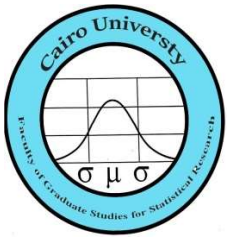
CAIRO UNIVERSITY
FACULTY OF GRADUATE STUDIES FOR STATISTICAL RESEARCH



A New Crowdsourcing Requirements Elicitation Method for eLearning Systems

By
Nancy Mohamed Rizk

Under the supervision of
Dr. Mervat Gheith
Dr. Eman Nasr

A thesis submitted to the Department of Information Systems in partial fulfillment of the requirements for the degree of Doctor of Philosophy in Information Systems.

2020

Table of Contents

List of Figures

List of Tables

List of Abbreviations

AMT	Amazon Mechanical Turk
B2B	Business to Business
CCRE	Crowd- Centric Requirements Engineering
COVID	Corona VIrus Disease
CREeLS	Crowdsourcing Requirements Elicitation for eLearning Systems
eLS	eLearning Systems
GSD	Global Software Development
GRC	Governance Risk and Compliance
IDF	Inverse Document Frequency
IS	Information System
LCMS	Learning Content Management System
LDA	Latent Drichlit Allocation
LMS	Learning Management System
LO	learning Object
LSA	Latent Semantic Allocation
ME	Method Engineering
MOOC	Massive Open Online Course
NFRs	Non-Functional Requirements
NLP	Natural Language Processing
NLTK	Natural Language ToolKit
POS	Part of Speech
UCL	University College London
RE	Requirements Engineering
SCORM	Shareable Content Object Reference Model
SLR	Systematic Literature Review
SNA	Social Network Analysis
SNS	Social Networking Site
TF	Term Frequency
TF-IDF	Term Frequency-Inverse Document Frequency
QoS	Quality of Service
WAEU	Wide Audience End-Users

Acknowledgment

All praise and gratitude are due to **Allah** (glorified and exalted be He) the actual helper and supporter. I would like to express my deep and truthful gratefulness to my supervisor **Dr. Mervat Gheith** for her understanding, and encouragement. Also, I would like to thank her for the constructive comments, and important support throughout this work.

I also owe my most sincere gratitude to my supervisor **Dr. Eman Nasr** who was abundantly helpful and offered invaluable assistance, constant support and continuous guidance. Her wide knowledge and her logical way of thinking have been of great value for me. Words can't describe how grateful I am for my dear **father** and my beloved **mother;** without their encouragement and understanding, it would have been impossible for me to finish this work. My special gratitude is due to my **brothers** for their loving support. I owe special gratitude to **Karim El-Goraisy,** my husband for his continuous support and motivation, and special thanks to my beloved kids for their pure and endless love. I'm very grateful to **G2 Company** in USA for sharing with us its Learning Management Systems users' reviews data.

List of Publications

1 - Rizk, N. M., Nasr, E. S., & Gheith, M. H. (2016). "Requirements' Elicitation Needs for eLearning Systems," in *The Proceedings of The 12th International Computer Engineering Conference* (*ICENCO*), (28-29 Dec.) Giza, Egypt, pp. 142-147.

2 - Rizk, N. M., Nasr, E. S., & Gheith, M. H. (2018). "A Proposed Plan for Crowdsourcing as a Requirements Elicitation Method for eLearning Systems" in The Proceedings of The 53rd Institute of Statistical Studies and Research, Giza, Egypt, pp. 15-23.

3 - Rizk, N. M., Zaki, A. M., Nasr, E. S. & Gheith, M. H. (2019), "CREeLS: Crowdsourcing based Requirements Elicitation for eLearning Systems," International Journal of Advanced Computer Science and Applications, vol. 10, no. 10, pp. 242-251.

4 - Rizk, N. M., Nasr, E. S., & Gheith, M. H (2019) "Enhancing CREeLS the Crowdsourcing based Requirements' Elicitation Approach for eLearning Systems Using Bi-gram Evaluation," in The Proceedings of *The 15th International Computer Engineering Conference (ICENCO)*, (29-30 Dec.), Giza, Egypt, pp. 222, 226.

Abstract

eLearning is gaining more importance, and becoming more popular nowadays. The outbreak of the COrona VIrus Disease (COVID-19) pandemic across the globe has forced educational institutions to shut down to control the spread of the virus. This happening forced the spread out of eLearning or online learning, in which teaching professionals and students are virtually connected. It has become mandatory for all educational institutions like schools, colleges, and universities all around the world. eLearning Systems (eLS) have a special nature with special characteristics, such as the large number and diversity of users who could be geographically dispersed. eLS are in continuous need for improvements to meet their users' requirements. Hence requirements elicitation is of major importance. Requirements elicitation is an activity within the Requirements Engineering (RE) phase, the early stage in the software engineering process, which is responsible for deriving the system requirements and their evolution. Requirements engineers should pay attention to the special eLS requirements in the software development phases and specially in the requirements elicitation phase for better and fast satisfaction of stakeholders' needs. Current requirements elicitation techniques, e.g. interviews, questionnaire, and observation have many limitations and challenges and can't satisfy the continuous and fast changing demand of eLearning users. Among those limitations are the inadequate involvement of users and stakeholders who are geographically dispersed. Crowdsourcing could offer a solution for that. It is defined as the use of the power of the crowd to achieve different tasks. The use of crowdsourcing is one of the new ways used for solving the problems of current requirements elicitation approaches because it focuses on involving stakeholders and there are supported tools to overcome the geographically dispersed stakeholders. Also, crowdsourcing can be advantageous if used in a continuously changing environment such as the eLS environment. It is used in the field of RE as an emerging concept to help simplifying its activities. To the best of our knowledge, there is little evidence that a crowdsourcing-based RE approach or method especially tailored for eLS that can address their special characteristics exists. In this research we attempt to fill in this gap. We propose a new Crowdsourcing based method for Requirements Elicitation for eLS (CREeLS),

which is made up of a framework and five phases. CREeLS considers the special nature of eLS during the requirements elicitation activity. The framework is for the general recommended tools of the crowdsourcing and the phases are to follow it to overcome the challenges and limitations of the requirements elicitation for eLS. The framework is composed of the necessary elements of crowdsourcing, which are; the crowd, users' feedback, interactivity, text mining tools and social collaboration. For each element of the framework there is a suggested tool to be used to conduct the crowdsourcing concept. CREeLS' phases aim to implement the framework, which are: 1- create a channel for users to post their feedback, or show their interactions, 2- extracting users' interactions or feedback, 3- analyzing users' interactions or feedback, 4- evolving refined software requirements, and 5- categorizing and consolidating the requirements. The proposed method was evaluated through analyzing about 4000 users' reviews for three educational Learning Management Systems (LMS) which are Blackboard, Google Classroom, and Canvas. We succeeded to extract keywords that represent users' requirements by the use of topic modeling techniques. First, we used Latent Drichlit Allocation (LDA) uni-gram topic modeling, the results were evaluated by manually reviewing users' text and the extracted features. The results of the evaluation process were found to be coherent with an average of 0.79 precision, 0.44 recall and 0.56 f-measure. Then, we further evaluated the method to enhance the results using LDA bi-gram topic modeling to better extract eLearning users' requirements and help in the requirements elicitation and evolution of eLS. The bi-gram evaluation is used on the same number of LMS products with the exact number of text reviews. We compared the results with manual extraction of the requirements; the average results were 0.76 precision, 0.61 recall and 0.68 f-measure. The extracted requirements were more understandable and relevant. CREeLS results were compared to published results for some of the related work, which applied almost the same evaluation techniques and measures as used in CREeLS but in other application domain, and the results were found comparable. Hence, we contend that CREeLS can help requirements engineers of eLS to analyze users' opinions and identify the most common users' requirements for better software evolution.

1 Introduction

1.1 Overview

Crowdsourcing is known to be the process of obtaining the needed services by outsourcing them to the crowd. The word crowd is defined in the English language as a group of people with a common interest[1]. Crowdsourcing as a term was coined by Jeff Howe in Wired in June 2006. In his article "The Rise of Crowdsourcing" (Howe, 2006) he described how small businesses are getting successful by using "the power of the crowd", rather than the traditional professional ways, to cut their costs. Howe mentioned that crowdsourcing can be used at any time when needed in an organization (Howe, 2006).

