All	0.625	0.599
20	All	0.639	0.601
25	All	0.634	0.617
30	All	0.644	0.620
29	All	0.663	0.624
32	All	0.649	0.622
35	All	0.659	0.626
27	All	0.648	0.634
28	All	0.664	0.623
27	NOUN VERB ADJ	0.639	0.615
28	NOUN VERB ADJ	0.647	0.613
27	All	0.649	0.625
28	All	0.646	0.631
Australasia			
25	NOUN VERB ADJ	0.627	0.601
30	NOUN VERB ADJ	0.609	0.594
35	NOUN VERB ADJ	0.604	0.579
25	All	0.635	0.612
30	All	0.627	0.600
35	All	0.610	0.594
22	All	0.638	0.618
20	All	0.656	0.619
15	All	0.637	0.617
18	All	0.642	0.620
Europe			
15	All	0.695	0.639
20	All	0.665	0.639
22	All	0.662	0.637
25	All	0.662	0.646
Continued on next page			

Table 4.2 – continued from previous page

Topics	POS	Best Score	Mean Score
27	All	0.648	0.620
30	All	0.668	0.627
35	All	0.645	0.616
38	All	0.644	0.612
10	All	0.636	0.588
13	All	0.663	0.610
17	All	0.669	0.629
14	NOUN VERB ADJ NUM	0.646	0.617
16	NOUN VERB ADJ NUM	0.664	0.607
15	NOUN VERB ADJ NUM	0.659	0.614
17	NOUN VERB ADJ NUM	0.676	0.622
30	NOUN VERB ADJ NUM	0.637	0.612
15	NOUN VERB ADJ	0.677	0.623
17	NOUN VERB ADJ	0.657	0.625
14	All	0.659	0.626
16	All	0.654	0.622
15	All	0.653	0.611
17	All	0.669	0.641
25	All	0.675	0.642
25	NOUN VERB ADJ	0.660	0.639
LatinAmerica			
5	All	0.474	0.417
6	All	0.467	0.422
7	All	0.447	0.415
8	All	0.459	0.413
9	All	0.465	0.422
MiddleEast			
18	All	0.673	0.646
20	All	0.672	0.633
22	All	0.636	0.621
25	All	0.648	0.630
15	All	0.674	0.642
28	All	0.628	0.608
30	All	0.643	0.612
10	All	0.671	0.641
Continued on next page			

Table 4.2 – continued from previous page

Topics	POS	Best Score	Mean Score
12	All	0.648	0.622
NorthAmerica			
8	All	0.605	0.576
10	All	0.612	0.571
12	All	0.613	0.567
15	All	0.613	0.560
18	All	0.583	0.553
20	All	0.562	0.551
22	All	0.569	0.553
25	All	0.565	0.541
28	All	0.552	0.532
30	All	0.551	0.534
5	All	0.596	0.553
7	All	0.586	0.566
8	NOUN VERB ADJ NUM	0.594	0.548
10	NOUN VERB ADJ NUM	0.600	0.560
15	NOUN VERB ADJ NUM	0.566	0.550
12	NOUN VERB ADJ NUM	0.595	0.561
8	NOUN VERB ADJ	0.629	0.552
10	NOUN VERB ADJ	0.572	0.547
12	NOUN VERB ADJ	0.596	0.558
8	NOUN VERB ADJ	0.616	0.563
10	NOUN VERB ADJ	0.571	0.551
12	NOUN VERB ADJ	0.583	0.553

Table 4.2: LDA Experiments Results

After the preprocessing and topic modeling pipeline, we have 370,524 labeled reviews. We discover 31 topics in the app reviews. We discuss the topics in detail in the next chapter.

Chapter 5

Empirical Study

In this research, we want to systematically study the issues reported in the user reviews of different contact tracing apps across different continents. We explore the following research questions to develop an insight into the user feedback:

1. What are the topics discussed in COVID-19 contact tracing app reviews across the continents?
2. How does the popularity of the topics evolve over time?
3. How do the complaints and appraisals of the users vary across the topics?
4. How do the complaints and appraisals of the users evolve across the continents?

5.1 Topics in COVID-19 contact tracing app reviews (RQ1)

5.1.1 Motivation

A major benefit we may receive by analyzing the app reviews is to have insight about user requirement of new features, bug reports, user experience(UX) and security issues, etc. The categorization should be helpful to assign development teams for addressing those. Due to the substantial difference in the nature of development of different software components, e.g., UI, security mechanisms, etc., it would expedite the new release if issues under a topic are addressed by the team specialized in it.

5.1.2 Approach

We determine 31 topics from 7 continents with overlapping topics among continents. Individually, the topic count is 12 in Europe, 7 in North America, 16 in Australasia, 12 in Asia, 12 in Middle-East, 4 in Africa, and 5 in Latin America. We assign the dominant topic to a review. The dominant topic is the topic with the maximum probability in that review's topic distribution. We categorize the topics into seven software engineering specific categories, i.e., "Usability", "Reliability", "Performance", "Privacy and security", "Configuration", "Feature shortcoming", and "Version".

5.1.3 Results

[5.1](#) displays the topics determined by our approach. The first column holds the topics grouped into categories. The next column represents the percentage of topics across different continents. Each color denotes a continent, followed by the continent's topic frequency.

UX related reviews are noticed across all the continents, where the majority of them are satisfactory. Users also propose suggestions for the improvement of the app's user interface(UI). In Latin America, reviews under Usability mostly reveal UI issues such as complicated UI of the app. In Asia, maximum reviews are from the contact tracing app of India. Users are satisfied with the app's overall functionality; hence, 77% of the reviews show the feeling of positive UX. Users in North America, Australasia, and Middle East are concerned about the app's privacy threat. Security issues involving authorization and authentication also exist in reviews across all the continents. Another common pattern in reviews across most continents is the app's compatibility with devices and the operating system. In some cases, the app cannot operate on older devices. Also, the app does not work on the operating system of certain versions. In general, the apps of all the continents have a major drawback, i.e., they drain most of the battery power in less than acceptable time. Users also express concern about the app's impact on different components of the phone, e.g., battery, Bluetooth, GPS, which indicates that some performance optimization techniques may not be properly implemented in the quick development period.

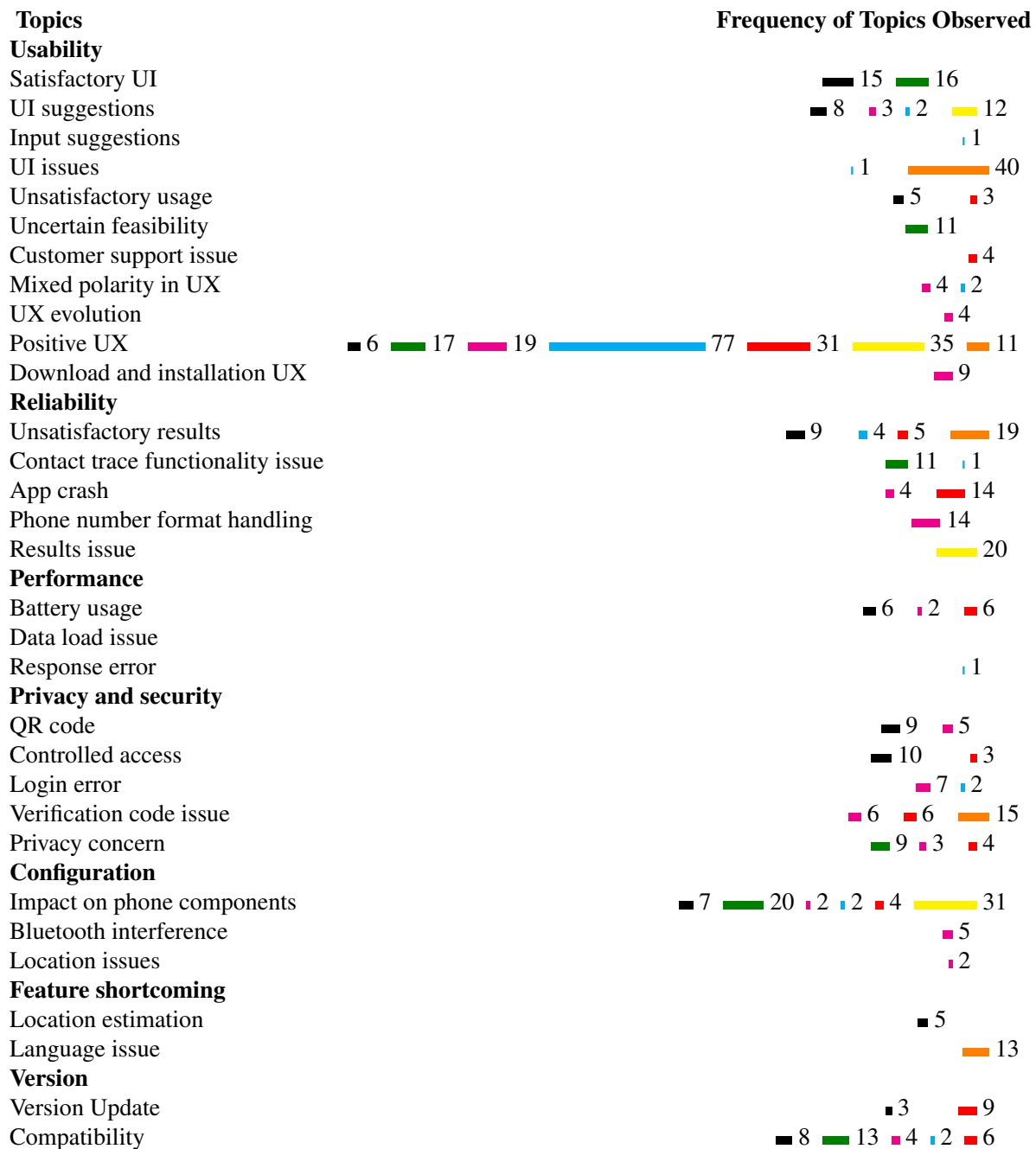
Table [A.1](#) shows a representative example of each topic.

