FACULTY OF INFORMATION TECHNOLOGY AND ELECTRICAL ENGINEERING

**Fuzel Ahamed Shaik**

# UNDERSTANDING USER BEHAVIOR ASPECTS ON EMERGENCY MOBILE APPLICATIONS DURING EMERGENCY COMMUNICATIONS USING NLP AND TEXT MINING TECHNIQUES

Master's Thesis
Business Analytics
Computer Science and Engineering
October 2023

**Shaik F. (2023) Understanding User Behavior Aspects on Emergency Mobile Applications during Emergency Communications Using NLP and Text Mining Techniques.** University of Oulu, Degree Programme in Business Analytics, 65 p.

# ABSTRACT

**The use of mobile devices has been skyrocketing in our society. Users can access and share any type of information in a timely manner through these devices using different social media applications. This enabled users to increase their awareness of ongoing events such as election campaigns, sports updates, movie releases, disaster occurrences, and studies. The attractiveness, affordability, and two-way communication capabilities empowered these mobile devices that support various social media platforms to be central to emergency communication as well. This makes a mobile-based emergency application an attractive communication tool during emergencies. The emergence of mobile-based emergency communication has intrigued us to learn about the user behavior related to the usage of these applications. Our study was mainly conducted on emergency apps in Nordic countries such as Finland, Sweden, and Norway. To understand the user objects regarding the usage of emergency mobile applications we leveraged various Natural Language Processing and Text Mining techniques. VADER sentiment tool was used to predict and track users' review polarity of a particular application over time. Lately, to identify factors that affect users' sentiments, we employed topic modeling techniques such as the Latent Dirichlet Allocation (LDA) model. This model identifies various themes discussed in the user reviews and the result of each theme will be represented by the weighted sum of words in the corpus. Even though LDA succeeds in highlighting the user-related factors, it fails to identify the aspects of the user, and the topic definition from the LDA model is vague. Hence we leveraged Aspect Based Sentiment Analysis (ABSA) methods to extract the user aspects from the user reviews. To perform this task we consider fine-tuning DeBERTa (a variant of the BERT model). BERT is a Bidirectional Encoder Representation of transformer architecture which allows the model to learn the context in the text. Following this, we performed a sentence pair sentiment classification task using different variants of BERT. Later, we dwell on different sentiments to highlight the factors and the categories that impact user behavior most by leveraging the Empath categorization technique. Finally, we construct a word association by considering different Ontological vocabularies related to mobile applications and emergency response and management systems. The insights from the study can be used to identify the user aspect terms, predict the sentiment of the aspect term in the review provided, and find how the aspect term impacts the user perspective on the usage of mobile emergency applications.**

**Keywords: Emergency communication, Topic modeling, Social media, BERT, Aspect Based Sentiment Classification, Sentiment analysis, Mobile applications, Empath categorization, Ontology**

# TABLE OF CONTENTS

# FOREWORD

This thesis builds on the Natural Language Processing (NLP) research conducted by the Center for Machine Vision and Single Analysis (CMVS), Faculty of Information Technology and Electrical Engineering (ITEE).

I'm grateful that my supervisor, Dr. Mourad Oussalah gave me the chance to work with him. I became really interested in NLP while taking Dr. Mourad's classes on social network analysis and natural language processing.

My thesis supervisor, Dr. Mourad Oussalah, stayed by my side the entire time I was collecting the dataset for emergency mobile applications, extending it, and working to enhance emergency communication. His impromptu coaching inspired me to better my work every day. Despite being a very busy, dedicated researcher, he spent hours supervising me. I would also like to extend my gratitude to the DIGeMERGE project for funding my thesis.

I am grateful that Dr. Olli Silven, the director of CMVS, and Dr. Jouko Niinimäki, the rector of the University, allowed me to pursue my aspirations by allowing me to study at the University of Oulu in Finland. I also want to thank Dr. Satu Tamminen, my instructor and tutor, for always being there for me when I needed her. I also want to express my thanks to Md. Saroar Jahan, one of my colleagues, was continuous support to me when I was working on my thesis.

Finally, I would want to convey my sincere gratitude to my parents, siblings, friends, peers, and all the researchers who have made contributions to this field for their spiritual support and blessings along this voyage.

Oulu, October 3rd, 2023

Fuzel Ahamed Shaik

# LIST OF ABBREVIATIONS AND SYMBOLS

| | |
|---|---|
| EC | Emergency Communications |
| ICT | Information Communication Technology |
| EENA | The European Emergency Number Association |
| EU | European Union |
| GPS | Global Positioning System |
| NLTK | Natural Language Toolkit |
| VADER | Valence Aware Dictionary for Sentiment Reasoning |
| ANEW | Affective Norms for English Words |
| LIWC | Linguistic Inquiry and Word Count |
| TF-IDF | Token frequency and Inverse document frequency |
| CSV | Comma Separated Value |
| LDA | Latent Dirichlet Allocation |
| EDA | Exploratory Data Analysis |
| NLP | Natural Language Processing |
| SMS | Short Message Service |
| JST | Joint Sentiment Topic |
| DS | Dimensional Score |
| TDS | Topic Document Sentence |
| ABSA | Aspect Based Sentiment Analysis |
| ATE | Aspect Term Extraction |
| RNN | Recurrent Neural Network |
| CRF | Conditional Random Fields |
| SVM | Support Vector Machines |
| KNN | K Nearest Neighbours |
| CNN | Convolutional Neural Networks |
| BERT | Bi-directional Encoder Representation Transformer |
| MHSA | Multi-Head Self-Attention |
| LCF | Local Context Focus |
| GCF | Global Context Focus |
| DO4MG | Domain Ontology for Mass Gatherings |
| MOAC | Management Of a Crisis Vocabulary |
| CROnto | Crisis Response Ontology |
| OWL | Web Ontology Language |
| TF | Term Frequency |
| DF | Document Frequency |
| IDF | Inverse Document Frequency |
| DTM | Document Term Matrix |
| TS | Topic Score |
| LSTM | Long Short-Term Memory Networks |
| MLM | Masked Language Models |
| ATEPC | Aspect Term Extraction Polarity Classification |
| BOI | Beginning, Outside, and Inside of the aspect |
| APC | Aspect Polarity Classification |
| ReLU | Rectified Linear Unit |

| | |
|---|---|
| PASDF | Positive Aspect Sentiment Data Frame |
| NASDF | Negative Aspect Sentiment Data Frame |
| MSS | Mean Sentiment Score |
| SR | Sentiment Rate |
| TAM | Technology Acceptance Model |
| UTAUT | Unified Theory of Acceptance and Use of Technology |
| AI | Artificial Intelligence |

# 1. INTRODUCTION

Communication is an important factor especially when we are dealing with an emergency. The effects of communication can have a huge impact on providing better emergency services. An emergency is a broad spectrum that includes various incidents such as terrorist attacks, climatic disasters, medical emergencies, accidents, wars, internal conflicts in a location, fire breakdowns, and many more. The common public is generally the victim of an emergency. Hence, every individual has to be aware of the necessary tools through which one can easily communicate with emergency service providers in order to provide quick help in such conditions. Over time, the medium of communication from the individual to the emergency service provider and vice versa has been varying according to the technological growth in society. The individuals who are considered as users have been adopting these technical advancements to get better help or to get information regarding the events around them. The main reason for these technical advancements in emergency communications is due to Information Commuinication Technology (ICT). ICT has changed the way things work in various domains. One such domain is Emergency communication.

In this thesis, we discuss a pipeline to study the user aspects of user reviews for various emergency mobile applications across Nordic countries and construct a vocabulary-based Ontology. The outcome of this thesis aims to act as a guide while assessing the emergency mobile applications based on user aspect sentiments by highlighting the aspects that affect positively and negatively which results in the creation of user-friendly emergency mobile applications in Nordic countries.

## 1.1. Background and Motivation

Humans are leveraging ICT-based solutions to solve various problems across different domains such as business transactions, industrial operations, security and privacy, aid assistance, and all the major aspects of our daily lives [1]. The rapid growth in the use of computers, the Internet, mobile applications, and other technologies is the result of development in ICT [2]. From another perspective, ICT can be seen as an information processing and sharing platform. ICT is itself a broad area that involves many aspects of technology. Social media and Web 2.0 are two major aspects of ICT that have been creating a huge impact on our society. Users are leveraging these technologies to gather information on various activities such as entertainment, recent market trends, climate forecasts, disaster awareness, precautions to take during the pandemic, and relevant information on many other things. These technological advancements are also helping in emergency communications by broadcasting information to a mass audience through social media means. The primary factors in emergency communication are information sharing and location gathering, these factors can help to provide an effective response system [3]. The latest technologies such as social media, the internet, and Web 2.0, among others, fill the gap between an individual and a given emergency service provider (a legally established institution [4] or the country's government) by creating a bidirectional flow of information and creating a public warning notification system to indicate the emergency.

Mobile devices have become an integral part of human life. On the other hand, mobile devices leverage ICT tools to contribute to information sharing. Mobile applications are one such tool that helps in information sharing. These are considered as the major sources of medium to communication during emergencies by the European Emergency Number Association (EENA). By considering the growth of mobile applications, EENA concluded that mobile-based emergency applications are in need [5]. These applications work similarly to SOS calls (112 calls in the EU). Apart from just calling services, these applications are designed to give information on relevant events such as weather conditions, disaster alerts, traffic notifications, location details through GPS, and many other services that help an individual to be prepared when an emergency occurs. The need for mobile-based emergency applications has been considered due to previous incidents in the Nordic countries. For instance, The Bomb Explosion in Myyrmanni, Finland 2002, where a 19-year student exploded a bomb inside a shopping mall [6]. Ortenwall et al., [6] highlighted the response time taken from the various units such as ambulance, rescue helicopter, pre-hospital medical team, police, and crisis management team. Mobile devices are one of the prominent ways of communication during emergencies. Apart from voice calls, mobile devices are well known for providing accurate location, relevant information on time regarding the event and the user, prediction of hazards and disasters, etc., in an efficient means which eventually improves the quality of help provided by emergency services [5]. The EENA stressed the need for a mobile application that is regulated and made accessible all over the European Union [4]. As of now, EENA accepts several mobile emergency applications in different countries. Some of the incidents are mentioned below:

1. **New Year's Storm, 1992**: The "New Year's Storm" of 1992 in Norway brought heavy snowfall and strong winds, causing widespread power outages and transportation disruptions. The widespread power outages due to heavy snow and strong winds made communication more challenging. Apart from that the transport network has been disrupted due to blocked roads. The government has prioritized the restoration of essential services such as electricity and communication networks. They also sent regular updates via radio broadcasts and emergency text messages to inform residents about the road conditions, shelter locations, and safety measures.

2. **2014 Floods in Sweden**: The flood of 2014 in Sweden, particularly in areas around Lake Mälaren, led to property damage and evacuation of residents. Ensuring effective communication with residents in flood-prone areas, disseminating evacuation orders, and providing information about emergency shelters and assistance resources became a major concern. The state utilized social media, emergency apps, and traditional media channels to convey updates and safety information to the public.

3. **Näsijärvi fire, 2018**: The Näsijärvi fire in 2018 in Finland resulted in significant forest areas being engulfed in flames, threatening nearby communities. The fires spread rapidly in remote forested areas. The major difficulty is to ensure accurate and timely information dissemination to affected communities. To encounter these challenges the emergency service providers communicated

evacuation orders through emergency alerts, radio broadcasts, and door-to-door notifications.

4. **Nycomed Plant Explosion, 2004**: The explosion at the Nycomed chemical plant in Denmark in 2004 resulted in casualties and environmental hazards. The challenge faced during this incident is to ensure swift and accurate communication between emergency responders, authorities, and neighboring communities about the nature of the incident, potential health risks, and necessary precautions. The state has leveraged sirens, loudspeakers, and emergency alerts to inform residents about evacuation procedures and safe zones.

The United Nations Office on Disaster Risk Reduction has carried out various studies to understand the effect of disasters in terms of revenue. In the case study Understanding Disaster Risk, the importance of early warning and early action systems has been highlighted. In one study carried out in 2009, in terms of preventing future harm, every dollar spent on readiness is worth $15. This pointed out that up to US $66 billion in gains from losses averted might be realized. Early warning can also help with overall climate service investments with one-tenth of the cost-benefit ratio. For example, Timely cyclone alerts in Bangladesh have drastically lowered the number of fatalities attributable to cyclones. Less than 20 people perished during Cyclone Matmo-Bulbul in 2019, while 2.1 million people were successfully moved to evacuation centers.

All these incidents and the responsive measures have intrigued us to understand user behavior regarding mobile-based emergency applications. In order to examine user evaluations posted in Google's Play Store and the App Store for this thesis, we took into account 5 distinct applications. The applications along with their descriptions are mentioned in Figure 1 and Table 1 respectively.


Figure 1. Emergency mobile applications are considered for our study. (a) 112 Suomi (b) Emergency plus (c) SOS alarm (d) Krisinformation.se (e) Hjelp 113

## 1.2. Research Scope and Contribution

EENA highlights the usage of mobile devices as a means of communication during emergencies, it seems 70% of emergency calls to emergency number 112 (in Europe) are from mobile devices [7]. This makes mobile devices one of the prominent ways of communication during emergencies. Apart from voice calls, mobile devices are well known for providing accurate location, relevant information on time regarding the event and the user, prediction of hazards and disasters, etc., in an efficient means which eventually improves the quality of help provided by emergency services [5]. The EENA stressed the need for a mobile application that is regulated and made

Table 1. Short description of selected applications

| Apps | Description | Play Store ratings | App Store ratings | Total reviews | Maxi-mum tokens | Mini-mum tokens |
|---|---|---|---|---|---|---|
| 112 Suomi | The software is intended to assist customers in swiftly locating the best service in crises and other difficult situations. | 3.4 | 4.0 | 1355 | 201 | 2 |
| Emergency Plus | Allows us to accurately communicate our location. | 3.9 | 4.0 | 972 | 442 | 1 |
| Krisinformation.se | Alert messages to the public regarding weather warnings and traffic. | 3.6 | 2.3 | 228 | 697 | 2 |
| SOS alarm | Emergency application based in Sweden. | 4.2 | 4.6 | 467 | 132 | 1 |
| Hjelp 113 | An overview of all the emergency numbers in Norway in one place so you can easily call the right emergency service. | 4.1 | 3.9 | 147 | 128 | 2 |

accessible all over the European Union [4]. As of now, EENA accepts several mobile emergency applications in different countries. In this study, we mainly focus on Nordic countries such as Finland, Sweden, and Norway.

The main goal of the thesis study is to identify the user aspects of emergency mobile applications and leverage the insights to improve emergency communication through mobile applications by providing a defined pipeline to analyze the user reviews scraped from the Google Play Store and the App Store. To achieve our aim, the study is divided into different analyses:

1. Define the polarities of the user reviews and identify version-level sentiment for each considered application. This helps us to have an idea of how the user sentiments change over each version of the application.

2. To analyze which factors affect user behavior over time. We adopted topic modeling techniques using the Latent Dirichlet Allocation (LDA) [8] model to identify the different themes or topics discussed among the users.

3. Later pre-process the user reviews scraped from two different mobile application stores to perform aspect term extraction tasks using state-of-art architectures. The output of this task is used as the input data for the below two analyses.

4. The reviews, aspect terms, and polarity of the aspect terms in the review are considered to perform a sentence pair classification task. The trained model predicts the polarity of the aspect in the review. Later analyze the reasons behind the user sentiments.

5. A vocabulary for emergency mobile application is constructed and the relation between the words is highlighted using a word association tree.

Each of the goals will be clearly discussed and the results from the analysis will be highlighted in the upcoming chapters 4 and 5.

## 1.3. Outline of the Thesis

This thesis is broken up into seven different chapters, the first of which discusses the history and inspiration of this piece and establishes the thesis's specifications. We give a literature study and examine the suggested techniques with previous work in the second chapter. We go into further detail about the data collection process and explore exploratory data analysis (EDA) in the third chapter. In the fourth chapter, we aimed to highlight the methodology followed to achieve the goal of the study. This section also contains the metrics we used to analyze the score and the models we used to perform predictive analysis. In the later chapter, which is chapter five, we display the results obtained from the methods we applied and also point out the insights. Then in the end chapters we discuss the results obtained from the study, highlight some limitations, suggest the future scope of this project, and conclude the study with an overall summary.

# 2. LITERATURE REVIEW

Over the past years, bidirectional communication using mobile emergency applications has become popular in many European countries [5]. Nordic countries are no exception in this. EENA highlights three significant means of access to emergency services during emergencies they are voice calls, emergency SMS, and mobile emergency applications [5]. Repanovici and Nedelcu, [9] studied the current state, barriers, and future potential of mobile emergency notification apps, Their study statistically shows that mobile applications are the ideal way of communication during an emergency. The authors use the Multi-Criteria Analysis [10] method a decision-making process that calculates the performance index to rank the mode of communication during an emergency. Their study concludes that mobile emergency applications are the preferred means of access over voice calls and emergency SMS. Tan et al., [11] provided a comprehensive view of using mobile emergency applications by analyzing 49 crisis informatics articles that focus on mobile apps during emergencies [11]. Their study highlights the thematic purposes of creating an emergency purpose app, later classified the apps under different categories depending on the interaction purpose of the app and divided the individual into various sectors depending on their need of usage of the emergency application. It also states that usage of social media and mobile apps already become a habit. According to Tan et al., [11] presentation and visualization are the two significant aspects that are to be presented in a mobile emergency communication app.

## 2.1. Sentiment Analysis

Sentiment analysis is part of Natural Language Processing (NLP) and Text mining methods that deal with the sentiment of the text. Over the years this method has been used to understand the user sentiment by their reviews. A few of the examples are, understanding user sentiment toward restaurants, analyzing the user opinion regarding a product in an e-commerce store, and defining the sentiment of sentences mentioned by a user in a meeting. Lexicon-based approach and machine learning approach are two different approaches to performing a sentiment analysis task [12]. The machine learning methodology is thought of as supervised learning since it incorporates training data, whereas the lexicon-based approach is thought of as unsupervised or semi-supervised learning.

Lexicon-Based Approach employs information about which words and phrases are favorable and which are negative from a sentiment lexicon [13]. There were many lexicon-based tools that could perform sentiment analysis. Natural Language Toolkit (NLTK) is an open-source Python library that performs many text-mining methods. The sentiment score of a sentence is calculated using the SentiWordNet method from the NLTK library. Apart from its better performance in defining polarity to the sentences, it fails to identify the correct sentence polarity for the text in microblogs such as Twitter, Facebook, Reddit, etc. To address this issue Hutto and Gilbert [14] developed a simple rule-based lexicon model to extract the sentiment of the text and they named it Valence Aware Dictionary for Sentiment Reasoning (VADER). VADER has performed better on microblogging or social media data. Its effectiveness has been compared with many existing models such as SentiWordNet, Affective Norms for English Words (ANEW), and Linguistic Inquiry and Word Count (LIWC) [15]. Hence for this study, we employed VADER to perform sentiment analysis.

## 2.2. Topic Modeling

Through sentiment analysis, one can only say the overall sentiment score of the text or a sentence. To identify the various themes discussed in the complete text researchers developed various topic modeling techniques. An approach known as "topic modeling" uses a number of algorithms to disclose, identify, and annotate the subject organization in a collection of texts [16]. LDA techniques have been used in many use cases to extract user opinions by defining various topics. Ramamonjisoa [17] leveraged LDA techniques to model different topics in user's comments. The author also integrated time series and hierarchy in the LDA model to extract specific terms that are unrelated to the new articles[17]. A customer reviews analysis using LDA to identify various topics in the dataset has been experimented with in Farkhod et al., [18] study. The authors developed a Topic Document Sentence (TDS) model that is based on Joint Sentiment Topic (JST) and LDA topic modeling techniques [18]. Nguyen et al., [19] employed a topic modeling methodology using LDA to define various topics discussed in the user reviews of a US website for smark speakers. Nguyen et al., [19] stress the importance of the $k$ value in LDA modeling, which defines the optimal number of topics to be selected. The authors also mentioned a new metric called Dimensional Score (DS) to measure brand performance which we later employed in our study. The score is used to understand the performance of various brands for different sets of topics. In this thesis, we leveraged LDA, a topic modeling technique to identify the topics discussed among the user reviews and highlight important topics using the Topic Score metric.

