

USABILITY CONSTRUCT FOR MOBILE APPLICATIONS: A CLUSTERING BASED  
APPROACH

A Dissertation  
Submitted to the Graduate Faculty  
of the  
North Dakota State University  
of Agriculture and Applied Science

By

Pratap Kotala

In Partial Fulfillment of the Requirements  
for the Degree of  
DOCTOR OF PHILOSOPHY

Major Program:  
Computer Science

December 2013

Fargo, North Dakota

North Dakota State University  
Graduate School

---

**Title**

Usability Construct for Mobile Applications – A clustering based  
approach

---

**By**

Pratap Kotala

---

The Supervisory Committee certifies that this *disquisition* complies with North  
Dakota State University's regulations and meets the accepted standards for the degree of

**DOCTOR OF PHILOSOPHY**

SUPERVISORY COMMITTEE:

Dr. Jun Kong

---

Chair

Dr. William Perrizo

---

Dr. Gursimran Walia

---

Dr. Limin Zhang

---

Approved:

5/14/2015

---

Date

Dr. Brian Slator

---

Department Chair

## **ABSTRACT**

The growth of mobile applications that run on cell phones and other handheld devices has introduced a broad range of usability challenges that were not faced by the web and standalone PC environments. The current usability models for mobile applications are mostly based on the experience of the usability experts and users that were collected through surveys and field studies. Many usability researchers and practitioners have developed conceptual usability frameworks that utilize either different or overlapping usability attributes. Moreover, the usability frameworks in existence they are limited in scope and do not consider all the usability dimensions. There is no consensus among usability researchers and standard organizations regarding what constitutes a usability model or framework. This research attempts to utilize a novel, computational, linguistic approach in order to identify the semantic relatedness between different usability attributes. We use text-mining and information-extraction techniques to mine for usability attributes in a large collection of published literature about mobile usability. A hierarchical clustering analysis is performed to cluster semantically related usability attributes. The results are utilized to develop a usability taxonomy and a unified usability construct for mobile applications.

## **ACKNOWLEDGEMENTS**

I would like to express my profound gratitude and appreciation to my major adviser, Dr. Jun Kong, for his support, patience and guidance. Additionally, special acknowledgements are given to my committee members: Dr. William Perrizo, Dr. Gursimran Walia and Dr. Limin Zhang. I would like to extend my special thanks to Dr. Kendall Nygard who always sought and encouraged me to complete my Ph.D. and who has been very kind to me all the years that I have known him. My friend Joseph Latimer deserves my gratitude for all his support during the last few weeks of my dissertation's completion. I would like to extend my special appreciation to Dr. Brian Slator and my colleagues in the department for their support and encouragement throughout my Ph.D. program.

## DEDICATION

I would like to dedicate this dissertation to the people who have been my biggest cheerleaders and without whose love, support and sacrifice I would not have had the honor of

completing my doctoral degree:

my beautiful wife, Kavitha, who always believed and trusted me;

my gorgeous kids, Sri and Abhi, for all the long hours I was gone;

my loving mother who never lost hope in me;

my late father for all the opportunities he helped to create for me all my life;

my late brother and best friend, Sridhar, whose dream it was to see me complete my doctoral

degree;

finally, my *Bhagwan, Sri SaiBaba* who walked and carried me every step of the way and guided

me till the end.

## TABLE OF CONTENTS

ABSTRACT .....	iii
ACKNOWLEDGEMENTS .....	iv
DEDICATION .....	v
LIST OF TABLES .....	viii
LIST OF FIGURES .....	ix
CHAPTER 1. INTRODUCTION .....	1
CHAPTER 2. BACKGROUND AND LITERATURE REVIEW .....	6
2.1. Challenges Presented by the Mobile Domain .....	6
2.2. Current State of Defining and Identifying Usability Dimensions .....	9
2.3. Text Mining for Knowledge .....	20
2.3.1. Clustering .....	25
2.3.2. Classification .....	27
2.3.3. Association Rule Mining .....	27
CHAPTER 3. OVERVIEW OF THE APPROACH .....	29
3.1. Collection Phase .....	31
3.2. Attribute Identification Phase .....	31
3.3. Pre-Processing Phase .....	35
3.4. Document Granularity .....	36
3.5. Tokenization .....	37
3.6. Filtering and Stemming .....	38
3.7. Extraction Phase .....	40

3.7.1. Vector Space Model.....	44
3.8. Analysis Phase .....	48
CHAPTER 4. RESULTS AND DISCUSSION.....	52
4.1. Discussion about Clustered Usability Terms and the Usability Taxonomy .....	59
4.1.1. System Dimension .....	59
4.1.2. Performance Dimension.....	66
4.1.3. Reliability Dimension .....	71
4.1.4. Ergonomic/Context Dimension .....	75
4.1.5. Satisfaction Attributes.....	80
4.1.6. Easy-to-Learn/Retain Dimension.....	82
4.1.7. Protection Dimension.....	84
4.2. Cross-Cutting Usability Attributes .....	87
4.3. Cross-Cutting Usability Themes.....	94
4.4. Proposed Unified Usability Construct .....	96
4.5. Significance and Implications for the Research.....	101
CHAPTER 5. SUMMARY AND CONCLUSIONS .....	104
REFERENCES .....	107
APPENDIX A. STOP WORD LIST OF 395 WORDS.....	118
APPENDIX B. LIST OF USABILITY ATTRIBUTES AND THEIR SEMANTIC MATCHES FROM LEXICOGRAPHIC DATABASES. ....	121
APPENDIX C. LIST OF 76 ATTRIBUTES USED FOR HIERARCHICAL CLUSTERING. ....	122

## LIST OF TABLES

<u>Table</u>	<u>Page</u>
1. Collected usability dimensions and attributes <sup>[32]</sup> .....	15
2. Variance decomposition for different cluster sizes.....	55
3. Usability terms grouped into seven clusters. ....	57
4. Semantically similar cross-cutting usability terms .....	92



## LIST OF FIGURES

<u>Figure</u>	<u>Page</u>
1. The top 10 search results of semantically similar terms for accuracy .....	35
2. Matrix representation of document vector .....	40
3. Positional index example <sup>[61]</sup> .....	42
4. An algorithm for proximity intersection <sup>[61]</sup> .....	43
5. Vector Space Model of documents, d1 and d2, assuming there are only two terms .....	46
6. Variance versus number of clusters .....	53
7. Dendrogram produced from hierarchical clustering of 76 usability attributes .....	56
8. Mind map showing the clustering of 76 usability terms.....	58
9. WordNet database showing the two senses for the word "fulfillment" .....	69
10. A hierarchical structure of the usability taxonomy depicting the seven usability dimensions (themes) .....	89
11. Relationship between the Cross-cutting Usability themes.....	95
12. Unified Usability Construct showing the relationship between the seven usability dimensions. ....	98

## CHAPTER 1. INTRODUCTION

The last few years have seen the explosive growth of mobile applications that run on cell phones and other handheld devices. Mobile applications have presented new opportunities and challenges for software engineering. This growth and the widespread use of mobile devices is expected to continue as they become more powerful with greater processing abilities; memory capabilities and new, emerging technologies.

The mobile-device environment is different from the conventional web and standalone PC environments. The unique nature of mobile applications with ubiquitous uses, small screen sizes, portability, device limitations, limited interactivity, changing contexts, and input and output challenges presents unique usability challenges that need to be addressed. Although a great volume of research on general usability exists, there are few studies that have been conducted on mobile usability [1] [2] [3] [4]. The development of usability principles that apply to mobile devices of various domains has not kept pace with the tremendous growth of mobile applications in the last few years. Human-Computer Interaction field has to come to terms with the opportunities and challenges offered by mobile computing.

Usability has been defined in various ways by different usability researchers and practitioners. It is a multi-dimensional concept, and different researchers have emphasized various dimensions of usability, some of which overlap. Many researchers and practitioners have utilized various usability dimensions/attributes to formulate usability conceptual frameworks in order to develop usability standards, principles and guidelines for different types of computer-based systems [5] [6] [7] [8]. These usability concepts and frameworks have been refined and adapted to apply to specific domains, such as web-based and mobile applications. While some

conventional usability principles are still valid for mobile applications, most of these principles fail to reflect mobile-phone characteristics comprehensively [2] [4] [6].

It is necessary to investigate whether the tools, techniques, principles and guidelines that are at the interface designers' disposal are adequate to meet these challenges and whether additional guidelines, principles and refinements to existing human-computer interaction principles are needed to meet these unique challenges.

The usability principles and guidelines that are being applied to mobile applications were identified through an ad-hoc, subjective process where the usability experts identified and grouped attributes that would be relevant to mobile devices. Most usability principles for mobile applications are either: a) partially done or are not specified in a concrete or precise way [9] [10], focusing on predictability [11], or b) are too domain specific to be applied to other fields [12]. Different authors have used various words, terms and meanings to identify relevant usability attributes resulting in a scenario where the same usability attribute was grouped with multiple high-level, usability dimensions by different researchers and practitioners. Hence, there is not a unique list or grouping of usability attributes that are specific for mobile applications.

This investigation takes a novel computational approach that is based on computation linguistics to identify the semantic relatedness of usability attributes (terms) by utilizing text-mining and information-extraction techniques. In computational linguistics, a fundamental idea for understanding semantic relationships between words is based on the concept that “words with similar meanings will occur with similar neighbors if enough text material is available” [13]. The *distribution hypothesis* is widely accepted in computation linguistics, and it states that words with similar distributional patterns are semantically related [14]. Therefore, studying the patterns of target words' neighbors helps to understand the semantic relatedness with other words in a

text because words with similar meanings tend to occur in the same context. Our research leverages this concept of words' distributional properties in order to derive the semantic relatedness between different usability attributes in the text. To our knowledge, this approach has not been tried in the area of usability.

This research proposes to develop a usability taxonomy and a usability construct for mobile applications built on the existing body of published HCI articles. The proposed methodology takes a computational linguistics approach to study the semantic relatedness between usability attributes on mobile usability.

Our approach involves identifying usability attributes which are commonly mentioned in the mobile usability literature. The usability attributes proposed by Coursaris and Kim (2011) [15], through their extensive review of publications about mobile usability, were adopted for this study. The usability attributes (terms) were searched through lexicographical databases, UMBC WebBase Corpus and the LDC English Gigawords Corpus of American newswire services, to determine the highly associated, semantically similar terms for each usability attribute. All terms extracted from the lexicographical databases, along with the original usability attributes, were regarded as the target words.

Next, a large number of published articles on mobile usability that had been collected from different sources were pre-processed using a tokenization, filtering and stemming process. The *tokenization* process helps split the text into words based on their word boundaries. It also handles non-text characters such as tab spaces and punctuation. In order to reduce the documents' size, frequently occurring *stop words* were filtered from the documents. This step was followed by morphological normalization using the *Porter stemming* algorithm [16] to chop

suffixes and prefixes from each word in order to reduce words to their base-normal form called the *stem*.

Text-mining and information retrieval (IR) techniques were employed to extract words from the processed documents to generate term-document *vectors* for a *Vector Space model*. Because the objective of this investigation was to identify the *semantic* relatedness between usability terms, the term-document vectors were used to build the *Semantic word space model* containing term-term vectors. These term-term vectors were scanned for *target words* (usability terms) with the purpose of building *co-occurrence matrix* usability terms that stored the *cosine similarity measures* between usability term pairs.

A hierarchical clustering of the usability terms was performed to cluster terms. The clustering-analysis results helped discover the semantic relatedness between the different usability attributes, some of which were obvious while others were hidden. This analysis was utilized to generate a usability taxonomy and a unified usability construct.

The approach adopted in this research was considerably different from other usability researchers' methodologies to study usability attributes. The methodology was not directly based on statistics but, instead, utilized the measure of semantic relatedness to discover the relationship between usability attributes.

The objectives of the research were as follows:

1. Identify usability attributes relevant to the usability of mobile applications.
2. Generate clusters of usability attributes that are semantically related using a novel computational-linguistics approach.
3. Develop a taxonomy of usability attributes for mobile usability.

4. Develop a usability construct for mobile applications that captures all the relevant usability paradigms.

The findings of this study were expected to have real-world implications for usability researchers and practitioners. Some of the implications are as follows:

1. Developing usability evaluation techniques and design guidelines that are accurately based on relevant usability attributes for mobile applications.
2. Helping usability designers can better align their goals based on the usability attributes that capture all aspects of mobile usability.
3. Classifying usability problems for mobile applications based on their semantic relatedness.
4. Developing new metrics for important usability attributes that were being overlooked.

The rest of the dissertation is organized as follows. In Chapter 2, the issues that impact, and the related challenges of, mobile usability are discussed along with an extensive survey of the literature to understand usability issues. There will be a discussion about existing approaches being used to identify usability attributes along with the proposed computational approach. Finally, a survey of text-mining techniques and challenges for mining text data are presented. Chapter 3 provides a detailed discussion about the proposed methodology, followed by details for the approach. Chapter 4 provides a detailed analysis of the results, a discussion of the results and the proposed usability construct for mobile applications. Chapter 5 has a summary of the research and the conclusions.

## **CHAPTER 2. BACKGROUND AND LITERATURE REVIEW**

Recent years have seen tremendous growth for software applications that run on mobile devices. The convergence of ubiquitous connectivity as well as powerful and affordable mobile devices are beginning to impact every aspect of our lives. Mobile applications, simply referred to as “Apps,” are software systems that run on a mobile device, such as a smartphone or tablet, and are expected to grow and to evolve at a rapid pace as more powerful mobile devices are developed. Mobile applications have also flourished in different areas of business, such as finance, management, logistics and supply chains. The widespread use of mobile applications by personal and business users requires that the mobile apps not only be useful, but also usable [10]. The growth of mobile devices is expected to outpace desktops in the next few years.

Mobile devices are different from other computing systems in terms of the tasks they perform, their design and their operation [17]. The problems and issues associated with traditional computer systems are compounded by the multichannel requirement of mobile systems [17]. There are many different aspects of the user interface along with interactions that users perform with a mobile device which are significantly different from traditional computer-based systems [18]. The usability factors related to standalone computer design principles and guidelines do not reflect the physical interface features of mobile phones [19]. The fast-paced development and evolution of new physical features for mobile devices present different challenges. The next section discusses the Challenges Presented by the Mobile Domain.

### ***2.1. Challenges Presented by the Mobile Domain***

The unique features of mobile devices pose significant challenges when designing usability principles for mobile applications. The traditional methods and guidelines utilized

determine the usability of desktop systems need to be modified for a mobile environment [20].

Some challenges are as follows:

1. **Context:** Because mobile devices, by definition, are mobile, the context changes continuously. The mobile context includes the location where the device is used and the condition of the mobile user. A mobile user could be simultaneously interacting with other objects, people and the surrounding environment, impacting the device's usability. There are four aspects to the mobile problem: the mobile user, the mobile device, the mobile application and the mobile network [17] [21].
2. **Small Screen:** Mobile devices' small screen size impacts the usability of mobile applications [22]. The smaller screen sizes enable the devices to be portable. When adapting a desktop's usability guidelines and principles to mobile devices, there is a need to reduce and simplify the features. "Feature shedding" is a critical aspect of designing mobile applications; trying to fit all of a desktop application's functionality in or directly mapping its task/information hierarchy to a mobile application will cause more usability and utility problems [23].
3. **Limited Connectivity:** Mobile devices are often characterized by limited and unreliable connectivity. Wireless connections also suffer from low bandwidth and are unreliable in nature because of environmental factors such as physical obstructions, weather, etc. The unreliable nature of wireless connectivity impacts the Quality of Service (QOS). The QOS effect plays an important role when designing mobile applications; developers have to know how to make the applications continue to operate as mobile devices connect and disconnect from the network intermittently and frequently [17].



4. **Limited Processing and Storage Capacity:** The physical-size limitations impact mobile devices' storage and computational abilities. The size limitation means that the applications' capabilities are also limited [3]. As newer and more complex apps are being added, they are beginning to starve for computational resources. On the other hand, modern desktop computers are not limited by processing capabilities, memory or storage. When designing desktop applications, the impacts for these factors are limited.
5. **Limited Power Supply:** The primary power source for mobile devices is batteries. The device's size constraints limit the battery size and capacity. The mobile nature means that the device is unplugged from an electric outlet for long times and that applications have to run with minimal power usage. The desirability of using batteries, combined with size constraints, creates another constraint: a limited power supply which should be balanced with processing power, storage and size constraints [17].
6. **Restricted Input Methods:** The small and mobile nature requires different input methods for mobile devices compared to desktops. The hardware constraints limit the methods that could be used to provide inputs for the applications. Providing input for a small mobile device is tedious and requires a certain level of dexterity [23]. The smaller button size for the physical or on-screen keyboard as well as smaller labels limit the user's effectiveness and efficiency to enter data, which may reduce input speed and increase errors [20]. Moreover, each key can support multiple functions. Newer input technologies, such as touch, hand-writing recognition, voice commands and hand-gesture input methods, have been ameliorating the problem to some extent.
7. **Platform Heterogeneity:** The small nature of these devices, coupled with lower hardware costs, means that manufacturers can assemble the devices at lower costs and

with fewer resources. The small nature of these devices has led to an abundance of mobile devices with varied operating-system platforms. The availability of different platforms for mobile devices requires that the designed and developed applications be platform independent [17]. Mobile-device users also have to adapt to new forms of interactions as they move from one platform to another [3].

8. **Privacy and Security:** Privacy is defined as “the ability of an individual to control the terms under which their personal information is acquired and used” [24]. In the mobile context, “users find themselves in a variety of spaces, in a variety of situations, and in a variety of infrastructures” [3]. Thus, the mobile context raises privacy and security concerns.

The issues detailed earlier are primarily caused by the device limitations and the user’s context. These factors need to be considered while designing and developing mobile applications. The usability principles for standalone computers do not take these factors into consideration. The current usability guidelines for mobile devices are relatively unproven [25], and even the ones that exist are “isolated and disintegrated” [10]. In spite of the limitations, mobile devices present great opportunities that could impact every aspect of today’s life. Bertini and Kimani [26] identified ubiquitous access, portability, the personal nature of use, democratization of information access, opportunistic interaction, and reduced complexity as some advantages that mobile devices provide.

## ***2.2. Current State of Defining and Identifying Usability Dimensions***

Researchers have adopted different approaches for specifying usability elements/dimensions. The published studies contain numerous definitions of usability, usability characteristics and elements [17] [27] [28] [29], and few guidelines exist about how they are

related [30] [25] [19]. According to the ISO 9241-11 standard, there are three aspects of usability: effectiveness, efficiency and satisfaction. Nielsen [29] classified usability dimensions: learnability, efficiency of use, memorability, errors and satisfaction into two groups: objective and subjective. Shneiderman and Plaisant [28] identified five usability measures: time to learn, speed of performance, rate of errors by users, retention over time and subjective satisfaction. McGee et al. [31] developed a user-defined definition of usability: “Usability is your perception of how consistent, efficient, productive, organized, easy to use, intuitive and straightforward it is to accomplish tasks within a system.”

Heo et al. [19] developed a hierarchical model of usability factors to generate a framework for evaluating the usability of mobile phones. This model organizes the usability factors that impact mobile phones into four abstraction levels: usability, usability indicator, usability criteria, and usability property. The researchers identified and organized 136 usability factors on people’s opinions to classify and abstract usability problems.

Ji, Park, Lee and Yun (2006) [32], developed a usability checklist by utilizing research material about style guides obtained from various mobile phone companies to elicit “User Interface (UI) elements.” Using these UI elements, the researchers constructed a hierarchical structure which formed the basis of the usability checklist. Next, the researchers performed a literature survey for the usability principles and classified them by selecting, deleting and integrating them. Finally, the researchers matched the usability elements with usability principles by utilizing the survey results that involve usability experts and mobile-phone developers.

Many researchers tend to use usability and user experience interchangeably although user experience is a much broader concept than usability [33]. User experience covers the entire gamut and considers the individual’s interaction with products, as well as subjective attributes

such as thoughts, perceptions and feelings. Usability is mainly focused on the ease of use that is impacted by design features of interactive products [34].

McGee, Rich and Dumas [31], in their research on developing a usability construct, administered a study with 46 users to rate how integral the usability characteristics were to people's concept of usability. Utilizing the survey results, the researchers performed a hierarchical cluster analysis to organize the usability characteristics into a classification structure. The authors developed usability taxonomy by analyzing the cluster using expert knowledge. The usability characteristics were delineated into two clusters that were highly integral to usability (labeled Usability) and those that were not (labeled Not Usability). The Usability cluster had the following hierarchy names:

- a) Core Usability (Strategic): important usability characteristics that are considered important by users include consistent, efficient, organized, and easy and intuitive
- b) Secondary Usability (Tactical): containing the usability characteristics such as effective, familiar, controllable, complete beneficial, and useful.
- c) Tertiary Usability (General Qualities): elements related to accomplishing tasks such as expected, natural, worthwhile and flexible, and friendly.

Hornbæk and Law [35] performed meta-analysis to aggregate correlations among usability measures by utilizing raw data from 73 published studies. The measures were categorized according to ISO 9241-11: effectiveness, efficiency and satisfaction. The study reported weak correlations among efficiency, effectiveness and satisfaction. The correlation between effectiveness and efficiency was 0.247 +/- .059, between efficiency and satisfaction was 0.196 +/- .064, and between effectiveness and satisfaction was 0.164 +/- .062. Task complexity did not seem to have any influence on correlations. The study concluded that dependence among

various aspects of usability; the associations were too low for aggregating metrics into a summary score.

Ham et al. [6] proposed a conceptual framework and model for identifying and organizing the usability impact factors for mobile phones. The researchers proposed five views to reflect different aspects of the interactions that users have with a mobile phone: user view, product view, interaction view, dynamic view and execution view. The usability impact factors were derived from these five views and then organized into a hierarchical model in terms of goal-means relationships. Based on their review of literature, they derived five indicators to estimate the usability of mobile phones: effectiveness, efficiency, learnability, satisfaction and customization.

Hornbæk [36] classified 54 different usability measures using the ISO 9241 standard: effectiveness, efficiency and satisfaction. In each group, the measures were further classified into subgroups based on the usability measures identified through the usability measures mentioned in prominent usability and behavioral science textbooks as well as the similarities found in the usability measures utilized by the reviewed studies. The identified subgroups were not wholly exclusive. In order to assess the reliability of the classification, the authors had independent raters reclassify the same measures and found that there was an average 89.7% agreement for this classification.

A study about the challenges and issues with mobile applications by Zhang and Adipat [20] listed nine usability attributes that were collected and compiled from existing usability studies. These attributes were not validated, thus raising an interesting and challenging research question about how the usability of mobile applications can be effectively evaluated.

Hussain and Ferneley [37] developed a set of usability guidelines that were bundled with usability metrics using the Goal Question Metric (GQM) approach to assist developers who are designing mobile applications. The researchers employed a four-step method developed by Leavitt and Shneiderman [38] to shortlist the usability guidelines. According to Basili, Caldeira, and Rombach (1994), a GQM model is a hierarchical structure that starts with a goal (specifying purpose of measurement, object to be measured, issue to be measured and viewpoint from which the measure is taken). The goal is refined into several questions which usually divide the issue into its major components. Each question is then refined into metrics, some of which are objective while others are subjective. The same metric can be used to answer different questions for the same goal.

Garcia et al. [39] employed Artificial Intelligence (AI) algorithms to extract knowledge from usability evaluation questionnaires. In this study, a number of evaluators were asked about their opinion for some aspects of the webpage. Each questionnaire was a hierarchically structured tree of questions where each question was associated with one or more criteria. AI techniques were used to cluster the results for the questions and the questions themselves, and to further develop models for evaluators. The researchers used machine learning to predict webpage usability in an automated way. This study was limited to webpage usability evaluations.