5.2 Evolution of the popularity of topics over time(RQ2)

5.2.1 Motivation

There is a difference in culture and regulations of the countries of different continents. For example, the privacy concern of the app users in Europe or North America may vary from that of underdeveloped countries. The types/configurations of phones used by the mass population

Table 5.1: Distribution (frequency) of topics in our dataset per category. Each colored bar denotes a continent (Black = Europe, Green = North America, Magenta = Australasia, Blue = Asia, Red= Middle East, Yellow = Africa, Orange = Latin America)



depend on the economic condition of the country. The Government of some countries can enforce safety policies in a more strict way than the others where the privacy concern of the citizens is given high importance. The impact of such differences on app users' experience is expected to be revealed from the analysis of reviews. This would benefit the developers of an app targeting a specific continent to anticipate the expectation of the users of the target region.

5.2.2 Approach

In order to examine the time-specific trends of popularity, we compare the absolute impact and relative impact of the topics under different categories. The choice of these metrics is made from the consideration that frequency based metrics only consider the dominant topic with the highest probability and may miss the involvement of other topics. The absolute impact score of a topic in a particular month is the sum of the topic's probabilities for all reviews in that month. Therefore, the absolute impact of a category sums up the absolute impact of all topics under this category. We calculate this metric for all categories in the time period of February to September of 2020. The relative impact score of a topic in a specific month is the corresponding absolute impact scaled by the total reviews in that month. We compute the relative impact of a category by summing up the relative impact of topics under the category. Finally, we plot the relative impact score of different categories between the aforementioned time interval.

5.2.3 Results

The absolute impact score trend and the relative impact score trend of different categories between the period February 2020 and September 2020 are shown in 5.1 and 5.2 respectively.

It shows that the absolute impact of Usability was significantly higher compared to other categories during the period of April-July. A possible reason is that user satisfactions are likely to be expressed immediately at the initial stages after the release (March in this context). Also, users might get used to the UX limitations within a few months, and negative UX reviews are likely to decline in number, too.

The relative impact score plot in 5.2 shows the temporal evolution of topics under a particular category. Usability is again the dominating category for the whole period. The relative scores of the categories Reliability, Performance, and Version remain more or less stable over time. Interestingly, there is an indication of an increase in the trend of Privacy and security in the next two months.

5.3 User complaints and appraisals across topics (RQ3)

5.3.1 Motivation

User feedback can be associated with a number rating in the range of 1 to 5. Typically, users give a high rating when they are pleased with the app, and they give a low rating when they find the app problematic. Star rating is the simplest metric to analyze the users' emotional expressions. Analyzing the star rating across different categories of the topics would help understand the

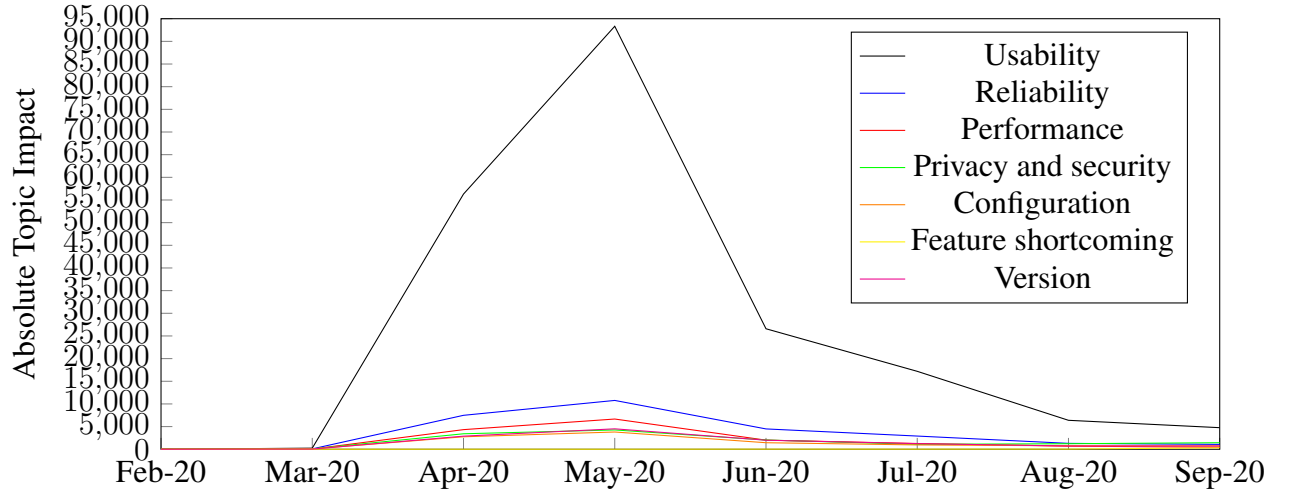


Figure 5.1: The absolute impact scores of different categories between February 2020 and September 2020

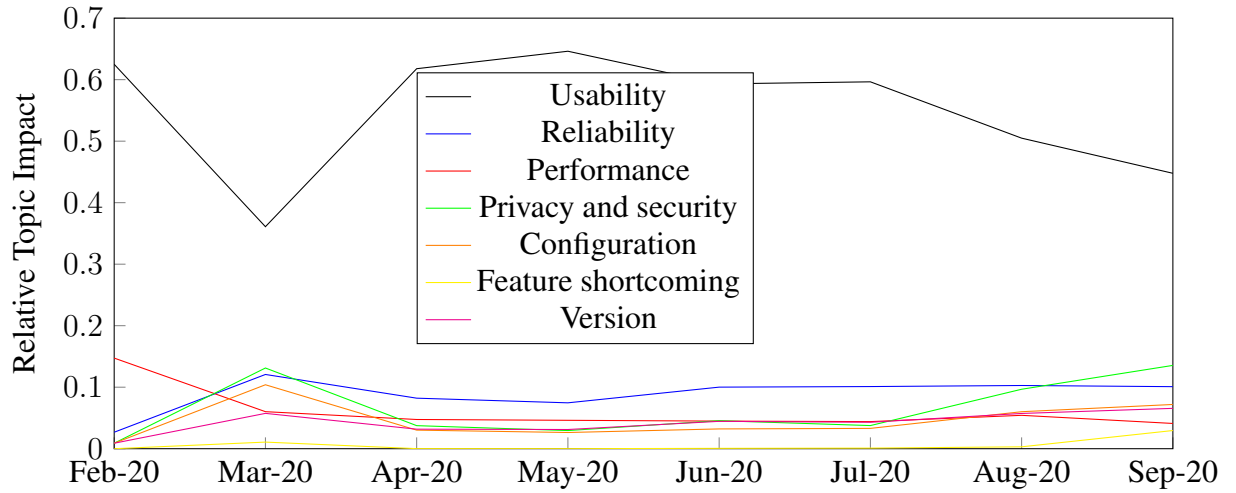


Figure 5.2: The relative impact scores of different categories between February 2020 and September 2020

degree to which users appreciate or dissent the app. This study outcome would help developers assign resources to address the issues belonging to the topics with a high ratio of complaints.

5.3.2 Approach

For each of the seven categories, we calculate the proportion of appraising reviews and complaining reviews. We consider a review as an appraisal if the corresponding star rating is greater or equal to 3. Otherwise, we consider the review as a complaint.