## 2.3. Aspect Term Extraction

Since the primary aim of a mobile emergency application is to provide efficient help on time to the users, the apps need to be updated according to the user's needs. The need for analysis of public behavior and motivation towards the use of mobile emergency applications has been highlighted by Tan et al., [11]. One way to understand public behavior and motivation is to analyze the user reviews of the app. Natural Language Processing (NLP) and text mining is a well-known way to handle text data and exact information from text data. There have been many NLP algorithms that aim to predict the sentiment of each user review, this task is termed as sentiment analysis. This task helps only to classify the review on a positive, negative, and neutral basis. To understand the factors that are mentioned in the review we need to utilize a high-level analysis such as Aspect Based Sentiment Analysis (ABSA). Aspect term extraction (ATE) is one of the ABSA tasks that aims to extract the aspect terms from the text. In the early literature, there have been many advancements in performing aspect extract tasks. Majorly, ATE tasks are unsupervised learning tasks. In NLP, Topic modeling is an unsupervised learning task that unfolds to identify hidden details from the text. When topic modeling is employed in sentiment analysis, the outcome is the identification of the entities and attributes that constitute the text's topics [20]. The Latent Dirichlet allocation (LDA) [8] model to extract discrete data from the text corpus increased the scope of topic modeling. LDA models are also been leveraged to perform aspect extraction tasks. Titov and McDonald [21] provided a model to recognize aspects from reviews, based on reviews. Brody and Elhadad [22] employed an intuitive LDA implementation to represent its features and sentiments. Every phrase is considered a completely distinct document, and the model's aimed output is the distribution of aspects for each sentence, which tends the model toward localized aspects rather than global subjects in the text. Hu and Liu [23] employed a frequent pattern-mining approach to extract aspect terms from the text by identifying the most common nouns in the sentence. To understand the context of each term and identify it as an aspect term is highly challenging. To fill this gap, deep learning techniques have been leveraged. Irsoy and Cardie [24] developed a deep Elman-type Recurrent Neural Network (RNN) to identify user opinions that outperformed CRF, semi-CRF, and shallow RNN [25]. Liu et al. [26] presented discriminative models built on Recurrent Neural Network (RNN) architecture and word embeddings. Such models can be successfully applied to fine-grained opinion mining tasks without a requirement for task-specific manual feature engineering [26].

ABSA tasks were initially introduced in SemEval2014 Task 4 by Kirange et al., [27]. Kirange et al., [27] aim to identify the aspect term and category, and the classification of sentiments for restaurant reviews. The authors used the Support Vector Machine (SVM) model to extract the information and compare the results with a traditional KNN classifier. According to Kirange et al., [27] ABSA has three subtasks namely Aspect Term Extraction, Aspect Term Polarity, and Aspect Category Detection. Liu [28] & Zhang and Liu [29] defined an aspect term or category as (a) a part or component of an entity (e.g., for an entity laptop, the battery can be one of the aspect terms) (b) an attribute of entity (e.g., for entity laptop, the price can also be an aspect term) (c) an attribute of a part or component of entity (e.g., for entity laptop, the battery life can also be an aspect term) [30]. SemEval2015 Task 12 discusses the methodology to identify the opinions expressed about specific entities and their aspects on three different datasets namely restaurant, laptop, and hotel reviews [31]. In contrast with the SemEval2014 task 4, SemEval2015 task 12 takes the whole review and extracts entity and aspect pairs and their polarities as output. SemEval2015 task 12 deals with two subtasks namely in-domain ABSA and out-domain ABSA. The former deals with the seen data domain whereas the latter tests the trained model on the different domain data. In SemEval2016 task 5, the study was further expanded into sentence-level ABSA and text-level ABSA. The English test data were annotated using a web-based annotation tool called BRAT by an expert linguist and undergraduate computer science students, later These annotations were verified and rewritten by another expert linguist for all the above-mentioned 3 SemEval tasks [9]. Poria et al., [32] proposed a new rule-based framework to identify various aspects of the reviews using common sense and sentence dependency trees to differentiate direct and indirect aspects [33]. A semantical aspect classification approach is proposed by Mukherjee et al., [34] to group the aspects into different aspect categories. Poria et al., [32] presented a 7-

layered deep CNN-based opinion-mining approach to extract aspect terms from the sentence by pinning each word as an aspect or non-aspect. Later He et al., [35] designed a word embedding method to extract the co-occurrence distribution of tokens and utilized an attention mechanism to imitate the importance of the irrelevant tokens which further highlights the important aspect term in the sentence. Wang et al., [36] build an end-to-end solution for aspect term extraction using a deep neural network model. Hoang et al., [37] use a pre-trained language model BERT, along with a fine-tuning model for an out-domain ABSA task [37]. Pre-trained language models provide context to the word in a sentence or text. These models learn from the previous occurrences and representation of words from training data [37]. The Bidirectional Encoder Representation of a Transformer (BERT) is a language model that identifies the context of the words from left to right simultaneously [38]. BERT can also be used to perform ABSA tasks mentioned in SemEval2014 task 4. Hoang et al., [37] fine-tuned BERT for a sentence pair classification task to achieve a sentiment polarity classification task that aims to predict the polarity between the topic and aspect [37]. Yang et al., [39] proposed a model that integrates BERT with a multi-head self-attention (MHSA) and local context focus (LCF) mechanism. These studies inspired us to implement a framework leveraging a state-of-the-art BERT model to extract aspects from the user reviews of mobile emergency applications. The results of this BERT model are later analyzed to identify the factors that are to be considered while developing a mobile emergency app or planning an emergency management system using mobile technology.

## 2.4. Ontology

After analyzing the factors that affect user behavior, we are tasked to relate the terms and understand how these terms affect each other. To achieve this task we planned to create a word association vocabulary for emergency mobile applications. Till now the existing ontologies mostly deal with the action and reaction mechanism during an emergency. Over the years Ontologies have provided an easy and faster way for querying and information retrieval capabilities. Kontopoulos et al., [40] define an ontology for climate crisis management which is aimed to support decision-making during climatic disasters. The authors formulated beAWARE, an EU-funded project to help in disaster forecasting and management [40]. A fire-related ontology termed 'EmergencyFire' is proposed by Bitencourt et al., [41]. This ontology shows fire emergency response protocols for evacuation purposes. Domain Ontology for Mass Gatherings (DO4MG) aims to fill the gaps in the communication between medical emergency personnel was proposed by Haghighi et al., [42]. Later a Linked data-based ontology named Management Of a Crisis Vocabulary (MOAC) is proposed, this acts as a vocabulary manual for practitioners and experts to link crisis management activities [3]. Bannour et al., [3] developed a CROnto, Crisis Response Ontology by integrating crisis features, crisis effects, and crisis response to provide a complete and sharable knowledge base for the crisis response stakeholders [3]. Babitski et al., [43] proposed a semantic technological support system called SoKNOS, which aims to highlight the use of semantic technologies in developing disaster management software. Wu et al., (2022) leveraged GPS data generated by mobile devices to propose an evacuation decision-making model during wildfires. Baytiyeh [44] shows the importance of the mobile application, WhatsApp, during emergencies by highlighting the importance of location, event conditions, sharing information such as images and videos, and extending moral support to family and friends. Apart from these, MIDEP, COCCC, mlO!, and CONON are some of the context Ontologies that discuss the design and creation of mobile applications. The Ontology, use case, role of Ontology, and common context terms between our study and existing Ontology are described in Table 2.

This intrigued us to formulate a semantic and contextual-based vocabulary for mobile emergency applications to understand the factors that are influential in improving the usage of these by individuals. The aspect terms extracted from the previous analysis are used to construct a word association tree by highlighting the classes and objects from the corpus data. An object is defined as a sub-class of a class. The classes and the objects are identified with the help of the previous ontologies related to emergency communications.

Table 2. Previous ontologies related to mobile applications and crisis management

| **Ontology** | **Use Case** | **Role of Ontology** | **Common concepts and terms** |
|---|---|---|---|
| MIDEP | Patterns of user interface design and ontology modeling for flexible mobile apps [45]. | Facilitate the choice of depending on the characteristics of the user and the surrounding context, alongside dynamically adapting user interfaces during runtime. | font size, position, interface, and map. |
| COCCC | Context ontology in mobile applications [46]. | Designed for Android mobile applications, to formalize the contextual knowledge embedded within them. | GPS, location, battery, interface, and application. |
| mlO! | A context ontology for mobile environments [47]. | Enables processing and utilization of the context for configuring, discovering, executing, and enhancing various services that may be of interest to the user. | interface, location, network, and coordinates. |
| CONON | OWL-based ontology-based context modeling and reasoning [48]. | Offers an overarching contextual ontology, encompassing fundamental concepts related to basic context, while allowing the flexibility to incorporate domain-specific ontologies in a hierarchical structure. | location, application, network, latitude, and longitude. |
| beAWARE | Crisis Management Procedures for Climate Events Represented Ontologically [40]. | A better "all-around" compact taxonomy that drastically streamlines choice-making and integrates a number of crucial elements is proposed. | location and natural disaster |

# 3. DATA COLLECTION AND ITS DESCRIPTION

## 3.1. Data Collection

Data used in this study are user reviews for emergency applications in Nordic countries. The reviews mentioned in both the Play Store and App Store are retrieved using scraping libraries in Python such as 'google_play_scraper' and 'app_store_scraper'.

We considered five different emergency applications across three countries. Each country has developed its emergency application and runs under the regulations of a government body. '112 Suomi' and 'Emergency Plus' are two such applications that are mostly used in Finland whereas 'Hejlp 113' is heavily used in Norway and Sweden people are relied on 'SOS alarm' and 'Krisinformation.se' emergency applications. The data distribution of the selected applications has been illustrated in the below figures 2, 3, 4, 5, and 6.

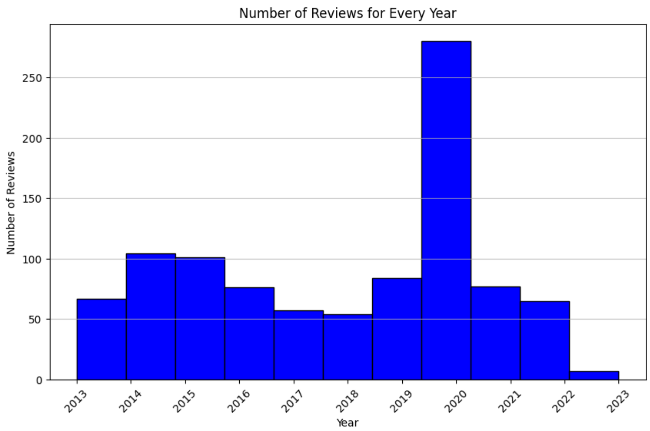


Figure 2. Distribution of reviews collected over the years for Emergency Plus application.

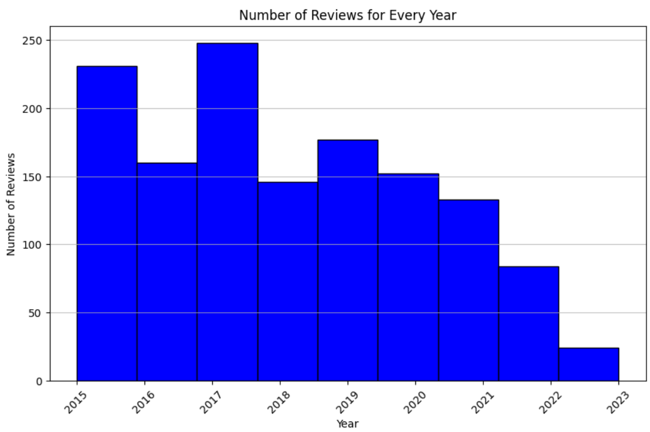


Figure 3. Distribution of reviews collected over the years for 112 Suomi applications.

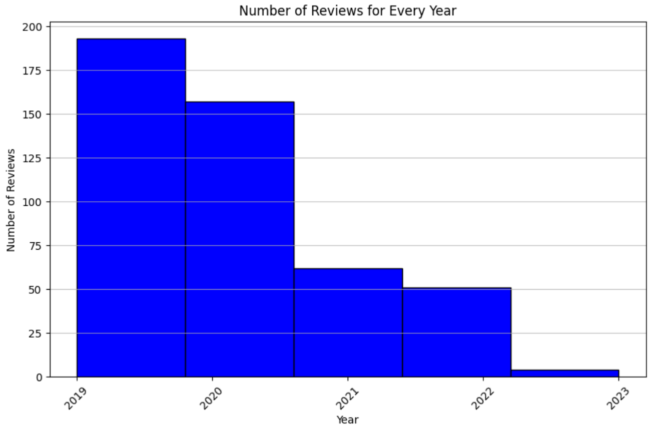


Figure 4. Distribution of reviews collected over the years for SOS alarm application.

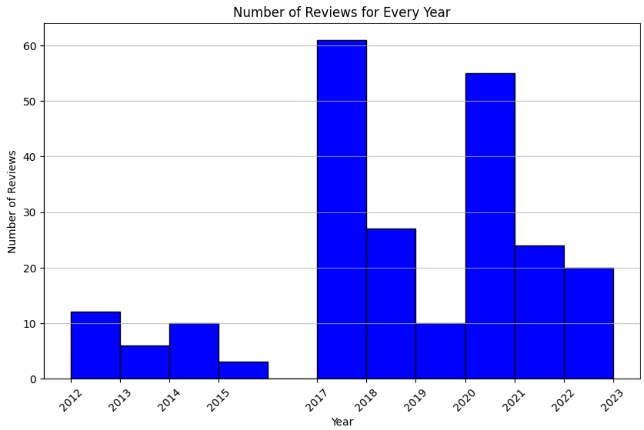


Figure 5. Distribution of reviews collected over the years for Krisinformation.se application.

### 3.2. Data Description

This section describes the data variables considered for the study. The object class of the Google Play scraper library takes googleId, googleLanguage, and googleCountry of the respective emergency app as inputs whereas on the other hand, the object class of the App Store scraper library takes appStoreName, appleAppId, and appleCountry as input. Both methods return the complete data regarding the user and the user's review. In this study, we are interested in Review ID, User Name, Review, Rating, Date of Review, and Review Created Version. This data is stored in a Common Separated Values format (CSV) for easy access and transformation. While scraping the reviews are translated into English using the GoogleTranslator method from the deep_translator python library. Table 3 lists the data variables in the dataset where Review ID is the unique identifier, the name of the user is noted under the username column, review text is mentioned under the review column, respective rating and date of review posted in the rating and date of review column, and review created version contains the version of application when the review posted.

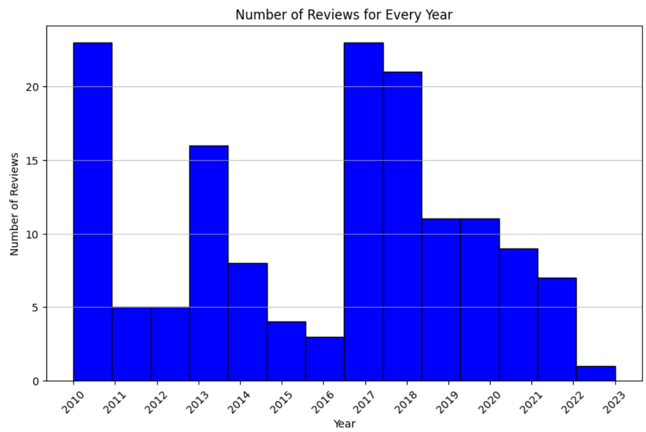


Figure 6. Distribution of reviews collected over the years for Hjelp application.

Table 3. Data variables considered for the analysis

| **No** | **Data Variables** |
|---|---|
| 1 | Review Id |
| 2 | Username |
| 3 | Review |
| 4 | Rating |
| 5 | Date of review |
| 6 | Review created version |

# 4. METHODOLOGY

## 4.1. Polarity Labeling Using VADER

Sentiment analysis is an NLP task that predicts the polarity of the text. In our thesis, we are required to label the user reviews extracted according to their sentiment respectively to understand the user sentiments. To perform such tasks VADER sentiment tool can be leveraged. VADER provides a compound score that combines all the lexicon scores and normalizes between -1 and +1 which indicates the extreme negative and extreme positive respectively. VADER has been very efficient while handling social media data [12]. The unlabeled reviews are labeled with different polarity scores using the VADER sentiment tool. The labels are $l \epsilon \{-1, 0, 1\}$, later decoded into Negative, Neutral, and Positive respectively. The overall polarity score is calculated based on the threshold given by the researcher. The considered threshold for this study is mentioned below

- if compound_score >= 0.05 then 'Positive'
- Else if compound_score <= 0.05 then 'Negative'
- Else 'Neutral'

Later a Mean Sentiment Score (MSS) is calculated for each application. The MSS is defined as the mean sum of the sentiments of each version for a particular application. MSS is shown in a simple equation form below

$$MSS_i = \sum_{j=version} \frac{\text{sum of sentiment score}}{\text{total number of observations}} \tag{1}$$

Figure 7 illustrates the simple NLP task pipeline to extract insights from the user reviews. This pipeline helps us to achieve our goal of providing useful information regarding user behavior and sentiment towards emergency apps and also highlights the most important themes discussed among the users which eventually helps us to understand the user needs and preferences.

## 4.2. Topic Modeling

The pre-processing phase mentioned in the Figure 7 is briefly illustrated in Figure 8. Pre-processing of the texts contains the removal of irregular terms from the texts, eliminating the English stopwords, and trimming the word to its root word using lemmatization techniques. Because of their frequent presence in natural language and little semantic meaning, stopwords are frequently filtered out of or omitted from text processing and analysis activities. Whereas lemmatization is a text normalization process to reduce the words to their base word, this reduces the dimensionality and improves the accuracy of NLP tasks.

The reviews were extracted from the Play Store and App Store under data pre-processing as mentioned in the previous section 3.3. Once the preprocessing

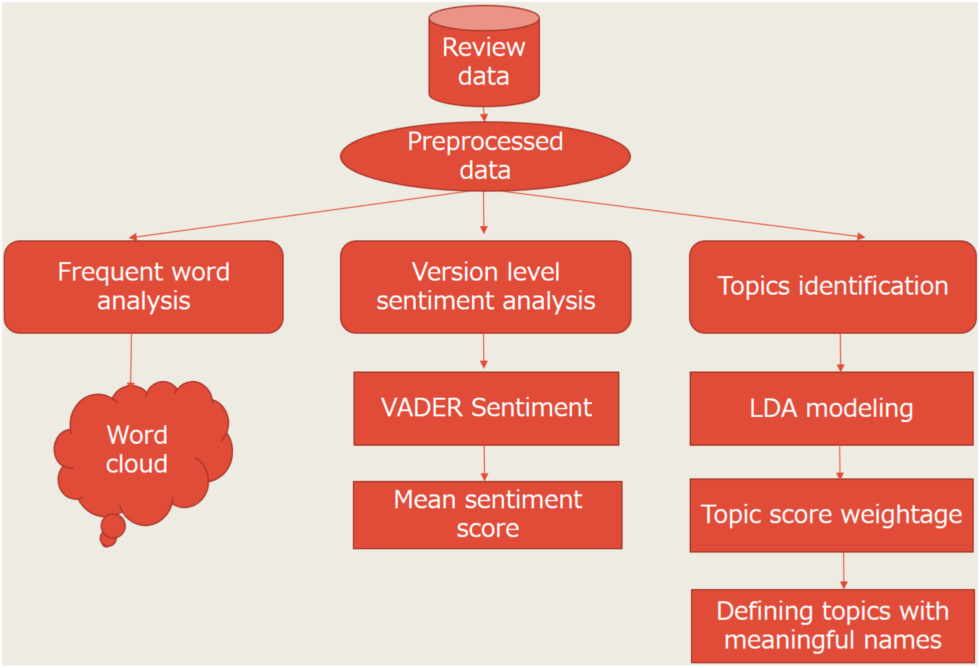


Figure 7. Basic text mining using sentiment analysis and topic modeling pipeline.