The use of "the power of the crowd" to achieve specific tasks is gaining more and more ground every day. The evolution of Web 2.0 enables internet applications to allow sharing and collaboration opportunities to people, emphasizing user-generated contents (O'Reilly, 2005). Web 2.0 blurred the line between content creators and content consumers and as a result empowered large crowd of users to collaborate, organize and share knowledge (Karataev & Zadorozhny, 2017). Web evolution made crowdsourcing used in solving the problems in the field of Requirements Engineering (RE) too help in simplifying the activity of requirements elicitation, which usually involves various stakeholders (Groen, et al., 2015) (Groen & Koch, 2016) (Hosseini, et al., 2014) (Sharma & Sureka, 2017) (Mao, et al., 2016). According to Sommerville (Sommerville, 2015). RE involves "all life-cycle activities devoted to identification of user requirements, analysis of the requirements to derive additional requirements, documentation of the requirements as a specification, and validation of the documented requirements against user needs, as well as processes that support these activities" ((DoD), Department of Defense, 1991). RE activities are: requirements elicitation, requirements analysis, requirements specification and requirements validation. This research will focus on the requirements elicitation activity only.

[1] https://dictionary.cambridge.org/dictionary/english/crowd

Requirements elicitation is an early software development activity within the RE phase. It is concerned with understanding and learning stakeholders' needs (Zowghi & Coulin, 2005). It is a very important activity for the success of a software development project, where detecting errors at the early stages of development can save money and time (Standish Group International, 2013). According to the Standish Group CHAOS Report (Standish Group International, 2013), users' involvement is one of the critical success factors in any software development project and this is usually performed in the requirements elicitation activity. Among the limitations of the traditional requirements elicitation techniques (e.g. interviews, questionnaire, and observation) are the narrow concept of stakeholders, the limited involvement of users with knowledge in requirements prioritization, and the bias of a requirements engineer who focus on certain types of requirements, in addition to, the geographically dispersed stakeholders, and the continuously changing technical and social environment e.g. culture, stakeholders' opinions (Snijders, et al., 2015) (Srivastava & Sharma, 2015) (Abdul Rahman & Sahibuddin, 2011). Crowd-based RE, which was coined as a term by Groen et al., is a highly interactive approach; it can get user requirements in less time, helps in getting new ideas for software evolution, and has the potential to increase the quality of requirements elicitation (Groen, et al., 2015).

Among the information systems that were highly affected by the Web evolution are the eLS. eLearning is well known to be the use of technology in the delivery of education, where in some cases the learning resources are accessed online anywhere and anytime (Casey & Wilson, 2007), (Holmes & Gardner, 2006). COrona VIrus Disease (COVID-19) pandemic has created the largest disruption of education systems in history, affecting 94 percent of the world's student population due to the school closure (United Nations, 2020). On the other hand, this crisis has stimulated innovation within the education sector (Li & Lalani, 2020). Education has changed dramatically, with the rise of eLearning, it has become mandatory for all educational institutions like schools, colleges, and universities all around the world (Radha, et al., 2020). eLearning has different tools, types, and information systems. eLS have two main types of management systems; Learning Management Systems

(LMS) and Learning Content Management Systems (LCMS). LMSs are concerned with the administrative process of learning, such as scheduling, testing, billing and registering learners, e.g. Moodle, and Blackboard (Horton & Horton, 2003). LCMSs combine the administrative processes of LMS with the authoring and content creation dimensions (Gheorghiu, 2017) (Irlbeck & Mowat, 2007). Hence eLS can be defined to be the systems that are concerned with the administering or content authoring tools to help students, instructors and management in the learning process. eLS can be used in an educational context or a corporate training context.

Among the characteristics of eLS are the large number and diversity of eLearning users in terms of background, geographical locations, and culture. Also, the high interactivity nature of the learning process leads to an on-going demand of requirements that should be fulfilled for better improvement of the learning process and satisfaction of stakeholders. Limitations of the traditional requirements elicitation approaches also exist when requirements elicitation are performed on eLS (AlKhuder & AlAli, 2017), (Abdul Rahman & Sahibuddin, 2011). Hence, we propose crowdsourcing to be used in the eLearning context to handle the eLS characteristics and serve in the requirements elicitation activity. To the best of our knowledge, there is little evidence that a crowdsourcing-based requirements elicitation approach or method especially tailored for eLS addressing their special characteristics exists. In this research we attempt to fill in this gap to increase the quality of eLS' requirements elicitation and be able to get user requirements in less time, or get new ideas for software evolution by reaching greater number of stakeholders no matter their location or culture. We claim that crowdsourcing can use "the power of the crowd" through the power of Web 2.0 technologies to better elicit the stakeholders' requirements for eLS.

In this thesis a proposed Crowdsourcing based Requirements Elicitation for eLS (CREeLS) is presented. Requirements elicitation acronym RE is only used within CREeLS name and shouldn't be confused with the acronym of Requirements Engineering. CREeLS is made up of a framework and phases. The framework is composed of the necessary elements of crowdsourcing, suggesting specific tools for each element, while the phases aim to

implement the framework in the requirements elicitation activity for eLS. The method is based on crowdsourced eLS stakeholders' inputs that would be analyzed by the requirements engineers for the evolution for eLS. Figure 1-1 provides an overview of the discussed method to extract eLS requirements. The method extracts eLS requirements through analyzing users' feedback for eLS users' reviews, which will be considered by requirements engineers who prepare eLS requirements reports.

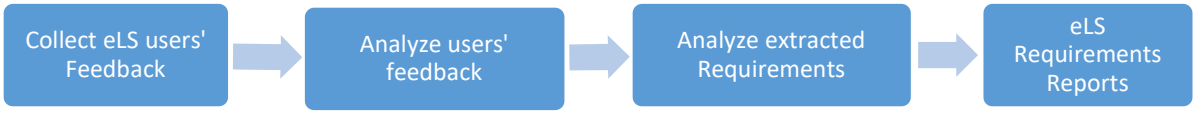


Figure 1-1. Method overview

An experimental study was conducted to evaluate the validity of the proposed method. In addition, manual reviewing of user's requirements was used for the evaluation of the experimental study. We first used LDA uni-gram topic modeling. LDA is chosen because it is a popular method for fitting a topic model. It treats each document as a mixture of topics, and each topic as a mixture of words. This allows documents to "overlap" each other in terms of content, rather than being separated into discrete groups, in a way that mirrors typical use of natural language (Silge & Robinson, 2017). The other option is Latent Semantic Analysis (LSA), LSA focuses on reducing dimension of classification while LDA solves topic modeling problems (Ma, 2018); the latter is what we focus on. LDA represents topics by word probabilities. the results were evaluated by manual text reviewing and the extracted features were found to be coherent. The results have an average of 0.79 *precision*, 0.44 *recall and 0.56 f-measure*. Then we further evaluated the method to enhance the results using LDA bi-gram topic modeling to better extract eLearning stakeholders' requirements and help in the requirements elicitation and evolution of eLS. The bi-gram evaluation is used on the same number of LMS products with the exact text reviews and we compared the results with manual extraction of the requirements. The

average results were *0.76 precision, 0.61 recall* and *0.68 f-measure*. The extracted requirements were more understandable and relevant. CREeLS results then used to be compared to published results of some of the related work which apply almost the same evaluation techniques and measures like what is used in CREeLS, the results were found comparable. Hence, we contend that the proposed method can help requirements engineers of eLS to analyze users' opinions and identify the most common users' requirements for better software evolution.

1.2 Research Motivation

eLearning seems to be the forthcoming trend; it has been spreading out widely. eLearning is best suited for everyone, depending on their availability and comfort, many people choose to learn at a convenient time. This enables the learner to access updated content whenever they want. The evolution of the Web and mobile applications affect the number of stakeholders and end users to become very huge. Among the information systems that were highly affected by the Web evolution are the eLS. eLearning is an old term that has different tools and techniques that were used for ages to support the physical learning process, however nowadays eLearning is boosting and becoming the first preference for corporates and individuals. eLearning for corporates increased by 900% from 2001 to 2017; in 2015 around 50% of the students worldwide agreed that they had enrolled in an online course in the preceding twelve months (Chernev, 2019). COrona VIrus Disease (COVID-19) pandemic has created the largest disruption of education systems in history, affecting 94 percent of the world's student population due to the school closure (United Nations, 2020). On the other hand, this crisis has stimulated innovation within the education sector. Education has changed dramatically, with the distinctive rise of eLearning, whereby teaching is undertaken remotely and on digital platforms (Li & Lalani, 2020). eLearning has become mandatory for all educational institutions like schools, colleges, and universities all around the world due to the pandemic crisis of COVID-19. eLearning is the domain of research for this thesis.