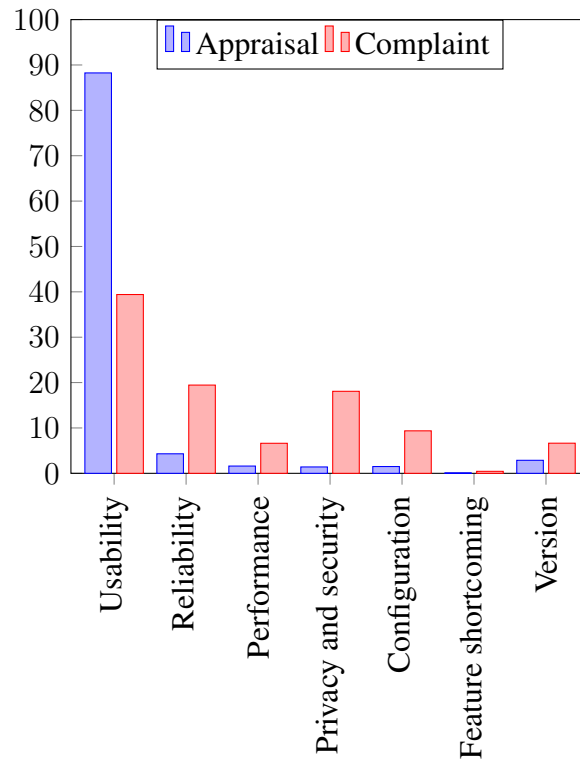


Figure 5.3: The star rating of topics across different categories

5.3.3 Results

5.3 shows the proportions of appraisals and complaints for each topic category. As expected, most of the appreciative ratings are from the reviews under Usability. It was evident from the results of RQ1 in Table 5.1 that most of the reviews in Usability are about positive UX. Therefore, 88% of users reviewing UX gave high ratings. The other topic categories contain more complaints than appraisals, as reviews of these categories involve the app’s failure to live up to the users’ expectations.

5.4 Evolution of user complaints and appraisals across the continents (RQ4)

5.4.1 Motivation

Conducting a comparative analysis of the star ratings of different continents can capture how the overall feeling of the users about the app varies from continent to continent. The outcome of this analysis, along with that of RQ2, will allow the developers of an app for a particular continent to understand the expectations of their potential users.

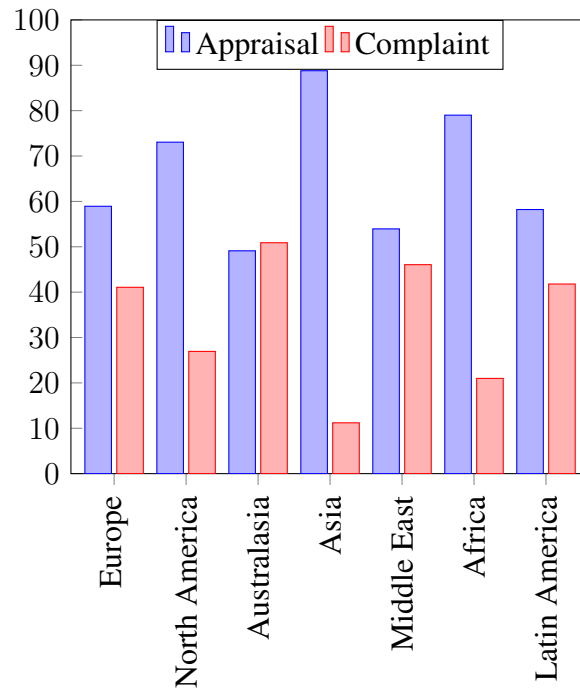


Figure 5.4: The star rating of reviews across different continents

5.4.2 Approach

For the two classes of reviews, i.e., appraisal and complaint, we calculate the percentage of reviews belonging to each class for the seven continents and graphically show the results.

5.4.3 Results

We observe that all continents except Australasia have more reviews about appraisals than complaints. For Australasia, the percentage of the two classes differ slightly. Asia has the highest portion of appreciative reviews among all continents. Again, the contact tracing app of India contributed to this high percentage.

5.5 Discussions

This study is expected to provide actionable insight to the developers of software/mobile applications as well as benefit the users. The implications of our findings are discussed below.

5.5.1 Implications to the contact tracing app developers

The major findings of this study are related to the user reactions expressing satisfaction about certain features of different apps and also their complaints about some issues or lack of features. Contact tracing application developers are the direct recipient of the insight obtained from this systematic study. They can better plan to address the issues in upcoming releases. They can be cautious while developing new features that users do not experience the same problems again. They also understand the differences in expectations of users or their experienced issues related to the surrounding environment of different geo-political regions.

5.5.2 Implications to the emergency response app developers

Contact tracing apps were developed hastily as an emergency response. These apps need to be developed without enough spending effort to design features, follow best programming practice, and test comprehensively as the developers have to meet a crying need. Many of the findings of this study would benefit the engineers of any app developed under similar circumstances. An example at hand is the current need for apps for the management of vaccination across the countries. Many issues found from the feedback of contact tracing apps are relevant here. Hence, the benefit of this study to the developers is beyond the immediate impact on contact tracing app providers.

5.5.3 Implications to the users

The most prevalent issues identified through this study are expected to attract the attention of developers, and they can prioritize their fixing plans accordingly. Consequently, there will be a quick release of more robust and user-friendly versions. The users will be relieved of their concerns and feel comfortable to use the apps when they find that their needs are addressed. This is likely to create an overall positive reputation that should increase the number of users significantly. The success of the attempts like contact tracing through apps depends on the size of a large user-base. Hence, this study is expected to contribute to widening the user-base and thus benefit the ultimate cause to a great extent.

Chapter 6

Threats to Validity

We discuss the threats to validity of our studies following common guidelines for empirical studies [52].

6.1 External Validity

The primary external threat to validity is the generalizability of our findings. Our study revolves around the reviews of COVID-19 contact tracing apps. Therefore, the results of our research may not generalize to non-COVID domains. Moreover, our analysis involves reviews from February 2020 to September 2020. A drastic change in the review patterns could make our approach inapplicable to the new reviews in the coming months.

6.2 Internal Validity

The internal threats to validity concern experimental bias and errors while conducting the analysis. The internal threats involving topic modeling are the performance degradation of LDA on short texts and selecting the optimal number of topics. LDA leverages word co-occurrence patterns to infer the latent topic distribution of documents. However, the lack of word co-occurrence information in short text causes a decrease in LDA performance. In our dataset, the LDA model fails to assign a particular topic to some short reviews. For example, the review "spy app" is assigned to multiple topics of more than one category with the same probability. Choosing the optimal number of topics is a complicated process. To address this threat, we use the four-stage topic coherence metric in [51] to select the appropriate value of the number of topics. Moreover, LDA renders different results on the same corpus for multiple runs. Hence, we run LDA models ten times and consider the overall best number of topics that yields the

desired score. We noticed a minimal difference in the coherence value for multiple runs on the optimal number of topics.

6.3 Construct Validity

The threat of Construct Validity, in our context, stems from the possibility that labeling the resulting topics from the LDA might not reflect the reviews associated with the topics. To minimize this threat, we individually examined 15 reviews having the highest probability belonging to each topic to assign labels to that topic. Then we discussed with two experts including our supervisor sir to reach a consensus about the labels to ensure that the labels correctly represent the topics. To merge the topics into subcategories and categories, similarly, we did the initial assignment, and those were validated by detailed discussion involving all four individuals. Since this is a standard practice in SE research, we believe it substantially reduces the threat to construct validity.

Chapter 7

Sentiment Analysis

Sentiment analysis is the computational analysis of affective states and subjective information in text documents. Usually, the goal is to quantify the underlying sentiment polarity(positive, negative, or neutral) of the text.

7.1 Related Work

Sentiment analysis has been utilized in the software engineering(SE) context to inspect app reviews [53], tweets on software development [54], discussions in collaborative development platforms such as GitHub [55], [56], Jira [57], [58], and Stack Overflow [59]. Initial works applied existing sentiment analysis tools(SentiStrength [60], Alchemy, NLTK, Stanford CoreNLP, and EmoTxt) from non-SE domain. The shortcomings of such tools [61], [62] motivated the development of SE-specific tools: SentiStrengthSE, Senti4SD and SentiCR. Islam et al. and Novielli et al. compared sentiment analysis tools in SE domain. Zhang et al. conducted a comparative study on sentiment analysis tasks between five existing SE-specific tools and four pretrained Transformer models: BERT, RoBERTa, XLNet, and ALBERT. Their results show that pretrained Transformer models outperform prior tools. Transformer is an encoder decoder architecture that relies solely on attention mechanism for dependencies between input and output. The encoder and decoder layers consist of two sublayers: multi-head self-attention and position-wise fully connected feed-forward network. In addition, the decoder does multi-head attention over encoder outputs. Of the four models, we have finetuned BERT for sentiment analysis on our dataset. BERT stands for Bidirectional Encoder Representations from Transformers. BERT pretrains a deep bidirectional transformer on "masked language model" and "next sentence prediction". The pretrained parameters can be easily finetuned for downstream tasks.

Continent	Positive	Negative	Neutral
Europe	35.5891	36.4350	27.9758
North America	49.2981	23.4699	27.2318
Australasia	26.3074	53.5994	20.0931
Middle East	40.1518	47.9737	11.8744
Africa	60.2389	17.7474	22.0136
Latin America	49.2537	25.3731	25.3731
Asia	79.7366	9.1390	11.1242

Table 7.1: Sentiments of reviews across continents

7.2 Fine Tuning BERT

We manually labeled 7060 reviews for sentiment classification. We use a train and test set of 5400 and 1660 reviews, respectively. The training set has 1800 reviews for three sentiment classes: Positive, Negative, and Neutral. We fine tune BERT-base on the training dataset for 400 epochs. Finally, we achieve a F1-score of 0.78 on the test dataset.