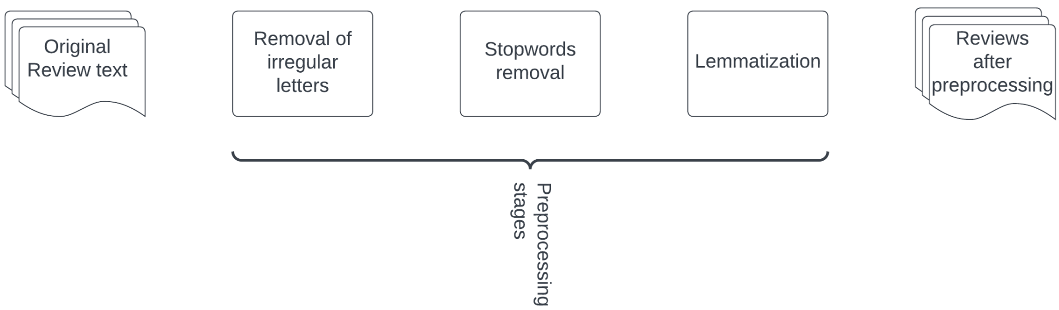


Figure 8. Preprocessing pipeline adopted in this thesis.

processing is complete to understand the word association we leveraged bigram and trigram models. In the context of NLP, n-grams are continuous sequences of n elements, often words. In order to identify certain language patterns within the text, we concentrated on creating n-grams in this study. The following actions were taken throughout the n-gram creation process:

1. **Choosing the Value of n**: The study's goals and the intended level of linguistic comprehension were taken into consideration while choosing the parameter n. Unigrams (n=1), bigrams (n=2), trigrams (n=3), and higher-order n-grams are typical values.

2. **Sliding Window**: A sliding window of size n was applied to the tokenized text. The window moved through the text one token at a time, capturing sequences of tokens to form n-grams.

3. **N-gram Construction**: N-grams were constructed by concatenating the tokens within the sliding window. For example, in the case of bigrams (n=2), consecutive pairs of words were combined to create the n-grams.

N-grams (bigram and trigram) are essential for language modeling because they help with word sequence probability prediction. From the trigram model output, a TF-IDF corpus is created which later passes as an input to the LDA model. A popular approach called TF-IDF gives words in a corpus of text weights depending on how important they are both inside and across individual documents. The following stages were taken during the TF-IDF calculating process:

1. **Term Frequency ($tf$)**: For each document, the frequency of each word was calculated. This indicated how often the word appeared in the document.

2. **Document Frequency ($df$)**: The number of documents containing a specific word was counted across the entire dataset. This reflected the word's ubiquity across the corpus.

3. **Inverse Document Frequency ($idf$)**: The inverse document frequency was calculated for each word using the formula
$$idf(word) = \log\left(\frac{N}{1 + df(word)}\right), \tag{2}$$
where $N$ represents the total number of documents in the corpus.

4. **TF-IDF Calculation**: The TF-IDF weight for each word within each document was computed using the formula
$$tfidf(word, document) = tf(word, document) \times idf(term), \tag{3}$$
where $tf(word, document)$ is the word frequency within the document.

A high-dimensional representation of the dataset was produced by the computed TF-IDF weights, where every record was delimited by a vector of TF-IDF values. To extract latent themes from the data, TF-IDF vectors were utilized as the input for topic modeling methods like LDA. In this study, LDA is employed as the main topic modeling method. LDA is a generative probabilistic framework that makes an inference that each text consists of a variety of subjects, each of which is expressed by a distribution over words. The model looks for underlying topics and the word distributions that go along with them in order to reveal latent semantic structures within a corpus. The following steps outline the implementation of the LDA model:

1. The preprocessed text corpus is used to produce a lexicon of distinctive terms. The topic-word distributions are created using this vocabulary as the foundation.

2. Each row in the document-term matrix (DTM) represents a document (user review), and each column is an individual word in the lexicon. The entries in the matrix showed how frequently each term appeared in a document.

3. In LDA, the number of topics ($K$) is an important hyperparameter. Using methods like perplexity analysis and coherence metrics, several values for $K$ are investigated in order to identify an optimal value.

4. The LDA model is trained using a Gensim library. The aim of the training is to make the model learn the topic distributions of each document and the word distributions for each topic.

5. After training, the topics were decoded by examining the terms most likely to be connected to each topic. This entailed determining the terms in each topic's word distribution that had the highest probability.

The selected topics and the most likely terms inside each subject made up the LDA model's final output. These topics gave information on the recurring themes and debates in the user reviews. The most frequently occurring terms and their contextual significance within each subject were manually analyzed in order to understand the key concepts.

Later, to analyze the topic scores for different sentiment labels among different datasets a new metric is constructed. Nguyen et al.,[19] used a similar metric to measure brand performance using the dimensional score. Here in our study, the Topic Score $TS$ is used to measure the importance of the user-defined topics across the datasets. The Topic Score $TS$ formula is

$$TS_{a,t_k} = \frac{\frac{1}{N_a}\sum_{r=1}^{N_a} p_r, t_k}{\frac{1}{N}\sum_{r=1}^{N} p_r, t_k}, \tag{4}$$

where Topic Score $TS$ is defined as the mean frequency of the probability of the word $r$ of a given topic $k$ in either a positive or negative dataset divided by the frequency of the probability of the same word $r$ of a topic $k$ in the whole dataset.

### 4.3. Bidirectional Encoder Representation of Transformer (BERT)

BERT is a transformer-based model that, due to its bidirectional nature, is able to comprehend the context of words in a phrase by taking into account both the left and right context [38]. By utilizing the attention mechanism found in transformers, BERT processes text as a whole as opposed to traditional language models like RNNs and LSTMs, which process text sequentially. BERT utilizes the attention mechanism within transformers to analyze text as a whole.

1. **Input representation**: BERT uses tokenized input text, dividing words into subword tokens. Each token has an embedding that incorporates token embeddings, positional embeddings, and segment embeddings (to discriminate between various phrases in a pair). Figure 9 shows the embedding layers of the BERT model.

2. **Transformer Encoder Layers**: BERT is made up of many layers of stacked transformer encoders. The model can evaluate the relative weights of various words in the input phrase because each layer has a multi-head self-attention mechanism. Additionally, feedforward neural networks are implemented in each layer to process the attended representations.

3. **Pre-trained Contextualized Embeddings**: BERT is pre-trained on a sizable corpus of text using two unsupervised tasks: next-sentence prediction NSP and masked language modeling MLM (which predicts masked words). Through this method, BERT is able to develop rich contextualized representations that accurately capture the meanings of words in relation to their surroundings.

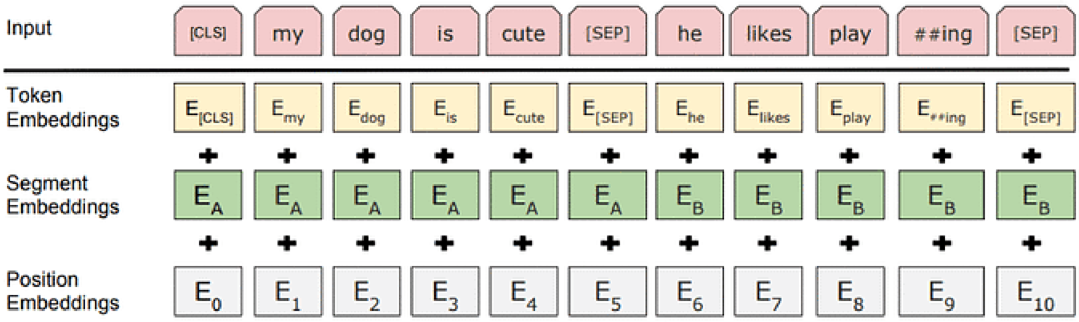


Figure 9. Embedding layers of BERT [49].

### 4.4. Aspect Term Extraction Polarity Classification (ATEPC) Task

One of the objectives of the study is to identify aspect terms in the reviews and their respective sentiments to understand the user opinion on the usage of mobile emergency applications point out the reason for negative user behavior and later define an ontological vocabulary for emergency apps. To achieve this task we employed a pipeline. The diagrammatic illustration of the outline is shown in Figure 10.

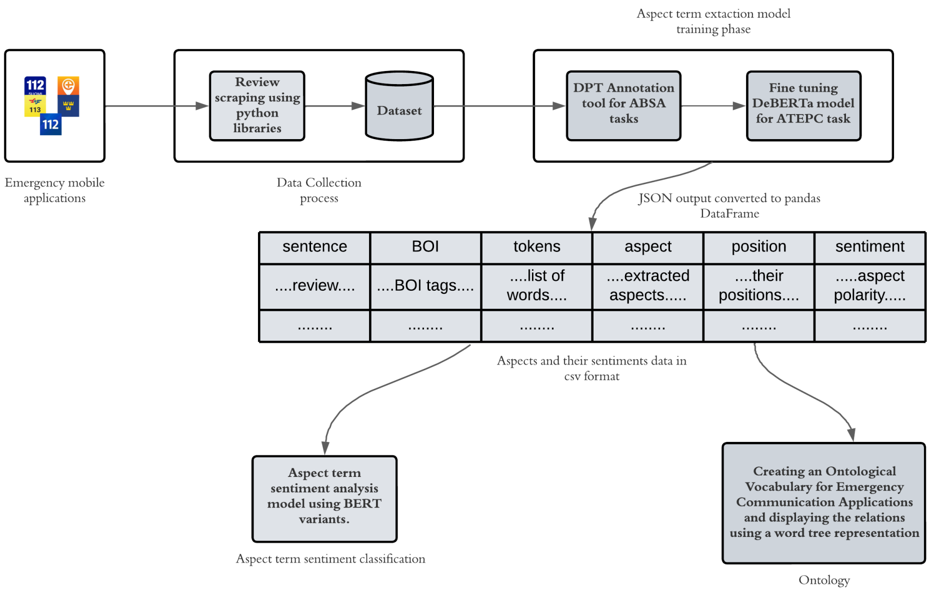


Figure 10. Outline of the study.
BOI - Beginning of the aspect, Outside the aspect, and Inside the aspect.
DPT - a web-based annotation tool.

Traditional sentiment analysis methods can be used to forecast the overall polarity of a sentence, but they are unable to isolate its components. From the user reviews of emergency applications, the ATEPC job assists in extracting the aspect phrases and their corresponding polarity. In our investigation, the input will be user reviews. An ATEPC task model then outputs a list of distinct aspect phrases together with their emotion in the context of the sentence or review. As was already explained, the BERT language model concurrently recognizes the context of a phrase from left to right [38]. We need to annotate the training data in order to fine-tune a BERT model for an ATEPC job. After data annotation, the annotated data is given as the input to obtain the desired results. The detailed explanation of annotation and the fine-tuned BERT model are discussed in the below sections 4.5 & 4.6.

### 4.5. Data Annotation for ATEPC Task

Data annotation is a process to prepare training data for the model. Sometimes this process can be tedious. Every model needs training data to train and learn the patterns required for the problem statement. In this study, we annotate the text data which is review data accordingly. To perform the aspect extraction task the data should be labeled manually by highlighting the aspect terms and the polarity of the aspect term in the review. PyABSA[1] is a modular framework to perform ABSA tasks [50]. Yang et al. [39] also mentioned that data annotation is a major significance of the PyABSA framework. DPT [39] is a data annotation tool for aspect-based sentiment analysis (ABSA) tasks initially mentioned in Yang et al. [39], which helps us to annotate the data and download the annotated data for aspect polarity classification (APC) task format later the APC format data is converted to an ATE format. The sample data annotation is shown in below Figure 11.

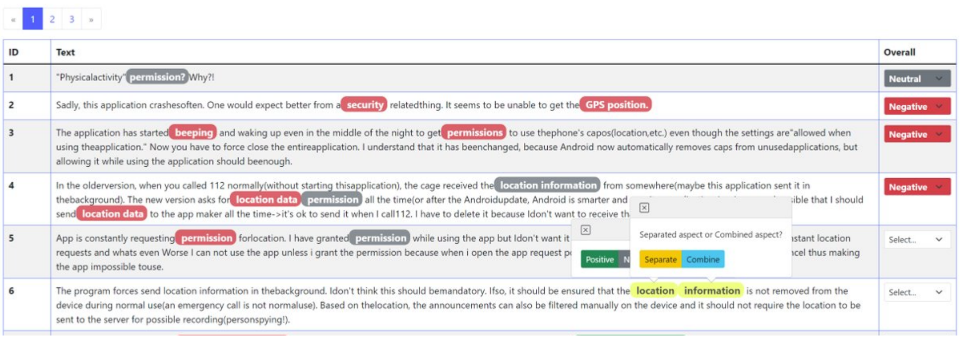


Figure 11. Web-based interface of the data annotation tool DPT.

*Note:* DPT a data annotation tool adopted by [50] and its Github link - `https://github.com/yangheng95/ABSADatasets.git`.

DPT is a web-based tool used for the manual text annotation process. Once the annotation is done, three files will be created a CSV file training set for the basic sentiment analysis process, a TXT file training set for ABSA tasks, and a JSON file for saving unfinished annotations. The downloaded TXT file is for the APC task, To convert the APC task training file to the ATE task training file we leveraged the convert-apc-set-to-atepc-set method from PyABSA. The data in the final output file follows the BOI tagging scheme, This scheme is widely used to perform ATE tasks which are Beginning, Inside, and Outside of the aspect terms. For example, the sentence 'Fine with dark mode but where do you switch back to light mode?' will be divided into tokens of words $\{w_1, w_2, \ldots, w_n\}$, $n = 14$. According to the BOI scheme, this sentence will be labeled as {O, O, $B_{asp}$, $I_{asp}$, O, O, O, O, O, O, O, $B_{asp}$, $I_{asp}$, O}.

[1]Github: `https://github.com/yangheng95/PyABSA.git`

### 4.6. Fine-Tuning BERT for ATEPC Task

This section deals with the fine-tuning of the BERT model for our task. BERT works on the Masked Language Modeling (MLM) technique. MLM masks the tokens in a given sentence by replacing them with [MASK] value. Later the model is trained to predict the [MASK] value based on the context of previous and forthcoming values in an input vector. Hence the model learns the context of the word in the sentence. In contrast to word2vec [51];[52] [53];[54] and GloVe [55], each word in a sentence has different BERT embeddings, these word embeddings will change from sentence to sentence according to the context of the word in the sentence [56]. DeBERTa [57] is a variant of the BERT model, which leverages disentangled attention mechanisms and decoding enhanced training to achieve a state-of-art performance on various NLP tasks which includes ABSA tasks. In DeBERTa, attention weights of words are calculated using the disentangled matrices. Disentangled matrices contain the contents and relative positions of a word in a sentence encoded into two vectors to represent each word, this mechanism is known as the disentangled attention mechanism [57]. This makes DeBERTa a state-of-art model to consider for our study.

LCF-ATEPC is a framework proposed by Yang et al., [39] to perform ABSA tasks. The Local context feature (LCF) generator and Global context feature (GCF) generator are two different units in the LCF-ATEPC model which contain two different pre-trained BERT models. Aspect sentiment polarity is predicted after combining the output from the LCF and GCF using the Feature interactive learning (FIL) layer [39]. Only GCF helps to extract the aspect terms. The input of this model is annotated with ATE and APC labels, the former denotes if the word relates to an aspect term, and the latter indicates the sentiment of an aspect term in the given input sentence [39]. The data annotation part is explained in section 4.5.

### 4.7. Aspect Terms Sentiment Classification

Hoang et al., [37] highlight the sentence pair classification to perform the aspect term sentiment analysis task. Song et al., [58] proposed an attention-based encoder model to identify the semantic modeling between context and target. In this study, we fine-tune different pre-trained BERT variants to perform supervised learning tasks in which the context and aspect will be the dependent variables and sentiments (positive, negative, and neutral) as the independent variables. Base BERT contains 12 layers of encoder blocks (transformer blocks), 768 hidden units, and 12 attention heads in each transformer layer with a total of approximately 110 million parameters. To reduce the computation time and cost, smaller version BERT models are proposed such as tinyBERT, DistilBERT, RoBERTa, and ALBERT. A classification layer with a dropout rate of 0.1 is added to the pre-trained BERT variants along with a ReLU activation function. The input array for the models is created using a pre-trained BERT tokenizer by joining the context and aspect term with a [SEP] token. For example, 'The location access is good' and 'location' is context and aspect respectively, and the input token vector is structured as [CLS]+'The'+'location'+[MASK]+'is'+'good'+[SEP]+location+[SEP]. The tokenizer output encoded pairs are token_ids, attention_masks, and token_type_ids. These variables are provided as input to the model to predict the probability of the sentiment class.

After the sentiment classification task, we employed a pipeline to extract deeper insights from the reviews. The pipeline is inspired by the Arhab et al., [59] study on Twitter data to understand user behavior in car parking analysis. The dataset is divided into two different data frames according to the aspect term sentiment values namely the Positive aspect sentiment data frame (PASDF) and Negative aspect sentiment data frame (NASDF). Perform an Empath categorization methodology on these two different data frames to understand which category occurred more frequently. Empath client [2] offers category ratings that express the degree to which a text is associated with each empathy category. These ratings were used to determine how frequently themes connected to empathy appeared in our dataset. The pipeline of the analysis is displayed in Figure 12.

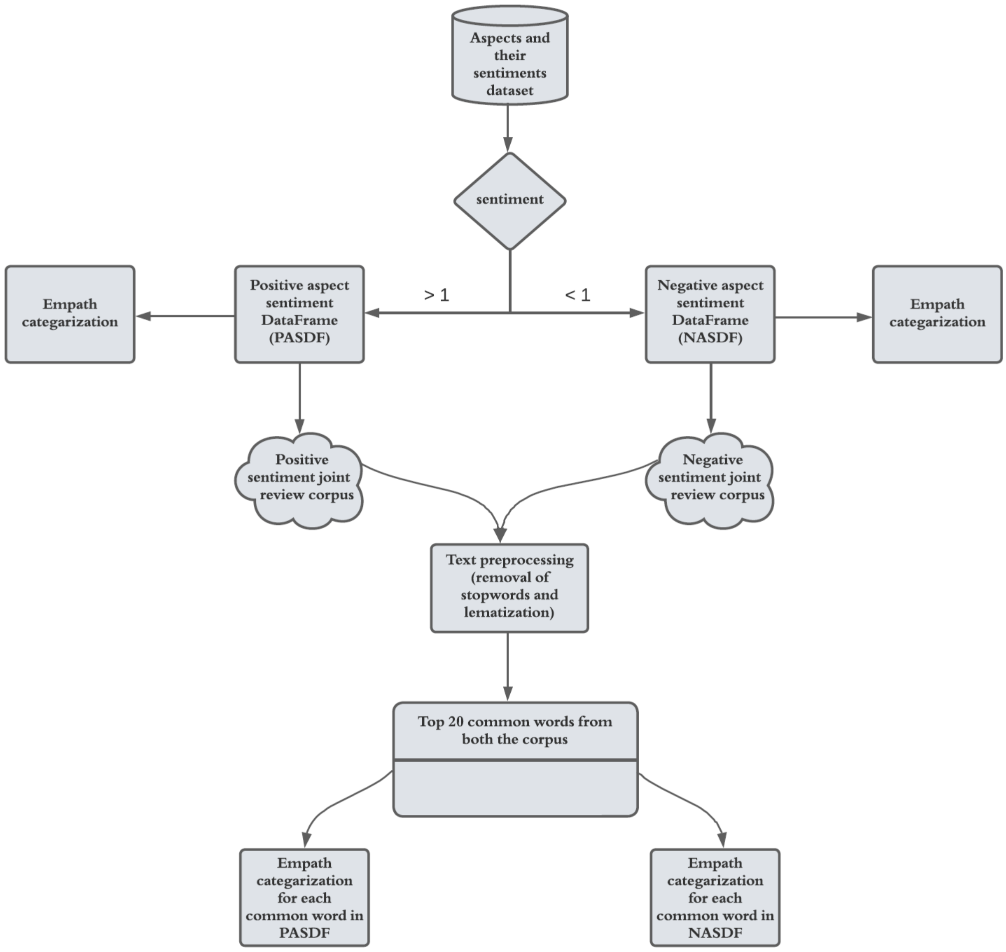


Figure 12. A flowchart representation for empath categorization to identify the categories in both positive and negative aspect sentiment datasets.