Alharithi et al. (Alharthi, et al., 2019) have confirmed that eLearning as software is a special type of system that has some characteristics, which leads to continuous demand and evolution of requirements. These requirements should be contented for better improvement of eLS and satisfaction of its stakeholders. Software engineers should pay attention to the special eLS requirements in the software development phases for better and fast satisfaction of stakeholders needs. Especially for both the requirements elicitation as an early activity in software development process and the requirements evolution phase in which the stakeholders became aware of missing requirements after the system is introduced to the market. The process of requirements elicitation is generally accepted as one of the critical activities in the RE process (Zowghi & Coulin, 2005). Getting the right requirements is considered a vital but difficult part of software development projects (Capers, 1996). The Standish group Chaos report 2013 (Standish Group International, 2013), in Table 1-1 illustrates the results of the studied 50,000 projects around the world, ranging from tiny enhancements to massive systems re-engineering implementations. The report includes an enhanced definition of success looking at some additional factors which were covered in previous surveys. The results indicate that there is still work to be done around achieving successful outcomes from software development projects. RE plays a very important role in software development. For some years now, it has been recognized that problems associated with RE are among the major reasons for software project failures where the end product does not meet the real needs of the project owners (Hull, et al., 2011). The first factor of the software projects challenges is due to the lack of users' inputs and the two top factors for the projects to be impaired are the incomplete requirements, and the lack of users' involvement.

Table 1-1. Chaos statistics on IS projects success (Standish Group International, 2013)

	2011	2012	2013	2014	2015
SUCCESSFUL	29%	27%	31%	28%	29%
CHALLENGED	49%	56%	50%	55%	52%
FAILED	22%	17%	19%	17%	19%

Requirements elicitation activity has some limitations and threats e.g. the limited involvement of users with knowledge in requirements prioritization, and the geographically dispersed stakeholders, and the uncertain technical and social environment (Snijders, et al., 2015) (Srivastava & Sharma, 2015) (Abdul Rahman & Sahibuddin, 2011). The eLS development process suffers from the same problems of the requirements elicitations activity. Hence the use of a method that relies on the eLearning stakeholders for the elicitation of eLearning requirements will help in both the eLS development and evolution. Requirements elicitation and evolution are the domain of research in this thesis.

1.3 Research Scope

We aim to find a new method of RE. Figure 1-2 illustrates the intersection between the three main fields of research under study in this thesis, which yields the new proposed method (CREeLS). Recently, crowdsourcing was investigated as an opportunity in the requirements elicitation phase (Mao, et al., 2016) (Snijders, et al., 2015). Crowdsourcing is used as the approach of the proposed framework and phases.

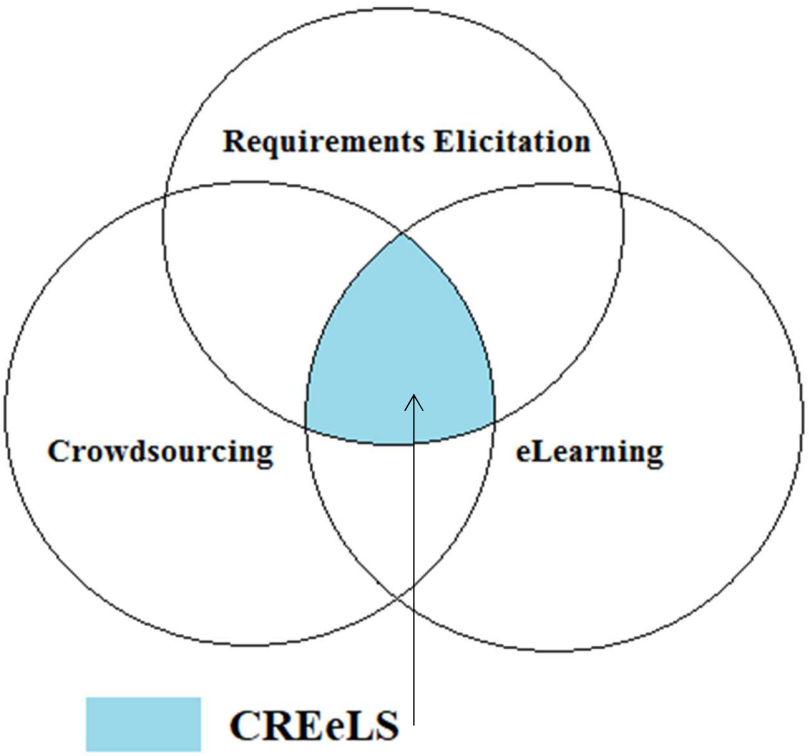


Figure 1-2. Research scope

1.4 Problem Statement

The evolution of the Web and mobile applications increase the number of stakeholders and end users to become very huge. Current requirements elicitations' techniques and approaches are having different challenges and limitations. The narrow concept of stakeholders, the limited involvement of users with knowledge in requirements prioritization, and the bias of a requirements engineer who focus on certain types of requirements, In addition to, the geographically dispersed stakeholders, and the uncertain technical and social environment are among the limitations and threats of the current requirements elicitation approaches (Vogel & Grotherr, 2020) (Snijders, et al., 2015) (Abdul Rahman & Sahibuddin, 2011) (Srivastava & Sharma, 2015).

Requirements elicitation for eLS have almost the same issues of the traditional RE techniques or approaches (AlKhuder & AlAli, 2017), (Abdul Rahman & Sahibuddin,

2011). There is a gap in research for eliciting requirements for eLS. Also, there is a need for newer and up to date activities that take benefits from the existing concepts of sharing in Web 2.0 and social networking to enhance the elicitation phase.

1.5 Research Objectives

The objective of this research is to propose a new requirements elicitation method for eLS. The method should help to support the elicitation of requirements from a large number of stakeholders for developing a new eLearning software project, or for enhancing and evolution of existing eLS. The method is based on the use of crowdsourcing, which is one of the new ways used for solving the challenges and limitations of the current RE approaches because it focuses on involving stakeholders and there are supported tools to overcome the geographically dispersed stakeholders. Also, crowdsourcing can advantageous if used in a continuously changing environment such as the eLS environment.

1.6 Research Methodology

The research objectives mentioned in the last section are realized by the following:

- Assessing the literature for the different techniques, approaches, and methods for RE for eLS.
- Assessing the literature for the existing challenges of requirements elicitation in general and for requirements elicitation for eLS in particular.
- Identifying the challenges that the research can provide a solution for it.
- Identifying the special properties that characterize eLS.
- Comparing the different researches found, their recommendations, and identifying the current unsolved problems.
- Assessing the literature for crowdsourcing in RE.
- Proposing a crowdsourcing-based method of requirements elicitation for eLS.
- Evaluating the proposed method.
- Enhancing the method based on the results of the initial experimental study.

- Analyzing results, stating the study findings, limitations and threats to validity.

1.7 Thesis Outline

The thesis consists of six chapters. Chapter 1 declares the problem and the objectives of the thesis. Chapter 2 introduces eLearning, the application domain in this thesis, also provides a short background about requirements elicitation, requirements evolution and crowdsourcing term. Chapter 3 It provides a survey for eLearning in requirements elicitation. It also provides literature survey to cover the use of crowdsourcing in RE in general and requirements elicitation in particular. and an overview of the use of crowdsourcing in eLS. Chapter 4 introduces CREeLS the proposed method to use crowdsourcing in eliciting requirements for eLS. Chapter 5 discusses the experimental studies and evaluation of the proposed method. Finally, the conclusion and the future work are presented in chapter 6.

2 Background

2.1 Introduction

This chapter presents background information about requirements elicitation and requirements evolution activities. eLearning and eLearning as a system is introduced. In addition, the crowdsourcing as a term and as a helping tool in RE is also introduced.

2.2 Requirements Engineering

RE is an early stage in the software engineering process. The following definition is one of the oldest, most long-standing, and comes from US Department of Defense (DoD) software strategy document in 1991 ((DoD), Department of Defense, 1991), it covers the RE processes which are; identification, analysis, development and validation of requirements. DoD RE definition is "involves all life-cycle activities devoted to identification of user requirements, analysis of the requirements to derive additional requirements, documentation of the requirements as a specification, and validation of the documented requirements against user needs, as well as processes that support these activities". Newer RE activities, illustrated in Figure 2-1, can be concluded according to DoD definition and the following Sommerville RE activities (Sommerville, 2015) as:

- Feasibility study: it's a study considers whether the proposed system will be cost-effective from a business point of view and whether it can be developed within existing budgetary constraints.
- Requirements elicitation and analysis: This is the activity of deriving the system requirements through the use of different techniques along the potential users and procurers. This helps the analyst understand the system to be specified.
- Requirements specification: This is the activity of translating the information gathered during the analysis activity into a document that defines a set of requirements.