Hussein and Kutar [30] conducted an extensive review of usability, and they identified 17 popular guidelines by utilizing effectiveness, efficiency and satisfaction as the quality characteristics. Their guidelines included completeness, accurate, less or no error, ease to input the data, ease to use output, ease to install, response time, simple, time, ease to learn, application size, battery power used, wireless connectivity, features available, satisfaction with interface, provide help/support and safety.

Through an extensive review of literature for developing an empirical, conceptual framework for mobility, Coursaris and Kim [15] identified 16 usability dimensions; they found that the constructs of efficiency, ease of use, effectiveness, satisfaction and learnability were most commonly measured in empirical mobile usability studies. Based on relative appearance of measures in the reviewed literature, they identified three core constructs to measure usability: efficiency (33%), effectiveness (62%) and satisfaction (20%). These three characteristics reflect the ISO 9241 standard because they are the measure attributes for that standard. The remaining measures, including learnability (11%), accessibility (7%), operability (4%), memorability (2%), flexibility (2%) and acceptability (2%), were classified as peripheral dimensions. The authors also identified methodologies that were utilized by different researchers in mobile-usability research they included questionnaires (61%), device data (33%), direct observation (7%), focus groups (7%), discussions (3%) and voice-mail dairies (2%).

Usability attributes are features that describe the quality of the application's interface. Researchers have adopted different approaches to specify usability elements. As a result, there are many usability attributes that have been identified. Ji et. al. [32] surveyed the published research on usability issues surrounding mobile devices and identified the usability dimensions/attributes that need to be considered for the mobile-application design process. The usability attributes that were identified are presented in Table 1.

Table 1. Collected usability dimensions and attributes<sup>[32]</sup>

Reference	Usability Dimensions/Attributes
Constantine [40]	Structure principle, simplicity principle, visibility principle, feedback principle, tolerance principle and reuse principle
Nielsen [29]	Visibility of system status; match between system and the real world; user control and freedom; help users recognize, diagnose and recover from errors; recognition rather than recall; aesthetic and minimalist design; pleasurable and respectful interaction with the user; consistency and standard
Treu [41]	Effort
Dix, Finaly, Abowd and Beale [27]	Learnability, predictability, synthesizability, familiarity, generalizability, consistency, flexibility, dialog initiative, multithreading, customizability, task migration, substitutivity, robustness, observability, recoverability, responsiveness and task conformance
Laueseen and Younessi [42]	Ease of learning, task efficiency, ease of remembering, understandability and subjective satisfaction
Preece, Rogers and Sharp [43]	Effectiveness, efficiency, safety, utility, learnability and memorability
Coursaris and Kim [15]	Efficiency, effectiveness, satisfaction, accessibility, learnability, workload, enjoyment, acceptability, quality, security, aesthetics, utility, memorability, content, flexibility and playfulness
Harrison, Flood and Duce [4]	Effectiveness, efficiency, satisfaction, learnability, memorability, errors and cognitive load



Kjeldskov and Paay (2012) surveyed the research methods that were used along with the purpose of research in the mobile HCI field starting from 2000-2003 [44] and later in 2009 [45]. Their findings for papers published in 2009 indicated that 49% of the mobile HCI research fell under the *lab experiment* category (research in a controlled environment), *field studies* (experiments in real-world social and cultural contexts) accounted for 35% of the research, followed by *applied research* (30%), *paper reports* from survey research (21%), *case studies* (6%), *basic research* (4%) and *normative writing* (2%). The findings regarding the purpose of the research showed that 68% of mobile research was on *usability evaluations*, followed by 40% involving *building systems* (focused on building new systems or parts of mobile interfaces), of which 72% involved *applied research* (using trial and errors, intuition, experience, and deduction and induction). Finally, 31% of mobile research was conducted for *understanding* (focus on finding the meaning of the studied phenomenon using frameworks and theories from collected data), of which 43% was done using *field studies* and 32% through *surveys*. Only 7% of the research papers described aspects of mobile HCI (mobile HCI elements and methods). The researchers' exhaustive survey concluded that papers with a purpose of *understanding* rarely appeared with the purpose of *engineering*. Engineering was predominantly based on trial and error, and papers with a purpose of *understanding* mobile HCI phenomena seldom led to new systems or designs. Finally, a very small percentage of research was being conducted to describe various aspects of mobile HCI.

As the review of literature indicates, the concept of usability has been studied differently by various researchers and practitioners. Moreover, the predominant research methodology was applied research involving trial and error. The predominant purpose was to evaluate mobile applications with a very small share of research about describing the mobile HCI aspects. Most

usability principles were borrowed from standalone computer systems and applied to mobile applications in an ad-hoc manner. The additional variables and restrictions for mobile devices, such as device parameters like screen size, low processing power, memory, etc., that were discussed earlier, in particular the context of use, were either ignored or not actively considered. There was also the issue of different terminologies being utilized for same usability terms. Thus, there is need to apply a systematic approach in order to build identify usability attributes that are relevant to mobile applications.

In our approach, we developed a new usability construct for mobile applications by taking a systematic computation approach. We created a novel, computational, linguistic approach to understand the relationship between usability attributes based on the distributional patterns of the usability terms in published mobile usability articles. Next, a hierarchical clustering of various usability attributes was performed to cluster them based on their semantic relatedness. The approach involved extracting a previously identified list of usability attributes (target words) and their semantically similar terms in order to study their co-occurrence based on semantic similarity. This analysis was utilized to generate a usability taxonomy.

The target usability attributes that was searched in the published documents was based on the usability attributes that were identified by Coursaris and Kim [15] through their extensive review of published articles over many years. This list was related to empirical research on mobile usability. These 16 usability attributes are described as follows:

1. *Learnability* refers to how easily users are able to learn and quickly complete a task the first time. A user should be able to predict the outcome of future interactions based on the learning in past interactions. With mobile devices, the small screen size means a reduced set of choices, thus decreased complexity. This reduced complexity can lead to better

learning and retentions functions. For instance, the number of menu items that a small-screen application can host is significantly reduced compared to the standard environment. Therefore, the number of options through which a user must search is also reduced, thus improving learning time and access cost. In the case of contextual interaction, pre-emptive behavior can create predictability problems because the interaction is triggered by external events in an opportunistic way [26]. In this case, it would be harder to predict how the system will behave, and in some cases, it will be impossible to predict when an interaction will happen. It would be interesting to explore how to overcome this inherent lack of predictability that is introduced in context-aware systems.

2. *Efficiency* is defined as how fast people can accomplish a task while using an application. The time it takes to accomplish a task is important; the responsiveness is crucial. Most of the time, mobile-computing users perform a task with a sense of urgency. They realize that there are various aspects at stake. There are also cases where mobile-computing tasks are not the main tasks. Rather, they are supportive tasks. Naturally, people would like to spend time on main tasks rather than supportive tasks.
3. *Memorability* refers to the level of ease with which users can recall how to use an application after discontinuing its use for some time [26]. The main idea is to measure how well people can recall the skills to use a mobile application. To increase the memorability, familiarity is important. Familiarity is very important and not very well supported on mobile devices. Different mobile devices appear with different keyboard layouts or modalities: keyboard vs. stylus and touch screen vs. buttons. Each device has its own special access keys that are unique to the model which can affect the

understanding of the interface, thus decreasing the memorability. In the case of location-independent applications, familiarity with the applications' PC versions can be exploited. Apart from familiarity, consistency is another factor that impacts memorability. Using indirect input with a keyboard leads to flaws. The user learns that some keys are always utilized to trigger certain actions, such as opening a menu; if, in special cases, the keys do not work as they are expected then there is a consistency problem [3]. Another issue with keyboards is that the button functions are overloaded.

4. *Satisfaction* reflects people's attitude toward using a mobile application.
5. *Effectiveness* is defined as completeness and accuracy with which users achieve certain goals. It can be measured by comparing user performance with the required levels. The effectiveness attribute is also used to assess the improvement for a new version of a mobile application.
6. *Accessibility* is the ability of a system to support different kinds of users regardless of their capabilities.
7. *Adaptability* is the ability of an application to make necessary changes in response to the hardware and software changes around it.
8. *Workload* refers to the amount of physical energy a user has to spend to complete a task on a system.
9. *Enjoyment* is a desirable goal that systems should seek. It refers to the pleasure a user derives through the interactions with a system.
10. *Content* or contentment refers to the state of being satisfied. It is closely associated with satisfaction.

11. *Acceptability* means whether the produce will be used, and how it will be used in the real world [46].
12. *Aesthetics* is a “branch of philosophy associated with art and beauty and is concerned with how individuals perceive object and make judgements based upon information received as five human sensory inputs” [47].
13. *Utility* is the value that a system provides to a user – the usefulness and applicability of a system.
14. *Flexibility* is “the ease with which a system or a component can be modified for use in applications or environments other than those for which it was specifically designed” [48].
15. *Playfulness* refers to the entertainment value of a system that a user experiences as he/she interacts with the system.
16. *Quality* according to [49], is “the totality of features and characteristics of a product or service that bear on its ability to satisfy stated or implied needs”. In simpler words, one can say that a product has good quality when it "complies with the requirements specified by the client”.

In Section 2.3, a review of the text-mining process is discussed in detail. Finally different type of text mining approaches including clustering, classification, and association rule mining are presented.

### ***2.3. Text Mining for Knowledge***

Text mining seeks to extract useful knowledge by using automated methods and techniques that can parse large number of words and structures in unstructured, natural-language text data. The text-mining methods and techniques are interdisciplinary and are borrowed from

various fields such as information retrieval, machine learning, statistics, computer linguistics and data mining [50].

Text mining aims to extract unknown knowledge from the text data. On the other hand, information retrieval methods focus on uncovering existing relationships between concepts in the text data. Text mining borrows heavily from the data-mining area; they use similar algorithms and techniques to extract unknown information. The difference lies in the data source. Data-mining techniques are applied to structured or semi-structured data that are usually stored in databases. On the other hand, text-mining techniques are used on unstructured, natural-language text data.

Natural-language text data possess multiple challenges that are usually not encountered by the entire data-mining area. The complexity of a natural language is a major challenge. There are different spoken languages in the world. Each one has its own vocabulary, grammar, syntactic and semantic rules. Most languages allow for flexibility in how sentences are constructed and allow the use of different words to convey the same meaning. This flexibility presents its own challenge: *ambiguity* [51]. There can be a lot of ambiguity in terms of multiple *senses* for the words and their synonymy. A *sense* of a word refers to a specific meaning of a word. Many words have multiple *senses* because the same word can acquire different meanings in varying contexts. Here is an example of the multiple senses of the word *bat*: “*A bat can fly.*”, “*A baseball player uses a bat.*” and, “*You can bat away the fly.*”

Text mining is a relatively nascent field that has made good progress in dealing with the ambiguity problem but the challenge has not been resolved completely. Another challenge posed in text mining is the nature of the data itself. They are unstructured and come from different types of documents and media such as emails, webpages, social media, twitter feeds, paper and

electronic documents, etc. Moreover, text data also suffer from missing data and an abundance of noise. The high dimensionality of text data is another challenge when there are tens and thousands of words with only a small percentage of them used in most documents.

Most text data require pre-processing using the methods from Natural Language Processing (NLP). These methods help translate documents from one language to another. NLP also provides techniques to utilize with tokenization of text, stemming algorithms, lexical analysis and semantic analysis [51].

Information-Extraction (IE) approaches extract information from texts. These techniques are mostly concentrated in the extraction of facts and event entities as well as deducing the relationship between these entities. Text-mining techniques use IE methods to extract information from the text. The extracted information is later stored in a database for running queries or for data mining.

The text-mining process involves multiple steps and borrows methods and techniques from various fields as discussed earlier. The steps in the process are explained next.

1. Collecting text data: Because text data can come from multiple forms, platforms, formats and sources, the first step involves determining the criteria for data selection. For example, with our study, the criterion was to collect published papers from named sources from the years 2012-2015. Typically, the data used for text mining is enormous and requires a lot of effort.
2. Pre-processing of text data: For most applications, text data require some pre-processing which includes the following:

- a) Converting into an appropriate format: Most text-mining methods require the text to be stored in a simple format. In our research, the documents are in PDF format, and they are converted to plain text.
- b) Tokenization: Tokening is a process of breaking long texts into smaller chunks where the size can vary from individual words to sentences or paragraphs. For the purpose of our study, we consider word boundary as the delimiter for tokenization.
- c) Dealing with bad data and noise: One of the important issues with text data is the bad data and noise that are embedded. It is important to eliminate noise, meta-data and other unwanted components from the text data. Our study requires that all meta-data for the files, figures, tables and reference sections be removed.
- d) Filtering, lemmatization and stemming: Because text data suffer from high dimensionality, sometimes, it is necessary to filter out the frequently used words that may not contribute to the information that is being extracted. There are many standard list of stop words available for various applications. Another way to reduce the size of text data is to use the process of lemmatization and stemming. These techniques help reduce the words to their base-normal form called a stem. A standard stop-word list is used in our research to reduce the dimensionality of the problem. Elimination of stop words is followed by stemming using the Porter's Algorithm [16].
- e) Other optional steps: Depending on the purpose and the application where data mining is being used, it might be necessary to use Part of Speech Tagging, a process of marking the words in a text with their corresponding parts of speech.



Another step that is required at times is Word Sense Disambiguation (WSD) which involves determining the *sense* of the word that is being used in the sentence.

3. Attribute generation: This step usually creates an appropriate text representation of the data. One of the common techniques that are used is Index Term Selection where selected keywords are utilized to describe the documents. The most common representation for an easy search and processing is to create term-document vectors where the term (word) represents the row and the document IDs in which the terms appear are the dimensions. An inverted index tree of the terms is built for easy access and quick searching of the index. For our research, term-document vectors are generated, and the terms are indexed. Using the docVectors and the termVectors, a term-term matrix of the usability attributes and their semantically related words is built.
4. Data mining/pattern discovery: At this step, the classical data-mining techniques can be applied because the data have been converted to a structured format and may be stored either on a flat file or in a database.
5. Interpretation and evaluation: The last step involves analyzing and interpreting the results to see if appropriate answers were found. To aid with the interpretation, the results can be presented visually with charts, graphs, etc.

Different types of data-mining techniques are employed for extracting useful information. Most data-mining techniques come from the field of statistics. The various data-mining techniques can be broadly classified into three categories: clustering, classification and Association Rule Mining (ARM).

### 2.3.1. Clustering

Clustering is a popular data-mining technique which involves grouping likeminded concepts or entities using a similarity measure. Clustering is also called *Unsupervised Learning*. When applied to textual data, clustering methods try to identify inherent groupings of the text documents so that a set of clusters is produced so that the clusters exhibit high intra-cluster similarity and low inter-cluster similarity [52]. Generally speaking, text-document clustering methods attempt to segregate the documents into groups where each group represents some topic that is different than the topics represented by other groups. Applications of document clustering include clustering retrieved documents to present organized and understandable results to the user, clustering documents in a collection (e.g., digital libraries), automated (or semi-automated) creation of document taxonomies (e.g., Yahoo and Open Directory styles), efficient information retrieval by focusing on relevant subsets (clusters) rather than entire collections and generating ontological clusters for a group of similar or related concepts.

The methods used for text clustering include decision trees, statistical analysis, neural nets, inductive logic programming and rule-based systems. These methods cross various research areas such as database, information retrieval, artificial intelligence including machine learning and Natural Language Processing. Any clustering technique relies on three concepts:

1. A data representation model,
2. A similarity measure,
3. A clustering algorithm that builds the clusters using the data model and the similarity measure.

Most document-clustering methods that are used today are based on the Vector Space Model (VSM) which is a widely used data model for text classification and clustering. VSM models are commonly utilized for information retrieval, information filtering, indexing and relevancy rankings. VSM is simple and based on linear algebra, use term weights instead of binary values, allow computing a continuous degree of similarity between queries and documents, and, allow partial matching [53].

The Vector Space Model represents documents as a feature vector of the terms (words) that appear in the document. Each feature vector contains term weights (usually term frequencies) of the terms that appear in the document. The vector representation allows us to compute distance or similarity between pairs of items using standard vector operations. Examples include the cosine measure and the Jaccard measure. Clustering methods based on this model only use single-term analysis; they do not make use of any word proximity or phrase-based analysis. Cluster analysis is typically used for grouping similar documents together so that documents in different groups are clearly distinct from one another.

Hierarchical clustering another unsupervised learning technique that is used to cluster objects in clusters in a hierarchical structure that tend to share high degree of similarity. There are two types of hierarchical clustering: agglomerative (bottom-up) and divisive (top down). The agglomerative algorithm starts by treating each object as a singleton cluster and then successively merges pairs of clusters until all the clusters have been merged into a single large cluster. There are different methods available for merging the clusters and some of the popular methods will be discussed in chapter 3.

### 2.3.2. Classification

Classification is the process of using a training data set with known class labels to generate a classifier that can predict the class labels for new test data. Classification is also called the Supervised Learning technique. Machine-based classification techniques have wide applications. They are adopted for text categorization such as spam filtering, fraud detection, bio-informatics and the natural language process. Some examples of classification techniques *include Decision Tree Classifier, Rule-based induction, Neural Networks, Genetic algorithms and Bayesian networks.*

### 2.3.3. Association Rule Mining

Association Rule Mining (ARM) is a data-mining technique which involves finding association rules in large data sets, matching user-defined interests such as support and confidence. This technique involves mining all frequent itemsets and then deducing rules from the frequent itemsets. A simple example of an association rule is as follows:

*“baby diapers  $\rightarrow$  beer”*

The rule suggests that a person who buys baby diapers is very likely to purchase beer. ARM techniques have applications in genomics to study gene-expression data, in supermarkets to study transactional data, in the stock market to make predictions about stocks, etc.

For our research, we adopt a hierarchical clustering technique to analyze the data and visually represent the results in the form of a dendrogram. A “bottom-up” agglomerative approach is taken; each entity starts its own cluster, and a hierarchy is built progressively by merging the clusters.

The cluster-analysis results are used to study the semantic relationship between different usability attributes. The relationship between the usability dimensions is used to build a usability taxonomy. Finally, a unified usability construct is created.

The next chapter provides an Overview of the Approach and methodology. It details the criteria used for selecting published articles on mobile usability as well as the different pre-processing steps that were taken to prepare the document for extraction, including filtering, stemming and tokenization. Finally, our approach for text segmentation and text representation is discussed.

### CHAPTER 3. OVERVIEW OF THE APPROACH

The advent of mobile computing brings new opportunities and challenges. Among the challenges, mobile usability is a prominent one. The presence of non-conventional aspects such as mobility, device limits, and changing contexts makes ensuring usability difficult [26]. While some conventional usability principles and methods could be applied to mobile settings, some need to be revised. Over the last few years, researchers have proposed usability principles that are relevant to mobile computing; most of them are either: a) partially done or are not specified in a concrete or precise way, or b) too specific to a particular domain/field to be applied to other cases [26]. In this study, a computational linguistics approach using text-mining and information-extraction techniques is utilized to identify and mine usability attributes that are semantically related with the purpose of developing a usability construct for mobile computing.

This research is based on the extensively used Word-Space Model, which is a computational linguistic model of word meanings that represents the *semantic similarity* between words in terms of spatial proximity [14]. This model is based on *distributional patterns of words* over a large corpus of text. This research seeks to build a Word-Space Model for terms that are frequently utilized for usability attributes to identify all the usability terms that are semantically related in terms of the context (spatial proximity) they appear in the published articles on mobile usability. The notion of semantic relatedness is more general than semantic similarity because it also takes considers different functional relationships such as meronymy, antonymy, etc. [54].

The study used 16 usability attributes (terms) identified by Coursaris and Kim [15] as target words. After collecting the published articles based on pre-defined inclusion criteria, information-retrieval techniques were used to retrieve the usability attributes that are semantically related to the 16 target words (*usability attributes*) based on their spatial proximity.

Finally, all the usability terms identified, including the 16 target words, were clustered using the hierarchical clustering method. This approach can be grouped into five phases: collection, attribute identification, pre-processing, extraction and analysis.

The *collection phase* involved identifying the published articles that were used for information extraction. Multiple sources of published articles were considered. The relevant articles that met a pre-defined criterion were collected in electronic format. During the *attribute identification phase*, the selected papers were studied exhaustively to identify the usability dimensions that were considered for the research. A predefined selection criterion was used to choose the usability dimensions. In the *pre-processing phase*, tools were developed using open-source frameworks in order to retrieve information from the identified papers. This phase is important because data analysis is dependent on it. As part of pre-processing, the selected documents (in PDF format) were converted to plain-text format, retaining only the main textual body of the documents and excluding the references and other sections. Each document was then parsed to remove frequently occurring stop words based on a standard list of stop words. Finally, text in each document was stemmed and lemmatized using the *Porter Stemming* technique. In the extraction phase, each document was tokenized into a stream of words and parsed for the target words to identify the semantically related terms. The identified terms were organized using *term-term vectors* for analysis. In the *analysis phase*, all the usability attributes were analyzed with hierarchical clustering to identify their semantic relationships to develop a usability construct. The various phases of this approach are described in the following sections.

### **3.1. Collection Phase**

The first phase of the approach involved collecting published articles from the identified sources. These sources primarily involved the Association of Computing Machinery (ACM) digital library and the Institute of Electric and Electronic Engineers (IEEE) digital library. An exhaustive search was performed using the “Mobile Usability Dimensions” string. Only the published articles from 2012-2015 were considered. The journals and conferences that provided the most source included *ACM Transactions on Human-Computer Interaction*, the Mobile HCI conference, the ACM Conference on Human Computing Systems, the *International Journal of Human-Computer Interaction*, the *International Journal of Mobile Human Computer Interaction* and the *IEEE Transactions on Mobile Computing*.

A total of 130 published papers were collected through the search process. From these published articles, the papers that met the following exclusion criteria were rejected.

- a) Papers that did not involve discussions about mobile usability
- b) Papers that focused on mobile technology and application-development methodologies
- c) Papers that only focused on social aspects of using mobile applications

A total of 46 reviewed papers met the inclusion criteria; they were obtained electronically. All the articles were in PDF format.

### **3.2. Attribute Identification Phase**

This phase involved identifying the usability attributes that were considered for research to identify their semantic similarity. Papers were screened for the usability attributes that occurred regularly. The criterion used for selecting the dimensions was that they were mentioned in a minimum of three reviewed papers. From this screening process, 43 usability dimensions



were selected for study. The usability attributes that Coursaris and Kim [15] identified were considered as the target words for the research. This target word list includes the following usability terms.

- Efficiency
- Effectiveness
- Satisfaction
- Accessibility
- Adaptability
- Learnability
- Workload
- Enjoyment
- Acceptability
- Quality
- Aesthetics
- Utility
- Memorability
- Content
- Flexibility
- Playfulness

In natural language, people do not use the same terms to describe the same concept. In order to make the search more efficient, automatic word associations can be a great help. There are two different methods for computing word similarities that are based on a lexicographic database or statistics about a large corpus. WordNet is a freely available, lexicographic

knowledge base of the English language that was created by a group of researchers working in the area of cognitive science at Princeton in 1985. The goal of the database was to model the way concepts might be stored in the human mind [55]. WordNet contains nouns, verbs, adjectives and adverbs used in the English language. WordNet can be considered as a large thesaurus that is organized as a graph structure. The words in WordNet are linked together by their semantic relationships. WordNet stores synonyms of the words. Synonyms are words that have almost the same meaning and, in most cases, can replace another word without the structure of the sentence. A synset (synonym set) represents a collection of all synonyms based on some underlying concept. The synonyms contained in the synset are called lemmas. Synsets contain a list of words that appear together, such as “living thing” and “sit out,” or that can be used for the same concept. An example of an adjective synset from Wikipedia is as follows [56]:

**good, right, ripe** – (most suitable or right for a particular purpose; “a **good** time to plant tomatoes”; “The **right** time to act”; “the time is **ripe** to take action now”)

Each synset expresses a distinct concept and is linked to other synsets, forming a hierarchical structure of concepts.