[2]Github: `https://github.com/Ejhfast/empath-client/tree/master`

### 4.8. Ontology for Mobile Emergency Applications

To define a relation between the extracted aspect terms, the latter are categorized into classes and objects. A class will have multiple objects associated with it. Inspired by the existing ontologies related to emergency communication, some of the terms are considered as classes. The relation between classes and objects is formed by calculating a semantic similarity score. The similarity score is calculated using cosine similarity given the word embeddings of class and object pairs, the pre-trained BERT base model is used to construct word embeddings. Cosine similarity is defined as the dot product of the two-word embedding vectors divided by the product of the magnitude of the two vectors [31]. Once the word association is constructed using the similarity score and context awareness, these are compared with the existing relevant ontologies. This analysis is not an action and reaction mechanism, it mostly identifies the factors that are important to be considered for a mobile emergency application. The similarity score $Sim$ is

$$Sim(C, O) = \frac{\sum_{i=1}^{n} \sum_{j=1}^{m} C_i * O_j}{\sqrt{\sum_{i=1}^{n} C_i^2} \sqrt{\sum_{j=1}^{m} O_j^2}}, \quad (5)$$

where $C$ represents classes and n denotes the number of classes in the corpus. The object is represented by $O$ and the number of objects in the corpus is denoted by $m$.

# 5. RESULTS

In this section, we report the findings from our thorough examination of the Nordic nations' emergency communication through mobile applications. Our research focuses on five different types of analysis, each of which offers a special perspective through which we may understand the subtleties and complexity of the communication environment in crisis situations. We have thoroughly examined the corpus to uncover multifaceted layers of information, shedding light on the crucial components that shape effective emergency communication strategies. Our methods include sentiment analysis, topic modeling, aspect term extraction, aspect sentiment classification, and vocabulary-based ontology. Through this thorough investigation, we not only get a deeper knowledge of the data but also add to the general discussion on how to best utilize communication efforts in emergency situations.

## 5.1. High-Level Data Exploration

In this thesis, we display six different word clouds, each of which captures a different aspect of our data study. These word clouds act as educational visual summaries that help us identify important trends and patterns that could otherwise go undetected. The first word cloud in Figure 13 shows the language that was taken from the emergency plus the application's good customer ratings. The predominance of words like "app", "location", "good", "phone", and "great" highlights the emphasis on functionality and accuracy offered by the app during crises. Figure 14 depicts words like "emergency", "wrong", and "address" which point to a lack of emphasis on group activities and assistance during emergencies while using mobile apps. This realization emphasizes the need for community cohesion for the effective communication of crucial information in times of emergency. Positive user influencing factors of the 112 Suomi app is the focus of the third-word cloud shown in Figure 15, which includes words like "application", "center", and "positioning". Figure 16 displays a word cloud with terms related to the 112 Suomi app's drawbacks, emphasizing "time", "location", and "phone". The fifth-word cloud in Figure 17 explores the positive phrases that SOS alarm app users have specified, and it reveals words like "GPS", "battery", and "notification". Lastly, the sixth-word cloud in Figure 18 also includes words like "use", "message", and "service" which reflects the users' dissatisfaction with the SOS alarm application.

Finally, these six-word cloud illustrations collectively reveal a complex story of emergency communication in Nordic nations. When combined, they offer a thorough overview of the topics, difficulties, and priorities that influence the communication landscape during crises from the user's point of view. We obtain a comprehensive awareness of the language environment as well as the intricacies and tactics that support efficient emergency communication through these representations.

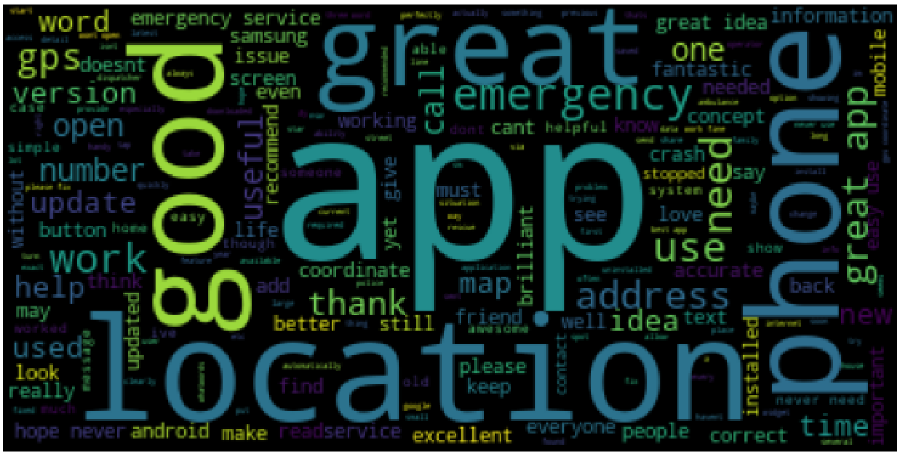

Figure 13. Word cloud for emergency plus app positive reviews.

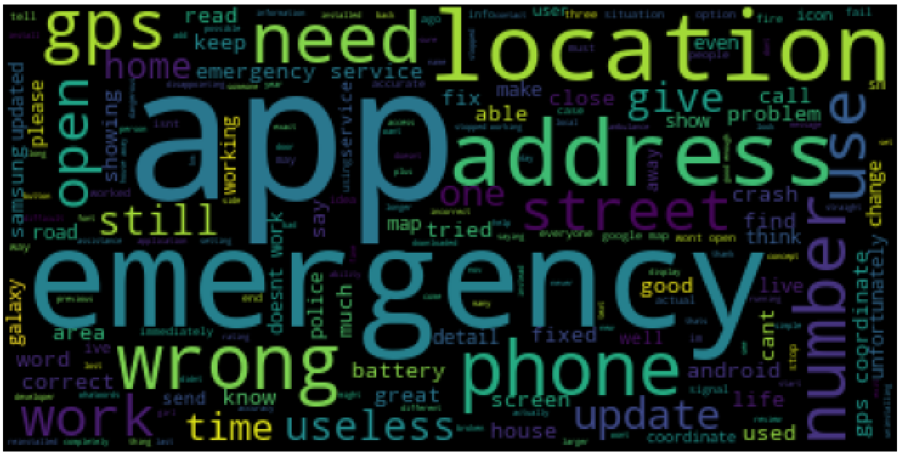

Figure 14. Word cloud for emergency plus app negative reviews.

### 5.2. Version Level Sentiment Analysis

MSS is used to understand the variation of the sentiments for each application over different versions. Through this one can also infer the change in sentiments of users regarding the applications. The MSS is calculated for each application and displayed using visualization libraries. Figure 19 represents the MSS score for the Emergency Plus app, we can observe that user sentiment of this app has never been less than 0, which means that the users exhibit positive sentiment for all the versions of this application. But that is not the case in other application MSS scores. Figures 20, 21, and 22 illustrate the MSS score plot for 112 Suomi, SOS alarm, and Hjelp 113 apps respectively. In these applications, the users exhibit both positive and negative sentiments for different versions.

### 5.3. Topic Modeling

As mentioned in section 4.2, we leveraged the LDA model to identify the topics discussed in the user review text corpus. The parameter $K$, is selected randomly to

Figure 15. Word cloud for 112 Suomi app positive reviews.

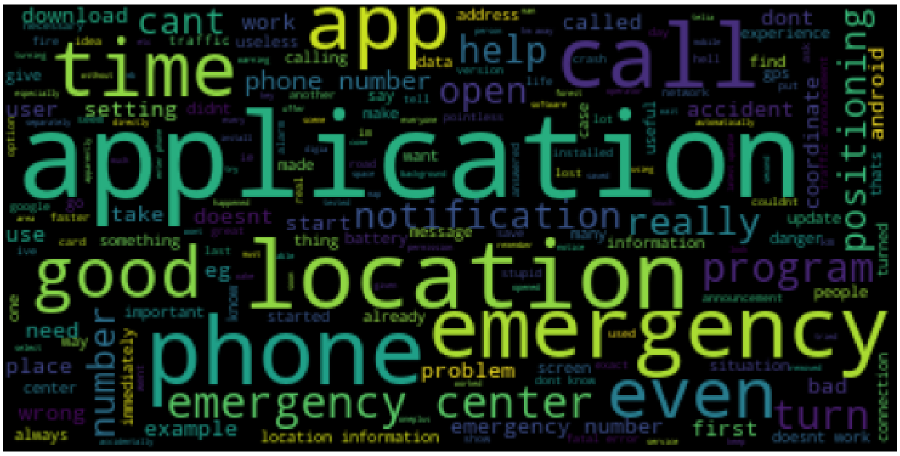

Figure 16. Word cloud for 112 Suomi app negative reviews.

train the initial topic model. Later multiple $K$ values are selected in between the range of 0 to 35, $K\epsilon[0,35]$. The coherence score at each $K$ value is noted and plotted in a line plot to consider the optimum $K$ value for the topic modeling. Figure 23 illustrates the coherence score of different $K$ values.

After plotting the coherence chart, we can observe that at $K$ value is equal to 8, $K = 8$ the model has a coherence score of 0.48. Hence we model the topic modeling with a total number of topics of 8. Each topic is represented as the sum of weighted terms from the corpus. The representation of the LDA model output is displayed in Figure 24.

In topic modeling, the major challenge is to identify the proper topic name. Here in this study, we named the topics by considering the terms mentioned in the topic vectors and the use case. The named topics along with the topic numbers are shown in Table 4.

After defining Topic Score $TS$, we divided the review corpus into two different data frames based on the sentiment of each review. $TS$ for each topic for two data frames i.e., positive and negative sentiment data frames are calculated for each country separately. The illustrations of $TS$ for two data frames across different countries are

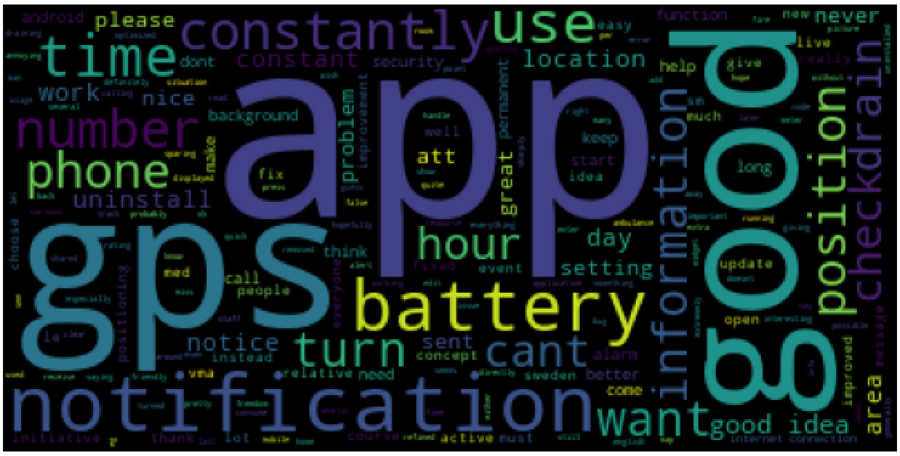


Figure 17. Word cloud for SOS alarm app positive reviews.

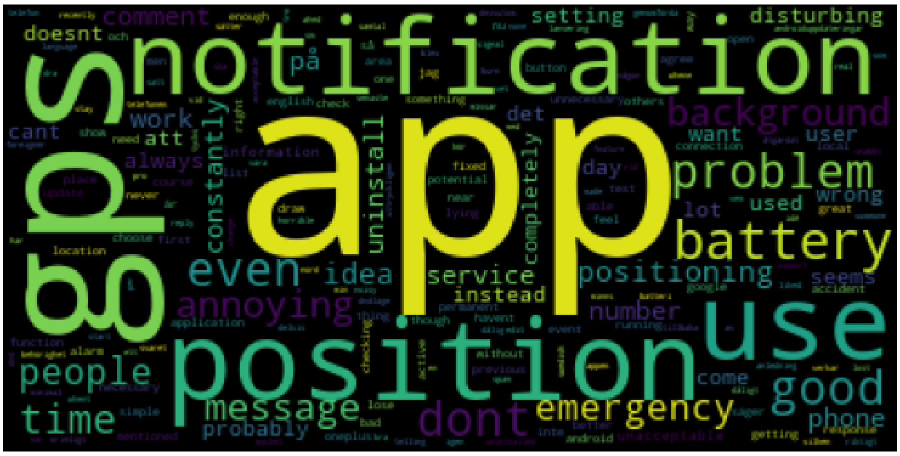


Figure 18. Word cloud for SOS alarm app negative reviews.

Table 4. Tabular description of user-defined topics and words belonging to each topic

| **Topic No.** | **Topic Name** | **Keywords** |
|---|---|---|
| 0 | Application usability | great, app, work |
| 1 | Application functionality | position, good, find, location |
| 2 | Application reliability | app, help, useful, support |
| 3 | User's satisfaction | important, easy, great, save lives |
| 4 | Application necessity | work, necessary, recommend |
| 5 | Information transferability | notification, work, save, call |
| 6 | Device compatibility | crash, phone, oneplus |
| 7 | User's trustworthiness | need, emergency, awesome |

displayed in Figures 25 and 26.

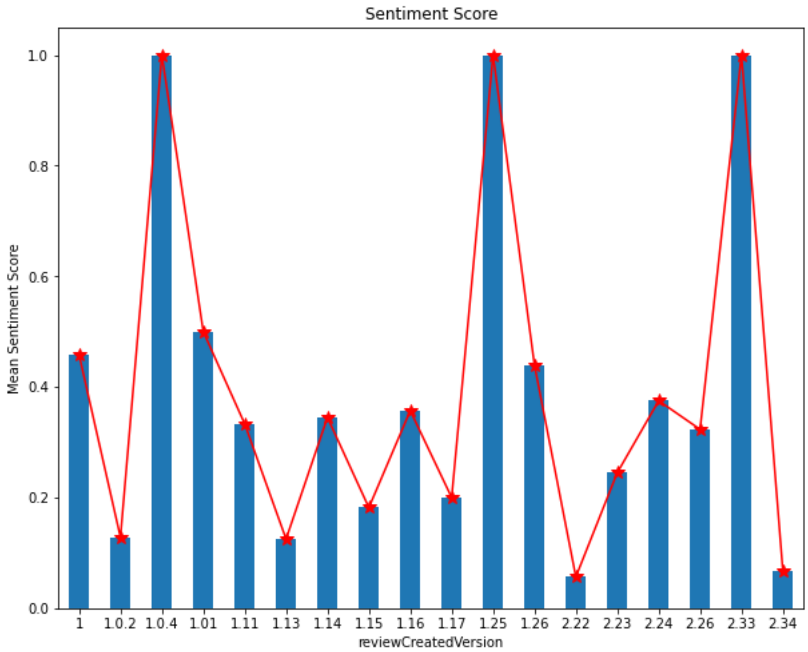


Figure 19. A combination chart of MSS for Emergency Plus app.

### 5.4. Aspect Term Extraction Task

The DeBERTa model mentioned in section 4.6 was used to perform the aspect term extraction (ATE) task. The output contains reviews, BOI tagging of each review, sentence tokens, aspect terms, the position of the aspect terms in the review, and their respective sentiments. The distribution of sentiments in the reviews can be seen in Figure 27.

We can observe that the dataset has more negative sentiments than positive and neutral sentiments. Later the sentiment rate ($SR_a$) of each aspect is calculated to identify the top 30 aspect terms from the whole corpus. The sentiment rate ($SR_a$) can be explained as the normalized weighted sum of the aspect in the dataset. The equation of $SR_a$ is

$$SR_a = \frac{n}{N} \sum_{i=1}^{n} w_i, \tag{6}$$

where $N$ is the total number of rows in the dataset, $n$ is the total number of reviews for the 'a' aspect and $w \in (-1, 0, 1)$ -1 for negative, 1 for positive, and 0 for neutral sentiment.

The top 30 aspects in the review data corpus are highlighted in Appendix 8.3 sorted by the frequency of the aspects in the dataset. The app, Location, Battery, Update, Idea, Emergency, etc., are some of the aspects. Among these 'App' has mostly occurred in the dataset. The $SR_a$ of each aspect is represented in a line plot in Appendix 8.3.

### 5.5. Aspect Sentiment Classification Task

This section describes the results obtained from the aspect term sentiment analysis using different BERT variants and later designs a pipeline to identify the impact categories using Empath categorization techniques. The sentiment analysis model is a sentence pair classifier model as discussed in section 4.7. The model takes context and aspect pair as the input and predicts the sentiment of the aspect term in a given context. We have considered four different BERT models which are BERT base, DistilBERT, RoBERTa, and ALBERT, and stacked a hidden layer on top of each model with a dropout rate of 0.1 and in the end, added a ReLU layer to predict the probability of labels (positive, negative, and neutral). The positive label is encoded into 1, neutral to 0, and negative to 2. Table 5 demonstrates the results from each variant BERT model.

**Base BERT** - According to Devlin et al., [38] BERT aims to pre-train deep bidirectional representations by considering the sentence context from left to right. This pre-trained model can be fine-tuned to various NLP tasks such as next-sentence prediction, question and answer, and language inference by just adding one additional hidden layer to the architecture [38].

**DistilBERT** - DistilBERT is trained using a combination of distillation and standard BERT pre-training. The distillation method allows the DistilBERT model to extract knowledge from a larger model and compress it into fewer parameters. According to Sanh et al., [11], DistilBERT extracts 97% of context by reducing the BERT model size by 40% and 60% faster.

**RoBERTa** - RoBERTa leverages larger batch sizes, and more training data, and dynamically changes the training data to construct a more robust language model which makes RoBERTa achieve better generalization and performance [60]. Liu et al., [60] also concluded that RoBERTa succeded in achieving state-of-art results on GLUE, RACE, and SQuAD datasets.

**ALBERT** - A Lite BERT (ALBERT) is an efficient and parameter-reduced version of BERT. The major concept behind ALBERT is to point out the inefficiencies of BERT by sharing parameters across layers which results in reducing the overall model size while maintaining or improving the overall performance of the model. ALBERT leverages a factorized embedding parameterization to share the embeddings and a cross-layer parameter sharing technique to share transformer layers parameters [61]. This helps ALBERT to achieve a more accurate and faster training process.

By considering the pros and cons of each BERT variant and its performance on our dataset we conclude that RoBERTa and ALBERT models gave better results compared to the other two variants which are Base BERT and DistilBERT. Since our dataset is small, large-batch training will make the learning process faster which might lead to overfitting scenarios. RoBERTa may not perform to its best on small datasets since there is limited data available for fine-tuning. Whereas ALBERT on the other hand can leverage the limited data during fine-tuning through its parameter-sharing technique and it requires limited computational resources. Hence we consider the ALBERT model for our aspect term sentiment classification task. The classification report of the ALBERT model on the test dataset is shown in Table 6.

**Performance metrics**: To measure the performance of the trained models many literatures have adopted the different performance metrics. Precision, Recall, and F1-Score are the metrics we employed from the literature to measure the performance of the models and displayed the results in a classification table 6.

1. **Precison**: precision is the percentage of accurately predicted positive instances. It emphasizes the precision of optimistic forecasts. Eq 7 defines the formula for how a precision score is calculated

$$Precision = \frac{TruePositives}{TruePositives + FalsePositives} \tag{7}$$

2. **Recall**: Recall figures out what percentage of all positive events were really anticipated correctly. It highlights how well the model can account for all instances of success. Recall equation is mentioned in eq 8

$$Recall = \frac{TruePositives}{TruePositives + FalseNegatives} \tag{8}$$

3. **F1-score**: The harmonic mean of recall and accuracy is known as the F1-score. When both false positives and false negatives are important, it offers a balance between precision and recall, which is helpful

$$F1 - Score = 2 \times \frac{Precision \times Recall}{Precision + Recall}, \tag{9}$$

Table 5. Results of different BERT variants for each label class Neutral, Positive, and Negative along with total accuracy

| **Model** | **Precision** | **Recall** | **F1-score** | **Accuracy** |
|---|---|---|---|---|
| Base-BERT | 1.00<br>0.76<br>0.78 | 0.11<br>0.86<br>0.97 | 0.20<br>0.81<br>0.86 | 0.78 |
| DistilBert | 0.33<br>0.52<br>0.85 | 0.12<br>0.87<br>0.69 | 0.17<br>0.65<br>0.76 | 0.66 |
| RoBERTa | 0.79<br>0.76<br>0.92 | 0.68<br>0.95<br>0.83 | 0.73<br>0.85<br>0.87 | 0.84 |
| ALBERT | 0.71<br>0.80<br>0.88 | 0.68<br>0.85<br>0.86 | 0.70<br>0.82<br>0.87 | 0.83 |

Empath framework has nearly 200 default categories and this framework also provides the flexibility to create our categories according to the problem statement. In this study, we defined 11 different categories namely application, coordinates,

Table 6. Classification report on test data from ALBERT model (the neutral class is represented as 0, Positive as 1, and Negative as 2)

| | **Precision** | **Recall** | **F1-score** | **Support** |
|---|---|---|---|---|
| 0 | 0.71 | 0.68 | 0.70 | 59 |
| 1 | 0.80 | 0.85 | 0.82 | 101 |
| 2 | 0.88 | 0.86 | 0.87 | 175 |
| accuracy | | | 0.83 | 335 |
| macro avg | 0.80 | 0.80 | 0.80 | 335 |
| weighted avg | 0.83 | 0.83 | 0.83 | 335 |

device_battery, emergency_services, exposure, fire_emergency, medical_emergency, miscellaneous, natural_site, network, and technology. The categories and the words associated with the category are shown in Table 7. The categories are considered with the help of ontological reasoning given in section 5.6.