- Requirements validation: This activity checks the requirements for realism, consistency and completeness. During this process, errors in the requirements document are inevitably discovered and solved.
- Requirements evolution: This activity is concerned with evolving the software according to changes in users' needs and requirements (Sommerville, 2015), (Ali, et al., 2011). Changing requirements are considered as one of the most significant risks for software systems development. On the other hands, these changing requirements also represent opportunities to exploit new and evolving business conditions (Ernst, et al., 2014).

In the next parts the focus will be on requirements elicitation and requirements evolution activities. We will consider both of the activities in the thesis investigations and experiments. References are considered old at some parts in terms of publication year because they address the basic terms and definitions of RE.

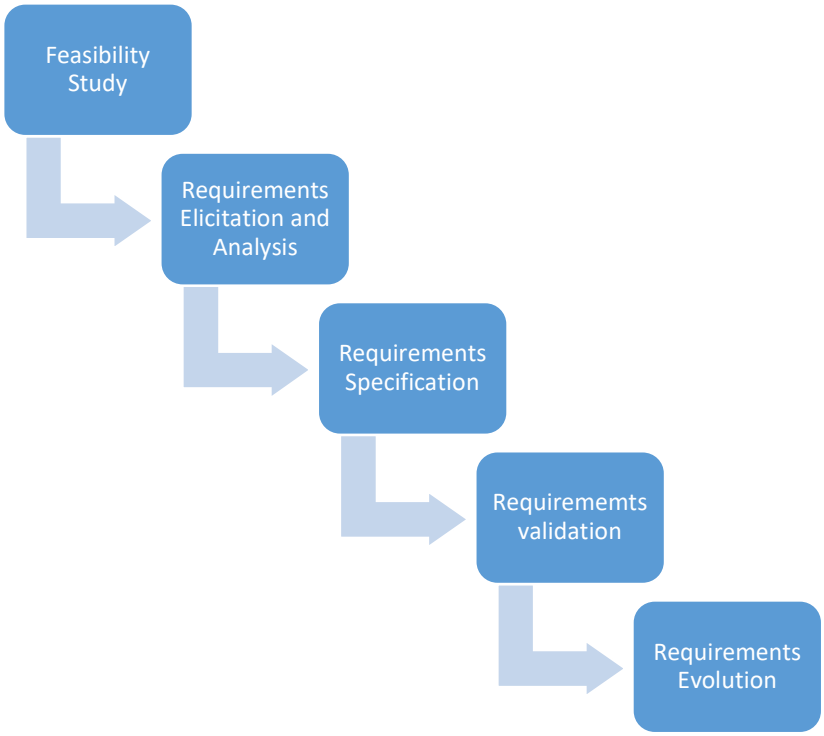


Figure 2-1. Adapted RE processes ((DoD), Department of Defense, 1991), (Sommerville, 2015)

According to Zowghi et al. (Zowghi & Coulin, 2005) there is no one agreed on definition for requirements elicitation, Hickey and Davis (Hickey & Davis, 2004) say its "learning, uncovering, extracting, surfacing, or discovering needs of customers, users, and other potential stakeholders", Zowghi and Coulin (Zowghi & Coulin, 2005) claimed that requirements elicitation is "concerned with learning and understanding the needs of users and project sponsors with the ultimate aim of communicating these needs to the system developers". Requirements elicitation is a social activity, because its nature is interacting with the system stakeholders to acquire their system's needs and requirements. Relatively old references are used across the thesis to discuss the basic terms, in other cases these references are used because they are considered as very important references in its field.

2.2.1 Requirements Elicitation Techniques vs. Approaches vs. Methods

There are a lot of different techniques, approaches or methods from a variety of sources that have been employed for requirements elicitation. In section 2.2.2 only some of those that are more widely used are presented.

In this section we attempt to present the difference between the three terms. We have to say that there is a great confusion in literature between the meanings of method versus approach; they have been used interchangeably in requirements elicitation literature.

A technique in the English language is defined by Merriam Webster [2]as "a method of accomplishing a desired aim". A technique in requirements elicitation is a precise strategy, a tested and trusted tip that's designed to help to reach goals. It is defined by (Brinkkemper, 1996) as "a procedure, possibly with a prescribed notation, to perform a development activity". A technique could be in form of an exercise or just any activity to complete a mission. It is defined by (Zowghi & Coulin, 2005) as "a way of doing something or a practical method applied to some particular task". From the previous definitions we can notice the different perspectives in defining "technique" within requirements elicitation. We will adopt the technique definition provided by Zowghi & Coulin.

[2] https://www.merriam-webster.com/dictionary/technique

An approach could be seen as a perspective, ideology, belief or theoretical stance on something. It encompasses a set of logical assumptions that could be made for better comprehension of issues. It could also be seen as a term that produce systematic plans and the strategies used to achieve particular objectives. “It can be a systematic arrangement, usually in steps, of ideas or actions intended to deal with a problem or situation” (Zowghi & Coulin, 2005). In the English language an approach is defined by Merriam Webster as “the taking of preliminary steps toward a particular purpose” [3].

A method in the English language is defined by Merriam Webster as “a procedure or process for attaining an object: such as: a systematic procedure, technique, or mode of inquiry employed by or proper to a particular discipline or art” [4]. A method as defined by (Nuseibeh & Easterbrook, 2000) is “a prescription for how to perform a collection of activities, focusing on how a related set of techniques can be integrated, and providing guidance on their use”; we adopt this definition in this thesis. It consists of heuristics and guidelines for the requirements engineer at different stages of a process. Likewise, Kramer and others define a method as “a grammar of steps and principles for applying them rather than just a collection of notations” (Kramer, et al., 1988). Brinkkemper definition of method is “an approach to perform a systems development project, based on a specific way of thinking, consisting of directions and rules, structured in a systematic way in development activities with corresponding development products” (Brinkkemper, 1996).

Concerning the requirements elicitation approaches and methods there is no agreed-on methods or approaches for requirements elicitation that are practically used, we surveyed the literature but we only found some studies that propose requirements elicitation methods.

The situational method is a type of method proposed by (Brinkkemper, 1996) in the light of his publication which explains method engineering of information systems development tools. Brinkkemper proposed situational method to be used in information system

[3] https://www.merriam-webster.com/dictionary/approach
[4] https://www.merriam-webster.com/dictionary/method

development however zowghi et al. (Coulin , et al., 2006) used this method to propose a systematic approach to the requirements elicitation in software development, based on collaborative workshops and the construction of a lightweight situational method, within a general process framework.

2.2.2 Requirements Elicitation Traditional Techniques

Next is explanation of some of the most common traditional requirements elicitation techniques according to (Zowghi & Coulin, 2005) Many of these methods have been borrowed and adapted from other disciplines such as the social sciences. Then we will state some of the traditional techniques' challenges and drawbacks.

- **Interviews**

Interviews are probably the most traditional and commonly used technique for requirements elicitation. Because interviews are essentially human based social activities, they are inherently informal and their effectiveness depends greatly on the quality of interaction between the participants. The results of interviews, such as the usefulness of the information gathered, can vary significantly depending on the skill of the interviewer.

- **Questionnaires**

Questionnaires are mainly used during the early stages of requirements elicitation. For them to be effective, the terms, concepts, and boundaries of the domain must be well established and understood by the participants and questionnaire designer.

- **Task Analysis**

Task analysis employs a top-down approach where high-level tasks are decomposed into subtasks and eventually detailed sequences until all actions and events are described.

- **Domain Analysis**

Examining the existing and related documentation and applications is a very useful way of gathering early requirements as well as understanding and capturing domain knowledge, and identification of reusable concepts and components.

- **Joint Application Development (JAD)**

Joint Application Development (JAD) involves all the available stakeholders investigating through general discussion both the problems to be solved, and the available solutions to those problems. With all parties represented, decisions can be made rapidly and issues resolved quickly.

- **Ethnography**

Ethnography or observation is the study of people in their natural setting. It involves the analyst actively or passively participating in the normal activities of the users over an extended period of time whilst collecting information on the operations being performed.

- **Prototyping**

Providing stakeholders with prototypes of the system to support the investigation of possible solutions is an effective way to gather detailed information and relevant feedback.

- **Scenario**

Scenarios are widely used in requirements elicitation and as the name suggests are narrative and specific descriptions of current and future processes including actions and interactions between the users and the system. .