The nouns in WordNet are organized hierarchically such that the ancestors (hypernyms) and descendants (hyponyms) of a noun are related. The hypernyms represents the supertype of a noun, and the hyponym represents the subtype of a noun. In the phrase “Car is a vehicle,” “vehicle” is the hypernym of the noun “car.” Alternatively stated, “car” is the hyponym for “vehicle.” The hypernyms and hyponyms of words provide a list of terms which can be used together. Each concept in WordNet is represented by a node in the hierarchy that is formed by the conjunction, or intersection, of all individual terms in the synset. Thus, the focus is more on

the relationships between concepts than the relationships between the words which represent them [55]. Thus, the same word can be part of another synset.

Because synsets are organized as a graph, it is possible to determine the semantic similarity between two synsets by determining the shortest path between them. The path distance is normalized into a distance metric and calculated by the following formula:

$$\textit{shortest\_distance} = \frac{1}{(\textit{synset}_x, \textit{synset}_y) + 1}$$

Apart from a lexicographical database such as WordNet, there are statistical methods that can discern word similarities by analyzing a large collection such as the Stanford WebBase Corpus at Sanford University, which is a large collection of snapshots of webpages from the Internet, and the English Gigaword Corpus, which is a comprehensive archive of newswire text data from different world news agencies that is managed by the Linguistic Data Consortium. The ebiquity Lab at the University of Maryland, Baltimore Campus (UMBC) developed a hybrid approach that combines both the thesaurus and statistical methods. UMBC statistical method is based on Latent Semantic Analysis (LSA) as well as the distributional similarity techniques applied with the WordNet database. The ebiuity lab provides a web-service Application Programming Interface (API) that allows users to search the Stanford WebBase Corpus as well as the English Gigaword Corpus for words that are semantically related.

UMBC's semantic service can be useful to identify the words that have semantically similarity with each usability attributes that was identified earlier. This information can be utilized to identify the sematic relatedness between a usability attribute and its semantically associated words as well as other usability attributes and their associated words. Our hypothesis is that if usability attributes are semantically related, then a similar relationship should exist between each of their semantically associated words. As an example, most usability researchers

and practitioners associate the usability element accuracy with effectiveness. There are other practitioners who use different terms, such as precision, to represent accuracy. Precision is one of the semantically related words that is part of accuracy. The expectation is that the similarity metric between effectiveness and precision would be close enough that they will form part of the same cluster group: Effectiveness.

The argument is that different usability practitioners have utilized various terminologies for the same or related usability concepts/dimensions/attributes. Thus, a hierarchy of usability terms can be extracted by utilizing Natural Language Processing. Figure 1 displays a screen shot of the UMBC Top-N Similarity Service interface showing search results for the usability attribute accuracy.

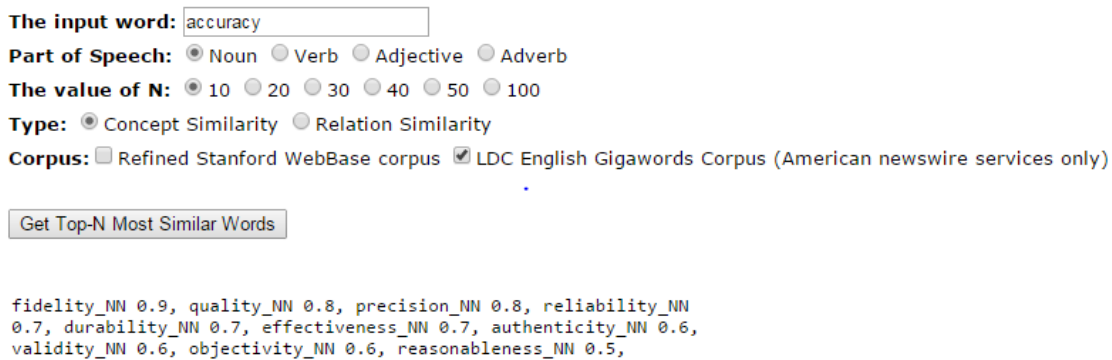


Figure 1. The top 10 search results of semantically similar terms for accuracy

Semantically related terms for each usability attribute were identified by searching the two corpuses. A list of semantically related terms for each of the 16 usability attributes are provided in Appendix B.

### 3.3. Pre-Processing Phase

Data analysis is dependent on appropriate pre-processing. Text mining is a form of data mining that retrieves knowledge from text data. Most text-mining tasks use information-retrieval

methods that require text documents to be pre-processed and converted to a simple format. The pre-processing steps used for text mining are different from the traditional methods used for pre-processing data in a relational database.

As a first step in pre-processing, the documents need to be stripped of all their meta-data. The documents which were downloaded electronically in a PDF format were converted to the plain-text format for information retrieval during the extraction phase. All the meta-data that indicate the document type, along with formatting details, the reference section, the keyword section, tables and figures, were stripped from the PDF files and later converted into plain text using PDF converters. The data stored in a plain file represents majority of the content that is in the document's body.

### ***3.4. Document Granularity***

It is necessary to define the document unit that will be used to organize the text data. Until now, each document file (article) was assumed to be a document. In a traditional UNIX email system, a file stores a sequence of email messages (an email folder) in one file. A better system would be to organize each email message as a separate, independent document. Further, emails may have attachments. Therefore, each email attachment needs to be its own document. For long documents, it is necessary to define the document's granularity. For example, it is possible to represent a book as a single, long document; a better organization would consider each chapter as a separate document. The granularity could be reduced further so that every section of the chapter, or even a sentence, can be a document unit. For the purpose of this study, we adopted each paragraph as a separate document. Thus, every paragraph in each chosen articles was treated as a separate document unit. Because a paragraph will serve as a good-sized unit with the usability terms well distributed and when extracting terms, spurious matches can be

avoided. If the granularity is further reduced, then it would be difficult to deduce important relationships with the terms. As the document unit was defined, the next step was to break each article into paragraph-sized chunks. Thus, all the articles were converted into documents that contained one paragraph.

### ***3.5. Tokenization***

In order to extract all the words, the long text in each document needed to be divided into separate units through a process called “tokenization,” a common task for text analysis. Most text-mining techniques require an unordered list of words called the “Bag of Words” as input.

As a first step in tokenization, it is necessary to define each token boundary, so each token could be stored as separate unit. With modern languages that are either Latin or Greek based, such as English, word tokens are delimited by a blank space. The commonly used delimiter for tokenization in Natural Language Processing is the whitespace. Some challenges with tokenization are how to handle punctuation such as commas, semicolons, apostrophes, periods and non-text characters. One strategy is to remove all the punctuations, tab spaces and other non-text characters with a single whitespace. In English and other Indo-European languages, a period follows immediately after the last word of the sentence. A whitespace after the period indicates that the period ends the sentence. On the other hand, if a period is followed immediately by a letter, it most likely indicates that the text is an abbreviation, so the period should be considered as an integral part of the sentence [57]. With hyphenated words, the hyphen is considered to be part of the token. There are challenges that deal with other formats, such as telephone numbers where the area code is inside brackets. For the purpose of this study, this challenge was ignored.

### **3.6. Filtering and Stemming**

An important step for pre-processing natural-language texts is *filtering* out the commonly occurring stop words. Stop words represent very frequently occurring and insignificant words that add little value to the knowledge being extracted. They are too common to search and, hence, are not indexed. Typically, about 40-50% of the total words in a document are removed with the help of a stop list [58]. As an example, the phrase “Student of Philosophy” contains the commonly occurring stop word “of”; eliminating it would result in a smaller number of terms with which to work: “Student” and “Philosophy”. There are different standard stop-words lists used to process natural-language texts. A standard stop-word list created by Lewis et al. [59] was used to parse the document to remove the stop words. A list of these stop words is provided in Appendix A.

Documents use different forms of a word, such as retrieve, retrieves and retrieving. In many situations, it is useful that a search for one of the words should return all the other words that are related. This standardization could be accomplished through the stemming process to morphologically normalize the words to their base forms.

Stemming is a process that strips the ends of words to remove the derivational affixes to produce the stem. The idea is to convert morphological forms of a word to its base-normalized form. In the area of computation linguistics, a stem represents the form the word which never changes, even when it is morphologically inflected. The stem of each word is obtained using a set of rules without considering the parts of speech. A word’s morphological forms are assumed to have the same meaning. All the variant forms for the terms would map to the stem after this process is completed.

As an example, the usability attribute *flexibility* includes the following the following morphological forms:

***Adjective forms:*** *flexible, flexile, hyperflexible*

***Noun forms:*** *flexibility, flexibleness, hyperflexibility, hyperflexibleness*

***Adverb form:*** *flexibly*

These morphological forms of the usability attribute *flexibility* can be normalized to the base form, flex, using an appropriate stemming algorithm. Each morphological form would match to the stem flex.

There are a number of stemming algorithms that have been developed in computer linguistics for different purposes. Broadly, they can be classified in three groups: truncating, statistical and mixed types. Algorithms in each category apply different methods to derive the stems for a word variant. For the purpose of this study, a popular truncating algorithm called Porter's Stemming [16] was adapted because it produces the smallest percentage of errors. The Porter's stemming technique truncates the suffixes and prefixes for each word. The technique is based on the idea that the suffixes in the English language are formed by a combination of simpler suffixes. The algorithm consists of five steps. Each step includes set of rules. The word that is stemmed is passed through each step, and as the rules apply, the word's suffixes are removed until the end of the last step [60].

The last step in pre-processing involves converting the entire text into the same case, so there is no issue of case sensitiveness. The text in all the documents is converted into the small case.



### 3.7. Extraction Phase

This phase includes steps to organize the text data and to make them conducive for information retrieval. An important step in the extraction phase requires that document data be organized in a format that is suitable for information retrieval. The most common format is to create a vector of words for each document. This format helps to reduce the documents' complexity, allowing easy handling during information retrieval. This approach is called a "Bag of words" model. Suppose  $D$  is the set of documents  $\{d_1... d_m\}$  and  $T = \{t_1, t_2... t_n\}$  is the set of all the different terms in  $D$ . Then, the frequency of a term,  $t \in T$ , in document  $d \in D$  is given by  $tf(d, t)$ . The term vector can be represented as  $t_d = (tf(d, t_1), \dots \dots \dots, tf(d, t_m))$  [50]. Next, term vectors and document vectors can be represented as a term-document matrix as shown in Figure 2.

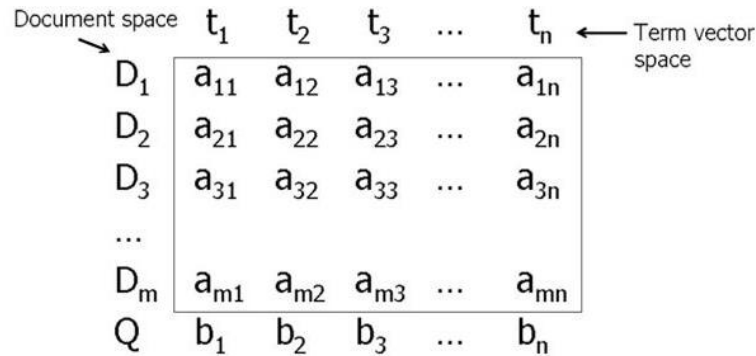


Figure 2. Matrix representation of document vector

Where,

Document  $D$  = a set of weighted keywords

Query  $Q$  = a set of non-weighted keywords

Vector Space = all the keywords found in the document, " $t_1, t_2, \dots, t_k$ "

Document  $D = \langle a_1, a_2, \dots, a_k \rangle$ ,  $a_i = \text{weight of } t_i \text{ in } D$

Query  $Q = \langle b_1, b_2, \dots, b_k \rangle$ ,  $b_i = \text{weight of } t_i \text{ in } Q$

Each document can be represented as a vector in three different ways: binary frequency, term frequency and term frequency-inverse document Frequency (*tf-idf*) weighing method.

1. Binary Frequency: With this method, binary values are stored in the matrix. A value of 1 indicates the term's presence in the document, and 0 indicates the term's absence.
2. Term Frequency: The matrix contains the frequency count of terms that appear in each document. The term frequency is based on the idea that words that occur frequently in a document represent its meaning well.
3. Term-Frequency and Inverse Document Frequency (*tf-idf*): This method uses a statistical measure that conveys the importance of a term to a document in a collection. The importance increases proportionally with the number of times a term appears in a document and is offset by the word's frequency in the collection.

Term-document vector matrixes are generally used for classifying documents rather than understanding the semantic or syntactic relationship between the documents. The documents' terms are mapped into n-dimensional space. The similarity between two documents can be found by computing the cosine similarity between their vector representations. This process is more useful for search engines where a term search returns all the documents that contain the word. A limitation for this approach is that term-document vectors do not consider the word's positional information in the document.

Another way to organize words in a document is to create positional indexes. In a *positional index*, for each *term*, the position(s) where its tokens appear is (are) captured. Below is an example of a positional index demonstrated by Manning et al. (2008) [61].

In Figure 3, the word “to” has a document frequency of 993,427, occurring 6 times in document 1 at positions 7, 18, 33, 72, 86 and 231. Suppose there is a query for “to be or not to be.” The position lists that need to be accessed are for the terms “to, be, or and not.” We start by intersecting the position lists for the terms *to* and *be*.

```

to, 993427:
  < 1, 6: <7, 18, 33, 72, 86, 231>;
    2, 5: <1, 17, 74, 222, 255>;
    4, 5: <8, 16, 190, 429, 433>;
    5, 2: <363, 367>;
    7, 3: <13, 23, 191>; ... >

be, 178239:
  < 1, 2: <17, 25>;
    4, 5: <17, 191, 291, 430, 434>;
    5, 3: <14, 19, 101>; ... >

```

Figure 3. Positional index example<sup>[61]</sup>

Next, we look for documents that contain both the terms *to* and *be*. In the example above, both terms occur in documents with ids 4 and 5. Then, we look for places in the lists where there is an occurrence of *be* with the token index 1 higher than a position of *to*, followed by looking for another occurrence of each term with token index 4 higher than the first occurrence. In the example lists, the pattern of occurrences is as follows:

```

to: < .....; 4:<....., 429, 433>; .....>
be: <.....;4:<....., 430, 434>; .....>

```

Below is the algorithm provided by Manning et al. (2008) [61] that captures the previously discussed process.

The algorithm described in Figure 4 is for proximity intersection of posting lists  $p_1$  and  $p_2$ . The algorithm finds places where the two terms appear within  $k$  words of each other and returns a list of tuples giving  $docID$  and the term positions in  $p_1$  and  $p_2$ .

```

POSITIONALINTERSECT( $p_1, p_2, k$ )
1   $answer \leftarrow \langle \rangle$ 
2  while  $p_1 \neq \text{NIL}$  and  $p_2 \neq \text{NIL}$ 
3  do if  $docID(p_1) = docID(p_2)$ 
4      then  $l \leftarrow \langle \rangle$ 
5           $pp_1 \leftarrow positions(p_1)$ 
6           $pp_2 \leftarrow positions(p_2)$ 
7          while  $pp_1 \neq \text{NIL}$ 
8          do while  $pp_2 \neq \text{NIL}$ 
9              do if  $|pos(pp_1) - pos(pp_2)| \leq k$ 
10                 then  $ADD(l, pos(pp_2))$ 
11                 else if  $pos(pp_2) > pos(pp_1)$ 
12                     then break
13                  $pp_2 \leftarrow next(pp_2)$ 
14                 while  $l \neq \langle \rangle$  and  $|l[0] - pos(pp_1)| > k$ 
15                     do  $DELETE(l[0])$ 
16                     for each  $ps \in l$ 
17                         do  $ADD(answer, \langle docID(p_1), pos(pp_1), ps \rangle)$ 
18                          $pp_1 \leftarrow next(pp_1)$ 
19                  $p_1 \leftarrow next(p_1)$ 
20                  $p_2 \leftarrow next(p_2)$ 
21             else if  $docID(p_1) < docID(p_2)$ 
22                 then  $p_1 \leftarrow next(p_1)$ 
23             else  $p_2 \leftarrow next(p_2)$ 
24 return  $answer$ 

```

Figure 4. An algorithm for proximity intersection <sup>[61]</sup>

A modification to this method is called a “sliding context window” approach where the positional index by contrast is iterated over the contents of a document and terms are indexed based on their neighboring terms. This approach is based on the idea that a word’s meaning is carried by the words that co-occur with it. The words inside a context window are considered

relevant and will be checked for consistency. The size of the “sliding context window” can vary from high granularity (a window) to low granularity (the entire document).

A context window size of  $N$  means that the window’s term frequency is  $N$  including the target term. Generally, the target term appears in the middle of the context window, so there about the same number of terms on either side. If the target term appears at the beginning or end of the window, then the total number of words in the context window will be less than  $N$ . A large context window increases the chance of finding words that are semantically related to the target term.

In our approach, the context window was a paragraph in the document. It was possible that number of words and sentences could vary from paragraph to paragraph. The document granularity was set as a single paragraph to reflect the context window size.

One of the objectives for this study was to cluster usability attributes that have the same semantic relatedness. This assumption was based on the conviction that *if any two usability attributes are semantically related, then their related words should also be related semantically.*

After documents were extracted into a stream of tokens (words), they needed to be organized in a format that is conducive to further analysis. They were arranged with a commonly used organization format called Vector Space model which organizes words into a term-document matrix.

### **3.7.1. Vector Space Model**

A document represents a “Bag of words.” Mathematically, a *bag* represents a collection of elements where duplicates are allowed and order does not matter. Suppose  $b_1 = \{a, b, b, b, c, c\}$  and  $b_2 = \{c, c, a, b, b, b\}$  are two “Bags of words”. Bag  $b_1$  can be represented using vector  $v_1 = \{1, 3, 2\}$ , where 1 represents the frequency count of element  $a$ , 3 is the frequency count of  $b$

and 2 is the frequency count of  $c$ . Similarly, bag  $b_2$  can be represented by vector  $v_2 = \{1, 3, 2\}$ ; then, bags  $b_1$  and  $b_2$  are considered to be equivalent.

As a large collection of documents, document vectors can be organized into a term-document matrix where row vectors correspond to terms and columns represent the document vectors. Suppose  $T$  represents our term-document matrix with  $n$  documents (columns) and  $m$  unique terms (rows). Let  $w_i$  represent the  $i^{\text{th}}$  term in the collection and  $d_j$  be the  $j^{\text{th}}$  column in  $T$ . The  $i^{\text{th}}$  row in  $T$  represents row vector  $x_i$ , and the  $j^{\text{th}}$  column in  $T$  represents the column vector. Element  $x_{ij}$  in  $T$  is the frequency count for the  $i^{\text{th}}$  term,  $w_i$ , in the  $j^{\text{th}}$  column,  $d_j$ .

Vector  $v$  is represented as sum of elements:

$$v = a_1 v_{i1} + a_2 v_{i2} + \dots + a_n v_{ik},$$

where  $a_k$  is weights and  $v_{ik}$  is elements.

In a Vector space model, a set of documents is represented as vectors in the vector space, where each axis represents a term.

Suppose  $V(d)$  represents the vector derived from document  $d$ , with one element of the vector being a term. In vector space, each term is represented by an axis. Figure 5 shows a Vector Space Model consisting of two documents,  $d_1$  and  $d_2$ , and a query vector,  $Q$ . The assumption is that the space only contains two terms,  $t_1$  and  $t_2$ , which represented by the axis.

The Vector Space Model is mostly used for determining document similarity measured by their cosine similarity. There are multiple applications of document similarity: document retrieval, document clustering, document classification, document segmentation and more [2].

In our approach, we seek to transform a vector space model into a word-space model. A word-space model provides a spatial representation of word meanings. This model is based on the idea that semantic similarity can be represented in  $n$  dimensional space. The model provides

an approach to perform Word Sense Discrimination, the process of automatically identifying the senses (meanings) of a word. The spatial proximity between words indicates their semantic relatedness. The concept is based on the metaphor that “two things that are deemed to be similar in some sense are conceptualized as being close to or near each other” [14]. The word-space model organizes words into term-term matrices which are suited for measuring the semantic

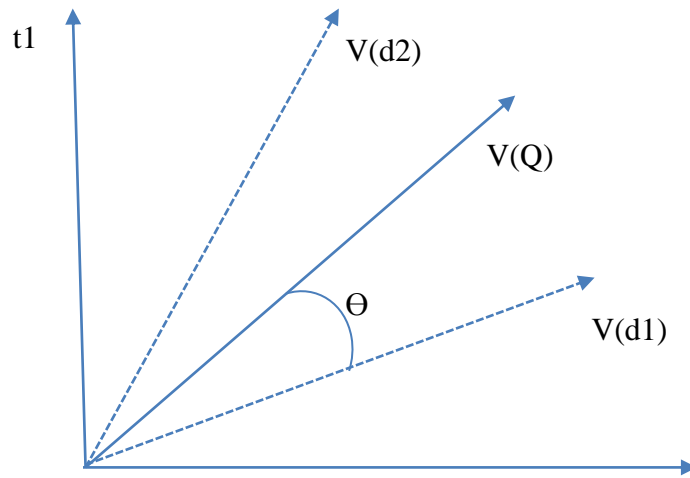


Figure 5. Vector Space Model of documents, d1 and d2, assuming there are only two terms

similarity between words. The idea behind this model is that a word’s meanings are carried by words that co-occur with it and that two words are semantically related if they tend to co-occur with the same words [62]. In this model, word similarity is measured using the cosine of the angle between row vectors in a word-context matrix [2]. Some common applications for measures of word similarity include word clustering, word classification, automatic thesaurus generation, word sense disambiguation, etc.

In our approach, a Java-based framework, Semantic vectors package [63], was used to construct the word-context matrices. They can be built using an unsupervised distributional

analysis of the text. The concept for the semantic vector model was derived from the Vector Space Model where each document is represented as a vector of terms.

One of the big problems for text processing is how to deal with the problem's high dimensionality. The Semantic vectors package [63] relies on a dimensional reduction technique called Random Indexing (RI). This technique allows semantic vectors to be built without any term-term matrix factorization. Thus helping improve dimensionality reduction efficiently. RI removes the need to create a huge co-occurrence matrix. Instead of collecting co-occurrences in a co-occurrence matrix and then extracting the context vectors, this technique incrementally accumulates context vectors which can easily be reconstituted into a co-occurrence matrix [14].

The co-occurrence matrix stores counts for the terms that occur in the same context. The cosine similarity measures the cosine of the angle between the two context vectors and computes the normalized vector similarity between two context vectors,  $\vec{x}$  and  $\vec{y}$ , as defined as follows:

$$sim_{cos}(\vec{x}, \vec{y}) = \frac{x \cdot y}{|x||y|} = \frac{\sum_{i=1}^n x_i y_i}{\sqrt{\sum_{i=1}^n x_i^2} \sqrt{\sum_{i=1}^n y_i^2}}$$

Cosine measure is one most frequently utilized similarity metrics in word space research and is efficient to compute [14]. The cosine measure corresponds to taking scalar products of the vectors and then dividing by the norms. This method is convenient that it provides a fixed measure of similarity between 1 and -1. A cosine measure of 1 means the vectors are identical; measure 0 illustrates that vectors are orthogonal; and measure -1 means that the vectors are pointing to opposite directions.

The term-term matrix containing the cosine measure between occurrences of the terms was generated, and a hierarchical cluster analysis was performed on them. Because the words were in their stemmed form, they were reconstituted in either their noun or adjective forms.