7.3 Sentiment Evaluation

We use our fine-tuned BERT model to evaluate the sentiments of reviews across all continents. Table 7.1 shows the results in percentage.

Chapter 8

Conclusion

This book presents a study on different topics the users discuss about the COVID-19 contact tracing apps across the continents by analyzing the reviews of the apps. It reveals specific problems users had to face, the concerns they have about the apps, and the positive/negative experience they share about the design and usage of the apps. We identified around 31 topics from the 35 apps under study from all over the world. Topics related to usability and performance of the apps are prevalent across all regions. Users often complained about the limitation of features and complex user interfaces in the apps, as well as the negative effect of such apps on their mobile batteries. Despite some negative experiences, many users considered the app as a utility tool to be wary of the potential sources of infection. Users also expressed concerns about the privacy implications of the app. Our study analyzes all these kinds of topics in a structured manner. The outcome of this study is expected to benefit the developers to address the issues in a planned way and release improved versions with informed decisions.

References

- [1] T. P. Velavan and C. G. Meyer, “The covid-19 epidemic,” *Tropical medicine & international health*, vol. 25, no. 3, p. 278, 2020.
- [2] K. T. D. Eames and M. J. Keeling, “Contact tracing and disease control,” *Proceedings of the Royal Society of London. Series B: Biological Sciences*, vol. 270, no. 1533, pp. 2565–2571, 2003.
- [3] P. O’Neill, T. Ryan-Mosley, and B. Johnson, “A flood of coronavirus apps are tracking us. now it’s time to keep track of them,” 2020. [Online] Available: <https://www.technologyreview.com/2020/05/07/1000961/launching-mitttr-covid-tracing-tracker/>.
- [4] M. Harman, Y. Jia, and Y. Zhang, “App store mining and analysis: Msr for app stores,” in *2012 9th IEEE Working Conference on Mining Software Repositories (MSR)*, pp. 108–111, 2012.
- [5] L. V. G. Carreño and K. Winbladh, “Analysis of user comments: An approach for software requirements evolution,” in *2013 35th International Conference on Software Engineering (ICSE)*, pp. 582–591, 2013.
- [6] D. Pagano and W. Maalej, “User feedback in the appstore: An empirical study,” in *2013 21st IEEE International Requirements Engineering Conference (RE)*, pp. 125–134, 2013.
- [7] R. Vasa, L. Hoon, K. Mouzakis, and A. Noguchi, “A preliminary analysis of mobile app user reviews,” in *Proceedings of the 24th Australian Computer-Human Interaction Conference, OzCHI ’12*, (New York, NY, USA), p. 241–244, Association for Computing Machinery, 2012.
- [8] N. Seyff, F. Graf, and N. Maiden, “Using mobile re tools to give end-users their own voice,” in *2010 18th IEEE International Requirements Engineering Conference*, pp. 37–46, 2010.
- [9] S. Vaudenay, “Centralized or decentralized? the contact tracing dilemma,” tech. rep., 2020.

- [10] E. M. Redmiles, “User concerns & tradeoffs in technology-facilitated contact tracing,” *arXiv preprint arXiv:2004.13219*, 2020.
- [11] V. Von Wyl, S. Bonhoeffer, E. Bugnion, M. A. Puhon, M. Salathé, T. Stadler, C. Troncoso, E. Vayena, and N. Low, “A research agenda for digital proximity tracing apps,” *Swiss medical weekly*, vol. 150, p. w20324, 2020.
- [12] L. Simko, R. Calo, F. Roesner, and T. Kohno, “Covid-19 contact tracing and privacy: Studying opinion and preferences,” *arXiv preprint arXiv:2005.06056*, 2020.
- [13] R. Sun, W. Wang, M. Xue, G. Tyson, S. Camtepe, and D. Ranasinghe, “Vetting security and privacy of global covid-19 contact tracing applications,” *arXiv preprint arXiv:2006.10933*, 2020.
- [14] M. A. Azad, J. Arshad, S. M. A. Akmal, F. Riaz, S. Abdullah, M. Imran, and F. Ahmad, “A first look at privacy analysis of covid-19 contact tracing mobile applications,” *IEEE Internet of Things Journal*, 2020.
- [15] J. Li and X. Guo, “Covid-19 contact-tracing apps: A survey on the global deployment and challenges,” *arXiv preprint arXiv:2005.03599*, 2020.
- [16] L. Reichert, S. Brack, and B. Scheuermann, “A survey of automatic contact tracing approaches,”
- [17] Q. Tang, “Privacy-preserving contact tracing: current solutions and open questions,” *arXiv preprint arXiv:2004.06818*, 2020.
- [18] H. Cho, D. Ippolito, and Y. W. Yu, “Contact tracing mobile apps for covid-19: Privacy considerations and related trade-offs,” *arXiv preprint arXiv:2003.11511*, 2020.
- [19] N. Ahmed, R. A. Michelin, W. Xue, S. Ruj, R. Malaney, S. S. Kanhere, A. Seneviratne, W. Hu, H. Janicke, and S. K. Jha, “A survey of covid-19 contact tracing apps,” *IEEE Access*, vol. 8, pp. 134577–134601, 2020.
- [20] P. Ralph, S. Baltes, G. Adisaputri, R. Torkar, V. Kovalenko, M. Kalinowski, N. Novielli, S. Yoo, X. Devroey, X. Tan, *et al.*, “Pandemic programming: How covid-19 affects software developers and how their organizations can help,” *arXiv preprint arXiv:2005.01127*, 2020.
- [21] L. Bao, T. Li, X. Xia, K. Zhu, H. Li, and X. Yang, “How does working from home affect developer productivity?—a case study of baidu during covid-19 pandemic,” *arXiv preprint arXiv:2005.13167*, 2020.
- [22] A. Rahman and E. Farhana, “An exploratory characterization of bugs in covid-19 software projects,” *arXiv preprint arXiv:2006.00586*, 2020.

- [23] T.-H. P. Chen, S. W. Thomas, and A. E. Hassan, “A survey on the use of topic models when mining software repositories,” *Empirical Software Engineering*, vol. 21, no. 5, pp. 1843–1919, 2016.
- [24] X. Sun, B. Li, Y. Li, and Y. Chen, “What information in software historical repositories do we need to support software maintenance tasks? an approach based on topic model,” *Computer and Information Science*, pp. 22–37, 2015.
- [25] X. Sun, X. Liu, B. Li, Y. Duan, H. Yang, and J. Hu, “Exploring topic models in software engineering data analysis: A survey,” in *17th IEEE/ACIS International Conference on Software Engineering, Artificial Intelligence, Networking and Parallel/Distributed Computing*, pp. 357–362, 2016.
- [26] H. Li, T.-H. P. Chen, W. Shang, and A. E. Hassan, “Studying software logging using topic models,” *Empirical Software Engineering*, vol. 23, p. 2655–2694, 2018.
- [27] B. Cleary, C. Exton, J. Buckley, and M. English, “An empirical analysis of information retrieval based concept location techniques in software comprehension,” *Empirical Software Engineering*, vol. 14, pp. 93–130, 2009.
- [28] D. Poshyvanyk, Y.-G. Guéhéneuc, A. Marcus, G. Antoniol, and V. T. Rajlich, “Feature location using probabilistic ranking of methods based on execution scenarios and information retrieval,” *IEEE Transactions on Software Engineering*, vol. 33, no. 6, pp. 420–432, 2007.
- [29] S. Rao and A. C. Kak, “Retrieval from software libraries for bug localization: a comparative study of generic and composite text models,” in *8th Working Conference on Mining Software Repositories*, p. 43–52, 2011.
- [30] H. U. Asuncion, A. U. Asuncion, and R. N. Tylor, “Software traceability with topic modeling,” in *Proc. 32nd Intl. Conf. Software Engineering*, pp. 95–104, 2010.
- [31] J. Hu, X. Sun, D. Lo, and B. Li, “Modeling the evolution of development topics using dynamic topic models,” in *IEEE 22nd International Conference on Software Analysis, Evolution, and Reengineering*, pp. 3–12, 2015.
- [32] S. W. Thomas, B. Adams, A. E. Hassan, and D. Blostein, “Studying software evolution using topic models,” *Science of Computer Programming*, vol. 80, no. B, pp. 457–479, 2014.
- [33] S. W. Thomas, B. Adams, A. E. Hassan, and D. Blostein, “Modeling the evolution of topics in source code histories,” in *8th working conference on mining software repositories*, pp. 173–182, 2011.