2.2.3 Problems of Traditional Requirements Elicitation Techniques

According to (Michael & Kyo, 1992) problems of requirements elicitation can be grouped into three categories:

- Problems of scope, in which the requirements may address too little or too much information. Requirements engineers should focus on identifying the requirements from the system's users rather than requirements that serve the design of the system. Requirements can serve the organizational goals, the environmental and project context of the target systems. Too narrow or too broad requirements elicitation will result in ambiguous, incomplete, unnecessary or unusable requirements.
- Problems of understanding, within groups as well as between groups such as users and developers. Problems of understanding in requirements elicitation include the variations of communities' backgrounds and experiences involved. Also, the language used to express the requirements weather too formal or too informal and the structure of the elicited information, which is also affected by the variations of elicitation communities.
- Problems of volatility, i.e., the changing nature of requirements. Requirements may need change overtime or we can call it the evolution of requirements. While the process of requirements elicitation is taking place requirements may change and evolve

The problems of the traditional requirements elicitation approaches are also include the narrow concept of stakeholders, the limited involvement of users with knowledge in requirements prioritization, and the bias of requirements engineers who focus on certain types of requirements, In addition to, the geographically dispersed stakeholders, and the uncertain technical and social environment (Snijders, et al., 2015) (Srivastava & Sharma, 2015) (Abdul Rahman & Sahibuddin, 2011). Tuunanen since 2003 believes that traditional requirements elicitation techniques and methods don't assist software engineers from approaching what he called Wide Audience End-Users (WAEU). WAEU are end-users for ISs which are results of new emerging technologies like java-embedded systems, and Web 2.0 (Tuunanen, 2003). In an attempt to study if the manual processing and analysis of large-scale text for users' reviews and feedback can work well to be efficient and scalable Groen et al. (Groen, et al., 2018) perform manual analysis of online user reviews of smart compact

cameras and determined the possibility to perform the process of deriving requirements from users' feedback manually. However, the required effort and fatigue occurring makes manual analysis does not scale well. Furthermore, Groen et al. determined that manually analyzing 2,000 user reviews per month is almost the upper limit when continuously monitoring the user feedback a product receives (e.g. identify problems and ideas for enhancements).

2.2.4 Requirements Evolution

Changing requirements is one of the greatest risks for large software development projects. Changing requirements usually take place where stakeholders keep changing their minds on what they want out of a project, and where their priorities lie. One of the greatest risks in software industry is the little attention paid to after installation requirements' changes which occur after a system is in operation. Changing requirements is a result of changing technologies, operational environments, and/or business needs. Requirements' changes are also referred as requirements evolution (Ernst, et al., 2014). Evolution of requirements refers to changes that take place in a set of requirements after RE phase; Changes in requirements are additions, omissions or modifications of requirements (Huuhka, 2003).

Requirements' evolution plays an important role in the lifetime of a product system in that they define possible changes to product feature requirements, which are one of the main issues that affect development activities (Zhao & Zhao, 2019). Requirements evolution is a main driver for software evolution. Traditionally, requirements evolution is associated to changes in the users' needs and environments (Ali, et al., 2011). Software evolution is important because organizations invest large amounts of money in their software, their systems are critical business assets and they have to invest in system change to maintain the value of these assets. Consequently, most large companies spend more on maintaining existing systems than on new systems development (Sommerville, 2015). During the evolution phase, the software is used successfully and there is a constant stream of proposed requirements changes.

One of the common challenges in software evolution includes short time-to-market and obtaining the set of requirements that drive the changes (Galvis Carreño & Winbladh, 2013). Users' feedback has been the driving force in software evolution (Galvis Carreño & Winbladh, 2013). Feedback systems used to allow users to provide feedback, which are used to measure the user satisfaction. The analysis of user satisfaction is helpful in improving products.

2.3 eLearning

eLearning is well known to be the use of technology in the delivery of education, where in some cases the learning resources are accessed online anywhere and anytime (Casey & Wilson, 2007), (Holmes & Gardner, 2006). eLearning has different tools, types, and information systems.

eLS have two main types of management systems; Learning Management Systems (LMS) and Learning Content Management Systems (LCMS). LMSs are concerned with the administrative process of learning, such as scheduling, testing, billing and registering learners, e.g. Moodle, and Blackboard (Horton & Horton, 2003). LCMSs combine the administrative processes of LMS with the authoring and content creation dimensions (Gheorghiu, 2017) (Irlbeck & Mowat, 2007). LCMS is used to author, approve, publish, and manage learning content, which is referred to as Learning Objects (LO) (Nichani, 2001). Hence eLS can be defined to be the systems that are concerned with the administering or content authoring tools to help students, instructors and management in the learning process. eLS can be used in an educational context or a corporate training context.

eLS are special software systems (Alharthi, et al., 2019). Studying their characteristics can help to better understand what is special about them, in order to best develop this type of systems especially after the Web evolution. Web 2.0 creates a new direction for Web applications and services. It is a change in Web attitude that shifts the focus of Web-based information from the creator or author of information to the user of that information (White, 2007). Web 2.0 is coined by O'Reilly (O'Reilly, 2005) resulting in an evolution in the

Web technologies e.g. wikis, social networking sites, and blogs. The evolution of the Web technologies led to a great increase in the number of people using some information systems. Among the information systems that were highly affected by the Web evolution are the eLS. The following section presents the characteristics of eLS.

2.3.1 eLS Characteristics

eLS are special software systems. Studying their characteristics can help us better understand what is special about them, in order to best develop this type of systems. This section presents the special properties of eLS, which are mainly as follows:

- **eLS Stakeholders**:
  eLearning is a type of software which is usually developed for a mass market with large number of customers instead of single customer, this means that there are unknown stakeholders with diverse backgrounds (Ambreen, 2019).
- **Collaboration issue**: Learning is a social process, it requires continuous collaboration between learners, instructors and courses contents (Allen, 2016). Figure 2-2 illustrates some forms of interactions between eLS' participants. Improving the social interaction in eLS lead to more satisfaction in the learning process (Abdul Rahman & Sahibuddin, 2010), which in turn leads to the success of the eLS under use. Collaboration can take many forms like open discussion, Storytelling, forums, project work, and brainstorming. (Abdul Rahman & Sahibuddin, 2010)
- **Diversity (background, culture, regulations and geographical):** We mentioned the large number of stakeholders for the eLS and the different forms and types of them, this leads to stakeholders' diversity in different aspects (Alharthi, et al., 2015), (Goldsworthy & Rankine, 2009) (Piccioli & Moriera, 2015) (Thi Tran & Anvari, 2016). Geographical diversity as learners and instructors may reside in different locations in the same country or different countries. The diversity in location leads to diversity in culture of people, their background, and the regulations that control each country.

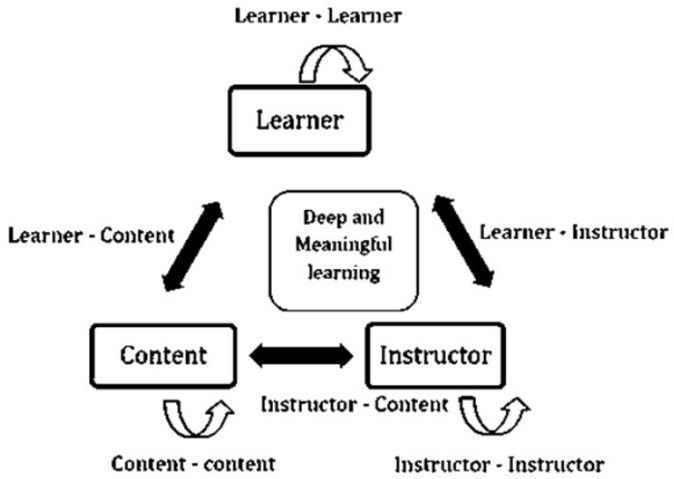


Figure 2-2. eLS participants' interactions (Garrison & Anderson, 2003)

2.3.2 eLS Requirements

Web 2.0 imposed new requirements for web based educational systems (Karataev & Zadorozhny, 2017). eLS have special nature; the thesis discussed earlier its common characteristics. Therefore, there must be special requirements during RE phase in software development process. These requirements are discussed next, first the functional requirements and then the non-functional requirements.

2.3.2.1 Functional Requirements

Functional requirements are product features or functions that developers must implement to enable users to accomplish their tasks. Some of the functional requirements of eLS are as follows

- **Social aspect requirements**: eLS as we mentioned before are social-based systems. It depends on the social interactions and participation between its participants. Improving social interaction in eLS can improve user satisfaction. Social interactions involve more collaborative activities. Social aspects needs in eLS are translated in RE process as social aspects requirements.
- **Learner-centered design**: since eLS' main users are the learners, so RE must focus on the learner's requirements. As we mentioned in the previous part, learning

depends on the social interactions. Accordingly, effective interaction is needed between user and the application to reduce the time needed. Thus, the eLearning developer must produce accurate learner-centered design that suits learners since they have various learning styles, behaviors and expectations towards the system.