Table 7. User define empath categories based on the aspect terms association

| **Empath category** | **Sample words associated with each category** |
|---|---|
| application | functionality, interface, new_software, format, update |
| coordinates | latitude, GPS, address, location, position |
| device_battery | power, battery_drain, power_consumption, batteries, energy_consumption |
| emergency_services | evacuation_plans, service_centers, fire_alarm, fire_engines, distress_call, phone_lines |
| exposure | notification, relevant_information, alerting, inaccurate_information, electronic_messages |
| fire_emergency | forest_fires, nuclear_explosion, outbreaks, amazon_rain_forest, wildfires, yellowstone_national_park |
| medical_emergency | cardiac_arrest, ambulance, nearby_hospital, heart_attack, CPR, liver_failure |

| Empath category | Sample words associated with each category |
|---|---|
| miscellaneous | permission, necessary, recommending, allowed, save_lives, guidelines |
| natural_site | hiking, mountains, cross_country_skiing, waterfalls, lake_michigan, valleys |
| network | signals, transmitter, coordinates, satellites, radio_signals, phone_signals, fiber_optic_cables |
| technology | initiative, safeguards, security, idea, urgent_need, need, flexibility |

The frequency charts of this analysis are illustrated in Figures 28 and 29. Later top 20 common terms from both data frames are noted. Before identifying the common terms, initial pre-processing such as removal of stopwords and timing the word to its root word using lemmatization a text-mining technique so that we can mitigate the redundancy of the words. The top 20 common words along with their frequency in two different datasets are illustrated in Table 8. Reviews associated with each common word are combined to undergo an empath categorization technique. Finally, the top empath categories associated with each common word along with their respective empath scores are mentioned in Table 9, the former displays the reviews from PASDF and the latter shows the reviews from NASDF.

Table 8. Top 20 common words from both the PASDF and NASDF

| Common Words | Number of occurrences in negative reviews | Number of occurrences in positive reviews |
|---|---|---|
| locate | 190 | 67 |
| work | 174 | 43 |
| batteri | 146 | 30 |
| updat | 141 | 22 |
| emerg | 124 | 78 |
| phone | 119 | 41 |
| good | 98 | 109 |
| use | 93 | 46 |
| notif | 88 | 15 |
| time | 79 | 20 |
| address | 75 | 26 |

| Common Words | Number of occurrences in negative reviews | Number of occurrences in positive reviews |
|---|---|---|
| number | 75 | 26 |
| open | 74 | 15 |
| would | 73 | 56 |
| posit | 65 | 15 |
| need | 64 | 62 |
| call | 60 | 45 |
| applic | 59 | 18 |
| great | 58 | 114 |
| even | 58 | 21 |

Table 9. Top 20 common words along with their respective top 3 empath categories and empath category score from PASDF and NASDF

| Word | Empath categories in NASDF | Empath categories in PASDF |
|---|---|---|
| locate | device_battery (1.97), application (1.48), miscellaneous (1.42) | technology (0.87), application (0.51), exposure (0.49) |
| work | application (3.59), exposure (0.88), device_battery (0.83) | application (0.79), technology (0.36), exposure (0.24) |
| batteri | device_battery (12.80), application (1.63), technology (0.95) | device_battery (2.38), technology (0.57), coordinates (0.15) |
| updat | application (11.59), device_battery (1.72), exposure (0.60) | application (1.50), exposure (0.31), miscellaneous (0.17) |
| emerg | application (1.05), exposure (0.76), technology (0.68) | technology (1.73), miscellaneous (0.63), application (0.51) |

| **Word** | **Empath categories in NASDF** | **Empath categories in PASDF** |
|---|---|---|
| phone | device_battery (1.93), application (0.74), technology (0.67) | technology (0.55), device_battery (0.34), application (0.30) |
| good | device_battery (3.24), technology (2.08), exposure (1.20) | technology (2.56), device_battery (2.55), exposure (1.32) |
| use | device_battery (3.04), application (1.02), miscellaneous (0.66) | exposure (0.65), device_battery (0.56), technology (0.46) |
| notif | exposure (5.84), technology (0.46), device_battery (0.35) | exposure (1.36), technology (0.44), fire_emergency (0.11) |
| time | device_battery (1.18), exposure (0.76), coordinates (0.37) | technology (0.55), application (0.24), exposure (0.12) |
| even | device_battery (0.85), application (0.78), exposure (0.66) | application (0.38), exposure (0.37), technology (0.20) |
| great | application (0.95), technology (0.66), exposure (0.39) | technology (4.60), application (1.30), exposure (0.80) |
| applic | device_battery (1.67), exposure (0.70), miscellaneous (0.69) | device_battery (0.46), exposure (0.11), medical_emergency (0.10) |
| call | device_battery (0.57), application (0.47), miscellaneous (0.35) | technology (0.93), application (0.45), network (0.42) |

| Word | Empath categories in NASDF | Empath categories in PASDF |
|---|---|---|
| need | technology (1.71), application (1.37), exposure (0.30) | technology (3.23), exposure (0.53), application (0.41) |
| posit | coordinates (2.73), device_battery (1.49), technology (0.58) | coordinates (0.46), technology (0.36), exposure (0.25) |
| would | exposure (0.70), technology (0.54), device_battery (0.51) | technology (0.74), exposure (0.60), application (0.49) |
| open | application (2.31), exposure (0.49), technology (0.19) | technology (0.28), application (0.20), exposure (0.09) |
| number | application (0.32), exposure (0.28), miscellaneous (0.28) | technology (0.38), exposure (0.21), application (0.18) |
| address | network (0.65), application (0.49), technology (0.36) | technology (0.87), application (0.51), network (0.19) |

## 5.6. Ontology on Emergency Communication Apps

The ultimate aim of mobile emergency application ontology is to provide knowledge on the factors that are important for a mobile application and expand the factors to identify the variables of each factor. The factors are termed as classes whereas variables are called as objects. The classes and objects are considered with the help of other ontologies and the aspect terms extracted from the ATEPC task. The word association tree is created after checking the similarity between the classes and objects as mentioned in section 4.8. As discussed in section 4.8 previous ontologies are constructed majorly highlighting the emergency management process, Our study defines the semantic relations of words regarding mobile emergency applications.

The ontology word representation, similarity score, and relation between words are shown in Figure 10. We have considered 9 different classes namely app, coordinates, initiative, concept, network, accidents, fire, emergency, and function. In word representation, each class contains one or more objects to expand the class characteristics. Some objects share different classes and some objects act as a class for a small cluster. The relation is called by the 'subClass of' notation. For example, in Figure 30. 'Idea' is an object which is a subclass of 'Initiative' and 'Concept'. That means the term 'Idea' can be either used as a 'Concept' or 'Initiative'. And also 'Medical Emergency' object act as a class for a small group of objects such as 'Heart attack', 'Serious illness', 'Ambulance', and 'First aid'.

Apart from this, we also classified some terms as action-reaction pairs. Action-reaction pairs are pairs that cause mutual effect when changed, for example in Figure 30. When the font size is changed the user experiences a change in reading the text which can either be a positive or negative effect. This implies that the change in font size affects the reading experience for a user hence, 'Font size' and 'Read' objects are related as an action-reaction pair. Other relations such as 'synonyms' and 'similar context pairs' are highlighted using different colors in the word representation shown in Figure 30.

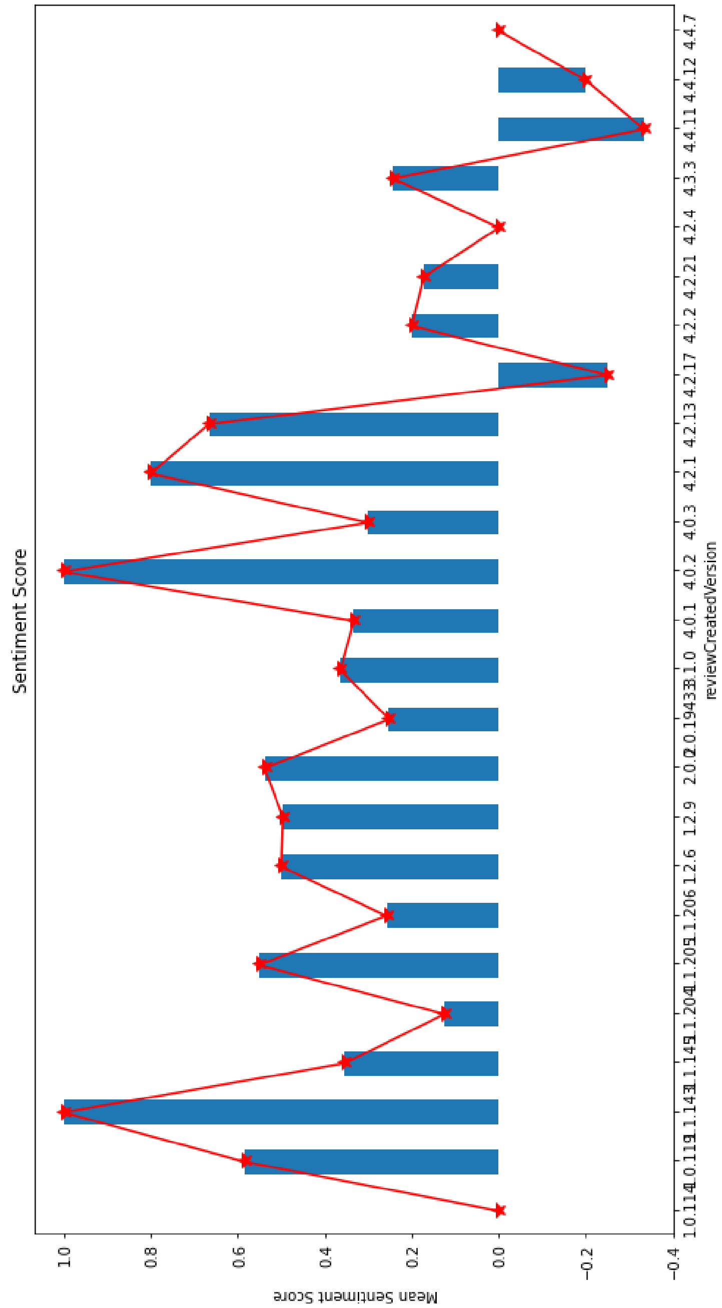


Figure 20. A combination chart of MSS for 112 Suomi app.

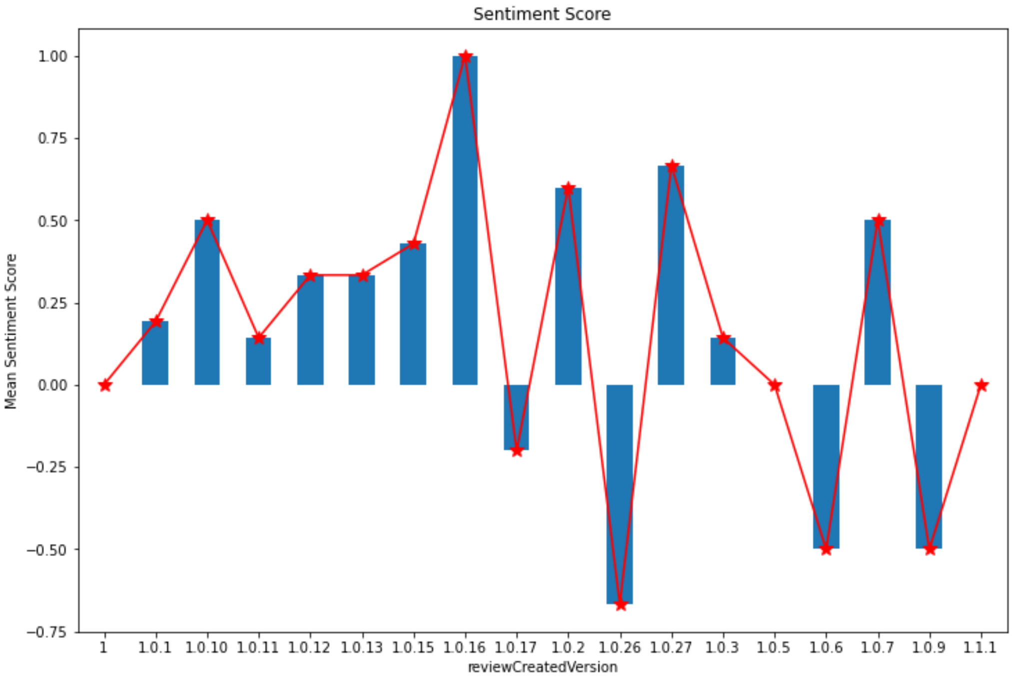


Figure 21. A combination chart of MSS for SOS Alarm app.

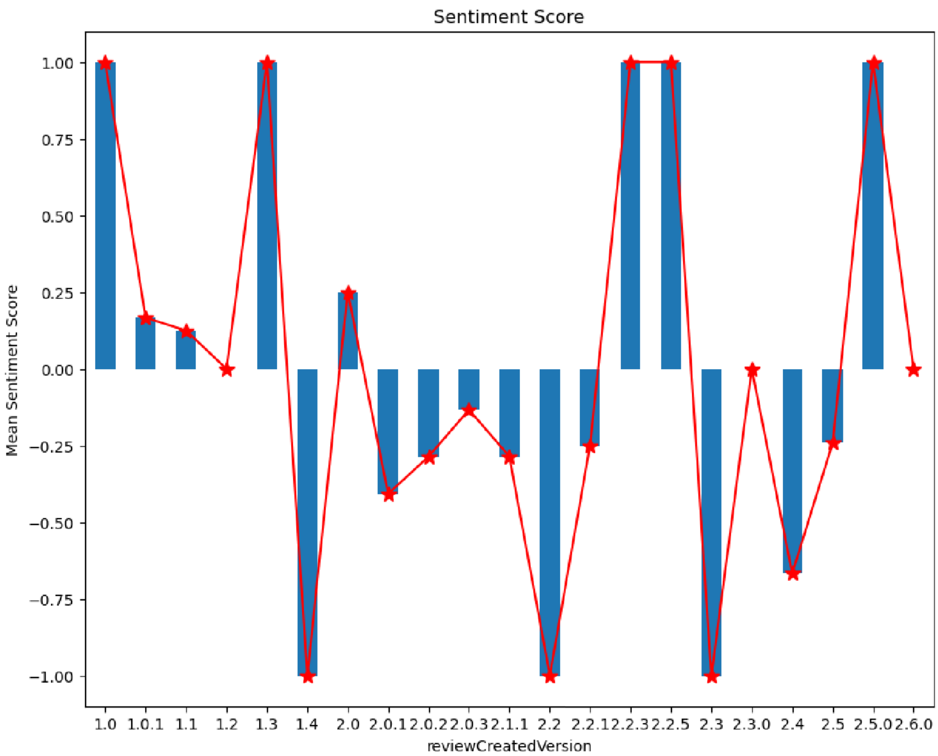


Figure 22. A combination chart of MSS for Hjelp 113 app.

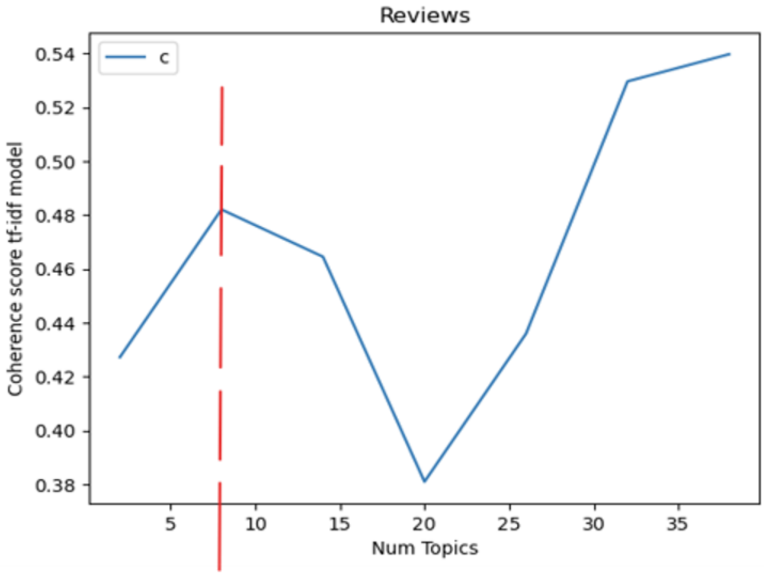


Figure 23. Coherence Scores for different values of $K$.

```
[(0,
  '0.034*"great" + 0.017*"app" + 0.012*"idea" + 0.012*"must" + 0.006*"work" + 0.006*"would" + 0.006*"live" + 0.006*"phone" + 0.
006*"find" + 0.005*"location"'),
 (1,
  '0.011*"need" + 0.010*"call" + 0.009*"app" + 0.009*"event" + 0.008*"help" + 0.007*"location" + 0.006*"phone" + 0.006*"informa
tion" + 0.006*"emergency" + 0.006*"get"'),
 (2,
  '0.031*"necessary" + 0.030*"useful" + 0.024*"safe" + 0.012*"really" + 0.011*"download" + 0.009*"brilliant" + 0.009*"security"
+ 0.009*"app" + 0.007*"good" + 0.007*"hopefully"'),
 (3,
  '0.074*"good" + 0.017*"battery" + 0.014*"app" + 0.011*"really" + 0.010*"application" + 0.009*"drain" + 0.008*"time" + 0.008
*"thing" + 0.007*"background" + 0.007*"would"'),
 (4,
  '0.024*"open" + 0.018*"update" + 0.013*"application" + 0.011*"notification" + 0.010*"work" + 0.010*"crash" + 0.009*"app" + 0.
008*"location" + 0.008*"turn" + 0.007*"fix"'),
 (5,
  '0.018*"emergency" + 0.015*"yet" + 0.013*"handy" + 0.012*"never" + 0.012*"hope" + 0.010*"recommend" + 0.010*"convenient" + 0.
010*"good" + 0.009*"need" + 0.008*"show"'),
 (6,
  '0.030*"work" + 0.017*"number" + 0.017*"phone" + 0.015*"awesome" + 0.011*"know" + 0.011*"call" + 0.010*"oneplus" + 0.010*"wel
l" + 0.008*"location" + 0.008*"save"'),
 (7,
  '0.030*"important" + 0.019*"excellent" + 0.016*"easy" + 0.011*"app" + 0.011*"happen" + 0.008*"nice" + 0.007*"say" + 0.006*"fa
st" + 0.006*"think" + 0.006*"really"')]
```

Figure 24. Weighted sum of words representation of topics created by LDA model.