- [34] K. Tian, M. Reville, and D. Poshyvanyk, "Using latent dirichlet allocation for automatic categorization of software," in *6th international working conference on mining software repositories*, pp. 163–166, 2009.
- [35] I. Alahi, M. Islam, A. Iqbal, and A. Bosu, "Identifying the challenges of the blockchain community from stackexchange topics and trends," in *2019 IEEE 43rd Annual Computer Software and Applications Conference (COMPSAC)*, 2019.
- [36] P. Chakraborty, R. Shahriyar, and A. Iqbal, "Empirical analysis of the growth and challenges of new programming languages," in *2019 IEEE 43rd Annual Computer Software and Applications Conference (COMPSAC)*, 2019.
- [37] G. Bavota, R. Oliveto, M. Gethers, D. Poshyvanyk, and A. D. Lucia, "Methodbook: Recommending move method refactorings via relational topic models," *IEEE Transactions on Software Engineering*, vol. 40, no. 7, pp. 671–694, 2014.
- [38] T.-H. Chen, S. W. Thomas, M. Nagappan, and A. E. Hassan, "Explaining software defects using topic models," in *9th working conference on mining software repositories*, pp. 189–198, 2012.
- [39] X. Sun, B. Li, H. Leung, B. Li, and Y. Li, "Msr4sm: Using topic models to effectively mining software repositories for software maintenance tasks," *Information and Software Technology*, vol. 66, pp. 671–694, 2015.
- [40] A. Panichella, B. Dit, R. Oliveto, M. D. Penta, D. Poshyvanyk, and A. D. Lucia, "How to effectively use topic models for software engineering tasks? an approach based on genetic algorithms," in *International Conference on Software Engineering*, pp. 522–531, 2013.
- [41] A. Panichella, B. Dit, R. Oliveto, M. D. Penta, D. Poshyvanyk, and A. D. Lucia, "Parameterizing and assembling ir-based solutions for se tasks using genetic algorithms," in *23rd IEEE international conference on software analysis, evolution, and reengineering*, 2016.
- [42] A. Barua, S. W. Thomas, and A. E. Hassan, "What are developers talking about? an analysis of topics and trends in stack overflow," *Empirical Software Engineering*, pp. 1–31, 2012.
- [43] S. Ahmed and M. Bagherzadeh, "What do concurrency developers ask about?: A large-scale study using stack overflow," in *Proceedings of the 12th ACM/IEEE International Symposium on Empirical Software Engineering and Measurement*, p. Article No. 30, 2018.
- [44] M. Bagherzadeh and R. Khatchadourian, "Going big: A large-scale study on what big data developers ask," in *Proceedings of the 2019 27th ACM Joint Meeting on European*

- Software Engineering Conference and Symposium on the Foundations of Software Engineering*, ESEC/FSE 2019, (New York, NY, USA), pp. 432–442, ACM, 2019.
- [45] A. Abdellatif, D. Costa, K. Badran, R. Abdalkareem, and E. Shihab, “Challenges in chatbot development: A study of stack overflow posts,” in *17th International Conference on Mining Software Repositories*, October 5–6, 2020, Seoul, Republic of Korea. New York, NY, USA, ACM, 2020.
- [46] D. M. Blei, A. Y. Ng, and M. I. Jordan, “Latent dirichlet allocation,” *Journal of machine Learning research*, vol. 3, no. Jan, pp. 993–1022, 2003.
- [47] M. Hoffman, F. Bach, and D. Blei, “Online learning for latent dirichlet allocation,” *advances in neural information processing systems*, vol. 23, pp. 856–864, 2010.
- [48] R. Řehůřek and P. Sojka, “Software Framework for Topic Modelling with Large Corpora,” in *Proceedings of the LREC 2010 Workshop on New Challenges for NLP Frameworks*, (Valletta, Malta), pp. 45–50, ELRA, May 2010.
- [49] S. Bird, E. Klein, and E. Loper, *Natural language processing with Python: analyzing text with the natural language toolkit*. ” O’Reilly Media, Inc.”, 2009.
- [50] D. Newman, J. H. Lau, K. Grieser, and T. Baldwin, “Automatic evaluation of topic coherence,” in *Human language technologies: The 2010 annual conference of the North American chapter of the association for computational linguistics*, pp. 100–108, 2010.
- [51] M. Röder, A. Both, and A. Hinneburg, “Exploring the space of topic coherence measures,” in *Proceedings of the eighth ACM international conference on Web search and data mining*, pp. 399–408, 2015.
- [52] C. Wohlin, P. Runeson, M. Höst, M. C. Ohlsson, B. Regnell, and A. Wesslén, *Experimentation in software engineering: an introduction*. Norwell, MA, USA: Kluwer Academic Publishers, 2000.
- [53] W. Maalej, Z. Kurtanović, H. Nabil, and C. Stanik, “On the automatic classification of app reviews,” *Requirements Engineering*, vol. 21, no. 3, pp. 311–331, 2016.
- [54] E. Guzman, R. Alkadhi, and N. Seyff, “A needle in a haystack: What do twitter users say about software?,” in *2016 IEEE 24th International Requirements Engineering Conference (RE)*, pp. 96–105, IEEE, 2016.
- [55] E. Guzman, D. Azócar, and Y. Li, “Sentiment analysis of commit comments in github: an empirical study,” in *Proceedings of the 11th Working Conference on Mining Software Repositories*, pp. 352–355, 2014.

- [56] D. Pletea, B. Vasilescu, and A. Serebrenik, “Security and emotion: sentiment analysis of security discussions on github,” in *Proceedings of the 11th working conference on mining software repositories*, pp. 348–351, 2014.
- [57] M. Mäntylä, B. Adams, G. Destefanis, D. Graziotin, and M. Ortu, “Mining valence, arousal, and dominance: possibilities for detecting burnout and productivity?,” in *Proceedings of the 13th International Conference on Mining Software Repositories*, pp. 247–258, 2016.
- [58] M. Ortu, B. Adams, G. Destefanis, P. Tourani, M. Marchesi, and R. Tonelli, “Are bullies more productive? empirical study of affectiveness vs. issue fixing time,” in *2015 IEEE/ACM 12th Working Conference on Mining Software Repositories*, pp. 303–313, IEEE, 2015.
- [59] F. Calefato, F. Lanubile, and N. Novielli, “How to ask for technical help? evidence-based guidelines for writing questions on stack overflow,” *Information and Software Technology*, vol. 94, pp. 186–207, 2018.
- [60] M. Thelwall, K. Buckley, and G. Paltoglou, “Sentiment strength detection for the social web,” *Journal of the American Society for Information Science and Technology*, vol. 63, no. 1, pp. 163–173, 2012.
- [61] R. Jongeling, P. Sarkar, S. Datta, and A. Serebrenik, “On negative results when using sentiment analysis tools for software engineering research,” *Empirical Software Engineering*, vol. 22, no. 5, pp. 2543–2584, 2017.
- [62] B. Lin, F. Zampetti, G. Bavota, M. Di Penta, M. Lanza, and R. Oliveto, “Sentiment analysis for software engineering: How far can we go?,” in *Proceedings of the 40th International Conference on Software Engineering*, pp. 94–104, 2018.

Appendix A

Example

Topic	Example(s)
Impact on phone components	I didn't open the app for a while and I have noticed that the database got updated in the moment I opened, this means is not correctly working in the background. A suggestion to prevent the app from going to sleep could be to implement "ongoing notifications", this is done by Google for the weather or Fitbit for MobileTrack. Another feature I've noticed from such apps that need to always run in the background is to disable battery optimisation. Good luck.
Bluetooth interference	Since installing this app, every other Bluetooth use for my phone is no longer working. I can't listen to music on Bluetooth speaker without it constantly cutting out and I can't use hands free in my car.
Location issues	Privacy policy says "No location data (data that could be used to track your movements) will be collected at any time." Then the app asks for permission to use location data. Not sure why it needs this permission if it does not use the data?
Location estimation	Using the first half of your postcode to work out your risk level is rubbish. My postcode area is mostly in Cumbria and Lancashire, with only a little bit of Yorkshire (where I live). It says 'my area or a neighbouring area' is in lockdown. I am not in lockdown. The information provided is not sufficient to be useful - it needs to say 'you are in lockdown' or 'you are not in lockdown'. Saying 'you might possibly be in lockdown' is a total waste of time...
Language issue	Why is the app description in English if the content isn't?
Continued on next page	