Each document was processed using the following steps:

1. Create a term-document vector for each document unit: As the first step of document processing, each document was passed through the Tokenzier to tokenize the terms using word boundaries.
2. Building term-document vectors: Each stream of tokens from a document produced by the Tokenzier was built into a term-document vector followed by an inverted index which mapped each term to the document that contained it. An inverted index speeds up the search and semantic interpretation.
3. Building a term-term matrix: The next step is to create a Term-Term matrix of all the usability attributes and their semantically similar terms. The matrix represents the cosine measures of usability attribute pairs. The term-term matrix was saved as a csv file for export to Excel.
4. Generate the cosine measure for co-occurrence: In this step the cosine measures of co-occurrences of usability terms is generated.
5. Restoring the words back from their stems: Finally, in order to allow for easy readability, the words are restored back from their stems.

### ***3.8. Analysis Phase***

There are several different methods available to measure the distance (similarity or dissimilarity) between rows or between columns of the data matrix, depending upon the measurement scale for the observations. In the term-term matrix obtained earlier, the distance between the terms is represented using the cosine similarity between term pairs.

Hierarchical clustering builds a hierarchy of structures based on a similarity function. The focus of hierarchical clustering is to build a binary tree by successively merging/splitting

similar/dissimilar groups of points. There are two approaches for hierarchical clustering: divisive (top-down) and agglomerative (bottom-up). With divisive clustering, all data points are initially considered as part of one large cluster, and they are split into two iteratively by using some measure of dissimilarity until there is a singleton cluster. A more popular approach is agglomerative clustering where each data point is treated as a singleton cluster, and then, pairs of similar clusters are iteratively merged based on the measure of similarity until there is single cluster. The results of hierarchical clustering can be presented visually as a binary tree-like structure called a dendrogram. With agglomerative clustering, there is one large cluster at the top that contains all the data points, and at the bottom of the tree, each data point is represented by a singleton cluster. For the purpose of this research, agglomerative hierarchical clustering is used to cluster 76 usability terms.

Agglomerative clustering algorithms uses three different methods to determine the pairwise similarity between the clusters: best-case similarity (single-linkage clustering), average similarity (average-linkage clustering) and word-case similarity (complete-linkage clustering).

- **Single-Linkage Clustering:** This method defines the best-case similarity where the shortest distance between any pair of data points between the two clusters is used to merge the clusters.

$$distance(A, B) = \min_{i \in A, j \in B} d_{ij},$$

where  $A$  and  $B$  are any two clusters,  $i \in A$ ,  $j \in B$ , and  $d_{ij}$  is the distance between points  $i$  and  $j$ .

- **Average-Linkage Clustering:** The method is based on the average similarity where the shortest average distance between all pairs of data points in the two clusters is used to merge the clusters.

$$distance = \frac{1}{N_A N_B} \sum_{i \in A} \sum_{j \in B} d_{ij},$$

where  $A$  and  $B$  are any two clusters,  $i \in A, j \in B$ , and  $d_{ij}$  is the distance between points  $i$  and  $j$ .  $N_a$  and  $N_b$  are the number of points in  $A$  and  $B$ , respectively.

- Complete-Linkage Clustering: This method is based on the worst-case similarity or, alternatively stated, the best-case dissimilarity. This method uses the maximum distance between any two pairs of points among cluster pairs for merging.

$$distance(A, B) = \max_{i \in A, j \in B} d_{ij},$$

where  $A$  and  $B$  are any two clusters,  $i \in A$ , and  $j \in B$ .

- Ward's method: The clustering method used for this dissertation is Ward's method, where the criterion used to merge a pair of clusters at each level is based on maximizing the sum of squares upon merging. The sum of squares reflects the cluster quality: how tight and cohesive is the cluster? The merging cost (SS) of two clusters,  $A$  and  $B$ , is calculated as follows:

$$SS(A, B) = \frac{n_A n_B}{n_A + n_B} \|c_A - c_B\|^2,$$

where  $n_A$  and  $n_B$  are number of data points, and  $c_A$  and  $c_B$  are centroids of clusters  $A$  and  $B$ , respectively. The goal of Ward's method is to minimize the cost as a pair of clusters are merged.

Apart from the clustering method, the other decision that needs to be made is where to cut the tree to determine the optimum number of clusters? There are different stoppage rules or criteria developed to select the optimum number of clusters. One popular stoppage criterion is the *sum-of-squares-based indexes* which measure the cluster quality in terms of how tight each cluster is and how much effect the clusters have on one another.

With the *sum-of-squares-based indexes*, the group variance within a cluster and between clusters is calculated as the sum-of-squares within cluster (*SSW*) and between clusters (*SSB*), respectively. Suppose  $D = \{d_1, \dots, d_n\}$  represents the data set with  $N$  data points, and the center of the data set is  $\bar{D} = \sum_{i=1}^N x_i / N$ . Let  $C = \{c_1, \dots, c_m\}$  represent the set of  $M$  clusters. *SSW* and *SSB* are calculated as follows:

$$SSW = \sum_{j=1}^M \sum_i^N \|d_i - c_j\|^2, \quad i \in c_j$$

$$SSB = \sum_{i=1}^M n_i \|c_i - \bar{D}\|^2$$

For the purpose of this research, an analysis was performed between variances (*SSB* and *SSW*) and the number of clusters as shown in Figure 6.

In the next chapter, the results of the hierarchical clustering are presented and analyzed. A usability taxonomy is developed by utilizing the identified usability attributes. Finally, a unified usability construct is created.

## CHAPTER 4. RESULTS AND DISCUSSION

One important objective for this research is to build a unified usability construct for mobile applications that captures all the usability dimensions and attributes that impact usability. Further, a taxonomy of usability attributes is created. The proposed usability construct is built with an existing body of HCI literature. The suggested, novel methodology is based on a text-mining technique that, in combination with computational-linguistic concepts, identifies the semantic relatedness between usability attributes.

The first objective of this research was to identify the usability terms that are relevant to mobile applications. The approach initially identified usability attributes that are commonly mentioned in the mobile-usability literature. A popular usability-attribute list generated by Coursaris and Kim [15], consisting of 16 usability attributes called seed terms, were considered. These 16 usability attributes were searched against 2 popular lexicographical databases by utilizing the created custom tools. The top 7 highly associated, semantically related terms for each of the 16 target usability words were identified. This search produced a total of 112 term matches. These matching terms were combined with the original 16 to produce 128 target words. The list of original usability terms, along with their matches, is provided in Appendix B.

The initial analysis of these target terms showed multiple overlaps because the same words appeared as matches for multiple usability terms. As an example, accuracy was a match for the usability terms effectiveness, efficiency and quality while *reliability* was a common match for *efficiency* and *effectiveness*. The duplicate matching terms were removed, thus the target word count was reduced to 106. After stemming, more duplicates were identified; for example, *accessibility* and *accessible* resulted in a common stem, *access*. Eliminating the duplicate stems resulted in 76 usability target terms. The resulting usability target terms met the

first objective of the research, which is identifying the usability attributes that were relevant to mobile applications. A listing of all 76 usability terms is provided in Appendix C.

The second objective of the study was to generate clusters of usability attributes that are semantically related using a novel computational-linguistic approach. In order to meet this objective, the 76 usability attributes identified earlier were text-mined, and processed with term-document and term-term vectors as discussed in Chapter 3. The output for this process was a term-term matrix of 76 x 76 usability terms.

Next, agglomerative hierarchical clustering using a cosine coefficient and the complete linkage method was used to cluster the usability terms. The stoppage criterion for clustering was using the sum-of-squares-based technique as discussed in Chapter 3. An analysis was performed between the variances (SSB and SSW), and the clustering results are shown in Figure 6.

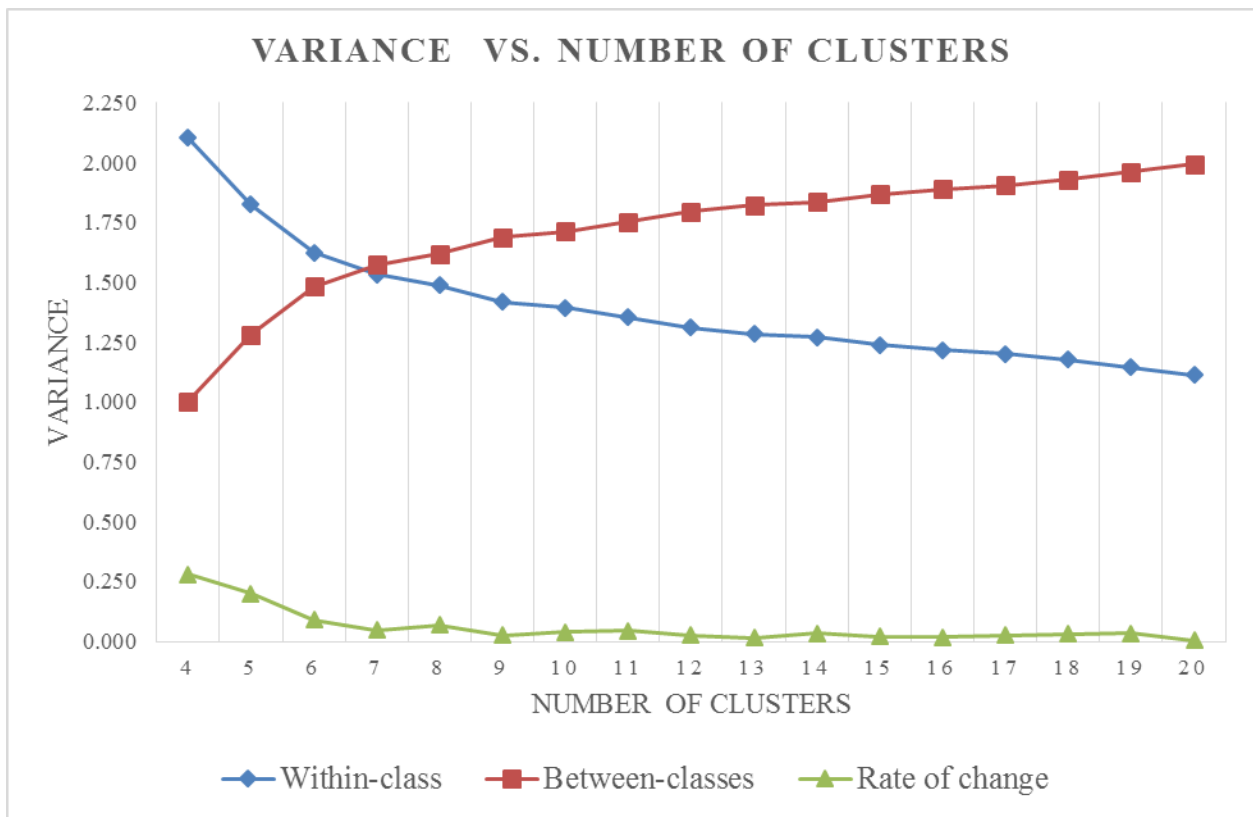


Figure 6. Variance versus number of clusters

The graph in Figure 6 traces the variance within cluster, variance between clusters and rate of change for the variance within-cluster against the number of clusters. The knee-point on the rate of change for the variance within-class graph indicates that the optimum cluster size is 7.

The variance decomposition for various cluster sizes is provided in Table 2. For the first time, the table indicates that the variance within-cluster (49.36%) becomes less than the variance between-classes (50.64%) for a cluster size of 7. The intersection between the graph's variance within-clusters and variance between-clusters also happens at a cluster size of 7 as depicted in Figure 6.

The sum-of-squares index analysis proves that the optimum cluster is 7. All 76 usability terms were clustered into 7 clusters. The resulting clusters are visually summarized by the dendrogram in Figure 7. The usability term members for each cluster are provided in Table 3

The labels used for each cluster are based on the common theme of the usability attributes that grouped together. The cluster labels provided are as follows:

1. System Dimension
2. Ergonomic/Contextual Dimension
3. Performance Dimension
4. Reliability Dimension
5. Satisfaction Dimension
6. Easy-to-Learn/Retain Dimension
7. Protection Dimension

Table 2. Variance decomposition for different cluster sizes<sup>1</sup>

Variance\Clusters	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20
<b>Within-cluster</b>	2.104 (67.75%)	1.824 (58.72%)	1.623 (52.27%)	<b>1.533</b> <b>(49.36%)</b>	1.486 (47.86%)	1.419 (45.68%)	1.393 (44.86%)	1.354 (43.59%)	1.310 (42.19%)	1.284 (41.34%)	1.270 (40.88%)	1.237 (39.83%)	1.218 (39.21%)	1.202 (38.69%)	1.176 (37.85%)	1.144 (36.83%)	1.111 (35.79%)
<b>Between-clusters</b>	1.001 (32.25%)	1.282 (41.28%)	1.482 (47.73%)	<b>1.573</b> <b>(50.64%)</b>	1.619 (52.14%)	1.687 (54.32%)	1.712 (55.14%)	1.752 (56.41%)	1.795 (57.81%)	1.822 (58.66%)	1.836 (59.12%)	1.869 (60.17%)	1.888 (60.79%)	1.904 (61.31%)	1.930 (62.15%)	1.962 (63.17%)	1.994 (64.21%)
<b>Total</b>	3.106 (100%)	3.106 (100%)	3.106 (100%)	<b>3.106</b> <b>(100%)</b>	3.106 (100%)	3.106 (100%)	3.106 (100%)	3.106 (100%)	3.106 (100%)	3.106 (100%)	3.106 (100%)	3.106 (100%)	3.106 (100%)	3.106 (100%)	3.106 (100%)	3.106 (100%)	3.106 (100%)

---

<sup>1</sup> Percentages are provided in parentheses. For cluster size 7 (displayed in bold), for the first time, the within-cluster variance becomes less than variance between clusters.



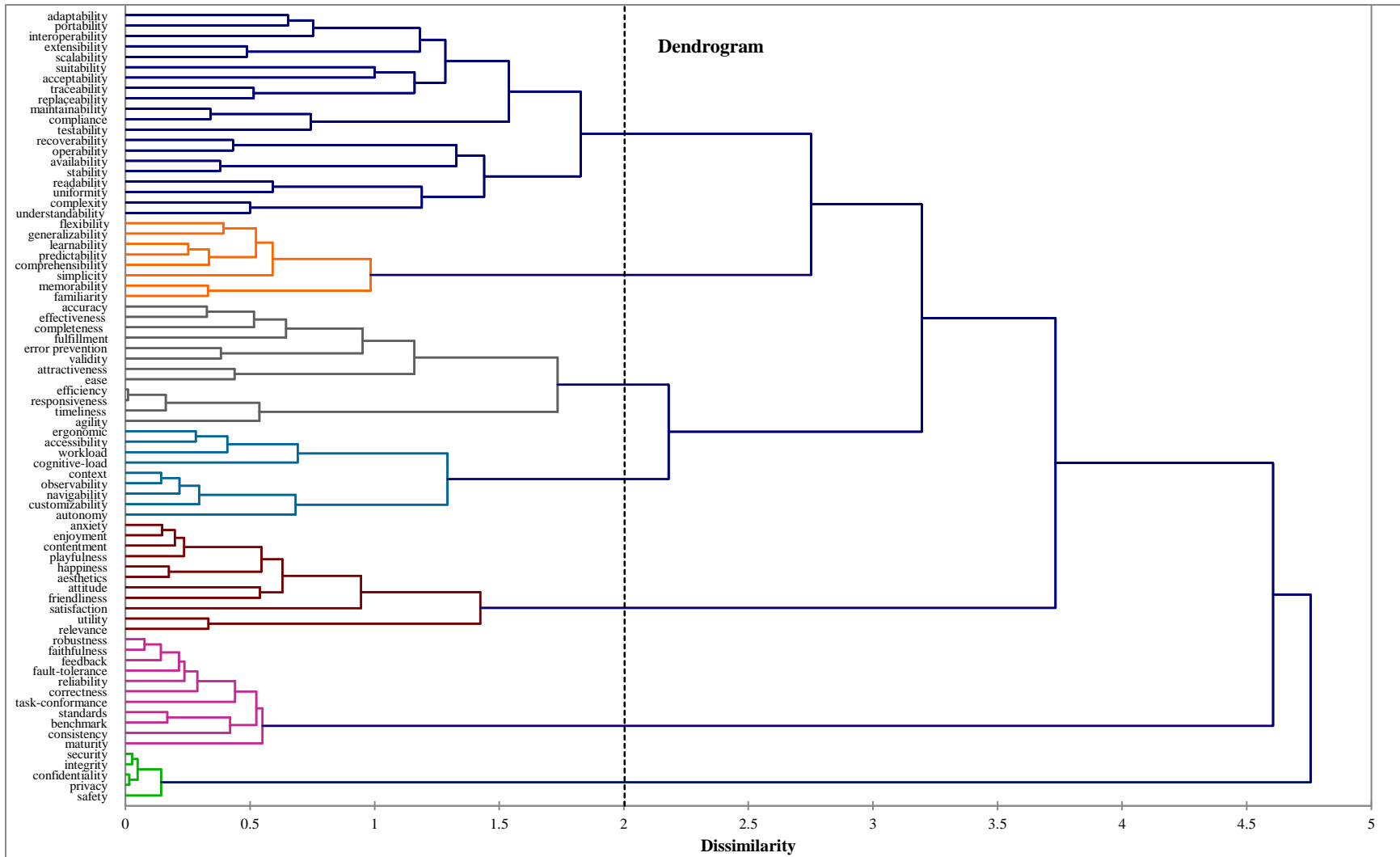


Figure 7. Dendrogram produced from hierarchical clustering of 76 usability attributes

Table 3. Usability terms grouped into seven clusters

System Dimension	Ergonomic/ Contextual Dimension	Performance Dimension	Reliability Dimension	Satisfaction Dimension	Easy-to-Learn/Retain Dimension	Protection Dimension
<b>Adaptability</b>	<b>Ergonomic</b>	<b>Effectiveness</b>	<b>Maturity</b>	<b>Contentment</b>	<b>Learnability</b>	<b>Privacy</b>
<i>interoperability</i>	<i>accessibility</i>	<i>validity</i>	<i>robustness</i>	<i>Enjoyment</i>	<i>Predictability</i>	<i>privacy</i>
<i>portability</i>	<i>ergonomic</i>	<i>completeness</i>	<i>correctness</i>	<i>Happiness</i>	<i>Comprehensibility</i>	<i>confidentiality</i>
<i>scalability</i>	<i>workload</i>	<i>accuracy</i>	<i>fault-tolerance</i>	<i>Anxiety</i>	<i>Learnability</i>	<b>Security</b>
<i>extensibility</i>	<i>cognitive-load</i>	<i>error-prevention</i>	<i>faithfulness</i>	<i>Playfulness</i>		<i>security</i>
<i>adaptability</i>		<i>effectiveness</i>	<i>maturity</i>	<i>Aesthetics</i>	<b>Memorability</b>	<i>integrity</i>
<i>suitability</i>	<b>Context</b>	<i>ease</i>	<i>reliability</i>	<i>Contentment</i>	<i>Simplicity</i>	<b>Safety</b>
<i>acceptability</i>	<i>context</i>	<i>fulfillment</i>	<i>feedback</i>	<i>Attitude</i>	<i>Flexibility</i>	<i>safety</i>
	<i>autonomy</i>	<i>attractiveness</i>	<i>task-conformance</i>	<i>Friendliness</i>	<i>Generalizability</i>	
<b>Maintainability</b>	<i>navigability</i>			<i>Satisfaction</i>	<i>Familiarity</i>	
<i>testability</i>	<i>customizability</i>	<b>Efficiency</b>	<b>Consistency</b>	<b>Utility</b>	<i>Memorability</i>	
<i>compliance</i>	<i>observability</i>	<i>timeliness</i>	<i>standards</i>	<i>Utility</i>		
<i>traceability</i>		<i>responsiveness</i>	<i>benchmarks</i>	<i>Relevant</i>		
<i>replaceability</i>		<i>agility</i>	<i>consistency</i>			
<i>maintainability</i>		<i>efficiency</i>				
<b>Availability</b>						
<i>stability</i>						
<i>operability</i>						
<i>recoverability</i>						
<i>availability</i>						
<b>Understandability</b>						
<i>understandability</i>						
<i>uniformity</i>						
<i>complexity</i>						
<i>readability</i>						

The largest cluster, *System Dimension*, consisted of 20 usability terms, followed by the *Performance Dimension* with 12 usability terms, *Reliability* with 11 terms, *Ergonomic/Context Dimension* and *Satisfaction Dimension* with 9 usability terms each, the *Easy-to-Learn/Retain Dimension* with 8 terms and the *Protection Dimension* with 5 usability terms. To help visualize how the usability dimensions are clustered, a mind-map is shown in Figure 8.

The next section provides a discussion about the usability attributes that congregated together and their common themes. This would be used to develop a usability taxonomy for mobile applications.

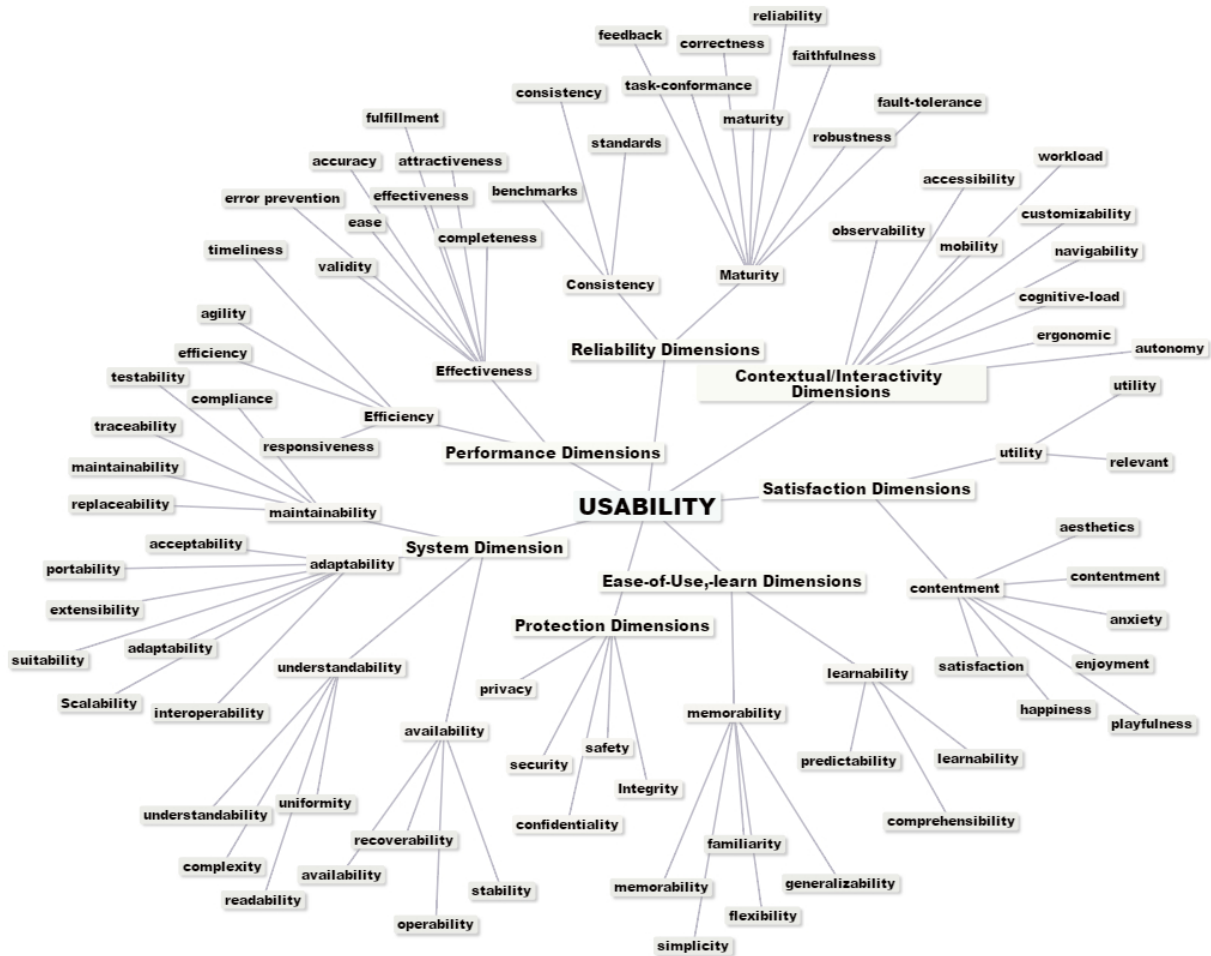


Figure 8. Mind map showing the clustering of 76 usability terms

#### ***4.1. Discussion about Clustered Usability Terms and the Usability Taxonomy***

The third objective of this research is to generate a usability taxonomy. Over the years, several definitions of usability and what constitutes it have been proposed by usability researchers and practitioners. The usability definitions and frameworks were reviewed in Chapter 2. There is no consensus yet on the concept of usability, either by the researchers or by the standard organizations [64]. Similarly, there is ambiguity about definitions of usability attributes and usability dimensions that they should they be grouped with. A number of usability classifications have been created for different purposes. With most of these classifications, the usability attributes' definitions are short and ambiguous, and the usability attributes tend to differ from one definition to another [65]. Thus, there is a strong need for an exhaustive taxonomy on usability terms. This research developed a hierarchical model for usability taxonomy that is based on the 76 usability attributes that were initially identified and the results of the hierarchical clustering.