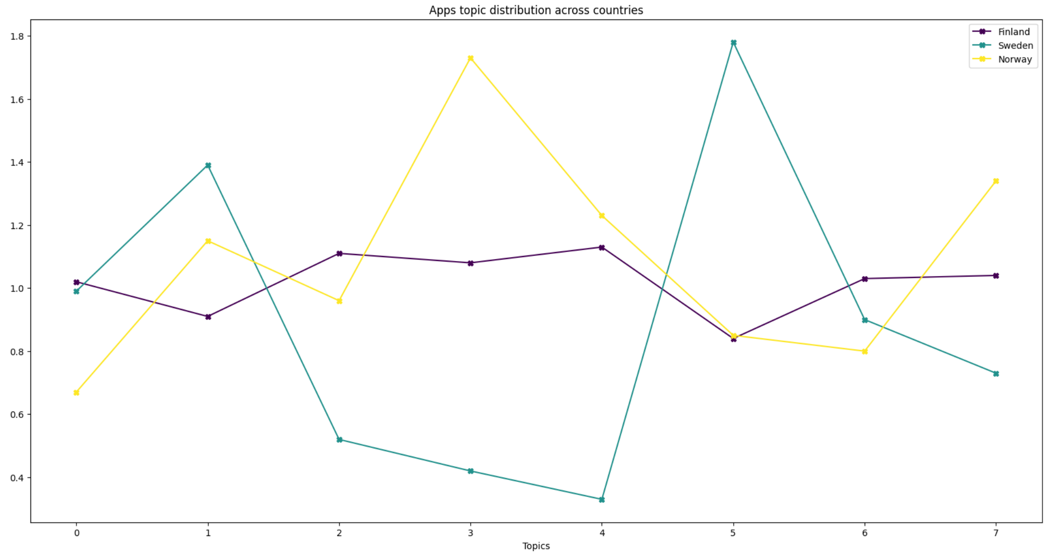


Figure 25. Distribution of various topics in the positive review corpus.

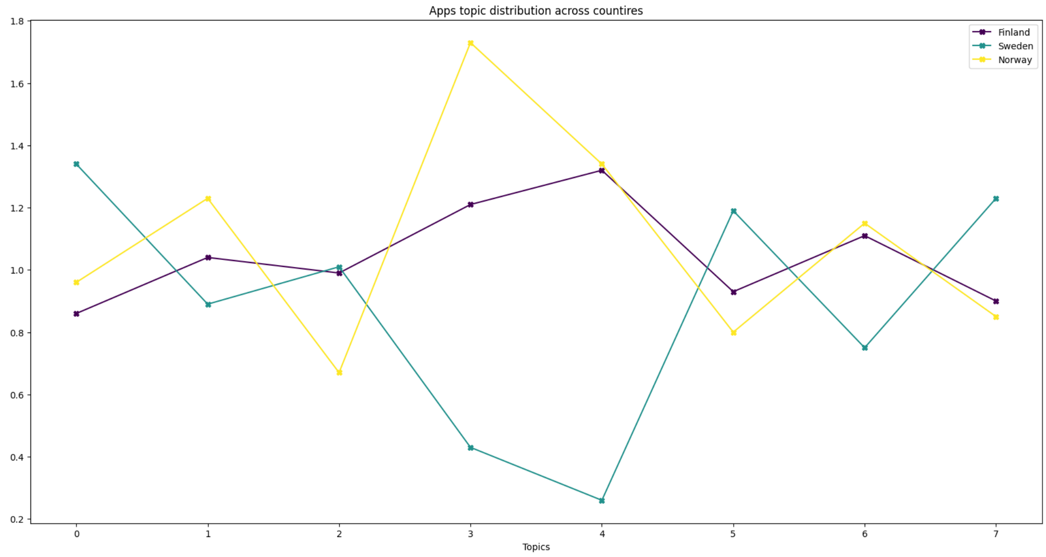


Figure 26. Distribution of various topics in the negative review corpus.

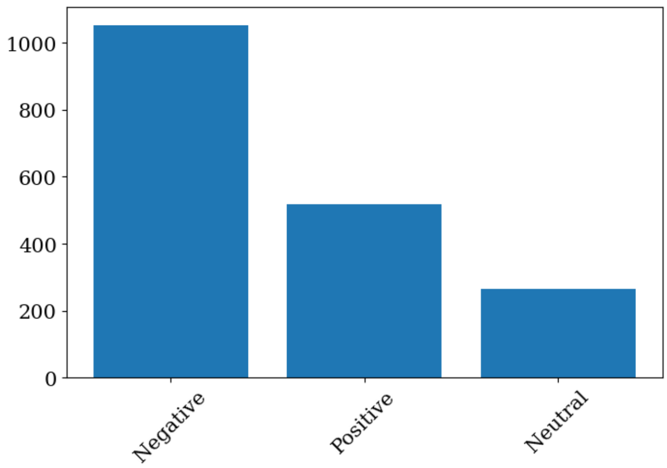


Figure 27. Frequency of each sentiment in the aspect term extraction task's output file.

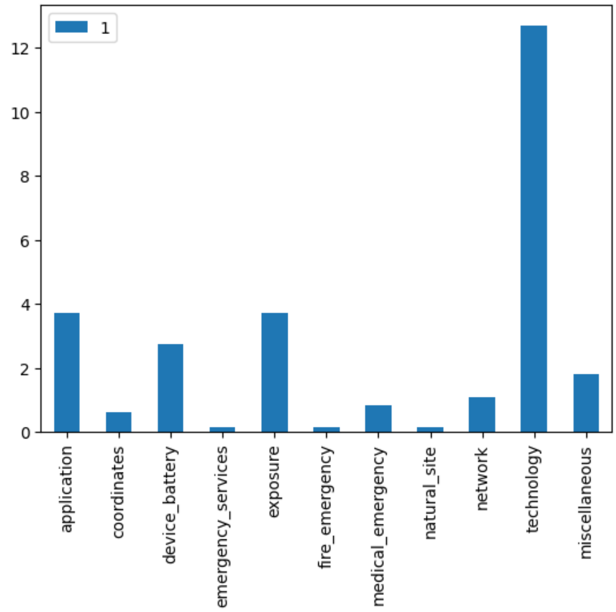


Figure 28. Frequency of empath categories in positive aspect sentiment data frame (PASDF).

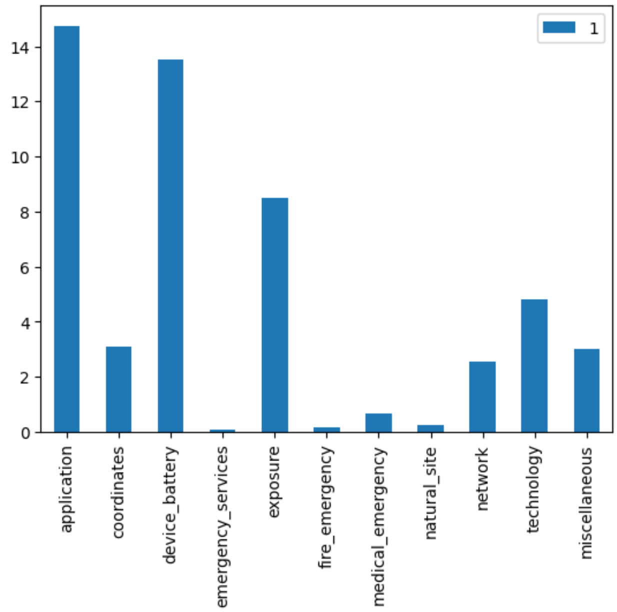


Figure 29. Frequency of empath categories in negative aspect sentiment data frame (NASDF).

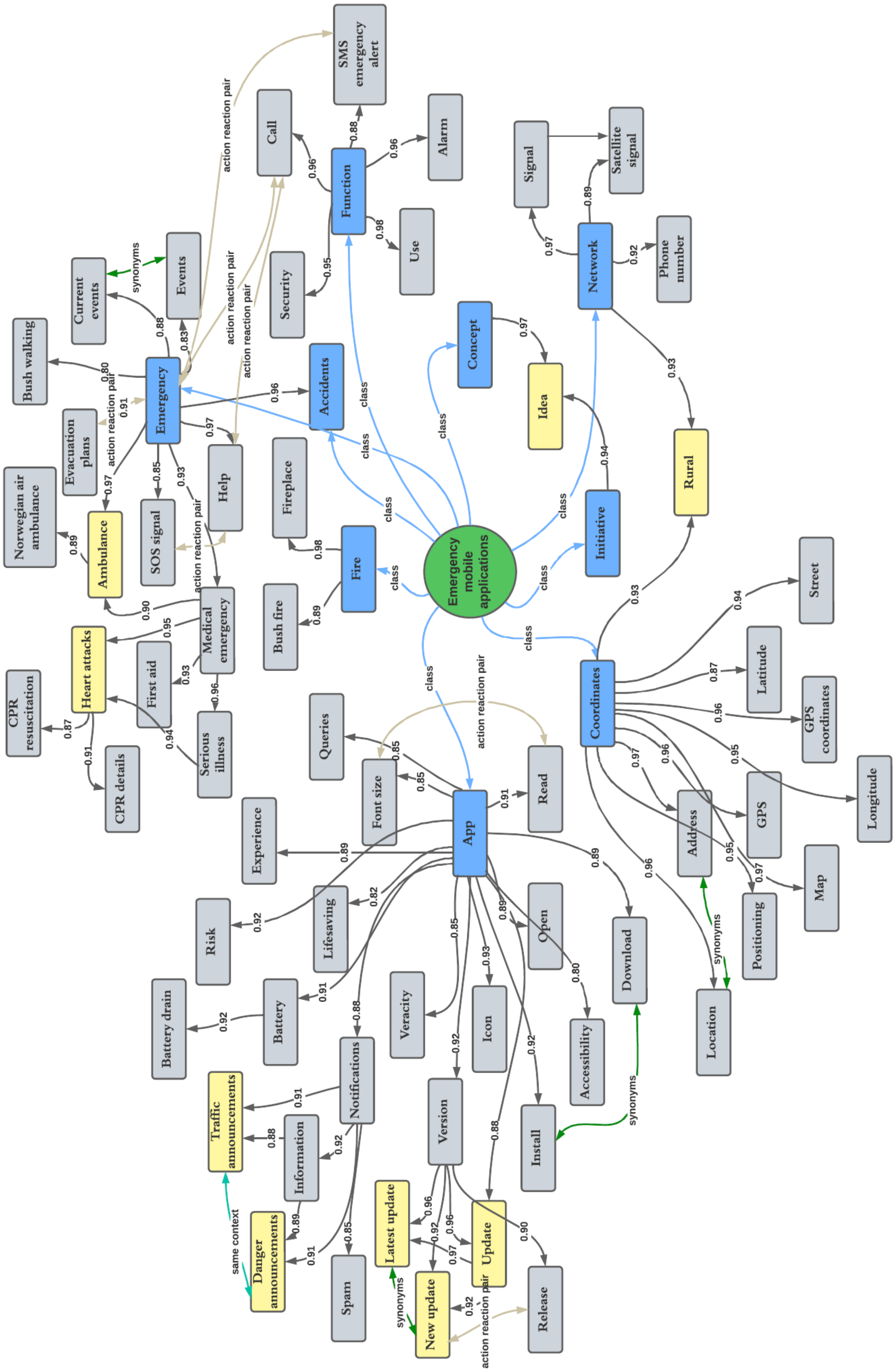


Figure 30. Word association tree representation.

# 6. DISCUSSION

The evolution of the Natural Language Programming models in order to extract the contextual information from the text has provided a larger scope in understanding the hidden information. This section point outs whether the implemented methodology has managed to achieved the desired goals mentioned in section 1.2.

Sentiment analysis tools such as VADER sentiment and TextBlob have helped in defining the sentence polarities. According to Min & Zulkarnain, [62] VADER sentiment tool has achieved better results than TextBlob in handling the sentiment polarities on social media textual data. In this study, I have leveraged the VADER sentiment tool to predict the polarity scores of each user review. The goal of the sentiment polarity prediction is to study the variations in user sentiments regarding each emergency application. The results are later plotted on a bar plot to see how the sentiments varied from version to version. Through this method, I aimed to answer, whether the users are satisfied with the functionalities and updates of emergency mobile applications or not. From section 5.2, we can observe that the users exhibit negative sentiments towards the recent versions of the 112 Suomi application, which makes the business or the state reconsider the functionality of the app. Apart from that, we have also noticed that the same negative sentiment from the users has been shown towards the recent versions of SOS Alarm and Hjelp 113 app. Overall, we can conclude that the users are ready to use these applications but there is a requirement of improving the functionality of these apps. From the distribution plots illustrated in 3.1, we can observe that the users in Finland are more willing to use emergency applications during emergencies. We can witness a spike in the Year 2020 for the Emergency Plus app, as shown in Figure 2. This high use of applications in the year can be due to the 2020 pandemic. We can also observe the spikes from the other application distribution plots in different years. These results helps to address the variation in user sentiment over different versions of each application. The relevant reason for the rapid use of emergency mobile applications can explained by analysis of the reviews yearly. In this thesis, we haven't discussed the year-wise analysis of user reviews. Such implementation can considered and further utilized in other studies to understand the reason behind the rapid usage of applications.

On top of the sentiment analysis, I have also performed some exploratory data analysis to identify the frequently occurring terms in the corpus. Word clouds can easily highlight the most repeated or mentioned term from a text corpus, in this study, these are used to quickly identify the most repetitive terms mentioned in the user reviews of the applications. The corpus is divided into two according to the sentiments of the reviews calculated using the VADER sentiment tool. A similar type of methodology has been implemented by Almjawel et al., [63] to identify the most common terms in different sentiments for Amazon books' reviews. Section 5.1 illustrates the word clouds. Even with the help of word clouds, we cannot validate the factors that cause the change in user sentiments. We can notice some terms such as location, program, application, battery, and GPS words mentioned in the positive word clouds. Notification, battery, and location can also be seen in the negative word clouds. By this, we can say that the user is exhibiting both positive and negative sentiments on notification, battery, locations, and some other terms as shown in respective word clouds. We can validate these results by comparing them with the various topics that can be defined by using topic modeling techniques.

To understand the factors that affect this change I employed the topic modeling technique described in section 5.3. Kwon et al., [64] employed sentiment analysis and topic modeling techniques to study the online user reviews of airlines by highlighting the factors that cause user dissatisfaction such as 'staff service', 'meals', and 'seat'. In this thesis, to identify which topics have been discussed or mentioned more in user reviews of different applications, we considered the Topic Score $TS$ metric defined in 4 and showed it in Figures 25 and 26. From those figures, we can observe that the 'Information transferability' topic in Sweden impacted users positively whereas users complained mostly about 'Application usability' negatively. In Finland, the users took a neutral stand mostly but showed slight adverse concern towards 'Information transferability' through emergency mobile applications. In Norway's case, we observe an almost equal number of users mentioned 'User satisfaction' both positively and negatively. This can be due to fewer data observations for Norway. Hence, we cannot conclude the topic's importance for Norway. The frequency of the dominant topic and topic weights are depicted in Appendix 8.2. With this topic modeling, we can infer the various themes that have been discussed in the reviews. These results provide an answer regarding the factors that can influence the change in user behaviour in a generic way. Apart from its advantages in highlighting the topics, it fails to extract much deeper information from the text. Farkhod et al. [18] leveraged LDA models to extract the topics from the corpus but also mentioned that the model fails to extract the implicit aspect terms from the text. Hence Yang et al. [39] employed ABSA tasks to extract the implicit words from the corpus using LCF and GCF models combined with the BERT. Inspired by the work I employed a similar model to extract user aspects from the emergency mobile applications reviews. Yanuar et al., [65] leveraged the Aspect-based Sentiment Analysis methodology to extract aspects from the tourist spot reviews dataset and achieved the best accuracy of 0.82.

Aspect term extract task by using the DeBERTa model instead of base-BERT is implemented in this thesis to extract aspects from the user reviews. Aspect Term Extraction task helps us to identify the user aspects from the reviews. From Figure 27, we can observe that the corpus has more negative aspect sentiments than positive and neutral. With the help of sentiment rate $SR_a$ metrics defined in the section 5.4, we can observe that mobile emergency applications are impacting 'battery' usage negatively, 'location' accuracy is questioned by the users, and users are not satisfied with the updates of the applications. On a positive note, the 'applications' have been performing well in informing the relevant information such as traffic announcements, current events, emergency calls, and security. The line plot of these results can be seen under Appendix 8.3. These results has provided an answer regarding the important aspect terms like the results from topic modeling which gave a general overview of user specific theme analysis. These results can help in constructing a better emergency communication ecosystem using mobile applications which can end up saving a huge expenditure and the lives of many people as mentioned in the example given in section 1.1 regarding timely cyclone alerts in Bangladesh. Once the aspect terms had been extracted along with their respective polarities, inspired by Hoang et al., [37] we employed a methodology to perform a sentence pair classification task which takes the user review and aspect as input and predicts the sentiment of the given aspect associated with the user review. Hoang et al., [37] implemented a sentence pair classification architecture on the SemEval-2016 dataset containing reviews along with entity and aspect pairs. In this thesis, only the aspect of user review and review text is considered. This model can help define the sentiments of each aspect of test dataset reviews. Later on, we understand the user empath, empath categorization methodology mentioned in section 4.7 is used. Firstly, a dataset containing reviews with positive sentiment aspects, and a second dataset containing reviews with negative sentiment aspects are created. The empath categorization methodology is inspired by Arhab et al., [59] in which they employed a pipeline to categorize the user tweets empath on car parking data. The aim of this analysis is to identify the sentimental variations in user behavior through user-defined empath categories.

From the empath categorization analysis results shown in Figures 29 and 28, the 'Application' empath category highest score in negative sentiment datasets followed by the 'Device Battery' and 'Exposure' empath categories. Empath categories application, device battery, and exposure account for an empath score of 14.73, 13.50, and 8.49 respectively. In the positive dataset, the 'Technology' empath category considers the highest empath score of 12.69. The application, device battery, and exposure contain an empath score of less than 4 in a positive data frame. Hence the users are ready to adopt the technology of mobile emergency applications. To improve the quality and user satisfaction of these apps, the user interface, functionalities, background usage which leads to battery drain, and relevant information providing utilities need to be reassessed. The common words identified from positive and negative sentiment corpus which are mentioned in Table 8 can provide an overview of the functionalities that a mobile emergency application should consist of. From the frequency of the terms mentioned in Table 8, we can conclude that the user exhibits more negative sentiment toward tracking location, the app's working, battery usage, and updates.

To find a relation between these terms and how these terms impact each other we constructed a word association representation by referring to some of the existing ontologies mentioned in section 2.4 and the aspect terms extracted. For example, the negative sentiment towards location and battery is because to access location 24/7 the user needs to run the app in the background which results in consuming more power. Hence the user sentiment towards these terms is negative. Similarly, the relation between other terms is also formulated. These insights help the developers to understand the action-reaction pairs. The results obtained from the analysis correlate with the existing ontologies as mentioned in section 2.4. Apart from that, the results of this study can also provide a comprehensive overview of how users react during an emergency and the likeliness of using mobile emergency applications for emergency purposes.

# 7. CONCLUSION AND FURTHER STUDY

In conclusion, this study was able to achieve the goal of highlighting the factors that are important to construct a efficient emergency communication ecosystem through emergency mobile applications. The use of these applications has been increased over the years and people are showing interest in leveraging this technology during emergencies. This statement can be validated by considering the results depicted in chapter 5. By taking this into consideration local governments or technical firms can focus on developing either existing or new emergency mobile applications which can be a life saving options during crisis. The methods implemented in this study which are elaborated in the chapter 4 helped in achieving the goals of study to address pros and cons on using these applications during emergencies in user perspective, some of them have mentioned in chapter 6.

For future study, a similar research can be carried out on other microblogging app data such as Twitter, Reddit, Threads, etc. Since the user activity on these social media platforms is high compared to the user activity in the Play Store and App Store, we can extract much information from the vast pile of data. Since this study has limited data availability, it has compromised in providing a better solution for identifying the aspect terms and analyzing their respective user sentiments. Apart from that, in this study a complete ontological representation is not explored. But suggests the ontological vocabulary terms related to Emergency Mobile applications which can act as a guiding manual to access which factors impact on which aspects of mobile applications. To add authenticity to our outcome, we could have considered the user behavior models, for instance, the Technology Acceptance Model (TAM), UTAUT, and UTAUT2 models, in order to perform a short study through questionnaires and compare the outcomes of both user behaviour analysis and ontological voabulary. The sole objective of this thesis is to leverage NLP and text mining techniques to highlight the problems encountered by users while using emergency mobile applications during an emergency.