Table A.1 – continued from previous page

Topic	Example(s)
Battery usage	App was fine until yesterday. Then my phone started to heat up and the battery drain was extremely rapid. I usually get 2 days use from a full charge. Yesterday the battery drained twice from a full charge in only 12 hours. This problem makes the app useless. I've uninstalled the app and cleared the cache but the drain continues. I may have to do a factory reset to recover my phone's usability
Data load issue	It takes too long time to update, some time not working properly
Response error	After entering my details the submit button is not taking to the next activity. And when i open it again it is asking me to enter details again..
QR code	Useless. Installed ok but every place I visit it cant scan and says QR code not in correct format...or place doesn't have QR codes yet ...like Mitre 10!
Controlled access	Pointless - it's the official launch date, but still can't access it as it's only for the IoW and people invited to trial it. Apparently loads of people signed up today - I've no idea how! Bloody useless.
Login error	Tried signing up and got an email for verification but nowhere to put the code. Went one step back and tried it says user already exists. Tried logging in says error.
Verification code issue	The code emailed to me to validate my account doesn't work when i enter it. I've repeatedly asked for a re-send code and it never works until I'm locked out. I want to do the right thing but the app doesn't make it easy
Privacy concern	There is no way this app doesn't collect your location. They know the IP address, device ID and nearby devices. I will not use this app due to privacy concerns. This app will not help protect my life more than a drops of hand sanitizer.
Unsatisfactory results	Doesn't show correct position of covid-19 patients, a family tested positive within 500m of my range and it shows no covid-19 patients within 5kms of my range. The no. of covid-19 cases shown near your area isn't also accurate.
Contact trace functionality issue	Wasnt really helpful. I got notified that I've been exposed in the last 14 days but I dont know if thats someone in a car beside me, when I was in contact with them, basically any information to tell me if I'm actually at risk or not. It would be more helpful if it said the day you were in contact with the person and the time so you can figure out if you're actually at risk
App crash	My EHTEAZ app keeps crashing everytime i open it, it wont let me even open the first page, it instantly closes it, please help as i am not able to enter any shops/malls without the app
Continued on next page	

Table A.1 – continued from previous page

Topic	Example(s)
Phone number format handling	Refuses my phone number.. Invalid phone number error message keeps appearing. Initially I was asked to try again later.. now with or without the zero I get invalid phone number
Results issue	The app is a great idea but this is South Africa, the majority of people who tested positive will not register themselves. This app should have been connected with the databases of all labs who have the contact details of people who tested positive. Obviously, their information will be kept confidential, people who downloaded the app will have the advantage of more accurate warnings. This app is currently a false security.
Satisfactory UI	Easy to use and well thought out judging by the user guide. It will be great if enough people use it to make the proximity feature worthwhile.
UI suggestions	Please add covid positive patient location on map view in application Also show my range area circle in map view thank you
Input suggestions	In profession column you didn't add bank employee who are highly exposed during the lockdown. Kindly add bank employee in the profession column.
UI issues	I've just downloaded the app. It is stuck on the first page where personal details are asked. It was asking about travel details but the input option is missing. So I cannot proceed further
Unsatisfactory usage	Worst App I Used Ever In My Life. Developers, Please Work On It. MOI Made This Mandatory Everywhere, And Its Not Working Properly.
Uncertain feasibility	I use it I wish more people would use it it's safe it doesn't collect your information and if 60% of Canadians actually use this app it would make a huge difference right now I think it's about 3%, or 10%, I don't know it's not nearly enough but come on people we're better than this
Customer support issue	It worked smoothly since last week. Now always asking to verify mobile and later mobile number blocked after reach out limit and asking to call 109. But they says wait for 2 days or use another number. How may time we can change number. Is it not possible to clear this issue from customer care center
Mixed polarity in UX	This is a third class aap not a suitable for a war of covide19, Well done work on this app
UX evolution	First time I used the app was not working, now that we are lvl3 again work much better.
Positive UX	Looks good. Glad to download and help keep the community safe.
Continued on next page	

Table A.1 – continued from previous page

Topic	Example(s)
Download and installation UX	I have tried to download the app for 3 nights and waited up to an hour for the download to start. I have given up trying and if most people experience the same trouble, then the government is not going to get the numbers required.
Version Update	I've had the app for almost three weeks, but the first page has been saying "13 of 14 days active" for the last few days. Reports in the press two days ago said this was a glitch and there was an update version 1.0.4 for Android that would fix it. I checked the RKI website but there is no such update.
Compatibility	Does not work unless you have a phone running at least Android 6 or IOS 13.5. So iPhone 5 won't work..

Table A.1: Example review for topics

Appendix B

Codes

B.1 Scraper code

```
1 from google_play_scraper import reviews
2 import pandas as pd
3 import pickle
4 import os
5
6 appids = [
7     'com.hamagen',
8     'com.gov.mcmc.projectcatur',
9     'sg.gov.tech.bluetrace',
10    'fr.gouv.android.stopcovid',
11    'covid.trace.morocco',
12    'au.gov.health.covidsafe',
13    'com.moi.covid19',
14    'it.ministerodellasalute.immuni',
15    'com.telkom.tracencare',
16    'co.gov.ins.guardianes',
17    'de.rki.coronawarnapp',
18    'nic.goi.aarogyasetu',
19    'com.covid19.cgig',
20    'hu.gov.virusradar',
21    'tn.onmne.e7mi',
22    'uk.nhs.covid19.production',
23    'net.hscni.covidtracker',
24    'is.landlaeknir.rakning',
```

```
25         'gov.vdh.exposurenotification',
26         'gov.georgia.novid20',
27         'my.gov.onegovappstore.mytrace',
28         'lv.spkc.gov.apturicovid',
29         'com.netcompany.smittestop_exposure_notification',
30         'bg.government.virusafe',
31         'no.simula.smittestopp',
32         'pl.gov.mc.protegosafe',
33         'com.covidtracker.hse',
34         'at.rotekreuz.stopcorona',
35         'nz.govt.health.covidtracer',
36         'ca.gc.hcsc.canada.stopcovid',
37         'bh.bahrain.corona.tracker',
38         'jo.gov.moh.aman',
39         'com.shohoz.tracer'
40 ]
41
42
43 start = 0
44 if not os.path.isfile('last.txt'):
45     currfile = open('last.txt', 'w')
46 with open('last.txt', 'r') as f:
47     startFrom = f.readlines()
48
49 print(startFrom)
50
51 if len(startFrom)>0:
52     start= appids.index(startFrom[0])
53
54
55 appids = appids[start:]
56 count = 200
57 continuation_token = None
58
59 if os.path.isfile('continuation_token'):
60     with open('continuation_token', 'rb') as tokenfile:
61         continuation_token = pickle.load(tokenfile)
62
63 for appid in appids:
```

```

64         print(appid)
65         dfIdx = 0
66         if not os.path.isdir(appid):
67             os.mkdir(appid)
68         else:
69             dfIdx = len(os.listdir(appid))
70         length = count
71         while length==count:
72             df = pd.DataFrame(columns=["reviewId", "userName", "conte
73             result, continuation_token = reviews(
74                 appid,
75                 count=count,
76                 continuation_token = continuation_token
77             )
78
79             for i, res in enumerate(result):
80                 df.loc[i] = [res["reviewId"], res["userName"], re
81
82             length=len(result)
83             dfIdx += 1
84             print('scraped_', length)
85             df.to_excel(appid+'/' +str(dfIdx)+'.xlsx', index=False)
86             with open('continuation_token', 'wb') as tokenfile:
87                 pickle.dump(continuation_token, tokenfile)

```

B.2 Preprocessing Code

```

1 import re
2 from time import sleep
3 import demoji
4 import os
5 import json
6 import argparse
7 import pandas as pd
8
9 from nltk import WordPunctTokenizer
10 punct_tokenizer = WordPunctTokenizer()
11
12 from langdetect import detect