This section discusses the clustering results, analyzing the usability terms that clustered together, in detail, and their relationship between each other. As the usability terms are described, a parallel usability taxonomy is also detailed.

Each of the seven clusters which were formed around a common theme were further divide into sub-themes reflecting the common thread between the small clusters of usability terms. These groups, based on sub-themes, were formed by utilizing the way the usability attributes are associated together (linkage distance) within a cluster.

##### **4.1.1. System Dimension**

The usability terms in this group seem to relate to some of the common, desirable system characteristics for mobile applications. They can also be regarded as non-functional attributes

that are addressed during the design phase of the development life cycle. Because there were 20 usability terms in this cluster, they were further grouped based on sub-themes. The identified sub-themes were Adaptability, Maintainability, Availability and Understandability.

a) Adaptability

The usability attributes that are part of this sub-theme relate to a system's ability to make appropriate changes in view of the changing environment. The usability terms that are part of this group include adaptability, portability, inter-operability, scalability, extensibility, suitability and acceptability.

Adaptability is a system's ability to adapt to a changing environment. The mobile application needs to be adaptable to different user needs and interactions. A close, semantically related term, *Portability*, according to ISO 9126-1, deals with the ease with which the software can *adapt* to changing environments and requirements. This ability allows a mobile application to be compatible with different platforms (operating systems and hardware) without much modifications. Unlike the PC domain, there is a wide choice of operating systems and hardware (processor or memory) available in the mobile domain. A mobile application needs to be portable on different platforms in order to support a wide user community.

According Wikipedia [66], extensibility "is a system design principle where implementation takes future growth into consideration." Extensibility is regarded as the measure of the ability to extend the system's functionality. A good mobile application should be extendible by either adding new functionality or modifying the existing functionality.

Interoperability is the software program's capacity to communicate, exchange and interpret data seamlessly. Most mobile applications are required to share resources and services

as well to exchange information with each other. The information exchange should happen with little input or knowledge from the user.

Another important usability term that follows the same theme of adaptability is Scalability. Scalability refers to the system's capacity to perform well under the increased volume of work to meet a user's need.

According to [46], acceptability simply refers to whether and how the product will be used in real world. It is important that mobile applications continue to provide value to the end user in terms of the task that they can accomplish. A product's acceptability is dependent on the end user's actual usage for a particular task and in a specific context [46].

Suitability, according to ISO 9126, is "the capability of the software product to provide an appropriate set of functions for specific tasks and user objectives" [49]. Suitability is measure of the extent to which an application can fulfill a specific user's needs. A mobile application is acceptable if it is suitable for use by a specific type of user in a particular context.

The relationship between the usability terms that are part of this group are reflected in their semantic relatedness. Adaptability, portability, scalability, and interoperability are semantically related to each other. On the other hand, adaptability is related to extensibility through another semantically related term, adjustability.

*Adaptability ↔ Portability ↔ Scalability ↔ Interoperability*

*Adaptability ↔ (Adjustability) ↔ Extensibility*

#### b) Maintainability

A good mobile application also needs to be easily maintainable. This sub-theme includes usability attributes related to maintainability, such as compliance, traceability, replaceability and

testability. Maintenance involves the software's ability to be easily updated and maintained with changes around it.

According to [67], "Adaptive maintenance is a process used to adapt software to new environments or new requirements due to the evolving needs of new platforms, new operating systems, new software, and evolving business requirements." Mobile applications must continue to provide quality and performance by easily and quickly adapting to changes.

One important attribute that is grouped under this sub-theme is *compliance*. According to [68], usability compliance is "the capability of the software component to adhere to standards, conventions, style guides or regulations relating to usability." It is important that mobile applications are compliant with the standards that might be related to the domain, or operating system, and other factors. A good example of domain-specific compliance, mobile health applications, such as mobile personal health records (PHRs), need to be in compliance with the Health Insurance Portability and Accountability (HIPAA) laws. Other examples of the compliance requirement include guidelines such as the Android Design Guidelines and the iOS guidelines which a developer would need to follow to publish his apps in their stores [69]. As the compliance requirements change, it is necessary that mobile applications be updated to be compliant.

Easy maintenance requires that all requirements be traceable through the different phases of the software development life cycle. Software requirements need to be traced through the design, coding and testing phases. Each requirement should be related to artifacts, code and tests through these phases. Traceability aids in easy maintenance and updates when making changes to a specific functional area of the mobile application. Traceability also helps with replaceability.

Replaceability is a system's ability to easily substitute a component with a new, better component with minimal changes to other system components. Traceability helps identify the impacted documents and artifacts, thus allowing for easy replaceability of the component.

Finally, the last usability attribute for this sub-theme is *testability* which refers to the ability of the system to be easily testable in order to identify problem areas quickly. As part of a good testing process, all test artifacts should be maintained appropriately any time that a system's functional area or component is updated. The usability terms that are part of maintainability sub-theme are also semantically related:

*Maintainability* ↔ *Testability* ↔ (*Verifiable, Validate*) ↔ *Compliance* ↔ (*Tractable*)  
↔ *Traceability* ↔ (*Flexibility, Substitutability*) ↔ *Replaceability*

Maintainability is related to testability which is related to compliance through terms that are not part of this cluster: verifiable and validate. Compliance is related to traceability through the term tractable. Tractable is related to replaceability through the terms flexibility and substitutability.

c) Availability:

The availability sub-theme deals with the system's ability to be available for users to request services. The usability terms that are part of this group include availability, stability, operability, and recoverability.

Availability is defined, according to IEEE Std. 610.12, as "the degree to which a system or component is operational and accessible when required for use." It is important that mobile applications are available and ready to provide the services and functionality that they are expected to provide when requested by users.



A mobile application must exhibit stability under changing conditions and must be able to provide the same performance to end users as they interact with the application. As software accumulates errors and changes over time, the performance and availability is impacted. Many times, mobile applications crash due to a lack of enough computing resources on the device, a miss-step on the part of the user or due to bugs that were not accounted for during the testing stage. It is important that stability is given importance during the specification and design phase of the mobile application's development.

Recoverability is the ability of a system to quickly recover from an error condition so that it is available to let the user perform tasks to accomplish the goals. Applications should not only be designed to prevent errors, but they should also be able to recover easily when users make mistakes [29]. Mobile applications should also allow end users to recognize errors through appropriate system feedback and to recover from them to continue accomplishing the goals. It is particularly important in the mobile context which reflects real world and in the real world, people are expected to make mistakes [3].

Another related term for availability is operability. According to ISO 9126 [49], operability is “the capability of the software product to enable the user to operate and control it.” In order for the system to be operable, it should be available for the end user in a safe and reliable functioning condition as per the operational requirements [70].

The usability terms that are part of this sub-theme, availability, are semantically related, some directly and a few indirectly. The semantic relationship between these terms is as follows:

*Availability ↔ Responsiveness*

*Availability ↔ (Suitability ↔ Feasibility) ↔ Operability*

*Availability ↔ (Suitability) ↔ Stability*

As seen earlier, availability and responsiveness are related semantically. There is no direct semantic relatedness between availability and operability and between availability and stability. Operability is indirectly related to availability through feasibility and suitability, whereas stability is related indirectly through suitability.

d) Understandability

Understandability is the property of a system to communicate the requested information, the output for a user's task, in a lucid manner without any ambiguity while promoting understandability. It is important for mobile applications to communicate information to users in a transparent and understanding manner. The usability terms that are part of this sub-theme relate to promoting understandability and include understandability, uniformity, complexity, and readability.

According to [49], understandability is defined as “the capability of the software product to enable the user to understand whether the software is suitable, and how it can be used for particular tasks and conditions of use.”

Uniformity is the system characteristic that promotes homogeneity and regularity which will enhance predictability and understanding. Uniformity breeds familiarity which, in turn, promotes user understanding.

Complexity is the system characteristic in terms of the difficulty the system poses for an end user to accomplish his/her tasks while interacting with the system. The lower the complexity, the greater the comfort level a user perceives while interacting with the system. A good design goal for usability is to reduce complexity and to promote simplicity. Thus a good design helps promote understandability.

The last attribute that is part of this sub-theme is readability. It is a quality that reflects the ease with which users understand the material presented to them. This attribute is related to comprehensibility which is grouped under the *Ease to Learn/Retain* cluster. The usability attributes that are clustered in this group are semantically related as follows:

*Understandability* ↔ *Uniformity*

*Understandability* ↔ *Complexity*

*Understandability* ↔ (*legibility*) ↔ *Readability*.

As seen earlier, understandability is directly related semantically with uniformity and complexity. On the other hand, understandability is semantically related to readability through legibility.

All the usability attributes that converged under this cluster are non-functional requirements that are typically addressed during the design phase. A usability practitioner, as a mobile application developer, will find the usability attributes that cluster under different sub-themes in this group useful for consideration, depending on the applications that are being built and their context for use.

#### **4.1.2. Performance Dimension**

The core usability attributes were grouped in this cluster. Most studies, as seen in the review of literature, identified efficiency and effectiveness as the most common usability attributes. Many of the usability attributes that congregated in this cluster are related to effectiveness and efficiency. The common theme for this cluster was related to the system's performance. According to [48], performance is defined as “the degree to which a system or component accomplishes its designated functions within given constraints, such as speed,

accuracy, or memory usage.” The cluster was divided into two groups, representing different sub-themes: effectiveness and efficiency. These sub-themes are discussed next.

a) Effectiveness

Effectiveness is the ability of a system to produce the desired goals for users. The usability attributes that are grouped in this sub-theme are related to effectiveness. They include effectiveness, accuracy, completeness, error prevention, ease, fulfillment, validity and attractiveness.

According to [49], effectiveness is “the accuracy and completeness with which users achieve specified goals.” As the definition shows, accuracy and completeness are strongly associated with effectiveness. This attribute is useful to assess the improvement of a mobile application’s new version [20].

Accuracy is a measure for the degree of correctness. It can be quantified as “the number of errors users make either during the process of completing tasks or in the solution to the tasks” [36]. The mobile applications’ usability designers need to aim for high accuracy so that users can easily perform tasks with the fewest number of errors and that the results produced by the application are as correct.

Completeness measures the extent to which a system completes the tasks. Unlike accuracy, where the focus is on the errors made by users while performing a task, this measure captures effectiveness in situations where users reached solutions for a different completeness [36].

Error prevention is the system’s ability to prevent problems from occurring in the first place [29]. This ability relates to providing continuous feedback to the user about the system’s status so that errors can be prevented. This attribute has a great impact on the system’s overall

usability when tasks are time-sensitive and completion-sensitive [2]. Han et al. [71] identified error prevention and effectiveness (accuracy and relatedness) as two measures for errors in usability analysis.

Validity refers to a system's ability to consistently generate correct results. A mobile application should be able to consistently produce the correct results for a specific task in a specific context.

Fulfillment is the usability attribute that requires that the purpose of the interaction that a user initiates would be completed and successful. It is semantically related in a close way to Satisfaction which is actually grouped under the heading Productiveness. When checked against the WordNet lexicographical database, fulfillment has two senses as shown in Figure 9 “the feeling of satisfaction at having achieved your desire” and “the act of consummating something (a desire of promise etc.).” Fulfillment is also semantically related to Performance, the execution of a task to successfully to fulfill its intended purpose.

Ease is another attribute that is part of the effectiveness sub-theme. Ease refers to the system's ability to allow the user to perform tasks comfortably with low mental and physical effort. Some examples of tasks being executed with ease include the ease to input data, the ease to use the output and the ease to learn [30]. The ease attribute directly impacts a system's effectiveness. Mobile applications should be designed to allow users to perform tasks with ease. This may also include providing users with various choices to accomplish the same task.

Attractiveness refers to the system's ability to keep the user engaged. The goal is to engage the end users to complete their tasks on the mobile application without any interruptions.

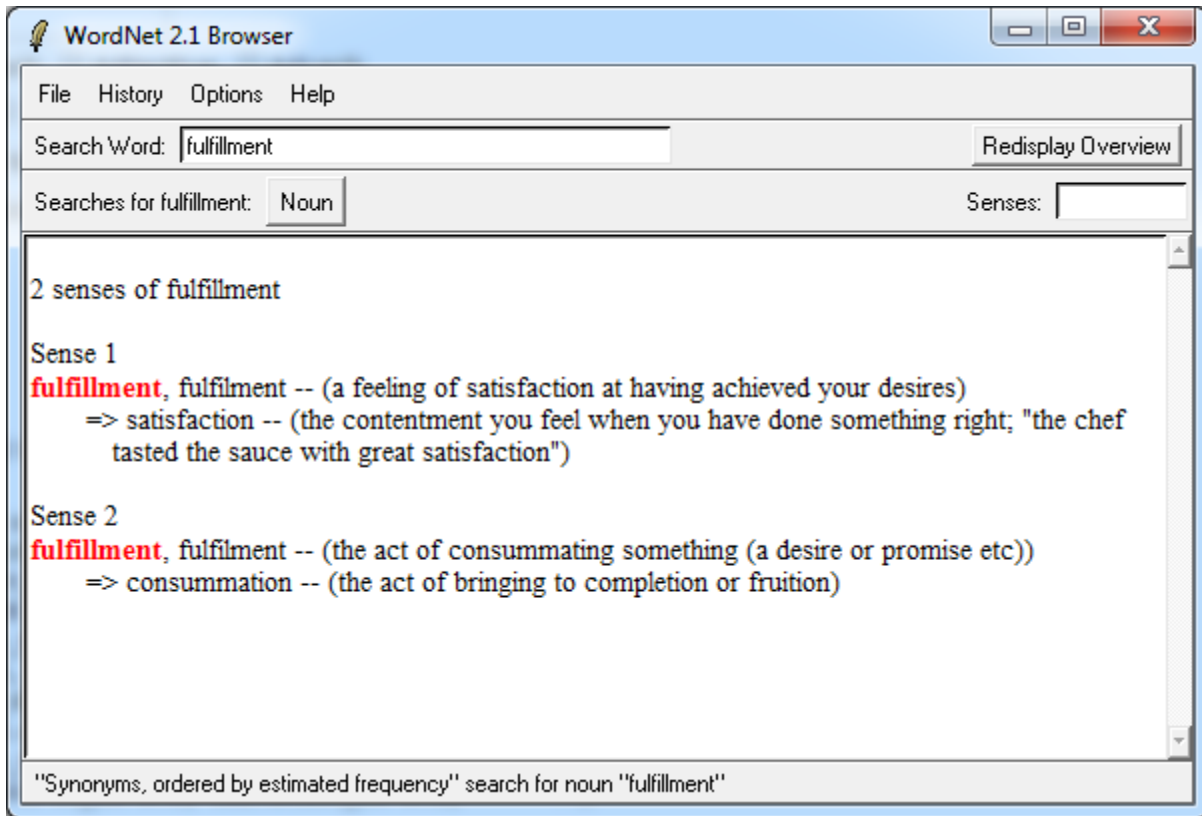


Figure 9. WordNet database showing the two senses for the word "fulfillment"

All the usability terms that are part of this sub-theme are also related semantically as shown:

*Effectiveness ↔ Validity*

*Effectiveness ↔ Accuracy ↔ (Inaccuracy) ↔ Errors*

*Effectiveness ↔ Fulfillment*

*Effectiveness ↔ Attractiveness*

*Effectiveness ↔ Ease*

Effectiveness is directly related semantically with validity, accuracy, fulfillment, attractiveness and ease. However, effectiveness is indirectly related to errors through the related term of accuracy-inaccuracy.

b) efficiency

Efficiency refers to the system's capability to complete tasks in a timely manner with minimal effort. The usability terms that are related to the efficiency sub-theme include efficiency, timeliness, responsiveness and agility.

Efficiency is defined as the speed with which users can accomplish tasks while using the application [20]. It is an important usability attribute that impacts a system's usability. The focus of this attribute is on the task performance, how quickly users can accomplish their tasks. A mobile application should be designed to allow a user to perform functions in an easy, quick and economical way [72].

Timeliness refers to the system's ability to allow the user to complete a task in the designated time as per the pre-defined requirements. Timeliness not only focuses on the duration of the entire task, but also on the parts of the task [36]. In many usability evaluation studies, time is the most frequently used variable because it is quantifiable [73].

Responsiveness refers to the speed with which a system responds to user actions. It is different from timeliness which measures the time taken to complete a task by a user, whereas responsiveness is the time taken for the system to respond to the user's action/task.

Agility refers to a system's ability to change its state in reaction to user actions. Apart from responsiveness, agility is an important attribute that impacts efficiency.

The usability terms that are part of the efficiency sub-theme are also related semantically.

*Efficiency ↔ Responsiveness ↔ Timeliness*

*Efficiency ↔ (speed) ↔ Agility*

Efficiency is directly related to responsiveness. Responsiveness, in turn, is related to timeliness through responsiveness. Agility is indirectly related semantically to efficiency through a related term, speed.

#### **4.1.3. Reliability Dimension**

The next cluster discussed is reliability, an important usability factor because the system's confidence and faithfulness are dependent on how reliable the system is. The common theme for the usability terms that are aggregated in this cluster is how a user can develop confidence in a mobile application's capabilities. The usability terms that are part of this cluster are grouped into two sub-themes: maturity and consistency.

##### a) Maturity

Maturity deals with a system's ability to continue to operate in spite of the erroneous conditions. Maturity helps improve the system's reliability. A user can be confident that the system is stable and continues to work in spite of error conditions or faults in its environment. The central theme of this group is a system's ability to recover from faults and errors. The attribute terms that are part of this group include maturity, fault-tolerance, robustness, feedback, correctness, faithfulness, task-conformance and reliability.

Maturity is "the capability of the software product to avoid failures as a result of defects in the software" [49]. There are faults that could occur from the user's input or the external environment surrounding the system.

According to the ISO/IEC/IEEE 24764 standard [48], robustness is "the degree to which a system or component can function correctly in the presence of invalid inputs or stressful environmental conditions." It is the system's ability to recover from a faulty condition to continue operating effectively and efficiently. Both fault tolerance and robustness are



semantically related terms. The focus for fault tolerance is the faulty condition of a system's component, whereas robustness is more related to faults introduced from the environment, such as incorrect input. Both fault tolerance and robustness are important factors for a critical system.

Fault tolerance is the system's capability to continue operating properly in the event of a failure (faults) for some of its components [74]. The idea for fault tolerance is that a system should not break down in the event of faults. Any performance reduction for the system should be proportional to the error's severity. Fault tolerance helps to improve the system's reliability. A fault-tolerant design most likely introduces redundancy in order to continue to function in the event of faults, possibly at a reduced level.

Feedback is the system's ability to provide regular progress indicators and messages to keep the user informed about the results of interactions, error conditions, and the system's state. It is an important usability attribute that is required of computer-based systems. Regular feedback helps to minimize the potential for user errors, thus improving system reliability. This ability of the system allows users to feel confident about the actions they utilize to perform a task by taking corrective measures if there is a faulty input.

Correctness refers to "the degree to which software, documentation, or other items meet user needs and expectations, whether specified or not" [48]. The user's confidence in the product depends on the correctness of the results produced by the user's actions.

Task-conformance refers to the degree to which a system supports the task a user would like to perform and the manner in which the user wishes to perform that task [3]. This feature is more applicable to mobile phones than desktop computers because each task is small, self-contained, generic and focused.

Reliability refers “to the ability of a system or component to perform its required functions under stated conditions for a specified period of time” [48]. According to the ISO/IEC 9126-1 standard [49], reliability is also defined “as the capability of a software product to maintain a specified level of performance when used under specified conditions.” Reliability is a very important usability attribute that impacts the confidence that end users have in the system’s performance and its outputs. All the usability attributes that were covered earlier in this sub-theme enhance the system’s reliability.

Faithfulness refers to the quality of being loyal. The usability attributes that were discussed earlier under the sub-theme of robustness help to promote faithfulness among the users about the system’s reliability. Users will be confident in the system’s performance and, thus, stay faithful to the system.

All the usability terms that are part of the maturity group directly impact the application’s reliability. It is important that usability designers consider these factors when designing mobile applications, particularly in the case of critical systems. These usability terms are also related semantically as follows:

*Reliability ↔ Correctness*

*Reliability ↔ Robustness*

*Reliability ↔ Task-conformance*

*Reliability ↔ (Fidelity) ↔ Faithfulness*

*Reliability ↔ (Responsiveness) ↔ Fault-tolerance*

*Reliability ↔ (Responsiveness) ↔ Feedback*

*Reliability ↔ (Assurance) ↔ Maturity*

Reliability is directly related semantically to correctness and robustness. Reliability is indirectly related to faithfulness through the term “fidelity.” Reliability is indirectly related to fault-tolerance and feedback through the term “responsiveness.” Reliability is indirectly related to maturity through the term “assurance.”

b) Consistency

This group of usability attributes subscribes to consistency in the system’s performance. Attributes that are part of this sub-theme are uniformity. The attributes that are part of this sub-theme include consistency, standards, and benchmarks.

Consistency is “the degree of uniformity, standardization, and freedom from contradiction among the documents or parts of system or component” [49]. Consistency refers to interface design choices (codes, naming, format, procedures, etc.) to be maintained in similar contexts and to apply different choices in varying contexts [75]. There are two types of consistency: internal consistency and external consistency. Internal consistency refers to the uniformity of the entire system in following the same rules and logic, whereas external consistency deals with how much of an application’s behavior and structure match the user’s experience with similar applications [76].

Standard is “a set of mandatory requirements established by consensus and maintained by a recognized body to prescribe a disciplined uniform approach or specify a product, that is, mandatory conventions and practices” [48]. Standards are based on the best practices in the domain. Standardization of the system artifacts and components promotes uniformity.

Ergonomics is “the scientific discipline concerned with the understanding of interactions among humans and other elements of a system, and the profession that applies theory, principles, data and methods to design in order to optimize human well-being and overall system performance”

[77]. Ergonomically designed applications improve the end users' physical comfort level. This usability attribute is important and needs to be actively considered by the usability designers who create mobile applications that are targeted toward the elderly and children. Consistency and improving the reliability of the product.