On the other note, gaming app industry is one such domain which can generate huge revenue. According to Statista, mobile applications in the gaming sector can generate a revenue of 352.1 million U.S. dollars within the 2019 to 2017 time periods. This industry holds a significant potential to improve the user experience and provide the customers with a better gaming experience one can leverage the insights obtained by following this study. For example, a similar methodology can be followed to identify user behavior and their aspects regarding gaming applications. Finally, this study leverage NLP techniques in order to answer the research questions such as how the user behaviour changed over the period regarding emergency mobile applciations, the factors that can influence the user experince of these applications, and how various user empaths differ from negative and positive aspect reviews.

# REFERENCES


[1] N. D. Oye, N. A. Iahad, and N. Ab. Rahim. "The history of UTAUT model and its impact on ICT acceptance and usage by academicians". In: *Education and Information Technologies* 19 (2014), pp. 251–270.

[2] A. T. Jimoh. "Students' attitude towards ICT in Nigeria tertiary institutions". In: *Education Focus* 1.1 (2007), pp. 73–79.

[3] W. Bannour, A. Maalel, and B. H. Ghezala. "Ontology-Based Representation of Crisis Response Situations". In: (Aug. 2019), pp. 417–427. DOI: 10.1007/978-3-030-28374-2_36.

[4] R. Repanovici et al. "Improvement of Emergency Situation Management through an Integrated System Using Mobile Alerts". In: *Sustainability* 14.24 (2022), p. 16424.

[5] R. Repanovici and A. Nedelcu. "Mobile emergency notification apps: Current state, barriers and future potential". In: 1009.1 (2021), p. 012049.

[6] P. Örtenwall, O. Almgren, and E. Deverell. "The bomb explosion in Myyrmanni, Finland 2002". In: *International Journal of Disaster Medicine* 1.2 (2003), pp. 120–126.

[7] D. Halliwell and C. Lumbreras. "Mobile Identity Platform for the Emergency Services". In: *EENA Operations Document* (2018).

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

[9] M. Pontiki et al. "Semeval-2016 task 5: Aspect based sentiment analysis". In: (2016), pp. 19–30.

[10] J. S. Dodgson et al. "Multi-Criteria Analysis: A Manual". In: (Jan. 2009).

[11] M. L. Tan et al. "Mobile applications in crisis informatics literature: A systematic review". In: *International journal of disaster risk reduction* 24 (2017), pp. 297–311.

[12] V. Bonta, N. Kumaresh, and N. Janardhan. "A comprehensive study on lexicon based approaches for sentiment analysis". In: *Asian Journal of Computer Science and Technology* 8.S2 (2019), pp. 1–6.

[13] Z. Nanli et al. "Sentiment analysis: A literature review". In: (2012), pp. 572–576.

[14] C. Hutto and E. Gilbert. "Vader: A parsimonious rule-based model for sentiment analysis of social media text". In: 8.1 (2014), pp. 216–225.

[15] S. Elbagir and J. Yang. "Twitter sentiment analysis using natural language toolkit and VADER sentiment". In: 122 (2019), p. 16.

[16] D. M. Blei. "Probabilistic topic models". In: *Communications of the ACM* 55.4 (2012), pp. 77–84.

[17] D. Ramamonjisoa. "Topic modeling on users's comments". In: (2014), pp. 177–180.

[18] A. Farkhod et al. “LDA-based topic modeling sentiment analysis using topic/document/sentence (TDS) model”. In: *Applied Sciences* 11.23 (2021), p. 11091.

[19] H. Nguyen and D. Hovy. “Hey Siri. Ok Google. Alexa: A topic modeling of user reviews for smart speakers”. In: (2019), pp. 76–83.

[20] M. Venugopalan and D. Gupta. “An enhanced guided LDA model augmented with BERT based semantic strength for aspect term extraction in sentiment analysis”. In: *Knowledge-based systems* 246 (2022), p. 108668.

[21] I. Titov and R. McDonald. “Modeling Online Reviews with Multi-Grain Topic Models”. In: (2008). DOI: 10.1145/1367497.1367513. URL: https://doi.org/10.1145/1367497.1367513.

[22] S. Brody and N. Elhadad. “An unsupervised aspect-sentiment model for online reviews”. In: (2010), pp. 804–812.

[23] M. Hu and B. Liu. “Mining and summarizing customer reviews”. In: (2004), pp. 168–177.

[24] O. Irsoy and C. Cardie. “Bidirectional recursive neural networks for token-level labeling with structure”. In: *arXiv preprint arXiv:1312.0493* (2013).

[25] H. Luo et al. “Improving aspect term extraction with bidirectional dependency tree representation”. In: *IEEE/ACM Transactions on Audio, Speech, and Language Processing* 27.7 (2019), pp. 1201–1212.

[26] P. Liu, S. Joty, and H. Meng. “Fine-grained opinion mining with recurrent neural networks and word embeddings”. In: (2015), pp. 1433–1443.

[27] D. K. Kirange, R. R. Deshmukh, and M. D. K. Kirange. “Aspect based sentiment analysis semeval-2014 task 4”. In: *Asian Journal of Computer Science and Information Technology (AJCSIT) Vol* 4 (2014).

[28] B. Liu. “Web data mining: exploring hyperlinks, contents, and usage data”. In: 1 (2006).

[29] L. Zhang and B. Liu. “Aspect and entity extraction for opinion mining”. In: (2014), pp. 1–40.

[30] M. Pontiki et al. “Semeval-2015 task 12: Aspect based sentiment analysis”. In: (2015), pp. 486–495.

[31] S. Polamuri. “Five most popular similarity measures implementation in python”. In: *Internet: http://dataaspirant. com/2015/04/11/five-most-popular-similarity-measuresimplementation-in-python* (2015).

[32] S. Poria et al. “A rule-based approach to aspect extraction from product reviews”. In: (2014), pp. 28–37.

[33] A. Wu et al. “Wildfire evacuation decision modeling using GPS data”. In: *International Journal of Disaster Risk Reduction* 83 (2022), p. 103373.

[34] A. Mukherjee and B. Liu. “Aspect extraction through semi-supervised modeling”. In: (2012), pp. 339–348.

[35] R. He et al. “An unsupervised neural attention model for aspect extraction”. In: (2017), pp. 388–397.

[36] B. Viver. “Emergency Communications and the EU Legislative Framework.” In: *EENA Operations Document* (2019).

[37] M. Hoang, A. O. Bihorac, and J. Rouces. “Aspect based sentiment analysis using BERT Proceedings of the 22nd Nordic Conference on Computational Linguistics, Linkoping University Electronic Press”. In: (2019).

[38] J. Devlin et al. “BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding”. In: (Oct. 2018).

[39] H. Yang et al. “A multi-task learning model for chinese-oriented aspect polarity classification and aspect term extraction”. In: *Neurocomputing* 419 (2021), pp. 344–356.

[40] E. Kontopoulos et al. “Ontology-based Representation of Crisis Management Procedures for Climate Events.” In: (2018).

[41] K. Bitencourt et al. “An Ontological Model for Fire Emergency Situations”. In: *IEICE Transactions on Information and Systems* E101.D (Jan. 2018), pp. 108–115. DOI: 10.1587/transinf.2017SWP0003.

[42] P. D. Haghighi et al. “Development and evaluation of ontology for intelligent decision support in medical emergency management for mass gatherings”. In: *Decision Support Systems* 54.2 (2013), pp. 1192–1204.

[43] G. Babitski et al. “SoKNOS – Using Semantic Technologies in Disaster Management Software”. In: (May 2011), pp. 183–197. DOI: 10.1007/978-3-642-21064-8_13.

[44] H. Baytiyeh. “The Uses of Mobile Technologies in the aftermath of Terrorist Attacks among Low Socioeconomic Populations”. In: *International Journal of Disaster Risk Reduction* 28 (Feb. 2018). DOI: 10.1016/j.ijdrr.2018.02.001.

[45] A. Braham et al. “User interface design patterns and ontology models for adaptive mobile applications”. In: *Personal and Ubiquitous Computing* 26 (Jan. 2021), pp. 1–17. DOI: 10.1007/s00779-020-01481-5.

[46] F. A. Norki, R. Mohamad, and N. Ibrahim. “Context ontology in mobile applications”. In: *Journal of Information and Communication Technology* 19.1 (2020), pp. 21–44.

[47] M. Poveda Villalon et al. “A context ontology for mobile environments”. In: (2010).

[48] X. H. Wang et al. “Ontology based context modeling and reasoning using OWL”. In: (2004), pp. 18–22.

[50] H. Yang and K. Li. “PyABSA: open framework for aspect-based sentiment analysis”. In: *arXiv preprint arXiv:2208.01368* (2022).

[51] T. Mikolov et al. “Efficient estimation of word representations in vector space”. In: *arXiv preprint arXiv:1301.3781* (2013a).

[52] T. Mikolov et al. “Distributed representations of words and phrases and their compositionality”. In: *Advances in neural information processing systems* 26 (2013b).

[53] Y. Goldberg and O. Levy. "word2vec Explained: deriving Mikolov et al.'s negative-sampling word-embedding method". In: *arXiv preprint arXiv:1402.3722* (2014).

[54] T. Mikolov et al. "Computing numeric representations of words in a high-dimensional space". In: (May 2015). US Patent 9,037,464.

[55] J. Pennington, R. Socher, and C. D. Manning. "Glove: Global vectors for word representation". In: (2014), pp. 1532–1543.

[56] B. Ghojogh and A. Ghodsi. "Attention mechanism, transformers, BERT, and GPT: tutorial and survey". In: (2020).

[57] P. He et al. "Deberta: Decoding-enhanced bert with disentangled attention". In: *arXiv preprint arXiv:2006.03654* (2020).

[58] Y. Song et al. "Attentional encoder network for targeted sentiment classification". In: *arXiv preprint arXiv:1902.09314* (2019).

[59] N. Arhab, M. Oussalah, and Y. Bounab. "Analysis of user's car parking behaviour through twitter hashtags". In: *Expert Systems* 40 (Dec. 2022). DOI: `10.1111/exsy.13218`.

[60] Y. Liu et al. "Roberta: A robustly optimized bert pretraining approach". In: *arXiv preprint arXiv:1907.11692* (2019).

[61] Z. Lan et al. "Albert: A lite bert for self-supervised learning of language representations". In: *arXiv preprint arXiv:1909.11942* (2019).

[62] W. N. S. W. Min and N. Z. Zulkarnain. "Comparative evaluation of lexicons in performing sentiment analysis". In: *Journal of Advanced Computing Technology and Application (JACTA)* 2.1 (2020), pp. 14–20.

[63] A. Almjawel et al. "Sentiment analysis and visualization of amazon books' reviews". In: *2019 2nd International Conference on Computer Applications & Information Security (ICCAIS)*. IEEE. 2019, pp. 1–6.

[64] H. J. Kwon et al. "Topic modeling and sentiment analysis of online review for airlines". In: *Information* 12.2 (2021), p. 78.

[65] M. R. Yanuar and S. Shiramatsu. "Aspect extraction for tourist spot review in Indonesian language using BERT". In: *2020 International Conference on Artificial Intelligence in Information and Communication (ICAIIC)*. IEEE. 2020, pp. 298–302.

[66] H. Schütze, C. D. Manning, and P. Raghavan. *Introduction to information retrieval*. Vol. 39. Cambridge University Press Cambridge, 2008.

# 8. APPENDICES

| | |
|---|---|
| Appendix A | Zipf law tables |
| Appendix B | Topic modeling visualizations |
| Appendix C | Top 30 aspect terms illustration |
| Appendix D | Source code[3] |

[3]Github: `https://github.com/FuzelahamedShaik/EmergencyCommunication.git`

## 8.1. Appendix A

Zipf's law was first introduced by linguist George Zipf in the mid-20th century. It is defined as an empirical regularity that characterizes the distribution of word frequencies within a given corpus. This occurrence has substantial implications for Natural Language Processing (NLP), providing important fresh perspectives on the fundamental structure and characteristics of natural language.

A power-law distribution or a rank-size distribution are other names for Zipf's Law. Its core tenet is that a word's frequency is inversely correlated with its position in the frequency table. Zipf's Law can be defined mathematically as:

$$f(w) \propto \frac{1}{r(w)^s} \tag{10}$$

In eq(10), $f(w)$ denotes the frequency of word $w$, $r(w)$ denotes the rank of word $w$ within the frequency table, and $s$ stands for the Zipf exponent ($s \approx 1$).

Understanding Zipf's Law makes it easier to spot high-frequency words with little semantic meaning, sometimes known as "stop words." This knowledge is helpful for preprocessing activities like text summarization since it allows for the selective omission of certain frequent terms [66]. In our study, we employed this method to highlight the frequently occurring terms in different datasets. The illustrations of the Zipf tables for different datasets are shown in Figures 31, 32, 33, 34, 35, and 36.

| Rank | Word | Actual Freq | Zipf Frac | Zipf Freq | Actual Diff | Pct Diff |
|---|---|---|---|---|---|---|
| 1 | app | 281 | 1/1 | 281.00 | 0.00 | 100.00% |
| 2 | great | 112 | 1/2 | 140.50 | -28.50 | 79.72% |
| 3 | location | 91 | 1/3 | 93.67 | -2.67 | 97.15% |
| 4 | emergency | 71 | 1/4 | 70.25 | 0.75 | 101.07% |
| 5 | would | 70 | 1/5 | 56.20 | 13.80 | 124.56% |
| 6 | phone | 69 | 1/6 | 46.83 | 22.17 | 147.33% |
| 7 | good | 68 | 1/7 | 40.14 | 27.86 | 169.40% |
| 8 | use | 64 | 1/8 | 35.12 | 28.88 | 182.21% |
| 9 | need | 59 | 1/9 | 31.22 | 27.78 | 188.97% |
| 10 | work | 52 | 1/10 | 28.10 | 23.90 | 185.05% |
| 11 | gps | 42 | 1/11 | 25.55 | 16.45 | 164.41% |
| 12 | address | 39 | 1/12 | 23.42 | 15.58 | 166.55% |
| 13 | idea | 38 | 1/13 | 21.62 | 16.38 | 175.80% |
| 14 | like | 34 | 1/14 | 20.07 | 13.93 | 169.40% |
| 15 | open | 34 | 1/15 | 18.73 | 15.27 | 181.49% |
| 16 | service | 31 | 1/16 | 17.56 | 13.44 | 176.51% |
| 17 | could | 30 | 1/17 | 16.53 | 13.47 | 181.49% |
| 18 | hope | 29 | 1/18 | 15.61 | 13.39 | 185.77% |
| 19 | one | 28 | 1/19 | 14.79 | 13.21 | 189.32% |
| 20 | useful | 27 | 1/20 | 14.05 | 12.95 | 192.17% |

Figure 31. Most frequent words from positive review corpus of Emergency Plus app after applying Zipf law.

## 8.2. Appendix B

This section shows more visualizations regarding topics and significant terms for each topic. Figures 37 and 38 represent the dominance of topics in whole documents and the importance of topic keywords in the corpus.

| Rank | Word | Actual Freq | Zipf Frac | Zipf Freq | Actual Diff | Pct Diff |
|---|---|---|---|---|---|---|
| 1 | app | 140 | 1/1 | 140.00 | 0.00 | 100.00% |
| 2 | emergency | 84 | 1/2 | 70.00 | 14.00 | 120.00% |
| 3 | location | 57 | 1/3 | 46.67 | 10.33 | 122.14% |
| 4 | address | 53 | 1/4 | 35.00 | 18.00 | 151.43% |
| 5 | gps | 41 | 1/5 | 28.00 | 13.00 | 146.43% |
| 6 | wrong | 35 | 1/6 | 23.33 | 11.67 | 150.00% |
| 7 | phone | 35 | 1/7 | 20.00 | 15.00 | 175.00% |
| 8 | need | 30 | 1/8 | 17.50 | 12.50 | 171.43% |
| 9 | number | 29 | 1/9 | 15.56 | 13.44 | 186.43% |
| 10 | work | 28 | 1/10 | 14.00 | 14.00 | 200.00% |
| 11 | open | 27 | 1/11 | 12.73 | 14.27 | 212.14% |
| 12 | use | 26 | 1/12 | 11.67 | 14.33 | 222.86% |
| 13 | update | 23 | 1/13 | 10.77 | 12.23 | 213.57% |
| 14 | street | 21 | 1/14 | 10.00 | 11.00 | 210.00% |
| 15 | service | 21 | 1/15 | 9.33 | 11.67 | 225.00% |
| 16 | coordinate | 19 | 1/16 | 8.75 | 10.25 | 217.14% |
| 17 | useless | 18 | 1/17 | 8.24 | 9.76 | 218.57% |
| 18 | give | 17 | 1/18 | 7.78 | 9.22 | 218.57% |
| 19 | would | 17 | 1/19 | 7.37 | 9.63 | 230.71% |
| 20 | still | 17 | 1/20 | 7.00 | 10.00 | 242.86% |

Figure 32. Most frequent words from negative review corpus of Emergency Plus app after applying Zipf law.

| Rank | Word | Actual Freq | Zipf Frac | Zipf Freq | Actual Diff | Pct Diff |
|---|---|---|---|---|---|---|
| 1 | good | 235 | 1/1 | 235.00 | 0.00 | 100.00% |
| 2 | application | 102 | 1/2 | 117.50 | -15.50 | 86.81% |
| 3 | app | 87 | 1/3 | 78.33 | 8.67 | 111.06% |
| 4 | phone | 86 | 1/4 | 58.75 | 27.25 | 146.38% |
| 5 | really | 78 | 1/5 | 47.00 | 31.00 | 165.96% |
| 6 | number | 67 | 1/6 | 39.17 | 27.83 | 171.06% |
| 7 | important | 51 | 1/7 | 33.57 | 17.43 | 151.91% |
| 8 | call | 48 | 1/8 | 29.38 | 18.62 | 163.40% |
| 9 | use | 47 | 1/9 | 26.11 | 20.89 | 180.00% |
| 10 | help | 46 | 1/10 | 23.50 | 22.50 | 195.74% |
| 11 | emergency | 45 | 1/11 | 21.36 | 23.64 | 210.64% |
| 12 | work | 44 | 1/12 | 19.58 | 24.42 | 224.68% |
| 13 | even | 42 | 1/13 | 18.08 | 23.92 | 232.34% |
| 14 | great | 41 | 1/14 | 16.79 | 24.21 | 244.26% |
| 15 | location | 40 | 1/15 | 15.67 | 24.33 | 255.32% |
| 16 | useful | 34 | 1/16 | 14.69 | 19.31 | 231.49% |
| 17 | need | 34 | 1/17 | 13.82 | 20.18 | 245.96% |
| 18 | time | 33 | 1/18 | 13.06 | 19.94 | 252.77% |
| 19 | know | 33 | 1/19 | 12.37 | 20.63 | 266.81% |
| 20 | dont | 31 | 1/20 | 11.75 | 19.25 | 263.83% |

Figure 33. Most frequent words from positive review corpus of 112 Suomi app after applying Zipf law.

## 8.3. Appendix C

After extracting the aspects from the reviews we highlighted the top 30 frequently repeated aspect terms in the corpus which have been displayed in Figure 39. The aspect terms are app, location, battery, update, idea, emergency, GPS, address, notifications, position, coordinates, concept, map, open, fire, information, application, use, latest update, network, GPS coordinates, current events, phone number, events, positioning, emergencies, icon, call, security, alarm, and notification. These aspects terms can be considered while building an emergency application and also validate the performance of the emergency mobile applications according to each aspect.