- **Internationalization of requirements**: is a process of developing an adapted software application to various languages and regions without engineering changes. Currently, some educational institutions provide, or others wish to provide, cross-countries learning material and eLearning tools for multicultural students. To support their learning activities, these institutions frequently use eLS which should grant specific internationalization features.
- **Pedagogical requirements**: as investigated by Hammad et al. (Hammad & Khan, 2013) there are major factors contributing towards the evolution of a typical eLearning environment in the layered structure shown in Figure 2-3. The 'learners' layer is at the core of eLearning evolution and this may be attributed to elements like learners' needs, attitudes and preferences, context and behavioral needs. These elements require necessary research in learning theories (e.g. behaviorism) and design considerations when developing eLS. The 'curriculum/tutor' layer includes various issues such as: approaches adopted to develop eLearning materials, evaluation mechanisms and types (formative, summative), their contribution to the learning process (i.e. the ways in which their results alter learners' learning paths), tutors' participations and effectiveness. The 'institution' layer embraces different points such as: institutional academic policies, procedures and processes followed to achieve these policies, and roles defined for various stakeholders. The 'community' layer refers to the environment and entities different with which institutions interact. It provides stimulus of multi-culture, various learning backgrounds and evolving market demands that mostly contribute to program/course development. Finally, the 'technology' layer provides every possible technique to achieve goals set by/for learners, tutors, institutions, and communities. Usually, changes at this level impose severe implications on the inner

layers. For instance, new pedagogical approaches can be developed to assimilate new technological inventions or recently invented technologies make the achievement of some pedagogical issues easier. All these layers are well connected and dependent on each other. For example, a new requirement at any layer has a ripple effect on its encompassing layers and hence requires appropriate measures to meet the requirement.

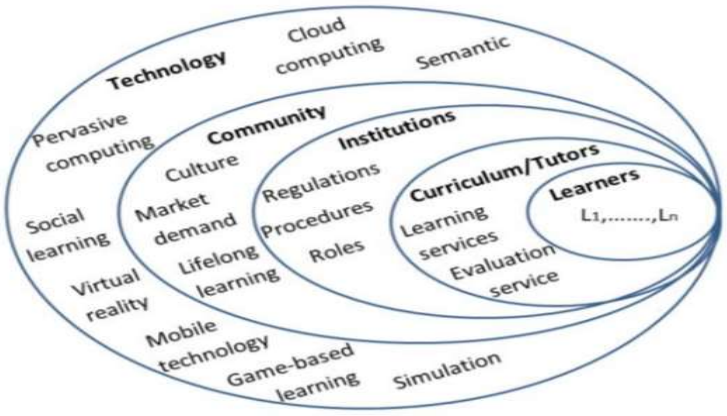


Figure 2-3. Factors affect eLS requirements (Hammad & Khan, 2013)

The evolution of the Web technologies led to a great increase in the number of people using eLS. Web 2.0 technologies unlike web 1.0 allow users to participate on their learning process in different ways e.g. comments, editing, chatting. This makes the eLearning process more enjoyable than before. Users are learners, instructors, management, admins, and parents.

2.3.2.2 Non-Functional Requirements (NFRs)

NFRs are considered as system attributes such as security, reliability, performance, maintainability, scalability, and usability. They serve as constraints or restrictions on the design of the system below are examples of the NFRs for the eLS:

- **Accessibility**: learning services include learning contents and tools such as progress monitoring widgets. These should be presented and accessible to learners using different devices/platforms (e.g. desktop and smart phones) within acceptable response times. In addition, learners should be able to search and

access related information (Batanero, et al., 2017) (i.e. pre-test, prerequisites if any, goals, duration, enrolment procedures/steps.).

- **Interoperability**: A learning artifact should be accessible from federations of eLS in a cloud environment to support learners' needs. Consequently, a desirable eLearning framework should provide seamless connectivity and interoperability with other systems or services for the purpose of data exchange and use of learning resources. Different eLearning standards such as SCORM, facilitates interoperability between different eLS. In addition, ontologies for semantic interoperability and OCCI standards for cloud interoperability can help to ensure inter and intra-cloud eLearning services provision and data exchange (Hammad, et al., 2013).
- **Personalization and customizations:** personalization can be seen as a mechanism to identify learners' preferences based on his/her skill set. This approach allows eLS to avoid a "one-size-fits-all" strategy and provides learners with customized interfaces, learning services, activities, and feedback based on their desire (Peng, et al., 2019).
- **Interactions**: this includes user-to-user, user-to- system, service to service interactions. It can be imagined that a dynamic and diverse eLS are capable of composing different learning services in a workflow to provide the required learning material to stakeholders based on a specific learning model. Stakeholders can interact with the eLS without knowing the details of background cloud service chaining based on specific service level agreements.
- **Resource Utilization:** eLearning is a continuous process involving learners, tutors, advisors, designers, administrators and other stakeholders. It generates special requirements in terms of its growing learning contents storage (i.e. multimedia educational resources), processing requirements to meet the demands of increasing number of participants (e.g. learners), Quality of Service (QoS) related requirements such as availability and accessibility of learning resources with minimal hardware requirements at the end user side so that the

learner can access these services from different devices such as smart phones, touch-pad, laptops, and others.

- **Communications and collaborations:** eLS should support synchronous and asynchronous communication amongst peers and their tutors via different tools such as content sharing, voice and video conferencing. Additionally, collaborative learning strategies need to be facilitated by providing access to tools such as wikis, blogs, sharing (files, desktop, and applications), tagging (Abdul Rahman & Sahibuddin, 2011).
- Administration: eLS should provide administrative privileges to concerned users to manage courses, services, user records, track users, enrolments.
- Security and Privacy: like any online system, eLS must be secure against internet-based security risks such as viruses and malware. In addition, eLS must have a comprehensive security and privacy policies implemented across the system in order to protect eLearning resources and user privacy with robust back up measures.
- Integrity with other services: eLearning services need to be integrated with other services or legacy systems of a particular institute (i.e. student records, HR systems, e-library, university portals, others) and hence require the necessary mechanisms to ensure data and resource integrity.

2.4 Crowdsourcing

Crowdsourcing term is coined by Jeff Howe the editor at Wired in June 2006 in his article "The Rise of Crowdsourcing" (Howe, Wired, 2006). It described how small businesses are getting successful from using the power of the crowd over the traditional professional ways to cut their costs. It also presented the use of it in research and development department to get new ideas for new products or features from the crowd. Howe then wrote an article named "Crowdsourcing: a definition" to make the term clearer to the audience and not to confuse it with other terms. Howe defined crowdsourcing as "I interpret crowdsourcing to be taking place any time a company makes a choice to employ the crowd to perform labor

that could alternatively be performed by an assigned group of employees or contractors, even if the company is just now putting up a shingle. In other words, crowdsourcing need not require an active shift from current employees (or again, contractors) to the crowd; it can start with the crowd" (Howe, 2006).

Crowdsourcing is a model that connects the power of a usually large and diverse number of people to share knowledge and solve problems. Crowdsourcing is motivated from the need of modern businesses for faster and cheaper solutions. Because of that, some crowdsourcing platforms have emerged and are really used for real-world software development e.g. Upwork, TopCoder, Elance, Odesk, Utest, Amazon Mechanical Turk (AMT), Ideal Scale (Mao, et al., 2015)

According to Hosseini (Hosseini M., Phalp, Taylor, & Ali, 2014) the four pillars of crowdsourcing are:

1- The **crowd**: who are the people who are involved in a crowdsourcing action.
2- The **crowdsourcer**: the entity (a person, organization) who look for the power of the crowd for doing a task.
3- The **crowdsourcing task**: the activity or action in which the crowd participates.
4- The **crowdsourcing platform**: the system (software based or non-software based) which a crowdsourcing task is accomplished within.

Crowdsourcing is used in requirements elicitation activity to give a new dimension and source of information to requirements engineers to accomplish this critical task within the software development process (Groen, et al., 2015). Crowdsourcing assists the finding, detecting and involvement of different stakeholders and users who can outline software requirements and the alternative ways for software to fulfill those requirements. Such activity increases and develops range of elicited requirements and, as a result, helps getting a whole idea of users' and other stakeholders' expectations from a software.