```

```
13 from googletrans import Translator
14 translator = Translator()
15
16 appIds = ['com.covid19.cgig',
17           'tn.onmne.e7mi',
18           'bg.government.virusafe',
19           'com.moi.covid19',
20           'nz.govt.health.covidtracer',
21           'co.gov.ins.guardianes',
22           'de.rki.coronawarnapp',
23           'is.landlaeknir.rakning',
24           'com.netcompany.smittestop_exposure_notification',
25           'uk.nhs.covid19.production',
26           'hu.gov.virusradar',
27           'pl.gov.mc.protegosafe',
28           'lv.spkc.gov.apturicovid',
29           'au.gov.health.covidsafe',
30           'it.ministerodellasalute.immuni',
31           'covid.trace.morocco',
32           'at.rokeskreuz.stopcorona',
33           'com.covidtracker.hse',
34           'gov.vdh.exposurenotification',
35           'no.simula.smittestopp',
36           'sg.gov.tech.bluetrace',
37           'com.telkom.tracencare',
38           'gov.georgia.novid20',
39           'bh.bahrain.corona.tracker',
40           'fr.gouv.android.stopcovid',
41           #'nic.goi.aarogyasetu',
42           'jo.gov.moh.aman',
43           'ca.gc.hcsc.canada.stopcovid',
44           'com.hamagen',
45           'com.shohoz.tracer',
46           'my.gov.onegovappstore.mytrace',
47           'com.gov.mcmc.projectcatur',
48           'com.moc.gh',
49           'za.gov.health.covidconnect',
50           'br.gov.datasus.guardioes',
51           'ar.gob.coronavirus',
```

```

52         'uy.gub.salud.planacovid19uy'
53     ]
54
55 stopwords = ['me', 'my', 'myself', 'we', 'our', 'ours', 'ourselves', 'you',
56             'he', 'him', 'his', 'himself', 'she', 'her', 'hers', 'herself',
57             "'s", 'it', "it's", 'its', 'itself', 'they', 'them', 'their',
58             'what', 'which', 'who', 'whom', 'this', 'that', 'these', 'tho
59             'am', 'is', 'are', 'was', 'were', 'be', 'been', 'being', 'hav
60             'do', 'does', 'did', 'doing', 'an', 'the', 'and', 'but', 'if'
61             'as', 'until', 'while', 'of', 'at', 'by', 'for', 'with', 'abo
62             'into', 'through', 'during', 'before', 'after', 'above', 'bel
63             'in', 'out', 'on', 'off', 'over', 'under', 'again', 'further'
64             'when', 'where', 'why', 'how', 'all', 'any', 'both', 'each',
65             'some', 'such', 'no', 'nor', 'not', 'only', 'own', 'same', 's
66             'can', 'will', 'just', 'don', 'should', 'now', 'll', 're', 'v
67             'wasn', "wasn't", 'weren', "weren't", 'won', "won't", 'would'
68             'omg', 'idk', 'app', 'please', 'fix', 'thank', 'thanks']
69
70 url_regex = re.compile('http[s]?://(?:[a-zA-Z]|[0-9]|[$-_@.&+]|[*\(\)\,]|
71 rep_regex = re.compile(r"([A-Za-z])\1{2,}", re.DOTALL)
72 phone_regex1 = re.compile('((samsung)|(iphone)|(ios)|(oneplus)|(huawei)|
73 phone_regex2 = re.compile('(samsung)|(iphone)|(ios)|(oneplus)|(huawei)|(r
74 phone_regex3 = re.compile('(phone(_)*)+')
75
76 def replaceRepeat(s):
77     return rep_regex.sub(r"\1\1", s)
78
79 def remove_url(s):
80     return url_regex.sub("_", s)
81
82 def modify_nums(s, numInfo):
83     s1 = str(s)
84     for num in numInfo:
85         s1 = s1.replace(num[0], num[1])
86     return s1
87
88 def remove_app_stopwords(s, stops):
89     s1 = str(s)
90     for stop in stops:

```

```

91         s1 = s1.replace("_"+stop+"_", "_")
92         if s1.startswith(stop+"_"):
93             s1 = s1.replace(stop+"_", "", 1)
94     return s1
95
96 def remove_device_names(s):
97     s = phone_regex1.sub("phone_", s)
98     s = phone_regex2.sub("phone_", s)
99     return phone_regex3.sub("phone_", s)
100
101 def tokenize(text):
102     #toks = word_tokenize(text)
103     toks = text.split()
104     tokens = []
105     for tok in toks:
106         if tok in wm_keys:
107             tokens += wordmap[tok].split()
108         else:
109             tokens += punct_tokenizer.tokenize(tok)
110
111     tokens = [tok for tok in tokens if tok.isalpha() or '_' in tok]
112     # lower case, remove number, punctuation
113     return tokens
114
115 def remove_stopwords(tokens):
116     stopped_tokens = [tok for tok in tokens if not tok in stopwords]
117     return stopped_tokens
118
119 if __name__ == '__main__':
120     parser = argparse.ArgumentParser(description='Preprocess')
121     parser.add_argument('--data', type=str,
122                         help='Data_Directory', default="../Data")
123     parser.add_argument('--out', type=str,
124                         help='Output_Directory', default="Preprocessed")
125     parser.add_argument('--stat', type=str,
126                         help='Completion_status_file', default="complete.")
127     parser.add_argument('--appInfo', type=str,
128                         help='App_info_stopwords_and_numbers_file', default="")

```

```
129             help='Word_map_file', default="wordmap.json")
130
131     args = parser.parse_args()
132
133     datadir = args.data
134     outdir = args.out
135     if not os.path.isdir(outdir):
136         os.mkdir(outdir)
137
138     stat = args.stat
139     lastApp = appIds[0]
140     start = 0
141     if os.path.isfile(stat):
142         with open(stat, 'r') as f:
143             done = f.read().split()
144             lastApp = done[0]
145             start = int(done[1])
146
147     with open(args.appInfo, 'r') as f:
148         appInfo = json.load(f)
149
150     global wordmap
151     global wm_keys
152     wordmap = {}
153     with open(args.wordmap, 'r') as f:
154         wordmap = json.load(f)
155
156     wm_keys = list(wordmap.keys())
157
158     appIds = appIds[appIds.index(lastApp):]
159     print(len(appIds), 'Left\n-----')
160     print(appIds)
161     print('-----')
162
163     #append one character word to stopwords
164     for ch in range(26):
165         stopwords.append(chr(ord('a')+ch))
166
167     for itr, appId in enumerate(appIds):
```

```
168         try:
169             print('-----\n', itr, appId, '\n')
170             file = appId + '.xlsx'
171             dest = os.path.join(outdir, file)
172             df = pd.read_excel(os.path.join(datadir, file))
173             nrows = df.shape[0]
174             print('Total:', nrows)
175             print('Start:', start)
176             if start==0:
177                 #new
178                 dfout = df.copy()
179                 #initialize preprocessed to empty
180                 dfout["preprocessedLDA"] = ""
181                 dfout["translation"] = ""
182                 #flag: if source language english: initially true
183                 dfout["isEnglish"] = 1
184             else:
185                 #resume
186                 dfout = pd.read_excel(dest)
187
188             info = appInfo[appId]
189             for i in range(start, nrows):
190                 comment = df.loc[i]["content"]
191                 if not isinstance(comment, str):
192                     continue
193
194                 #remove emoji
195                 comment = demoji.replace(comment, "_")
196                 #remove url
197                 comment = remove_url(comment)
198
199             while True:
200                 try:
201                     #google translator
202                     tr = translator.translate(comment)
203                     translated = tr.text
204                 except Exception as e:
205                     #network error
206                     print(e)
```

```

207             sleep(10)
208         else:
209             break
210
211         if tr.src!='en':
212             dfout.loc[i,"isEnglish"] = 0
213             #translation unsuccessful
214             #if comment.lower()==translated.lower():
215             #    continue
216
217             comment = translated.lower()
218             dfout.loc[i,"translation"] = comment
219             #replace letters occuring more than twice continuously to
220             comment = replaceRepeat(comment)
221             #check for kept numbers
222             comment = modify_nums(comment, info['numbers'])
223             #remove app related stopwords
224             comment = remove_app_stopwords(comment, info['stopwords'])
225             #replace device model with phone
226             comment = remove_device_names(comment)
227             #tokenize
228             comment = tokenize(comment)
229             #stopwords
230             comment = remove_stopwords(comment)
231             dfout.loc[i,"preprocessedLDA"] = ' '.join(comment)
232
233         dfout.to_excel(dest, index=False)
234         start = 0
235
236     except KeyboardInterrupt:
237         print(appId, i)
238         with open(stat, 'w') as f:
239             f.write(appId+'_'+str(i))
240         dfout.to_excel(dest, index=False)
241         break

```

B.3 LDA training code

```

1 from nltk.stem.snowball import SnowballStemmer

```

```
2 from gensim import corpora, models
3 from gensim.models.ldamulticore import LdaMulticore
4 #from gensim.models import LdaModel as LMSingle
5 from gensim.models.coherencemodel import CoherenceModel
6 import argparse
7 import pandas as pd
8 import random
9 import gensim
10 import json
11 import os
12 from statistics import mean
13 from pickle import dump
14
15 stemmer = SnowballStemmer("english")
16
17 def filter_pos(toks, pos1, pos2, select_pos):
18     if len(select_pos)==0:
19         return toks
20     tokens = []
21     for i in range(len(toks)):
22         if (pos1[i] in select_pos) or (pos2[i] in select_pos):
23             tokens.append(toks[i])
24     return tokens
25
26 def stem_tokens(tokens):
27     stemmed = []
28     for item in tokens:
29         stem_ = stemmer.stem(item)
30         stemmed.append(stem_)
31     return stemmed
32
33 def get_random_number():
34     return random.randint(0,50000)
35
36 class LDADocument:
37     def __init__(self, app, id, body):
38         self.app = app
39         self.id = id
40         self.body = body
```

```
41
42 class LDA_Model:
43     def __init__(self, training_data=None, num_topics=10, pos='ALL', data
44                 translated = False, use_multicore=True, coherence=0.6, c
45         self.num_topics = num_topics
46         self.pos = pos
47         self.translated = translated
48         self.datadir = datadir
49         self.use_multicore=use_multicore
50         self.target_coherence=coherence
51         self.workers=core
52         self.iterations=iterations
53         self.results = results
54
55
56         if (training_data is None):
57             self.training_data = self.read_data_from_oracle()
58         else:
59             self.training_data = training_data
60
61         self.models = self.create_model_from_training_data()
62
63         scores = []
64         outIdx = results.shape[0]
65         models_best =[]
66         for i in range(len(num_topics)):
67             maxScore = 0
68             nt = num_topics[i]
69             print('With_num_topics:', nt)
70             coherence=self.compute_coherence(i)
71             score = [coherence]
72             models_best.append(self.models[i])
73             for j in range(10):
74                 print('SEED:', self.seed)
75                 self.models[i]=self.prepare_model(nt)
76                 coherence=self.compute_coherence(i)
77                 score.append(coherence)
78                 print(j, '-----', coherence)
79                 if coherence>maxScore:
```

```

80             maxScore = coherence
81             models_best[i] = self.models[i]
82         meanScore = mean(score)
83         print('Coherence_Score_Max:', maxScore, '|Mean:', meanScore)
84         self.results.loc[outIdx] = [self.datadir[self.datadir.find('/')
85         outIdx+=1
86         print(self.results)
87     self.models = models_best
88
89
90     def get_model(self):
91         return self.model
92
93     def create_model_from_training_data(self):
94         document_ids = []
95         token_collection = []
96         for document in self.training_data:
97             document_ids.append(document.id)
98             token_collection.append(stem_tokens(document.body))
99
100         self.document_ids = document_ids
101         self.token_collection = token_collection
102
103         self.dictionary = corpora.Dictionary(self.token_collection)
104         #self.dictionary.filter_extremes(no_below=20, no_above=0.8)
105         self.corpus = [self.dictionary.doc2bow(text) for text in self.tok
106
107         self.models = []
108         for nt in self.num_topics:
109             self.models.append(self.prepare_model(nt))
110         return self.models
111
112     def prepare_model(self, topics_count):
113         self.seed = get_random_number()
114         if (self.use_multicore):
115             ldamodel = LdaMulticore(self.corpus, num_topics=topics_count,
116                                     passes=50, workers=self.workers, alpha='s
117                                     eta='auto', iterations=self.iteration
118         return ldamodel