The line plot drawn after calculating the sentiment rate of an aspect $SR_a$ is shown in Figure 40. 'App' aspect has created a good impression among the user reviews with high positive $SR_a$ whereas the 'location', 'battery', and 'update' aspects have observed negative sentiments among the user reviews in which the 'battery' aspect displayed more negative $SR_a$ than others, finally, the remaining aspects have mostly

| Rank | Word | Actual Freq | Zipf Frac | Zipf Freq | Actual Diff | Pct Diff |
|---|---|---|---|---|---|---|
| 1 | emergency | 88 | 1/1 | 88.00 | 0.00 | 100.00% |
| 2 | application | 87 | 1/2 | 44.00 | 43.00 | 197.73% |
| 3 | phone | 72 | 1/3 | 29.33 | 42.67 | 245.45% |
| 4 | location | 61 | 1/4 | 22.00 | 39.00 | 277.27% |
| 5 | number | 50 | 1/5 | 17.60 | 32.40 | 284.09% |
| 6 | call | 48 | 1/6 | 14.67 | 33.33 | 327.27% |
| 7 | time | 37 | 1/7 | 12.57 | 24.43 | 294.32% |
| 8 | app | 34 | 1/8 | 11.00 | 23.00 | 309.09% |
| 9 | center | 32 | 1/9 | 9.78 | 22.22 | 327.27% |
| 10 | even | 30 | 1/10 | 8.80 | 21.20 | 340.91% |
| 11 | good | 29 | 1/11 | 8.00 | 21.00 | 362.50% |
| 12 | would | 25 | 1/12 | 7.33 | 17.67 | 340.91% |
| 13 | work | 23 | 1/13 | 6.77 | 16.23 | 339.77% |
| 14 | turn | 21 | 1/14 | 6.29 | 14.71 | 334.09% |
| 15 | doesnt | 21 | 1/15 | 5.87 | 15.13 | 357.95% |
| 16 | program | 20 | 1/16 | 5.50 | 14.50 | 363.64% |
| 17 | dont | 19 | 1/17 | 5.18 | 13.82 | 367.05% |
| 18 | information | 19 | 1/18 | 4.89 | 14.11 | 388.64% |
| 19 | really | 18 | 1/19 | 4.63 | 13.37 | 388.64% |
| 20 | notification | 18 | 1/20 | 4.40 | 13.60 | 409.09% |

Figure 34. Most frequent words from negative review corpus of 112 Suomi app after applying Zipf law.

| Rank | Word | Actual Freq | Zipf Frac | Zipf Freq | Actual Diff | Pct Diff |
|---|---|---|---|---|---|---|
| 1 | app | 57 | 1/1 | 57.00 | 0.00 | 100.00% |
| 2 | good | 28 | 1/2 | 28.50 | -0.50 | 98.25% |
| 3 | gps | 28 | 1/3 | 19.00 | 9.00 | 147.37% |
| 4 | notification | 22 | 1/4 | 14.25 | 7.75 | 154.39% |
| 5 | battery | 16 | 1/5 | 11.40 | 4.60 | 140.35% |
| 6 | use | 12 | 1/6 | 9.50 | 2.50 | 126.32% |
| 7 | time | 11 | 1/7 | 8.14 | 2.86 | 135.09% |
| 8 | constantly | 10 | 1/8 | 7.12 | 2.88 | 140.35% |
| 9 | number | 10 | 1/9 | 6.33 | 3.67 | 157.89% |
| 10 | would | 9 | 1/10 | 5.70 | 3.30 | 157.89% |
| 11 | get | 9 | 1/11 | 5.18 | 3.82 | 173.68% |
| 12 | idea | 9 | 1/12 | 4.75 | 4.25 | 189.47% |
| 13 | phone | 8 | 1/13 | 4.38 | 3.62 | 182.46% |
| 14 | position | 8 | 1/14 | 4.07 | 3.93 | 196.49% |
| 15 | like | 7 | 1/15 | 3.80 | 3.20 | 184.21% |
| 16 | information | 7 | 1/16 | 3.56 | 3.44 | 196.49% |
| 17 | turn | 7 | 1/17 | 3.35 | 3.65 | 208.77% |
| 18 | want | 7 | 1/18 | 3.17 | 3.83 | 221.05% |
| 19 | drain | 6 | 1/19 | 3.00 | 3.00 | 200.00% |
| 20 | hour | 6 | 1/20 | 2.85 | 3.15 | 210.53% |

Figure 35. Most frequent words from positive review corpus of SOS Alarm app after applying Zipf law.

shown neutral $SR_a$ are the major insights from this graphical representation shown in Figures 39 and 40. To understand further the sentiment of the particular aspects we performed separate analyses on positive and negative sentiment datasets respectively.

| Rank | Word | Actual Freq | Zipf Frac | Zipf Freq | Actual Diff | Pct Diff |
|---|---|---|---|---|---|---|
| 1 | app | 38 | 1/1 | 38.00 | 0.00 | 100.00% |
| 2 | gps | 15 | 1/2 | 19.00 | -4.00 | 78.95% |
| 3 | position | 12 | 1/3 | 12.67 | -0.67 | 94.74% |
| 4 | use | 11 | 1/4 | 9.50 | 1.50 | 115.79% |
| 5 | notification | 11 | 1/5 | 7.60 | 3.40 | 144.74% |
| 6 | battery | 9 | 1/6 | 6.33 | 2.67 | 142.11% |
| 7 | problem | 7 | 1/7 | 5.43 | 1.57 | 128.95% |
| 8 | even | 7 | 1/8 | 4.75 | 2.25 | 147.37% |
| 9 | good | 7 | 1/9 | 4.22 | 2.78 | 165.79% |
| 10 | dont | 7 | 1/10 | 3.80 | 3.20 | 184.21% |
| 11 | get | 6 | 1/11 | 3.45 | 2.55 | 173.68% |
| 12 | would | 6 | 1/12 | 3.17 | 2.83 | 189.47% |
| 13 | like | 6 | 1/13 | 2.92 | 3.08 | 205.26% |
| 14 | time | 5 | 1/14 | 2.71 | 2.29 | 184.21% |
| 15 | annoying | 5 | 1/15 | 2.53 | 2.47 | 197.37% |
| 16 | message | 5 | 1/16 | 2.38 | 2.62 | 210.53% |
| 17 | background | 5 | 1/17 | 2.24 | 2.76 | 223.68% |
| 18 | emergency | 5 | 1/18 | 2.11 | 2.89 | 236.84% |
| 19 | people | 5 | 1/19 | 2.00 | 3.00 | 250.00% |
| 20 | positioning | 5 | 1/20 | 1.90 | 3.10 | 263.16% |

Figure 36. Most frequent words from negative review corpus of SOS Alarm app after applying Zipf law.

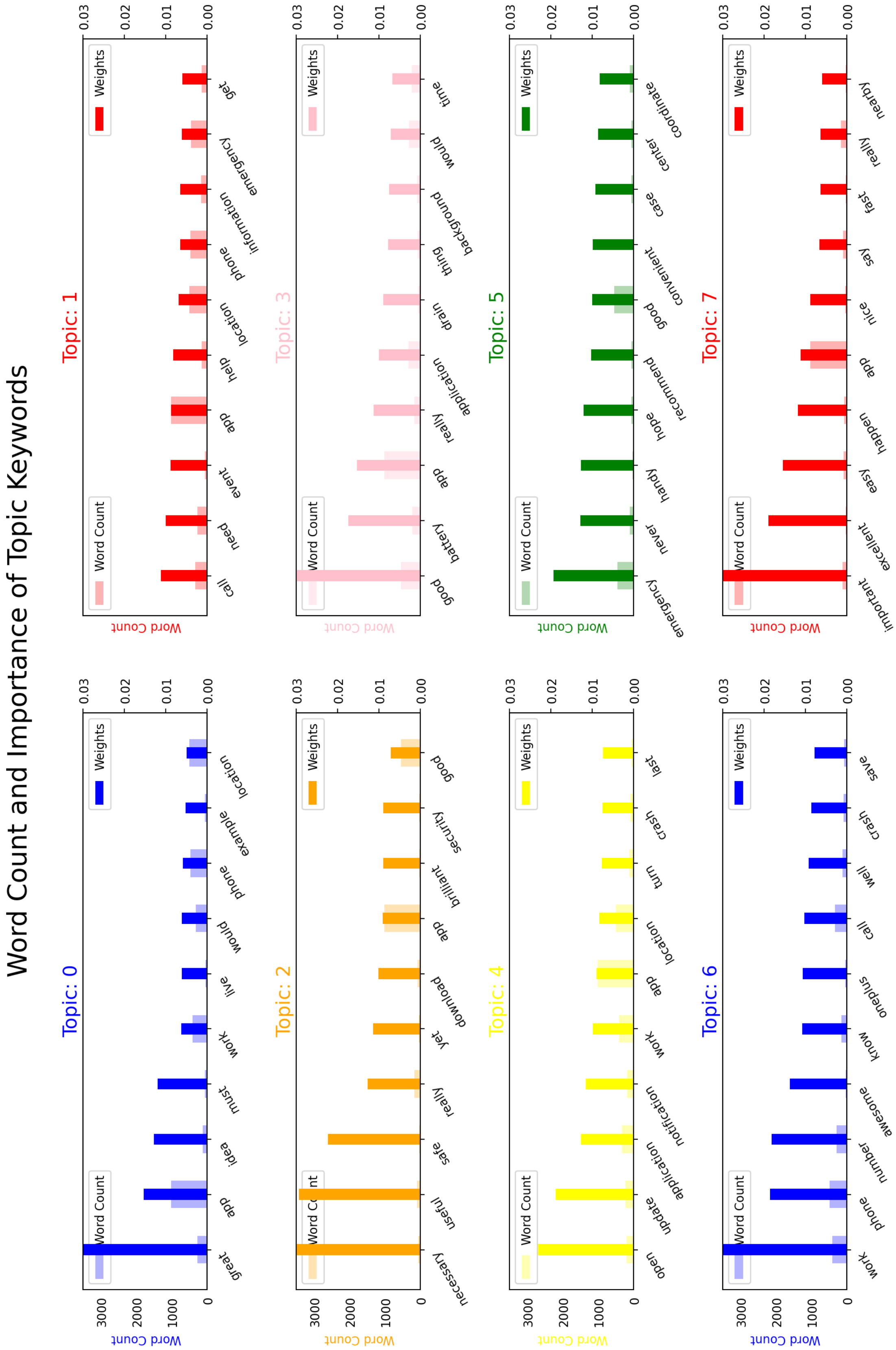


Figure 37. Word frequency and respective weights across different topics.

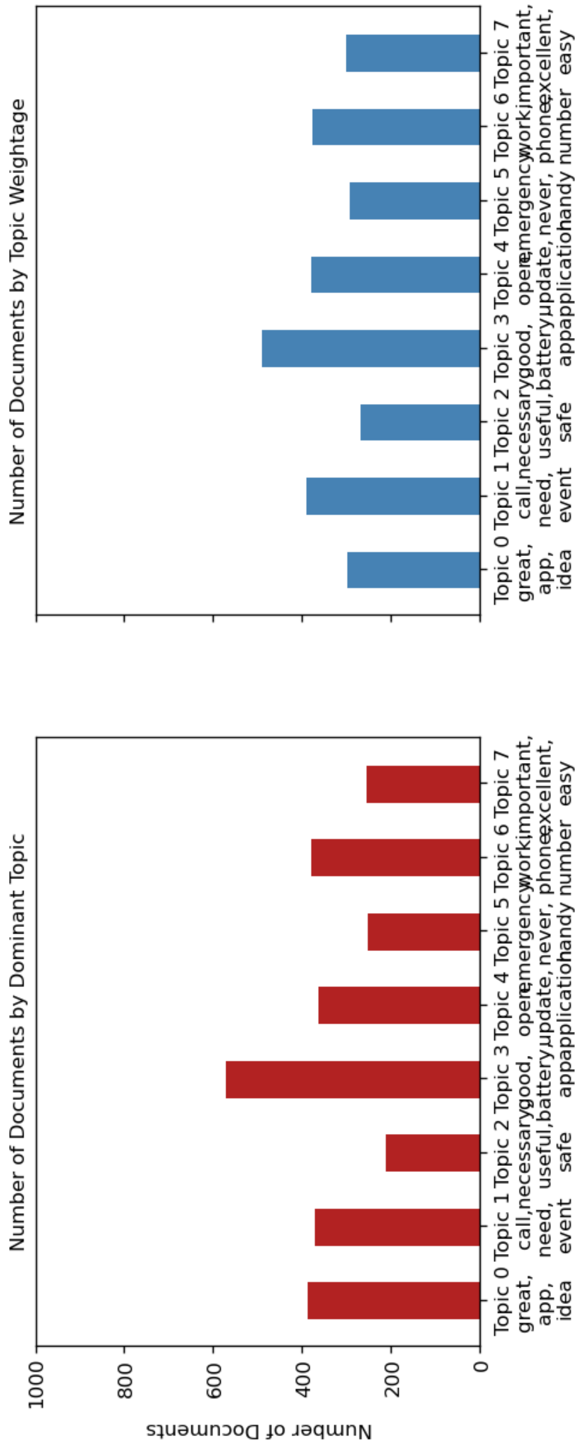


Figure 38. Number of documents by Dominant topic and topic weightage in review corpus respectively.

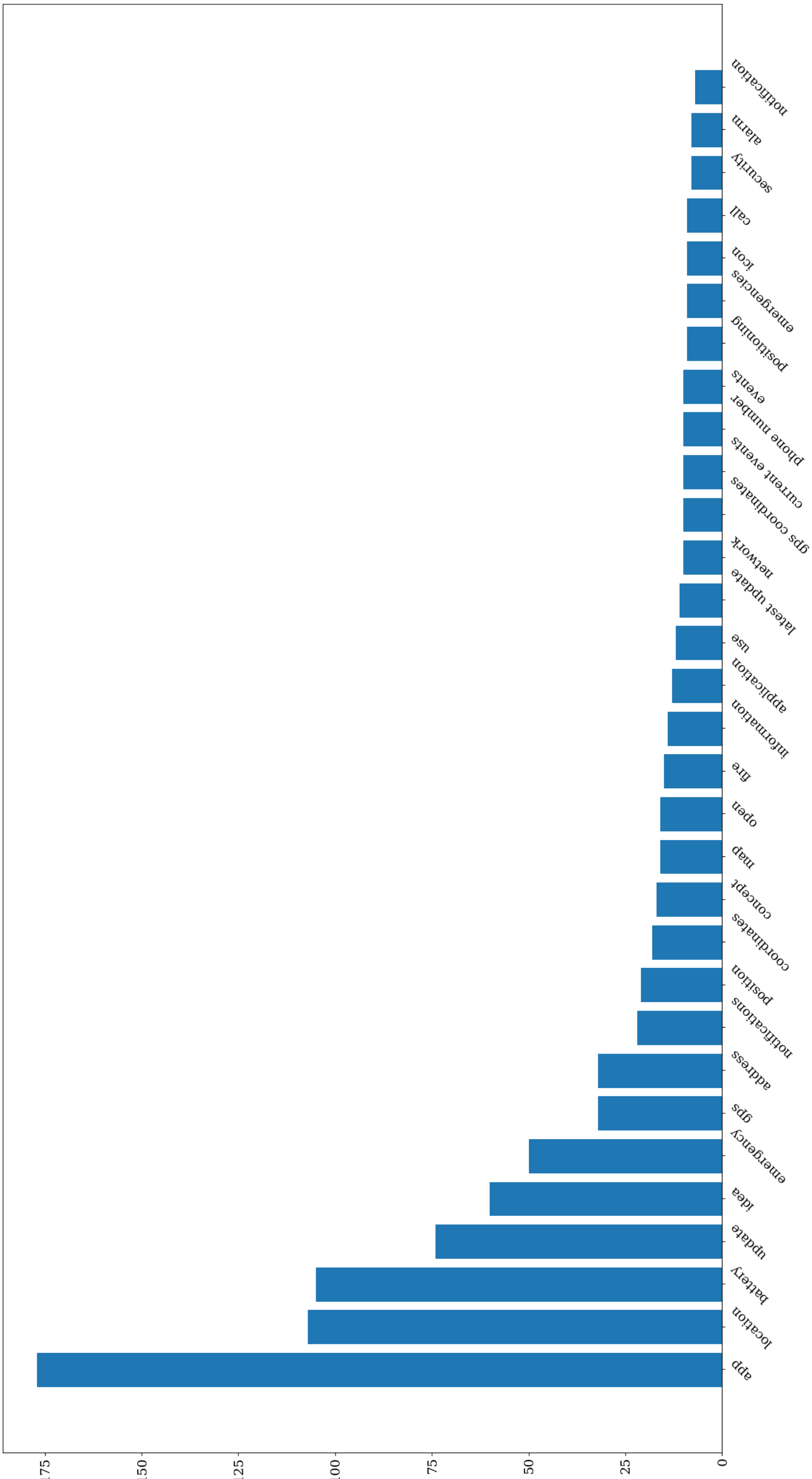


Figure 39. Top 30 frequently occurred aspects from the aspect term extraction task's output data frame and their respective frequencies sorted in descending order.

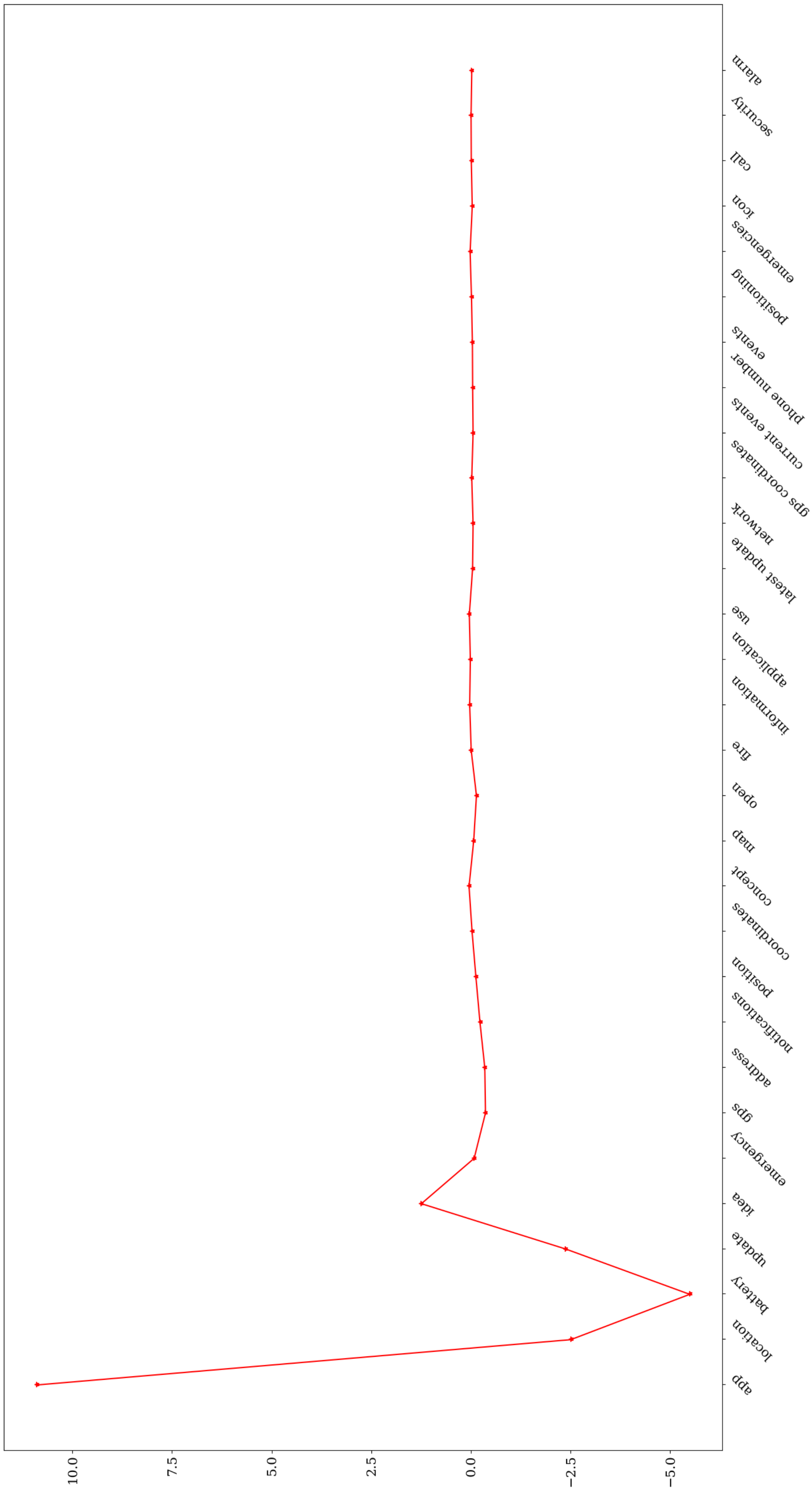


Figure 40. A line plot of the top 30 frequently occurred aspects from the aspect term extraction task's output data frame and their respective sentiment rate $SR_a$ scores.