2.4.1 Crowd-Based Requirements Engineering

Crowd-based RE, which was coined as a term by Groen et al. (Groen, et al., 2015) is a highly interactive approach; it can get user requirements in less time, help in getting new ideas for software evolution, and has the potential to increase the quality of requirements elicitation. It is defined as "a semi-automated RE approach for obtaining and analyzing any kind of "user feedback" from a "crowd", with the goal of deriving validated user requirements (Groen, et al., 2015). This definition is adapted and enhanced in Groen and Koch (Groen & Koch, 2016) to be "The combined set of techniques for analyzing data from the crowd using text- and usage mining, motivational techniques for stimulating further generation of data, and crowdsourcing to validate requirements". We adopt this definition in our work.

Wang et al. performed a mapping study for 44 researches on the use of users' feedback in the crowdsourcing of RE. The study reveals that explicit users' feedback is the main focus in the current researches. Requirements elicitation and requirements analysis are the most RE activities under study in crowdsourcing feedback (Wang, et al., 2019). What we found from the investigation in the use of feedback analysis for eLS requirements' elicitation that this research point is still in need for more researches.

2.4.2 Traditional vs. Crowdsourcing - based Approaches in Requirements Elicitation

Hosseini et al. provided initial work on investigation of crowd and crowdsourcing features in requirements elicitation and proved this investigation using focus groups and experts' survey (Hosseini, et al., 2014). Table 2-1 provides a comparison between crowdsourcing-based and the traditional requirements elicitation approaches. The comparison criteria are based on crowd and crowdsourcing features found in (Hosseini, 2014), also supported by Groen et al. (Groen, et al., 2015), (Ambreen, 2019) (Breaux & Schaub, 2014) in addition to some publications mentioned the traditional requirements elicitation approaches e.g. (Zowghi & Coulin, 2005), (Fuentes-Fernández, et al., 2010), (Soledade , et al., 2013), (Souza & Silva, 2015). This is a broad view comparison not a specific one for each technique because we need it to highlight the importance of crowdsourcing.

Table 2-1. Traditional vs. crowdsourcing-based approaches in requirements elicitation

Comparison criteria	**Current requirements elicitation approaches**	**Crowdsourcing- based Approach**
Number of users Involved	Small	Large (Crowd)
Unknown Stakeholders	Can't work with unknown Stakeholder	There can be unknown Stakeholders
Cost of activity	High because it needs Experts	Low because uses online social tools
Stakeholders Diversity	Great effort needed to deal with stakeholders diversity	Can deal with stakeholders diversity
Need of Motivation	Low need of motivation	High need of motivation
Feedback Analysis	Partially use feedback Analysis	Can rely on feedback Analysis
Quality Considerations	High quality standards	Low quality standards

- **Number of users involved**: most of traditional requirements elicitation techniques deal with small number of stakeholders. However crowdsourcing approach is mainly based on the use of large number of people so in systems such as eLS with large number of users, it's better to use the crowdsourcing approach.
- **Unknown stakeholders**: it happens at the time of requirements elicitation in software development that a number of stakeholders maybe unknown for the developing team. Late stakeholders' discovery can lead to imprecise requirements elicitation process. Crowdsourcing-based approach using supportive tools can help in this issue of stakeholders' discovery. Some new eLS can have unknown users, thus requirement elicitation using crowdsourcing approach is a way that can help.

- **Cost of activity**: current requirements elicitation approaches are time-consuming and require professional staff, which lead to high cost requirements elicitation process. On the other hands the use of crowdsourcing requires fewer numbers of professionals so it's less costly than the traditional approaches.
- **Stakeholders' diversity:** in large systems there are diverse stakeholders. Diversity of stakeholders can take many forms; geographical diversity, cultural diversity, or background diversity, this may need more effort from the development team when using current requirements' elicitation approaches. It requires the use of different techniques, or better communication skills. Crowdsourcing-based approach can solve this issue because every stakeholder feels free to communicate with his own way, using the available social tools.
- **Need of motivation:** the use of crowdsourcing-based approach requires motivating the stakeholders because stakeholders are volunteering the involvement in requirements elicitation process. On the other hands current approaches require less motivation because the developing team interacts more lively with the stakeholders.
- **Feedback analysis:** new systems have users' feedback mechanism. The use of users' feedback is very important and considered as one of the powerful tools of crowdsourcing-based approach. Conversely current approaches don't use feedback analysis in its techniques.
- **Quality considerations:** quality standards are well established in current approaches, however in crowdsourcing-based approach quality standards are not followed because of the involvement of crowd in requirements elicitation process.

Crowdsourcing in RE seems promising and it has been used already in solutions to obtain information from users, however; certain challenges are existing. According to Hosseini et al. (Hosseini, et al., 2014) and Groen et al. 2017 (Groen, et al., 2017) the general challenges of crowdsourcing are:

- **Malicious participants:** Due to the data protection and intellectual properties rules in certain environments anonymity makes users more honest in explaining their

opinions. However, it would allow malicious users or users intending for incentives only to join in.

- **Analyzing feedback:** Feedback comes from online platforms with anonymous users; it's hard to identify user subgroups (for example, by age). Current techniques have difficulties identifying all the relevant data, automatically analyzing multimodal feedback and estimating the quality of the text-based analysis. Certain topics can cause important results to be overlooked. Also, the issue of what feedback to give and when to do that in a way that it does not affect participants' opinion for the next steps and, also, does not overload them with unnecessary information.
- **Task trivialisation**: Ad hoc introduction of digital motivation might be seen as undermining the task and might adversely affect feedback's usefulness and truthfulness.
- **Dishonesty to win rewards:** Ensuring that the participants' goal is not solely to get incentives is a challenge. Measuring what the right incentives should be and how competence, intrinsic motivation and anonymity play a role in that are all still research challenges to investigate.

2.5 Summary

In this chapter background information on requirements elicitation, evolution and eLearning as systems with its special characteristics and requirements are presented. Also, crowdsourcing term was presented. In addition to introducing the concept of the crowd-based RE.

3 Related Work

3.1 Introduction

In this chapter a literature survey is presented on the requirements elicitation for eLS, and literature survey conducted to present the different studies of crowdsourcing for RE and its use in requirements elicitation and evolution activities. For the completeness of coverage, we will study some of the publications in crowdsourcing for eLS.

3.2 Requirements Elicitation for eLS

In our attempt to make a survey to study the limitations of the traditional requirements elicitation approaches for eLS; we only found very little publications that address requirements elicitation for eLS in particular.

Abdul Rahman and Sahibuddin (Abdul Rahman & Sahibuddin, 2011) discussed the challenges of RE for eLS, one of these challenges was the lack of traditional requirements elicitation technique to get all of the stakeholders' requirements. They mentioned that the requirements engineers only focus on the technical requirements but not the social requirements of users. The publication suggested the need for an adequate requirements elicitation mechanism to detect and enhance users' social requirements to keep the users' sustainability of the eLS.

Tran and Anvari (Thi Tran & Anvari, 2016) highlighted the lack of the availability of a framework to address eliciting requirements of eLS' stakeholders, and confirmed the special nature of eLS because of the great number and diversity of stakeholders. The study implied the need to open new insights in the perspectives of requirements elicitation by the software engineers, as well as the need for collaboration and communication in the requirements elicitation process. It focused on the questionnaire technique for corporate eLS in the context of Accounting Information Systems (AIS); a five-dimensional framework is proposed to guide the design of questionnaires that will be used in the requirements elicitation activity for eLS.

AlKhuder and AlAli (AlKhuder & AlAli, 2017) presented the importance of requirements elicitation activity for the eLS development, mentioned some of eLS characteristics that challenge the requirements elicitation activity e.g. the on-going demand of requirements of learners, and the variability of stakeholders. They also proposed some eLS requirements for different aspects of the system as an outcome of requirements elicitation activity. On the other hands, the authors were not clearly revealing the source of eLS requirements presented in their study.

Ali and Lai (Ali & Lai, 2017) addressed the importance of communication and collaboration between stakeholders in the Global Software Development (GSD) context. We found that stakeholders in this context are comparable to stakeholders in eLS context, they are diverse in cultures, geographically dispersed, and there are times zones and language barriers, which made difficulties in engaging into an effective communication.