According to the ISO/IEC/IEEE 24764 standard, a benchmark is “a standard against which measurements or comparisons can be made. It is also defined as a procedure, problem or test that can be used to compare systems or components to each other or to a standard” [48]. Testing the system and its functionalities against the benchmarks available for a specific domain helps to identify any errors and issues beforehand, so they can be accounted for, thus promoting reliability.

The usability attributes that are part of the consistency sub-theme promote the product's reliability by adopting the best practices in the domain and by elevating the product against the benchmark. The usability terms in this group are semantically related as follows:

*Standards ↔ Consistency*

*Standards ↔ Benchmark*

In this cluster, all the usability terms that congregated were either directly or indirectly related to reliability. A usability designer needs to consider these usability attributes in order to ensure high faithfulness for the product user.

#### **4.1.4. Ergonomic/Context Dimension**

This cluster includes usability attributes that are related to user interactions and ergonomic attributes. The main theme of this cluster is related to factors that impact the user's interaction with the application. The usability attributes that clustered in this group are as follows: *accessibility, ergonomic, workload, cognitive-load, context, autonomy, navigability,*

*customizability, and observability*. These attributes are grouped together based on the linkage distance between them; two sub-themes were created: Ergonomic and Context.

a) Ergonomic

The usability attributes that clustered into this subtheme are related to the users' physical and mental effort that is needed to complete the tasks in an application. There are four attributes that are part of this sub-theme: ergonomic, workload, accessibility, and cognitive load.

Ergonomically designed applications improve the end users' physical comfort level. This usability attribute is important and needs to be actively considered by the usability designers who create mobile applications that are targeted toward the elderly and children.

Accessibility is an important usability attribute that mobile applications should strive to achieve. Accessibility system's ability to provide support different users, regardless of their capabilities [48]. Mobile applications should aim to support people who use assistive technologies. Making the mobile application accessible needs to be part of the usability design. Accessibility may require solutions such as avoiding color schemes that are difficult for color-blind people to understand, providing different modes of interaction using voice for people with a hearing loss and having the ability to interact with screen readers to allow the content to be read for the vision impaired.

Workload also impacts a system's usability. Applications that are designed to require many inputs from the user will lower the usability. Larger workloads for the users will require larger cognitive loads. A usability designer's goal is to improve the autonomy of the application, so it can perform well with user inputs. Scapin and Bastein [75] identified workload as one of the important ergonomic criteria for design and stated that the goal should be to reduce the workload as well as to focus on brevity and conciseness. Brevity corresponds to the goal of reducing the

reading, inputs and actions that users have to perform, and conciseness concerns reducing the size of the entries that the user needs to make. The smaller the number of entries, the lower the probability of errors.

Cognitive load refers to the amount of mental effort being used in the user's working memory to perform a task on the system [78]. The goal for usability would be to reduce the cognitive load as much as possible, thereby improving the end users' comfort level. Reducing the short-term memory load is one of the golden design principles proposed by Shneiderman and Plaisant [28]. Cognitive load differs from user to user. Children and the elderly typically require a higher cognitive load for certain tasks; the same tasks can be performed by a youth and adults with a lower cognitive load. Therefore, usability practitioners need to consider the cognitive load, particularly for mobile applications that target children and the elderly.

#### b) Context

One of the most important usability factors that is often ignored is the context in which the user is interacting with the mobile application. An application's effectiveness can be verified by the context. For example, a user who is stationary can accomplish more tasks with a mobile application than someone who is walking. The usability attributes that are part of this sub-theme include observability, navigability, context, autonomy and customizability.

Observability refers to "the extent to which the user can evaluate the internal state of the system from the representation on the user interface" [3]. It is important that the user is always aware of where he/she is in the system while navigating and that he/she comprehends the information being displayed. The usability designer's goal would be to have consistency across the user interface in order to promote observability.

Context is an important concept that needs to be considered as an attribute of usability. Mobile devices, by definition, are mobile, meaning that the context changes continuously unlike fixed computers. Mobile applications are used by different user populations to meet certain goals by performing certain tasks. A mobile application is used within a certain range of technical, physical and social environments which may impact its use. It is important to consider the context in which the mobile application will be used along with its ergonomic and usable attributes [79]. The mobile context may include the location where the device is being used or the condition of the mobile user. The mobile user's condition may include the following examples: interacting with other people and the user performing other tasks at the same time. The mobile context is generally characterized by many interruptions, such as noise, constant movement and distractions on the user's part, along with the anywhere, anytime ability of mobile devices. Usability practitioners who design mobile applications should always consider the user's context.

Navigability refers to the user's ability to easily traverse an application to complete the tasks. Navigability is an important factor for usability. Navigability is impacted by the context. If a user is moving, the most likely mode of interaction with a mobile application is by utilizing audio commands. Navigability is also impacted by the user's experience. When designing for navigability, usability designers should consider the following concerns:

- a) Are the interfaces consistent?
- c) Are the users familiar with the navigation that is being used for the application?

If not, can they easily synthesize how to navigate from prior experience with other applications?

- d) Does the application support different modes of interaction?

- e) Have the target population and its cognitive capacities been considered?
- f) Has the mobile context been considered?

The system's navigation should be consistent and familiar so that users can learn to interact with mobile applications with little cognitive load.

Autonomy refers to the ability of the system to independently carry out most of the functions with little user intervention. Autonomy can lead to better performance, higher execution speeds, good predication, less user effort and fewer human errors [80]. It is important for usability developers to aim to improve the autonomy of their mobile applications, so users can experience a high comfort level when utilizing the application.

Customizability is a system's ability to allow users to tailor the system to their personal preferences. Unlike portability where the system adjusts to the external environment, customizability provides the ability for a user to make necessary changes to meet his/her tastes. It is important for users to be able to customize mobile applications. An example of customization may include a user changing the interface's color schemes to meet his/her preferences.

The usability attributes discussed in this cluster relate to the cognitive and physical load that users have to exert in a mobile context while interacting with the application. These usability terms are semantically related in the following manner:

*Accessibility ↔ Navigability*

*Accessibility ↔ Mobility*

*Accessibility ↔ (Convenient) ↔ Workload*

*Accessibility ↔ (Convenient) ↔ Ergonomic*

*Accessibility ↔ (Convenient ↔ Flexibility) ↔ Customizability*

*Mobility ↔ Autonomy*



*Mobility ↔ (Visibility) ↔ Observability*

*Observability ↔ (Tangible) ↔ Cognitive*

#### **4.1.5. Satisfaction Attributes**

This cluster consists of usability terms that reflect the users' subjective satisfaction when utilizing an application. There are nine attribute terms that were grouped in this cluster. The usability attributes were grouped into sub-themes: contentment and utility.

##### a) Contentment

This sub-theme consists of usability attributes that reflect the user's contentment regarding the product's usability. The attributes that are grouped under this sub-theme include aesthetics, enjoyment, happiness, anxiety, playfulness, contentment, attitude, friendliness and satisfaction.

Aesthetics "is a branch of philosophy associated with art and beauty and is concerned with how individuals perceive objects and make judgements based upon information received as five human sensory inputs" [47]. Aesthetic designs are perceived to be easier to use than less aesthetic designs. Kurosu and Kashimura [81] found that the apparent usability (aesthetic aspect) is less correlated with the inherent usability (functional aspect) compared to the apparent beauty. This outcome suggests that the user is strongly affected by the usability's aesthetic aspect more than the functional aspects. This observation is useful for interface designers to focus not only on inherent usability, but also on apparent usability.

Enjoyment, happiness, anxiety, and playfulness are related to the user's emotions and feelings that reflect the satisfaction with the tasks performed and an application's outcome. Davis et al. [82] defined perceived enjoyment as "the extent to which the activity of using the computer is perceived to be enjoyable in its own right, apart from any performance consequences that may

me anticipated.” All the usability terms except for anxiety reflect position emotions and feelings toward the application. Anxiety reflects the users’ anxiety when using the interface [36].

Attitude refers to the users’ perception or feelings towards the system and friendliness refers to the users’ attitude towards how easy is the system to use and interact.

Contentment and Satisfaction mean the state of being satisfied and content. These attributes reflects the users’ attitude toward the application after using it [20]. All the terms discussed in this sub-theme are semantically related as follows:

*Attitude ↔ Satisfaction ↔ Contentment ↔ Friendliness ↔ Happiness ↔ Enjoyment ↔ Anxiety*  
*Enjoyment ↔ (Pleasure) ↔ Playfulness*

Satisfaction, contentment, happiness, enjoyment, and anxiety are all directly, semantically related to each other. On the other hand, enjoyment is indirectly related to playfulness through the term “pleasure.”

#### b) Utility

This sub-theme is focused on the application’s usefulness from the user’s point of view. It is important that a mobile application not only be aesthetic, but it should also be functional. The two attributes that are grouped under this sub-theme are utility and relevance.

Utility is the value that a system provides to a user, the system’s usefulness and applicability. A mobile application may be designed for usability, but it should also provide utility for the end user in terms of the functionality that user seeks to fulfill his/her goals. Utility and relevance are semantically related through usefulness.

*Utility ↔ (Usefulness) ↔ Relevance*

#### **4.1.6. Easy-to-Learn/Retain Dimension**

This cluster includes usability terms that are related to learnability and memorability. The usability attributes in this cluster are important because they impact an application's effectiveness and efficiency as well as the satisfaction that a user derives when utilizing an application. In a mobile context, the applications are smaller and simpler. This complexity reduction can lead to better learning and retention functions [26]. The usability attributes are grouped into two sub-themes: learnability and memorability.

##### **a) Learnability**

This sub-theme is related to the ease-of-learning. Most current mobile applications provide simple functionalities. The interface should evoke consistency so that a user can easily learn how to navigate the application and perform the tasks with which he/she is interested. The usability attributes related to this theme are learnability, predictability and comprehensibility.

Learnability refers to ease with which a user can learn and remember functions and modalities that the system provides [3]. Learnability is a measure of how easily users can learn and perform a task quickly the first time. A user should be able to predict the outcome of future interactions based on learning from past interactions. This ability of the system helps novice users to easily start interacting with the application in order to perform some tasks. It is important that usability designers create an interface for high learnability in order to promote the intention to use the system [2]. Learnability is generally measured by utilizing the time a specific user takes to accomplish a task under a certain condition.

Predictability measures the ease with which a user can predict the outcome of future interactions based on the knowledge acquired during past interactions. The simple and small size of mobile applications can lead to improved predictability for users. Predictability can also be

increased by maintaining consistency across the application and by designing and developing applications based on best practices. A user who is used to a similar application should be able to easily migrate to a new application with little learning.

Comprehensibility refers to the user's ability to understand how to perform different application tasks with ease and to comprehend the feedback that the application provides in response to user interactions. Easy understanding about how to navigate the system will help with learnability and retention. The usability attributes for this sub-theme are related to the learnability attribute. These attribute terms are semantically related to each other as follows:

*Learnability ↔ Memorability ↔ Comprehensibility*

b) Memorability

The usability attributes that are grouped under this sub-theme are related to the ease of retention. This usability attributes becomes more relevant when applications which are not used regularly. A user should be able to easily recall how to navigate and how to perform different tasks with the application. The usability terms that are part of this group include memorability, simplicity, flexibility, generalizability and familiarity.

Memorability is the user's ability to easily perform an application task by recalling past interactions. This attribute refers to the ease of retention. Memorability helps a casual user with the ease of remembering so that he/she does not have to learn everything every time. It is possible that, with applications that are easy to learn, users may be willing to relearn them [4]. In such cases, memorability may not be as important as a goal. With applications that have significantly high learning curves, both learnability and memorability are important goals to achieve.

Simplicity refers to degree of comfort with which users find ways to accomplish their tasks. In simple words, simplicity means low complexity. This attribute is important and useful to assess the mobile applications' navigation design quality. Simplicity can promote increased memorability. Simplicity can be measured easily by utilizing the users' error rate for completing a specific task [4].

Familiarity is the ability of a user interface to allow a user to determine how to initiate an interaction based on their perception of the interface [26]. A factor that contributes to familiarity with the interface is to design screen flows that match the user's expectations and tasks.

Generalizability is a system's ability to allow users to migrate from other platforms and to interact with a system with no prior experience. It is an important usability attribute, particularly in the case of mobile phones because users tend to change mobile phones regularly. The mobile application should be so designed that it allows people to utilize the knowledge they acquired from interactions with other platforms and to transfer the skills to the new platform [3].

Flexibility is "the ease with which a system or component can be modified for use in applications or environments other than those for which it was specifically designed" [49]. Flexibility allows an application to be installed on different platforms and operating systems, thus letting users continue to use the application on other platforms.

#### **4.1.7. Protection Dimension**

This cluster was the smallest among the seven clusters and consisted of five usability attributes. The usability attributes that are part of this cluster are related to system protection. Mobile applications are vulnerable to virus attacks and hacking because the security protocols for wireless devices are not robust enough. Security and safety become an important concern, particularly for mobile applications that store sensitive and personal data. In spite of the

importance of security and safety concerns, the articles about mobile usability did not provide sufficient coverage. The usability attributes that are part of this cluster are privacy, safety, security, integrity and confidentiality. These attributes were further grouped into three sub-themes: privacy, security and safety.

a) Privacy

This sub-theme includes the usability attributes of privacy and confidentiality. This sub-theme deals with the use and dissemination of personal or organizational data and their protection against unauthorized access. There are two usability attributes that are part of this sub-theme: privacy and confidentiality.

Privacy refers “to the ability of a user to control the terms under which their personal information is acquired and used” [83]. Applications should help keep the user’s sensitive and personal data private and should limit access to the data based on the permissions that the user provides. Privacy is an important concern for the HCI field, particularly in the context of the widespread use of social networks. Confidentiality is related to privacy and refers to the system’s ability to protect personal and organizational data from unauthorized access.

b) Security

The usability attributes that are part of this sub-theme include security and integrity. This sub-theme is concerned with protecting the system from malicious intents and maintaining the application’s integrity so that it continues to perform as expected.

Security refers to “the protection of system items from accidental or malicious access, use, modification, destruction, or disclosure” [48]. The security protocols for wireless devices, including mobile devices, are not mature compared to the protocols used by desktop systems. Thus mobile applications more vulnerable to malicious attacks with consequences that may

include system failure, the system being hacked and no longer in the user's control, private data being compromised and other catastrophic consequences that lead to the loss of life and wealth. It is important that usability designers keep the application's security in mind while designing the usability requirements.

Integrity is “the degree to which a system or component prevents unauthorized access to, or modification of, computer programs or data” [48]. System integrity requires that the user's identity be authenticated each time the user launches the application. If the application is idle, it is a good idea for the application to log out automatically in order to prevent unauthorized usage.

### c) Safety

According to ISO/IEC/IEEE 24765 [48], Safety is “the expectation that a system does not, under defined conditions, lead to a state in which human life, health, property, or the environment is endangered.” Safety is an important concern for critical applications that can have severe consequences if they fail. Critical applications need to be highly reliable, ensuring that they are fault tolerant, robust, and are compliant with domain-related standards and conventions. In spite of its importance, safety does not show up in many usability classifications and frameworks [64].

The discussion in this section thus far was about the usability attributes and the relationship between usability attributes for the same cluster. Further, each usability dimension was divided into multiple sub-themes. Each sub-theme contained a set of usability attributes that depend on how they are grouped within the cluster. This relationship among the usability dimensions (themes), sub-themes and the usability attributes belonging to each sub-theme was organized into a hierarchical structure that shows the usability taxonomy as illustrated in Figure 10.

The top of the usability taxonomy contains the root node which represents the mobile usability. The seven dimensions (themes) are represented in the next level. Each dimension (theme) leads to multiple sub-themes that are shown in the circles. Finally, each sub-theme points to a set of usability attributes that are part of the sub-theme.

The usability taxonomy, developed through this research, is exhaustive because it considers 76 usability attributes which are organized in a hierarchical structure. This taxonomy can be extended by adding new usability dimensions, sub-themes and usability attributes as technology and contexts evolve. The next section discusses the usability terms that cut across clusters based on their semantic similarity.

#### ***4.2. Cross-Cutting Usability Attributes***

The clustering results in a previous section showed how the 76 usability attributes were grouped into 7 different clusters. The last section explored the relationship between different usability attributes within each group. In this section, we discuss the semantic similarity of the usability terms that cut across multiple clusters. There is always overlap between the meanings for different words that may be conceptually similar. The concept we used to cluster the usability terms was based on their distributional similarity which tends to cluster terms that are semantically related but not always semantically similar. The grouping of semantically similar terms is shown in Table 4.

Accuracy is related to faithfulness, reliability, and uniformity cross-cutting the usability attributes that belong to different dimensions. Accuracy is a measure of correctness which implies that the user's tasks lead to accurate results, actions and responses from the application. Accuracy increases the application's reliability, leading to greater confidence and faith for the user.



Validity is the application's ability to consistently generate correct results. Validity has a relationship with reliability, relevance, integrity and correctness. Reliability is impacted by the results' validity. A highly valid application performance increases its reliability. An integral system prevents unauthorized access to data and programs. An integral system can produce a more valid performance. Results that are valid would also be correct.

Effectiveness is related to eight usability terms that are semantically similar across the clusters. It is related to accessibility which was located under the System Attributes cluster. Exploring the relationship, it is obvious that these terms are related because, by improving an application's accessibility by designing it for users who have different capabilities (e.g., disabilities), the application's effectiveness can be improved. Suitability (a member of System Attributes cluster) measures the extent to which an application can fulfill a user's needs. Improving the suitability of the application can promote higher effectiveness. Availability (a member of System Attributes cluster) relates to the degree to which a system is operational and accessible for users. Increasing the availability will lead to improved effectiveness. Good safety features help a system avoid risk and damage, thus improving effectiveness. All the cross-cutting usability attributes impact effectiveness, thus effectiveness is subject to the effects of these usability attributes from different dimensions.

Correctness refers to the degree to which an application meets user needs and expectations. If the degree of correctness is high, it helps increase the application's effectiveness. For an application to be highly effective, it also needs to be reliable, so there is a high degree of confidence among the users who interact with the application to perform their tasks.

Acceptability measures the degree to which a user finds the application's performance and functionalities acceptable. Highly acceptable applications will improve their effectiveness.

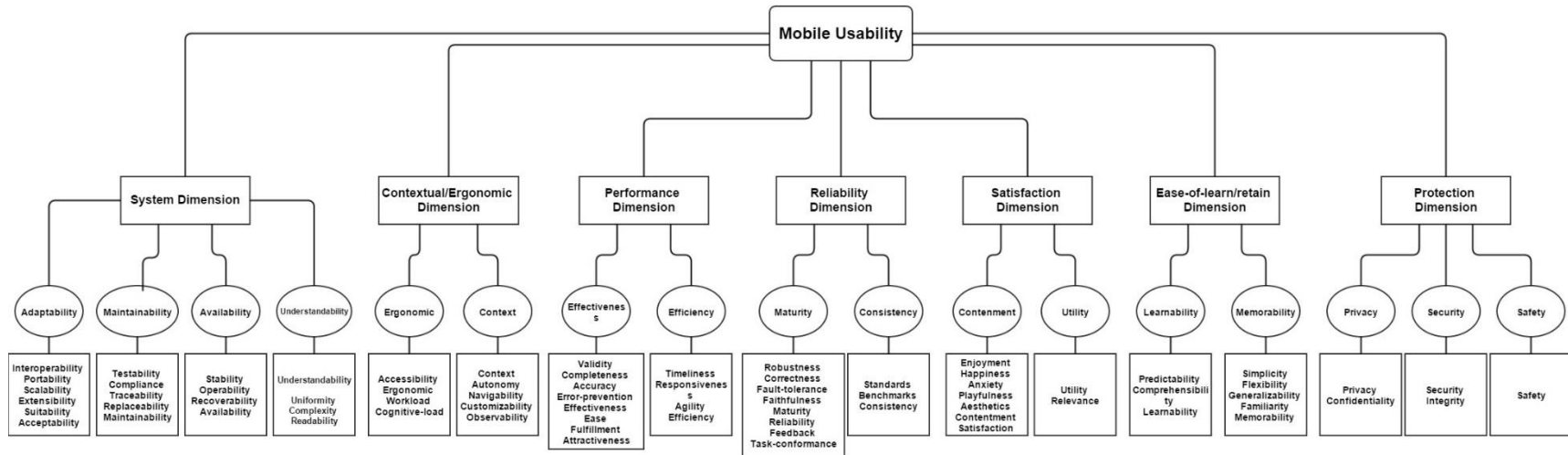


Figure 10. A hierarchical structure of the usability taxonomy depicting the seven usability dimensions (themes)

Efficiency is the application's ability to produce good results with a small amount of resources or time. Efficiency refers to the degree to which a product enables tasks to be performed in a quick, effective and economical manner [71]. The cross-usability attributes related to efficiency are accessibility, stability, simplicity, flexibility and reliability. Applications that are designed for users with different capabilities will be more efficient in terms of the time and resources that users have to commit. Highly accessible applications that support different user capabilities result in increased efficiency. Similarly, a stable and reliable system results in increased performance, thus efficiency. The usability attributes discussed impact efficiency, thus the application's efficiency is dependent on cross-cutting the usability concepts discussed earlier.

Fulfillment requires that user interactions are completed and successful, leading to overall satisfaction for the user. Comprehensibility enables a user to completely understand how to perform a certain task and the type of results that will be produced from his/her actions. Highly comprehensible applications can lead to greater completion and success rates.

An interesting relationship is shown between attractiveness and relevance. It can be argued that visually appealing applications may lead to greater relevance. There are studies that confirm how aesthetically designed applications hold the end users' attention better, increasing the application's relevance for the end user. The discovery of this relationship seems to substantiate the findings.

Responsiveness refers to the application's ability to react quickly and correctly in response to user or system actions. The cross-cutting related usability term is acceptability. This finding could indicate that responsive systems are preferable for and acceptable to the end users.

A flexible system can easily adapt to different user preferences and needs. On the other hand, an agile system allows the application to change its state quickly as a result of user actions. Flexibility and adaptability are semantically similar terms and are desirable usability attributes.

Consistency and uniformity are very semantically similar words that are part of the same synset in the WordNet lexicographic database. Consistency is defined as system uniformity and coherence [64]. The consistency and uniformity attributes help improve the application's predictability and comprehensibility. When the application's user-interface elements are consistent and are organized uniformly, a user is able to better predict the results of his/her actions based on past interactions. Consistency also helps improve the application's comprehensibility. Another interesting relationship is between consistency and safety. The review of literature did not yield any connection between these two usability attributes. The relationship may mean that a higher degree of consistency will aid in improving safety. It can be argued that applications that are highly consistent may decrease the possibility of errors and damage to users when the system is in use. This relationship between consistency and safety is interesting and needs to be explored further.

Accessibility and availability are related semantically. Improving the system's availability will help with accessibility for different user populations.

Reliability is related to acceptability, safety and security. The relationship between reliability and safety as well as between reliability and security is obvious. Promoting safety and security will improve the application's reliability. The relationship between reliability and acceptability is interesting. A reliable application generally is more acceptable to users. Increasing the application's reliability will lead to improved acceptance by the users.