```

```

119         else:
120             ldamodel = LMSingle(corpus =self.corpus, num_topics=topics_co
121                                random_state=self.seed, passes=50, alpha=
122             return ldamodel
123
124     def compute_coherence(self, idx):
125         coherencemodel = CoherenceModel(model=self.models[idx], dictionary=
126                                         coherence='c_v')
127         value = coherencemodel.get_coherence()
128         return value
129
130     def read_data_from_oracle(self):
131         print('----', len(os.listdir(self.datadir)), 'apps-----')
132         model_data = []
133         for file in os.listdir(self.datadir):
134             print(file[:-5])
135             df = pd.read_excel(os.path.join(self.datadir, file))
136             nrows = df.shape[0]
137             for i in range(nrows):
138                 if (not self.translated) and df.loc[i]["isEnglish"]==0:
139                     continue
140                 text = df.loc[i]["preprocessedLDA"]
141                 if not isinstance(text, str) or len(text)==0:
142                     continue
143                 review = filter_pos(text.split(), df.loc[i]["POS1"].split
144                 if len(review)>0:
145                     doc = LDADocument(file[:-5], df.loc[i]['reviewId'], r
146                     model_data.append(doc)
147         print(len(model_data))
148         #fname = self.datadir[self.datadir.find('/')+1:]+'.corpus'
149         #with open(fname, 'wb') as outfile:
150         #    dump(model_data, outfile)
151         return model_data
152
153     def save_results(self, name, dest):
154         self.results.to_csv(name)
155         length = len(self.models)
156         start = self.results.shape[0]-length
157         for i in range(length):

```

```
158         fname = os.path.join(dest, str(start+i)+'.model')
159         self.models[i].save(fname)
160
161
162 if __name__ == '__main__':
163     parser = argparse.ArgumentParser(description='LDA_Model')
164
165     parser.add_argument('--data', type=str,
166                         help='Data_Directory', default="Data/Europe")
167
168     parser.add_argument('--out', type=str,
169                         help='Output_Directory', default="Out")
170
171     parser.add_argument('--result', type=str,
172                         help='Results_file', default="results.csv")
173
174     parser.add_argument('--config', type=str,
175                         help='Configuration_File', default="config.json")
176
177     parser.add_argument('--multicore', type=bool,
178                         help='Iteration_count', default=True)
179
180     parser.add_argument('--coherence', type=float,
181                         help='Target_coherence', default=0.6)
182
183     parser.add_argument('--core', type=int,
184                         help='CPU_Threads', default=8)
185
186     parser.add_argument('--iteration', type=int,
187                         help='Number_of_iterations', default=100)
188
189     args = parser.parse_args()
190     datadir = args.data
191     outdir = args.out
192     multi_core=args.multicore
193     tgt_coherence=args.coherence
194     num_core=args.core
195     iters=args.iteration
196
```

```
197     with open(args.config, 'r') as f:
198         config = json.load(f)
199
200     if not os.path.isdir(outdir):
201         os.mkdir(outdir)
202
203     resFile = args.result
204     if not os.path.isfile(resFile):
205         results = pd.DataFrame(columns=["Location", "Num_Topics", "POS",
206     else:
207         results = pd.read_csv(resFile, index_col=0)
208
209     topics = config["Topics"]
210     pos_select = config["POS"]
211     useTranslated = config["UseTranslated"]==1
212
213     LDA = LDA_Model(num_topics=topics, pos=pos_select, translated = useTr
214                     datadir = datadir, use_multicore=multi_core, resu
215                     coherence=tgt_coherence, core=num_core, iteration
216     LDA.save_results(resFile, outdir)
```

B.4 LDA topics visualization

```
1 from gensim.models.ldamulticore import LdaMulticore
2 import pyLDAvis.gensim as ldvis
3 import pyLDAvis
4 from gensim import corpora
5 import pickle
6 from operator import itemgetter
7 import os
8
9 class LDADocument:
10     def __init__(self, app, orig, trn, id, body, tokens):
11         self.app = app
12         self.id = id
13         self.body = body
14         self.tokens = tokens
15         self.orig = orig
16         self.trn = trn
```

```
17
18 class CustomUnpickler(pickle.Unpickler):
19
20     def find_class(self, module, name):
21         if name == 'LDADocument':
22             #from settings import LDADocument
23             return LDADocument
24         return super().find_class(module, name)
25
26 corpusFile = 'MiddleEast'
27 with open('ALL--CORPUS/corpus_AllPOS/'+corpusFile+'.corpus', 'rb') as f:
28     data = pickle.load(f)
29
30 #modelFile = 'ALL--MODELS/model_'+corpusFile+'/1.model'
31 modelFile = 'ALL--MODELS/Out/4.model'
32 model = LdaMulticore.load(modelFile)
33
34 numTopics = len(model.print_topics(-1))
35 print("Number_of_topics=", numTopics)
36
37 corpus = [model.id2word.doc2bow(text.body) for text in data]
38
39 '''
40 lda_display = ldvis.prepare(model, corpus, model.id2word)
41 pyLDAvis.save_html(lda_display, "LDAvis/1_MiddleEast.html")
42 pyLDAvis.display(lda_display)
43 '''
44 all_topics = model.get_document_topics(corpus, per_word_topics=True)
45 print(len(data), len(corpus), len(all_topics))
46 print('-----')
47 topic_prb = []
48 for i in range(numTopics):
49     topic_prb.append([])
50
51 for i in range(len(all_topics)):
52     doc_topics = all_topics[i][0]
53     doc_topics = sorted(doc_topics, key=itemgetter(1), reverse=True)
54     maxProb = doc_topics[0][1]
55     for dt in doc_topics:
```

```

56         currProb = dt[1]
57         if currProb!=maxProb:
58             break
59         topic_prb[dt[0]].append([i, dt[1]])
60
61
62 for i in range(numTopics):
63     topic_prb[i] = sorted(topic_prb[i],key=itemgetter(1),reverse=True)
64
65 for i in range(numTopics):
66     length = len(topic_prb[i])
67     print('-----\nTopic:', i, '\n-----')
68     print(model.print_topic(i))
69     print('-----Top_15_Probalities-----')
70     for j in range(min(15, length)):
71         print('With_p=', topic_prb[i][j][1])
72         print('-----')
73         idx = topic_prb[i][j][0]
74         print(data[idx].app, data[idx].id)
75         print('-----')
76         print(data[idx].orig)
77         print('=====\n', data[idx].body)
78
79     print('-----Medium_10_Probalities-----')
80     mididx = int(length/2)
81     for j in range(mididx-5,mididx+5):
82         print('With_p=', topic_prb[i][j][1])
83         print('-----')
84         idx = topic_prb[i][j][0]
85         print(data[idx].app)
86         print('=====\n', data[idx].body)
87
88     print('-----Least_10_Probalities-----')
89     for j in range(length-11,length):
90         print('With_p=', topic_prb[i][j][1])
91         print('-----')
92         idx = topic_prb[i][j][0]
93         print(data[idx].app)
94         print('=====\n', data[idx].body)

```

```
95     print('~~~~~')
```

Generated using Undergraduate Thesis L^AT_EX Template, Version 1.4. Department of
Computer Science and Engineering, Bangladesh University of Engineering and
Technology, Dhaka, Bangladesh.

This thesis was generated on Saturday 20th February, 2021 at 5:35pm.