Table 4. Semantically similar cross-cutting usability terms

<b>Usability Terms</b>	<b>Semantically Similar Cross-Cutting Usability Terms</b>
<b>Accuracy</b>	<i>Faithfulness, Reliability, Uniformity</i>
<b>Validity</b>	<i>Reliability, Relevance, Integrity, Suitability, Correctness, Acceptability</i>
<b>Effectiveness</b>	<i>Accessibility, Suitability, Availability, Correctness, Reliability, Relevance, Acceptability, Safety</i>
<b>Efficiency</b>	<i>Accessibility, Stability, Simplicity, Flexibility, Reliability</i>
<b>Fulfillment</b>	<i>Satisfaction</i>
<b>Completeness</b>	<i>Comprehensibility</i>
<b>Attractiveness</b>	<i>Relevance</i>
<b>Responsiveness</b>	<i>Acceptability</i>
<b>Agility</b>	<i>Flexibility</i>
<b>Consistency</b>	<i>Uniformity, Predictability, Safety, Comprehensibility</i>
<b>Portable</b>	<i>Mobile</i>
<b>Accessibility</b>	<i>Availability</i>
<b>Reliability</b>	<i>Acceptability, Safety, Security</i>
<b>Simplicity</b>	<i>Complexity</i>
<b>Compliance</b>	<i>Safety</i>
<b>Integrity</b>	<i>Consistency, Reliability, Stability, Faithfulness</i>
<b>Predictability</b>	<i>Stability, understandability, Satisfaction</i>
<b>Privacy</b>	<i>Compliance, Reliability</i>
<b>Safety</b>	<i>Stability, Maintainability, Availability, Acceptability</i>
<b>Security</b>	<i>Anxiety, Standard, Compliance</i>
<b>Flexibility</b>	<i>Adaptable</i>

Simplicity and complexity are semantically related and appear to be antonyms of each other. There seems to be an adversarial relationship between them. The typical recommended practice is to keep things simple by reducing the complexity. However, there are specialized tasks and functionalities that applications support which are highly complex. An emerging theory called the simplicity theory proposes that there is a complementary relationship between simplicity and complexity [84]. From the design point of view, complexity tends to rise as more functionalities and features which solve specific problems are introduced to the system. On the other hand, simplicity with the user-interface design tends to improve the usability of such complex systems. Thus, a usability practitioner, instead of focusing on reducing the system's

complexity which is a reality for some specialized functions, needs to focus on simplifying the user interface to help users perform complex tasks with ease.

The integrity usability term seems to have a semantic relationship between reliability and stability. Stability promotes the system's integrity. Integrity promotes reliability. The other relationship between consistency and faithfulness is interesting. It can be argued that the applications' stability promotes faithfulness for the users. The relationship between integrity and consistency is not obvious. Integrity refers to a system's ability to continue to perform well under changing conditions, and consistency refers to a system's uniformity and coherence [64]. It may be argued that consistent systems that are uniform in terms of their various artifacts tend to better withstand and remain stable during the changing conditions. It would be interesting to explore this relationship further.

Predictability is related to the stability, understandability and satisfaction usability terms from other dimensions. Stable systems promote predictability about the outcomes for the users' actions, leading to overall satisfaction. Better understandability about the system's navigation and functions will help users make better predictions for their actions.

Other identified relationships between compliance and safety, flexibility and adaptability, security and standard, security and compliance, privacy and compliance, and privacy and reliability are obvious and have existed in the literature for a long time. The relationships between cross-cutting usability terms illustrated some interesting relationships that are either not given serious attention or considered. Just to summarize them, here are these interesting relationships between the cross-cutting usability terms.

1. Attractiveness and Relevance
2. Integrity and Faithfulness

3. Consistency and Safety
4. Integrity and Consistency
5. Reliability and Acceptance
6. Simplicity and Complexity

It would be worthwhile to explore these relationships and their impacts on the usability of mobile applications. Other relationships between the cross-cutting usability attributes show how the usability attributes from other dimensions impact each other. The cross-cutting relationships explain that usability attributes cannot simply be put in individual silos called dimensions.

### ***4.3. Cross-Cutting Usability Themes***

In an earlier section, we discussed the cross-cutting relationships between different usability attributes. Because we have already grouped these usability terms into themes, it might be interesting to study the relationship between the usability themes across the seven dimensions in order to uncover any interesting relationships that might exist at the theme level.

Figure 11 illustrates the relationship between different usability themes that cut across multiple dimensions. Each theme is identified as a circle with unidirectional arrows flowing-in and flowing-out, depicting the relationships between the themes. In order to build this relationship, it was assumed that, as long as the usability term  $u_x$  belonging to *theme*  $t_x$  has a semantic similarity (as explored in Section 4.2) with another usability term  $u_y$  belonging to *theme*  $t_y$ , we consider that a relationship exists between themes  $t_x$  and  $t_y$ .

Analyzing the relationship shows that, of all the themes, the effectiveness and efficiency have the most incoming arrows, meaning that they are most impacted by usability attributes from other themes. On the other hand, there are no outgoing arrows from effectiveness and efficiency, indicating that they do not directly impact any of the other usability themes.

A total of 10 usability themes directly affect effectiveness: learnability, safety, contentment, adaptability, availability, ergonomic/contextual, understandability, maturity, utility and security. The effect that the learnability attributes have on effectiveness is obvious. The learnability theme consists of attributes that impact the ease-to-learn. Improving the application's learning attributes leads to improved effectiveness. Other cross-cutting usability themes, such as contentment, ergonomic/contextual, maturity (reliability attributes), utility and security, have a beneficial effect on effectiveness. Effectiveness is positively impacted by increasing contentment of users, good system maturity (reliability), and tight security of the system. Similarly, better adaptable and available systems also have a positive impact on effectiveness. The interesting question is the relationship between the effectiveness and safety themes: how does improving the safety of a system increase its effectiveness? It can be argued that the more safe a system is, the

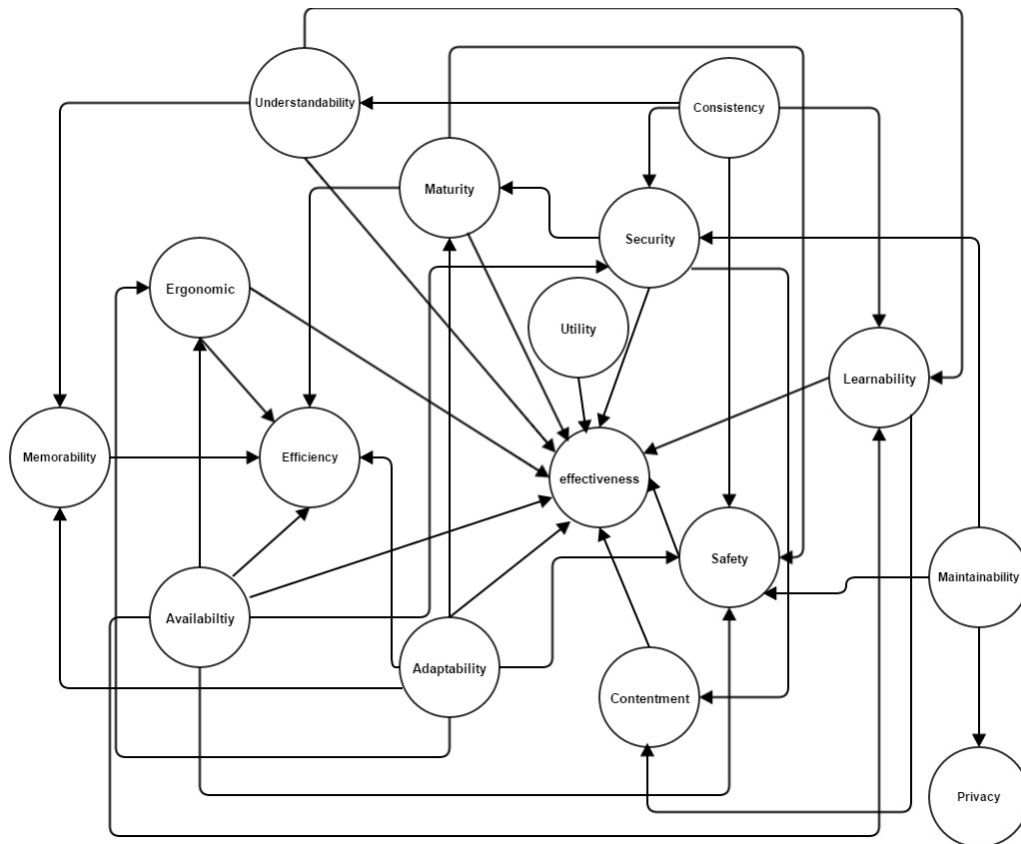


Figure 11. Relationship between the Cross-cutting Usability themes



more reliable it is and that the more reliable a system is, the more effective it will be. It would be interesting to explore this relationship further.

Efficiency is the other usability theme in the performance dimension that is impacted by five different cross-cutting usability themes. They are the adaptability, availability, memorability, ergonomic/contextual and maturity themes. The relationship between efficiency and these usability themes is obvious and well documented in literature. A highly adaptable, available and mature (reliable) system tends to be more efficient. Similarly, a greater focus on the memorability and ergonomic/contextual themes results in greater efficiency.

Safety is another usability theme that is impacted by five cross-cutting usability themes: maintainability, security, adaptability, availability and maturity. Here again, the relationships are apparent. Highly maintainable, secure, adaptable, available and mature (reliable) systems will contribute to increased safety for the application.

All the other relationships between the cross-cutting usability themes are obvious, well-reasoned and documented in the literature. The relationship between the cross-cutting usability themes seems to reconfirm earlier findings, but it still created an interesting question about the relationship between effectiveness and safety.

#### **4.4. *Proposed Unified Usability Construct***

The literature review on usability, as discussed in Chapter 2, is characterized by definitions that are brief, overlapping and do not capture the entire essence of usability. Most usability researchers have focused on a few important usability attributes, such as effectiveness, efficiency and satisfaction, and others have placed an emphasis on learnability and memorability.

The impact of other dimensions, such as ergonomics, privacy and security, have been considered independent of the usability. Most of the literature on ergonomics is discussed from the user experience (UX) point of view. Similarly, there are not many studies that relate the security and privacy dimensions to usability. In the last few years, there has been effort in that direction, notably by Dr. Lorrie Cranor of Carnegie Mellon University. The other constantly changing paradigm is the technological changes that create additional opportunities and challenges for usability practitioners. It is important that the concept of usability reflect these changes and continue to evolve.

One of the objectives of this research is to develop a unified usability construct for mobile applications that captures all the relevant usability attributes. This unified approach to usability is expected to comprehensively capture all the factors that impact usability. This framework is expected to aid usability practitioners in identifying relevant usability attributes to focus on during usability analysis, design and development.

The unified usability construct includes 7 dimensions that were identified through the unsupervised clustering of the 76 usability attributes that were extracted using text-mining techniques based on a computational linguistics approach. In this section, the results of clustering, as well as the cross-cutting usability attributes, is used to develop a usability construct for mobile applications.

Studying the semantic relationship between usability terms within the cluster allowed us to understand how the usability attributes grouped around central themes. On the other hand, the discussion on cross-cutting usability attributes helps us understand the usability terms that are semantically similar but are in different clusters (dimensions). Effectiveness, efficiency and reliability are a few important usability terms that are broad enough to show semantic similarity

with usability terms in other dimensions. Most usability studies regard effectiveness, efficiency and satisfaction as the most important usability dimensions [10]. These observations are also reflected in the results for this study, except for satisfaction. In our results, satisfaction clustered with many related terms in its own cluster and showed little semantic relationship with the cross-cutting usability terms.

Based on the results and discussions in two previous sections, a usability construct for mobile applications should consider not only the semantic relatedness between the attributes in each cluster (dimension), but also the semantic similarity between the cross-cutting usability attributes across different dimensions. The usability construct takes both these factors and reflects the relationship between dimensions. In this usability construct, each cluster that results

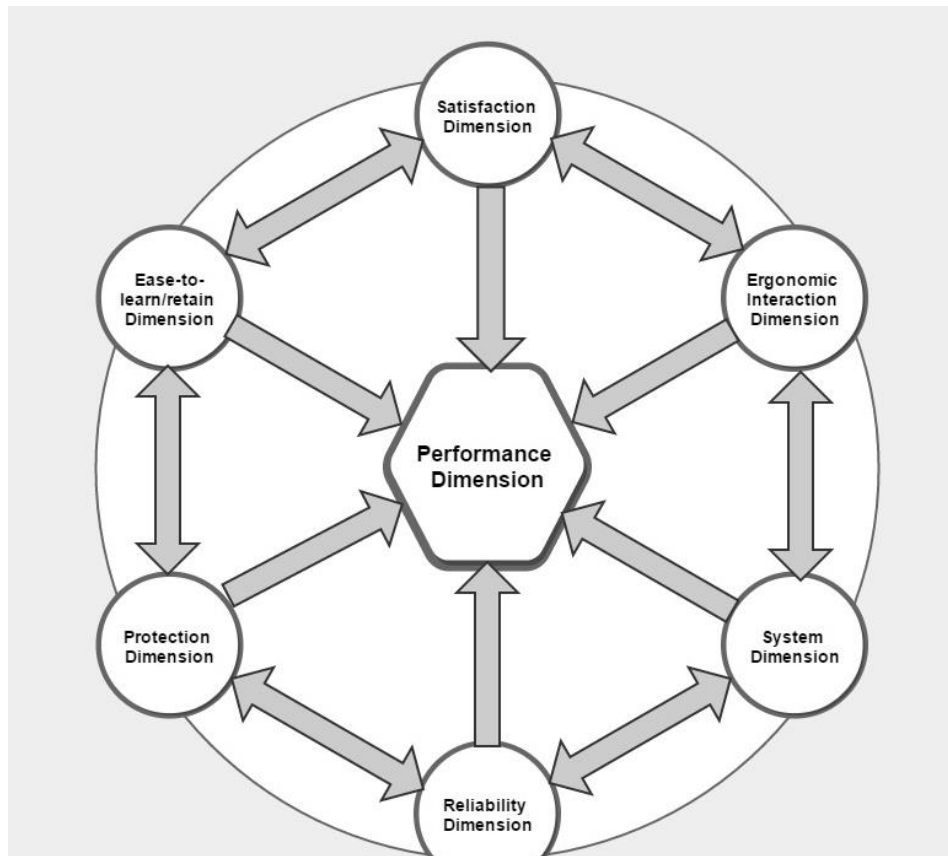


Figure 12. Unified Usability Construct showing the relationship between the seven usability dimensions

from hierarchical clustering was abstracted into an entity by itself. Therefore, each dimension is treated as its own entity with the usability terms inside each cluster tightly bound due to the semantic relatedness. As seen in Figure 12, this construct uses a Spoke and Wheel model approach. The hub, which is the center of the model, is represented as a hexagon.

The hub represents the performance dimension and is considered to be the most important dimension in this construct, hence it is at the center. The six sides of the performance dimension represent the six dimensions' impact on it. The six dimensions are represented by a circle at the end of each spoke. The outer circle that includes the entire structure represents the unified usability construct. It conveys the meaning that all seven dimensions are important and need to be considered by usability practitioners and researchers

The attributes that are part of the performance dimension have the largest number of semantically similar cross-cutting usability attributes across different dimensions. As discussed in Chapter 2, the attributes that are part of this dimension are regarded as the most important by number of usability researchers. As discussed in the last section, the effectiveness and efficiency attributes belonging to the performance dimension are impacted by multiple usability attributes from the other six dimensions. The six unidirectional arrows (spokes) from the six circles (dimensions) to the hub represent the impact that usability attributes for these dimensions have on the performance dimension. The bi-directional arrows that connect the six dimensions represent the impacts that these dimensions have on each other.

The unified usability construct suggests that all seven dimensions are important and should be part of any usability analysis, design and evaluation. The usability practitioners need to consider all the dimensions, but they have leeway with the usability attributes on which they

would like to focus depending on the mobile application's purpose, user population, environment and complexity.

Mobile applications vary in the type of services they provide. There are different categories for mobile applications: business, finance, communication, health, entertainment, games, travel, sports, etc. (For an entire category list, please see the following link for google developers: <https://developers.google.com/adwords/api/docs/appendix/mobileappcategories>.) Depending on the application type, a usability designer would focus on the pertinent usability attributes from different dimensions. As an example, for productive mobile applications, the focus needs to be more on the performance and reliability dimensions, whereas for health and finance related mobile applications, privacy and security will be a big concern.

The population that will use the mobile application also has a bearing on the usability attributes that need to be given more attention. Suppose the application is trying to reach larger segment of the population with different capabilities, then accessibility and ergonomics need to be actively considered by the usability practitioners. On the other hand, if the user population that is more likely to utilize the mobile application is younger, then aesthetics and other satisfaction attributes need to be the focus.

Other factors that may impact the choice of usability attributes to be considered include the application's environment (location, social and mobile contexts) and the complexity. The mobile context requires that the application be designed considering the context in which the application will be used along with condition of the user. A good example would be fitness apps that are typically utilized when the user performs physical activity such as running. For such apps, the reliability attributes are an important factor that need to be considered. The user should be confident about the health and fitness information that is collected by the app. The statistics

and other information displayed to the user should be reliable, so people can continue to trust the application. For more complex applications, there should be more focus on the learnability and memorability attributes.

#### ***4.5. Significance and Implications for the Research***

The research tries to address some perineal issues that have been identified by many different researchers and practitioners, but have not been addressed properly. The big issue is with the ambiguity in terminology that is utilized in the usability field to describe different usability factors and attributes. Moreover, there is no consensus in the field about what constitutes a usability construct/model/framework. Even though many researchers and practitioners have proposed usability definitions, models and frameworks, the researchers were limited by focusing on few core usability attributes. The proposed definitions are concise, limited and do not capture the entire essence of usability.

This research tried to address these challenges by, first, identifying an exhaustive list of usability attributes that are relevant to mobile usability, followed by developing a taxonomy for usability attributes and a unified usability construct that captures all the dimensions of usability.

To investigate and address the usability challenges, this research developed a novel computational approach, based on computation linguistics, to identify the sematic relatedness between usability attributes using text-mining and information-extraction (IE) techniques. This approach is based on the *distributional hypothesis* borrowed from computational linguistics. The common methods utilized for usability research are based on case studies, field studies, lab experiments, survey research, applied research, basic research and normative writing approaches [45]. This new technique was never tried in the usability research. This approach,

based on computational linguistics, should open another investigative methodology for usability studies.

The results of this study initially identified an exhaustive list of usability attributes that have been found to be relevant based on the literature analysis. This list should help usability practitioners to develop checklists for each of the identified attributes that are utilized for usability evaluation.

One of the important contributions of this research is a large taxonomy of usability attributes being laid out in a hierarchical fashion based on their relationship. Currently, there is no ontology for usability. A proposal for a usability ontology was made in 2007 by Lisa Battle [85], but, to our knowledge, there has been no progress. The exhaustive taxonomy of usability attributes is a first step leading to a usability ontology.

Another major contribution from this research is the unified usability construct developed for mobile applications. This usability model captures all the relevant usability attributes into different dimensions. The ergonomic/contextual and protection dimensions which are often neglected in most usability frameworks and models are given equal importance in this model. Unlike most other usability models, this usability construct does not put usability attributes in separate silos of usability dimensions with no cross impacts with other dimensions. The unified usability construct recognizes the cross-cutting impacts, interactions and relationships of usability attributes across dimensions.

This unified usability construct has real-world implications both for usability practitioners and usability researchers. It is expected to help the usability practitioners to better align their usability goals, depending on the tasks and user populations that will be supported as well as the application's complexity. The usability practitioner can pick and choose the usability

attributes to focus upon while still addressing all the dimensions of usability. The significance of this model for researchers is in helping to classify the usability problems accurately around the usability themes and sub-themes modeled in this usability construct. This usability construct also helps both the usability researchers and practitioners to develop usability metrics for the usability attributes captured in this usability construct.

Another advantage of the unified usability model is that it can evolve with technological changes and advances. Many usability frameworks that have been created over the years are no longer applicable because they do not reflect the technological and contextual changes. This model can evolve by easily adding new themes (dimensions), sub-themes and usability attributes.

The study discovered some interesting relationships between cross-cutting usability attributes that are either not given serious attention or considered. To summarize, here are these interesting relationships between the cross-cutting usability terms.

1. Attractiveness and Relevance
2. Integrity and Faithfulness
3. Consistency and Safety
4. Integrity and Consistency
5. Reliability and Acceptance
6. Simplicity and Complexity

It would be worthwhile to explore these relationships and their impacts on the usability of mobile applications.



## CHAPTER 5. SUMMARY AND CONCLUSIONS

This research addressed issues related to usability, in general, and mobile usability, specifically. One of the identified issues was that there is no clarity in what constitutes a usability construct. Over the years, usability researchers and practitioners described usability in limited and different ways. Apart from the usability framework/construct, there seemed to be ambiguity about the usability attributes themselves. Different researchers gave the same concept different names, and the same usability term was used to mean many different things. This ambiguity has led to a lot of confusion among practitioners, and there is no consensus to resolve this situation either by the researchers or standard organizations. This research tried to fill this gap, starting by identifying the most significant usability attributes that are relevant for mobile applications and then creating a *unified usability construct*.

This research tries to address these challenges by developing a *unified usability construct*. A novel computational approach based on computation linguistics was used to identify the semantic relatedness of usability terms by utilizing text-mining and information-extraction techniques. This approach is based on the idea that words with similar meaning and contexts will tend to occur with similar neighbors and contexts. The approach has its basis in the *distributional hypothesis* concept from the computational linguistics field. This research leverages the distributional properties of the usability terms to identify the relationships between them. To our knowledge, this novel approach has never been attempted in the usability field.

The investigation began by identifying a set of commonly referred to usability attributes in literature. The research identified a total of 76 usability attributes. Next, using a novel text-mining technique based on computational linguistics, a large collection of published papers on mobile usability were mined, and the results were organized into term-term vectors for further

processing. To our knowledge, this study was the first time such an approach was utilized for a usability study.

Using hierarchical clustering analysis, the study identified seven clusters. Each cluster that formed reflected a common theme. The usability attributes within the cluster were grouped into sub-themes based on how the usability attributes were organized within each cluster. Using the results of the clustering analysis, this research developed a usability taxonomy of usability attributes and how they are organized.

The usability attributes in each dimension (cluster) were studied further to understand the relationship between other usability attributes in the same cluster. The cross-cutting effect of the usability attributes/themes across different usability dimensions were identified and studied. This analysis discovered some obvious and a few interesting relationships which need to be explored by future research. The results of the cluster analysis and cross-cutting usability concerns led to the development of a *unified usability construct*. This framework was modeled as a Spoke and Wheel structure with the performance dimension as the hub and the other six dimensions connected to the hub through spokes. The impact of the different dimensions on the performance dimension was detailed. The intra-dimensional interactions were addressed by the model. This usability model also gave equal importance to the ergonomic/contextual and protection dimensions of usability which many other models and frameworks seem to ignore. Unlike most usability frameworks and models, the unified usability model did not limit the usability attributes into separate silos. The model recognized the cross-cutting impacts, relationships and impacts of the usability attributes from other dimensions.

The results of this research are expected to have real-world implications for usability researchers and practitioners. The unified usability construct identifies the different usability

dimensions that need to be considered from a usability point of view. Depending on the services that the mobile application provides, the target user population and the application's complexity, a usability analyst has the latitude to focus on the appropriate usability attributes while still considering all the dimensions. The framework is expected to help the usability designers better align their usability goals based on usability attributes that capture the context. This usability construct is expected to help researchers classify the usability problems accurately around the usability themes/sub-themes presented in the usability construct. This model also presents opportunities for researchers and practitioners to develop appropriate metrics and measures to evaluate the usability attributes. Many of the usability frameworks that were developed over the years are no longer applicable because they do not evolve with technological changes. The *unified usability construct* model is flexible enough to add new usability themes (dimensions), sub-themes and attributes as the technology evolves.

Because this usability construct was based on identifying and mining semantically related usability terms found in the literature, it was limited to only the semantic and syntactic relationships between words in the lexicographical databases. It would be interesting to utilize a different lexicographical database to see if additional semantically related terms can be extracted and incorporated in the model. For future research, a different clustering approach, such as fuzzy clustering, could study the usability attributes that tend to have wider impacts and implications across dimension boundaries.

## REFERENCES

- [1] K. C. Coursaris and D. Kim, *A research agenda for mobile usability.*, New York, NY: ACM, 2007.
- [2] O. Turel, "Contextual effects on the usability dimensions of mobile value-added services: a conceptual framework," *International Journal of Mobile Communication*, vol. 4, no. 3, pp. 309-332, 2006.
- [3] E. Bertini, T. Catarci, S. Kimani and A. Dix, "A review of standard usability principles in the context of mobile computing," *Studies in Communication Sciences*, pp. 111-126, 2005.
- [4] R. Harrison, D. Flood and D. Duce, "Usability of mobile applications: literature review and rationale for a new usability model," *Journal of Interaction Science*, vol. 1, no. 1, 2013.
- [5] R. Gafni, "Usability issues in mobile-wireless information systems," *Issues in Informing Science and Information Technology*, no. 6, pp. 755-769, 2009.
- [6] D. Ham, J. Heo, P. Fossick, W. Wong, S. Park, C. Song and M. Bradley, "Conceptual framework and models for identifying and organizing usability impact factors of mobile phones," in *Proceedings of the 18th Australia conference on Computer-Human Interaction: Design: Activities, Artefacts and Environments (OZCHI '06)*, New York, NY, USA, 2006.
- [7] A. Seffah, M. Donyae, R. B. Kline and P. H. K, "Usability measurement and metrics: a consolidated model," *Software Quality Journal*, vol. 14, no. 2, pp. 159-178, 2006.

- [8] G. M. Van der Welie, A. Eliens and G. Van der Veer, "Breaking down usability," in *Proceedings of Interact 99*, Edinburgh, Scotland, 1999.
- [9] V. Giller, R. Melcher and J. Schrammel, "Usability evaluations for multi-device applications development - three example studies," *Human-Computer Interaction with Mobile Devices and Services*, vol. 2795, pp. 302-316, 2003.
- [10] R. Baharuddin, D. Singh and R. Razali, "Usability dimensions for mobile applications - a review," *Research Journal of Applied Sciences, Engineering and Technology*, vol. 5, no. 6, pp. 2225-2231, 2013.
- [11] A. K. Dey, "Evaluation of ubiquitous computing systems: evaluating the predictability of systems," in *Workshop of Evaluation Methodologies for Ubiquitous Computing - UbiComp 2001*, Atlanta, GA, USA, 2001.
- [12] J. Mankoff, A. K. Dey, G. Hsieh, J. Kientz and S. Lederer, "Heuristic evaluation of ambient displays," in *Proceedings of CHI Conference 2003*, Ft Lauderdale, FL, USA, 2003.
- [13] H. Schutze and J. Pedersen, "Information retrieval based on word senses," in *Proceedings of the 4th Annual Symposium on Document Analysis and Information Retrieval*, Las Vegas, NV, USA, 1995.
- [14] M. Sahlgren, "The Word-Space Model using distributional analysis to represent syntagmatic and paradigmatic relations between words in high-dimensional vector spaces," in *Linguistics*, 2006, p. 156.
- [15] K. C. Coursaris and D. Kim, "A meta-analytical review of empirical mobile usability studies," *Journal of Usability Studies*, vol. 6, no. 3, pp. 117 - 171, 2011.

- [16] M. F. Porter, "An algorithm for suffix stripping," in *In Readings in information Retrieval*, San Francisco, USA, Morgan Kaufmann Publishers Inc., 1997, pp. 313-316.
- [17] R. B'Far, *Mobile Computing Principles: Designing and Developing Mobile Applications*, Cambridge, UK: Cambridge University Press, 2005.
- [18] T. Joklela, N. Iivari, M. Matero and M. Karukka, "The standard of user-centered design and the standard definition of usability: analyzing ISO 13407 against ISO 9241-11," in *Proceedings of the Latin American conference on Human-Computer Interaction*, Rio de Janeiro, Brazil, 2003.
- [19] J. Heo, D. H. Ham, S. Park, C. Song and W. C. Yoon, "A framework for evaluating the usability of mobile phones based on multi-level hierarchical model of usability factors," *Interacting with Computers*, vol. 21, no. 4, pp. 263-275, 2009.
- [20] D. Zhang and B. Adipat, "Challenges, methodologies, and issues in the usability testing of mobile applications," *International Journal of Human Computer Interaction*, vol. 18, no. 3, pp. 293-308, 2005.
- [21] R. Harrison, D. Flood and D. Duce, "Usability of mobile applications: literature review and rationale for a new usability model," *Journal of Interaction Science*, vol. 1, no. 1, 2013.
- [22] M. Jones, G. Marsden, N. Mohd-Nasir, K. Boone and G. Buchanan, "Improving web interaction on small displays," in *Proceedings of Eight International conference on World Wide Web*, Toronto, Canada, 1999.
- [23] R. Longoria, "Designing mobile applications: challenges, methodologies, and lessons learned," in *In Usability Evaluation and Interface Design: Cognitive Engineering*,

*Intelligent Agents and Virtual Reality*, New Jersey, Lawrence Erlbaum Associates Inc., 2001, pp. 91 - 95.

- [24] M. J. Culnam, "Protecting privacy online: is self-regulation working?," *Journal of Public Policy and Marketing*, vol. 19, no. 1, pp. 20-26, 2000.
- [25] H. Ferneley and A. E, "Usability metric for mobile application: A Goal Question Metric (GQM) Approach," in *The 6th International Conference on Advances in Mobile Computing and Multimedia*, Linz, Austria, 2008.
- [26] E. Bertini and S. Kimani, "Mobile devices: opportunities for users with special needs," *Human-Computer Interaction with Mobile Devices and Services - Lecture Notes in Computer Science*, vol. 2795, pp. 486-491, 2003.
- [27] A. Dix, J. A. Finlay, G. D. Abowd and R. Beale, *Human-Computer Interaction*, Harlow: Pearson Prentice Hall, 2004.
- [28] B. Shneiderman and C. Plaisant, *Designing the user interface: strategies for effective Human-Computer Interaction*, Boston, MA: Addison Wesley, 2005.
- [29] J. Nielsen, *Usability Engineering*, New York: Academic, 1993.
- [30] A. Hussain and M. Kutar, "Usability metric framework for mobile phone application," in *The 10th Annual Conference on the Convergence of Telecommunications, Networking & Broadcasting*, Liverpool, UK, 2009.
- [31] M. McGee, A. Rich and J. Dumas, "Understanding the usability construct: user perceived usability," in *Proceedings of Human Factors and Ergonomics Society 48th Annual Meeting*, New Orleans, LA, USA, 2004.

- [32] Y. G. Ji, J. H. Park, M. H. Yun and C. Leo, "A usability checklist for the usability evaluation of mobile phone user interface," *International Journal of Human-Computer Interaction*, vol. 20, no. 3, pp. 207-231, 2006.
- [33] D. Saffer, *Designing for interaction: creating smart applications and clever devices*, Berkeley, USA: New Riders, 2007.
- [34] T. Tullis and B. Albert, *Measuring the user experience*, Burlington, USA: Morgan Kaufmann, 2008.
- [35] K. Hornbaek and E. L. Law, "Meta-analysis of correlations among usability measures," in *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, San Jose, CA, USA, 2007.
- [36] K. Hornbaek, "Current practice in measuring usability: Challenges to usability studies and research," *Internal Journal of Human-Computer Studies*, vol. 64, pp. 79 - 102, 2006.
- [37] A. Hussain and E. Ferneley, "Usability metric for mobile application: A Goal Question Metric (GQM) approach," in *The Tenth International Conference on Information Integration and Web-based Applications Service*, Linz, Austria, 2008.
- [38] M. O. Leavitt and B. Shneiderman, *Research-based web design & usability guidelines*, Washington D.C., USA: U.S. Department of Health and Human Services, Washington, 2006.
- [39] E. Garcia, M. Sicilia, J. Hilera and J. Mesa, "Extracting knowledge from usability evaluation database," in *Human-Computer Interaction INTERACT '01*, IOS Press, 2001, pp. 713 - 714.



- [40] L. L. Constantine, "Collaborative usability inspections for software," in *Proceedings of the Software Development*, San Francisco, CA, USA, 1994.
- [41] S. Treau, *User interface evaluation*, New York, New York, USA: Plenum, 1994.
- [42] S. Lauessen and H. Younessi, "Six style for usability requirements," in *Proceedings of the 4th International Workshop on Requirements Engineering: Foundations of Software Quality*, 1998.
- [43] J. Preece, Y. Rogers and H. Sharp, *Interaction design*, New York, New York, USA: Wiley, 2002.
- [44] J. Kjeldskov and C. Graham, "A review of mobile HCI research methods," in *Proc. Mobile HCI 2003*, Udine, Italy, 2003.
- [45] J. Kjeldskov and J. Paay, "Hierarchical grouping of usability elements is a subjective process depending on the experience of the researcher," in *In Proceedings of the 14th international conference on Human-computer interaction with mobile devices and services (MobileHCI '12)*, New York, USA, 2012.
- [46] N. Bevan, J. Kirakowski and J. Maissel, "What is usability?," in *Proceedings of the 4th International Conference on HCI*, Stuttgart, Germany, 1991.
- [47] A. David and P. Glore, "The impact of design and aesthetics on usability, credibility, and learning in an online environment," [Online]. Available: [http://www.westga.edu/~distance/ojdla/winter134/david\\_glore134.html](http://www.westga.edu/~distance/ojdla/winter134/david_glore134.html). [Accessed 15 2015].
- [48] ISO/IEC/IEEE 24765, "Systems and software engineering - vocabulary," pp. 1-418, 2010.

- [49] ISO/IEC, International Standard ISO/IEC 9126-1., *Software engineering -- Product quality - Part 1: Quality model. International Organization for Standardization / International Electrotechnical Commission*, Geneva, 2001.
- [50] A. Hotho, A. Nürnberger and G. Paaß, "A brief survey of text mining," *LDV Forum - GLDV Journal for Computational Linguistics and Language Technology*, vol. 20, no. 1, pp. 19 - 62, 2005.
- [51] J. S and A. M, "Techniques, applications and challenging issue in text mining," *IJCSI International Journal of Computer Science*, vol. 9, no. 6, pp. 431 - 436, 2012.
- [52] K. M. Hammouda and M. S. Kamel, "Efficient phrase - based document indexing for web document clustering," *IEEE Transactions on Knowledge and Data Engineering*, vol. 16, no. 10, pp. 1279-1296, 2004.
- [53] Vector Space Model (n.d.), "In Wikipedia," [Online]. Available: [http://en.wikipedia.org/wiki/Vector\\_space\\_model](http://en.wikipedia.org/wiki/Vector_space_model). [Accessed 1 13 2013].
- [54] A. Budanitsky and G. Hirst, "Evaluating wordnet-based measures of lexical semantic relatedness," *Computational Linguistics*, vol. 32, no. 1, pp. 13-47, 2006.
- [55] D. Widdows, *Geometry and Meaning*, Stanford: CSLI Publications, 2004.
- [56] WordNet (n.d.), "In Wikipedia," [Online]. Available: <http://en.wikipedia.org/wiki/WordNet>. [Accessed 2 21 2015].
- [57] C. Trim, "The Art of Tokenization," 23 Jan 2013. [Online]. Available: <https://www.ibm.com/developerworks/community/blogs/nlp/entry/tokenization?lang=en>. [Accessed 3 10 2015].

- [58] G. Salton, *Introduction to Modern Information Retrieval*, McGraw-Hill, 1983.
- [59] D. Lewis, Y. Yang, T. G. Rose and F. Li, "RCV1: a new benchmark collection for text categorization research," *Journal of Machine Learning Research*, vol. 5, pp. 361-397, 2004.
- [60] G. A. Jivani, "A comparative study of stemming algorithms," *Journal of Computer Technology and Applications*, vol. 2, no. 6, pp. 1930 - 1939, 2011.
- [61] C. D. Manning, P. Raghavan and H. Schutz, *Introduction to Information Retrieval*, Cambridge University Press, 2008.
- [62] P. Bruza and L. Sitbon, "On the relevance of documents for semantic representation," in *Proceedings of 13th Australasian Document Computing Symposium*, Hobart, Australia, 2008.
- [63] D. Widdows and K. Ferraro, "Semantic vectors: a scalable open source package and online technology management application," in *Proceedings of the Sixth International Conference on Language Resources and Evaluation*, Marrakech, Morocco, 2008.
- [64] E. Mosqueira-Rey, D. Alonso-Rios and V. Moret-Bonillo, "Usability taxonomy and context of use taxonomy for usability analysis," in *Proceedings of the 2009 IEEE international conference on Systems, Man and Cybernetics*, San Antonio, TX, USA, 2009.
- [65] V. Brunom and G. Al-Qaimari, "Usability Attributes: An initial step toward effective user-centered development," 2004.
- [66] Extensibility (n.d.), "In Wikipedia," [Online]. Available: <http://en.wikipedia.org/wiki/Extensibility>. [Accessed 5 3 2015].

- [67] C. C. Chiang, "Software stability in software reengineering - information reuse and integration," in *IEEE International Conference on Information Reuse and Integration*, Las Vegas, NV, USA, 2007.
- [68] F. M. Bertoa, J. M. Troya and A. Vallecillo, "Measuring the usability of software components," *Journal of Systems and Software*, vol. 79, no. 3, pp. 427- 439, 2006.
- [69] B. C. Zapata, A. N. Hernandez, A. Idri, L. J. Fernandex-Aleman and A. Toval, "Mobile PHRs compliance with Android and iOS usability guidelines," *Journal of Medical Systems*, vol. 38, no. 8, p. 81, 2014.
- [70] Operability (n.d.), "In Wikipedia," [Online]. Available: <http://en.wikipedia.org/wiki/Operability>. [Accessed 4 12 2015].
- [71] H. S. Han, M. H. Yun, J. Kwahk and S. W. Hong, "Usability of consumer electronic products," *International Journal of Industrial ergonomics*, vol. 28, no. 3-4, pp. 143-151, 2001.
- [72] K. Kim, R. W. Proctor and G. Salvendy, "The relation between usability and product success in cell phones," *Behavior and Information Technology*, vol. 31, no. 10, pp. 969-982, 2012.
- [73] F. Balagatas-Fernandez and H. Hussmann, "A methodology and framework to simplify usability analysis of mobile applications," in *Proceedings of the 2009 IEEE/ACM International Conference on Automated Software Engineering*, Washington, DC, USA, 2009.
- [74] Fault tolerance (n.d.), "In Wikipedia," [Online]. Available: [http://en.wikipedia.org/wiki/Fault\\_tolerance](http://en.wikipedia.org/wiki/Fault_tolerance). [Accessed 4 10 2015].

- [75] L. D. Scapin and M. C. J. Bastien, "Ergonomic criteria for evaluating the ergonomic quality of interactive systems," *Behavior and Information Technology*, vol. 16, no. 4-5, 1997.
- [76] S. Winter, S. Wagner and F. Deissenboeck, "A comprehensive model of usability," *In Engineering Interactive Systems, Jan Gulliksen, Morton Borup Harning, Philippe Palanque, Gerrit C. Veer, and Janet Wesson (Eds.). Lecture Notes In Computer Science*, vol. 4940, pp. 106-122, 2008.
- [77] International Ergonomic Association (n.d.), "Definition and Domains of ergonomics | IEA website," [Online]. Available: <http://www.iea.cc/whats/index.html>. [Accessed 4 11 2015].
- [78] Cognitive load (n.d.), "In Wikipedia," [Online]. Available: [http://en.wikipedia.org/wiki/Cognitive\\_load](http://en.wikipedia.org/wiki/Cognitive_load). [Accessed 4 5 2015].
- [79] M. Maguire, "Context of use within usability activities," *International Journal of Human-Computer Studies*, vol. 55, no. 4, pp. 453-483, 2001.
- [80] E. Buie and D. Murray, *Usability in Government systems - User experience design for citizens and public servants*, Waltham, MA, USA: Morgan Kaufmann, 2012.
- [81] M. Kurosu and K. Kashimura, "Apparent usability vs. inherent usability - experimental analysis on the determinants of the apparent usability," in *In Conference Companion on Human Factors in Computing Systems (CHI '95)*, I. Katz, R. Mack, and L. Marks (Eds.). ACM, New York, NY, USA, 1995.

- [82] D. F. Davis, P. R. Bagozzi and M. R. Ryan, "Extrinsic and intrinsic motivations to using computers in the workplace," *Journal of Applied Social Psychology*, vol. 22, no. 14, 1992.
- [83] S. M. Angela and I. Flechais, "The Case for Usable Security," in *In L. Cranor and S. L. Garfinkel (eds), Designing secure systems that people can use*, O'Reilly and Associates, Cambridge, MA, USA, 2005.
- [84] Simplexity (n.d.), "In Wikipedia," [Online]. Available: <http://en.wikipedia.org/wiki/Simplexity>. [Accessed 4 28 2015].
- [85] Battle, L (n.d.), "Position Paper for: Creating a Usability Ontology," [Online]. Available: [http://www.ipgems.com/content/sample\\_position\\_paper\\_2007.pdf](http://www.ipgems.com/content/sample_position_paper_2007.pdf). [Accessed 4 5 2015].

**APPENDIX A. STOP WORD LIST OF 395 WORDS**

a	back	e	had	keep	Name	own	sec
able	be	each	happens	keeps	namely	p	section
about	became	ed	hardly	kept	nay	page	see
above	because	edu	has	kg	nd	pages	seeing
abst	become	effect	hasn't	km	near	part	seem
accordance	becomes	eg	have	know	nearly	particular	seemed
according	becoming	eight	haven't	known	necessarily	particularly	seeming
accordingly	been	eighty	having	knows	necessary	past	seems
across	before	either	he	l	need	per	seen
act	beforehand	else	hed	largely	needs	perhaps	self
actually	begin	elsewhere	hence	last	neither	placed	selves
added	beginning	end	her	lately	nevertheless	please	sent
adj	beginnings	ending	here	later	new	plus	seven
affected	begins	enough	hereafter	latter	next	poorly	several
affecting	behind	especially	hereby	latterly	nine	possible	shall
affects	being	et	herein	least	ninety	possibly	she
after	believe	et-al	heres	less	no	potentially	shed
afterwards	below	etc	hereupon	lest	nobody	pp	she'll
again	beside	even	hers	let	non	predominantly	shes
against	besides	ever	herself	lets	none	present	should
ah	between	every	hes	like	nonetheless	previously	shouldn't
all	beyond	everybody	hi	liked	noone	primarily	show
almost	biol	everyone	hid	likely	nor	probably	showed
alone	both	everything	him	line	normally	promptly	shown
along	brief	everywhere	himself	little	nos	proud	shows
already	briefly	ex	his	'll	not	provides	shows
also	but	except	hither	look	noted	put	significant
although	by	f	home	looking	nothing	q	significantly
always	c	far	how	looks	now	que	similar
am	ca	few	howbeit	ltd	nowhere	quickly	similarly
among	came	ff	however	m	o	quite	since
amongst	can	fifth	hundred	made	obtain	qv	six
an	cannot	first	i	mainly	obtained	r	slightly



and	can't	five	id	make	obviously	ran	so
announce	cause	fix	ie	makes	of	rather	some
another	causes	followed	if	many	off	rd	somebody
any	certain	following	i'll	may	often	re	somehow
anybody	certainly	follows	im	maybe	oh	readily	someone
anyhow	co	for	immediate	me	ok	really	somethan
anymore	com	former	immediately	mean	okay	recent	something
anyone	come	formerly	importance	means	old	recently	sometime
anything	comes	forth	important	meantime	omitted	ref	sometimes
anyway	contain	found	in	meanwhile	on	refs	somewhat
anyways	containing	four	inc	merely	once	regarding	somewhere
anywhere	contains	from	indeed	mg	one	regardless	soon
apparently	could	further	index	might	ones	regards	sorry
approximately	couldnt	furthermore	information	million	only	related	specifically
are	d	g	instead	miss	onto	relatively	specified
aren	date	gave	into	ml	or	research	specify
arent	did	get	invention	more	ord	respectively	specifying
arise	didn't	gets	inward	moreover	other	resulted	still
around	different	getting	is	most	others	resulting	stop
as	do	give	isn't	mostly	otherwise	results	strongly
aside	does	given	it	mr	ought	right	sub
ask	doesn't	gives	itd	mrs	our	run	substantially
asking	doing	giving	it'll	much	ours	s	successfully
at	done	go	its	mug	ourselves	said	such
auth	don't	goes	itself	must	out	same	sufficiently
available	down	gone	i've	my	outside	saw	suggest
away	downwards	got	j	myself	over	say	sup
awfully	due	gotten	just	n	overall	saying	sure
b	during	h	k	na	owing	says	

**APPENDIX B. LIST OF USABILITY ATTRIBUTES AND THEIR SEMANTIC  
MATCHES FROM LEXICOGRAPHIC DATABASES**

acceptability	suitability adequacy reasonableness desirability correctness usefulness validity
accessible	access readable available affordable comprehensible navigable adaptable
accuracy	fidelity precision quality reliability correctness consistency validity
adaptable	agile versatile resilient amenable approachable responsive tolerant
availability	accessibility available convenience suitability reliability utilization scarcity
completeness	completion finish accomplish fulfill accomplishment fulfillment
complexity	intricacy complex complicated complicate sophistication ambiguity subtlety
consistency	coherence cohesion solidity uniformity consistent quality similarity
ease	richness affluence simplicity thoroughness versatility clarity lightness
effectiveness	feasibility usefulness reliability quality validity acceptability accuracy
efficiency	reliability productivity efficient capacity performance accuracy competitiveness
error	mistake fault failure omission deviation lapse miscalculation
fidelity	faithfulness accuracy dedication quality sharpness adherence durability
flexibility	ability ease versatility leeway autonomy agility transparency
functionality	operational aesthetic conceptual practical compatible spatial
learn	read discover memorize practice train learning
operable	serviceable separable negotiable replaceable navigable
privacy	confidentiality secrecy anonymity separation disclosure freedom safeguard
quality	effectiveness suitability attractiveness usefulness accuracy credibility
reliability	dependable reliability accurate trustworthy, credible efficient affordable
responsiveness	responsive sensitivity quality friendliness helpfulness reliability
robustness	reliable flexible dependable durable secure adaptable credible versatile
satisfaction	gratification comfort contentment happiness fulfillment enjoyment
security	safety protection assurance privacy
simplicity	naturalness brevity clarity
standards	criterion benchmark guideline requirement rule regulation,
understandable	comprehensible intelligible justifiable readable legible evident
usefulness	practicality viability feasibility helpfulness utility relevance validity
utility	usefulness practicality feasibility viability
familiarity	acquaintance familiar intimacy conversant friendliness
safety	security protection protect standard privacy

**APPENDIX C. LIST OF 76 ATTRIBUTES USED FOR HIERARCHICAL CLUSTERING**

interoperability	Completeness	customizability
portability	Accuracy	observability
scalability	error-prevention	predictability
extensibility	effectiveness	comprehensibility
adaptability	Ease	learnability
suitability	fulfillment	simplicity
acceptability	attractiveness	flexibility
testability	timeliness	generalizability
compliance	responsiveness	familiarity
traceability	Agility	memorability
replaceability	efficiency	validity
maintainability	robustness	contentment
stability	correctness	satisfaction
operability	fault-tolerance	attitude
recoverability	faithfulness	friendliness
availability	Maturity	utility
understandability	reliability	relevant
uniformity	Feedback	privacy
complexity	task-conformance	confidentiality
readability	standards	security
accessibility	benchmarks	integrity
ergonomic	consistency	safety
workload	enjoyment	navigability
cognitive-load	happiness	aesthetics
context	anxiety	
autonomy	playfulness	