Alharthi et al. (Alharthi, et al., 2019) confirmed on the special nature and characteristics of eLS and conducted a Systematic Literature Review (SLR) to provide the current problems and state of art of the sustainability requirements for eLS. The review found that individuals in eLS context perform the most important role, which means that addressing individuals' feedback in eLS will lead to extracting many eLS requirements. Finding this publication supported our research point. Other publications show the challenges of the requirements elicitation for eLS and the need for tailored approaches or techniques without providing a complete solution (AlKhuder & AlAli, 2017) (Abdul Rahman & Sahibuddin, 2011) (Thi Tran & Anvari, 2016).

A summary is given in Table 3-1 that gives a list of publications with the contribution and limitation of each one.

Accordingly, there is a need of requirements elicitation approach to fulfill the collaborative needs and diverse context of stakeholders. The publication had presented a new method for requirements elicitation and analysis based on four stages involving some of the traditional

requirements elicitation techniques e.g. use case, scenarios. Finally, a preliminary evaluation was conducted through applying a case study on graduate students.

Table 3-1. List of eLS Publications with their contributions and limitations

Publication	Contribution	Limitation
(Abdul Rahman & Sahibuddin, 2011)	Highlighted the challenge of the lack of traditional requirements elicitation technique to get all of the stakeholders' requirements. Suggested the need for an adequate requirements elicitation mechanism to detect users' requirements.	No Solution is proposed
(Thi Tran & Anvari, 2016)	Proposed a five-dimensional framework to guide the design of questionnaires that will be used in the requirements elicitation activity for eLS.	The domain of research: focused on the questionnaire technique for corporate eLS in the context of Accounting Information Systems (AIS)
(AlKhuder & AlAli, 2017)	Mentioned eLS Requirements as an outcome of requirements elicitation activity	the authors were not clearly revealing the source of eLS requirements presented in their study
(Ali & Lai, 2017)	Addressed the importance of communication and collaboration between stakeholders in the (GSD) context. stakeholders in this context are comparable to stakeholders in eLS context	It Presented a new requirements elicitation method based on the traditional techniques.
(Alharthi, et al., 2019)	SLR provided the current problems and state of art of the sustainability requirements for eLS, it finds that individuals in eLS context perform the most important role,	No Solution for the addressed problems.

We can conclude that the surveyed publications confirm on the characteristics of the eLS and the need for new requirements elicitation approaches to overcome the limitations of the traditional ones in eLS context.

3.3 The Different Tools and Methods of Crowdsourcing in RE

According to Groen et al. (Groen, et al., 2015) crowdsourcing for RE is used with support of many tools to develop crowd-based solutions that can be categorized as illustrated next. We mentioned publications related to each category as a literature survey; however, the term crowdsourcing is not explicitly mentioned in some of publications because it was not yet coined, but its meaning is used, that's why we included these publications in our literature survey.

3.3.1 Social oriented collaboration tools

CrowdREquire is proposed by (Adepetu, et al., 2012). It's a platform that supports RE using the crowdsourcing concept. CrowdREquire specifies how RE can harness skills available in the crowd. CrowdREquire, involved the design of a crowdsourcing business model and market strategy for crowdsourcing RE. The CrowdREquire purpose is to allow the crowd submits requirement specifications as solutions to tasks submitted by clients. CrowdREquire is aimed at providing expertise through the crowd. The solution helps individuals and companies to find the best requirements specification for their proposed tasks and projects. Figure 3-1 illustrates the different participants or CrowdREquire and their functions they can perform on the platform. Although the publication presents the model, evaluation and output but the evaluation method is not clearly defined.

Srivastava and Sharma (Srivastava & Sharma, 2015) have proposed a crowdsourcing-based solution to a case study on MyERP software, a German company which faced competition from American startup ERP Company. MyERP wanted to reach for the requirements of non-German users with different geographies. Srivastava and Sharma proposed tasks to accomplish their crowd-based solution. The solution starts with identifying the crowd, who are the potential stakeholders for MyERP to collect their requirements; crowd can be domain experts or potential end users. The authors suggested LinkedIn to connect with the domain experts.

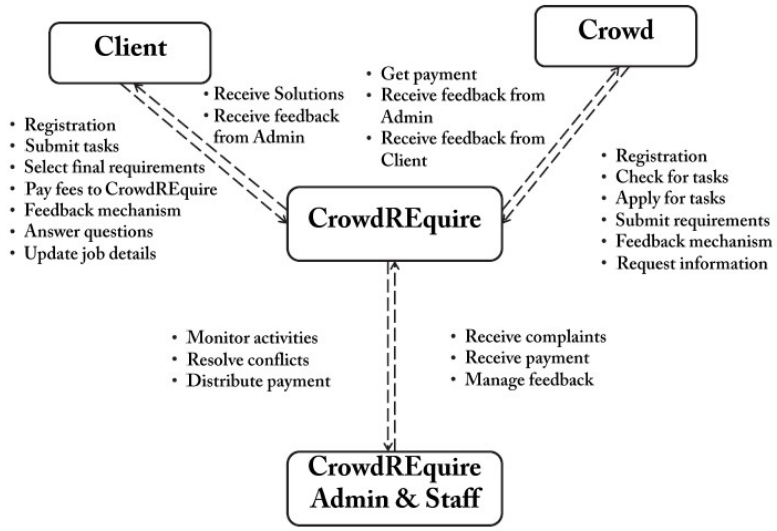


Figure 3-1. Crowd REquire participants and functions (Adepetu, Ahmed, & Al Ab, 2012)

Then, keeping the crowd involved, in this step, requirements should be gathered from the crowd with motivating them to achieve the task. Then, identifying the tasks to elicit ERP requirements, specific tasks are designed to elicit the users' requirements. Tasks are categorized according to the requirement nature; also, functional and non-functional requirements are included. Prioritizing and resolving requirements conflicts; different positions and levels of management in an organization reveal conflicts in the users' requirements, in these tasks conflicts should be resolved through prioritization ad sometimes negotiations to reach for agreement, also specific techniques can be used to speed up the task. Identifying duplicate requirements task come next, connection with crowd can take place to ensure that these duplicate requirements with the same meaning or not. Last step is to recognize cheaters to ensure the quality of the collected requirements. This task can be achieved through automated validations approaches. The publication only discusses the problem and how it can be solved but doesn't provide a complete, clear and detailed solution.

3.3.2 Web-based approaches with social network analysis or recommender systems

StakeSource is proposed by (Lim, et al., 2010), it is a web-based solution that automates stakeholder analysis. It crowdsources the stakeholders themselves for recommendations about other stakeholders and aggregates their answers using social network analysis. StakeSource helps experts from the burden of stakeholder analysis its first feature is identifying the stakeholders, this can be achieved by assigning specific stakeholders to each project then asks for recommending other stakeholders and specifying their roles in the project. Feature 2: StakeSource aggregates each stakeholder's validations on the other recommended stakeholders, then it draws a social network and links between stakeholders, then it calculates the weight of these links and performs social analysis measurements like betweenness centrality, load centrality, closeness centrality, page rank, degree centrality, in-degree centrality and out-degree centrality. Phase 3 is identifying potential problems, where stakeholders with problems in identifications to encourage them for more engagement in the website. Phase 4 is displaying stakeholders' information. StakeSource displays name, role, photo, who they recommended, stakeholder's position and rank through visualizations. StakeSoure is implemented in University College London (UCL) in RALIC project. Results show that it is a powerful tool it reduces the experts work in stakeholders' analysis, the tool is implemented and used in UCL Admissions System Project. The publication is well written and the evaluation method is clearly explained.

Lim et al. (Lim & Finkelstein, 2012) have proposed a novel method called StakeRare that uses social networks analysis to identify and prioritize requirements in large software projects. The method is based on building a social network of stakeholders and their recommendations of other stakeholders to reach a list of requirements using applied Social Network Analysis (SNA) measures. The system is evaluated by applying it on large size software project. The case study applied confirmed that StakeRare predicts stakeholder needs accurately and correctly prioritized. Lim et al. has evolved her research presented earlier and apply it on a large scale case study in a very well and detailed steps.

Renzel et al. (Renzel & Klamma, 2014) have proposed Requirements Bazaar, browser-based social software for Social RE. It brings together communities and service providers into a negotiation process. Requirements Bazar first aspect is requirements specifications, all information on a particular requirement along with its community participants. Second aspect is co-creation workflow, aims to continuously integrate communities into the entire service development process. The co-creation operations are: reporting new requirements, refining by adding artifacts or contributing to discussion, negotiating by voting or commenting, providing/testing a prototype/solution and acknowledging a solution. Third aspect is workspace integration; Requirements Bazaar provides different means to integrate with end-user and developer workspaces to lower entry barriers. Fourth step is personalizable requirements prioritization; it uses a modular extensible requirements ranking framework. Initial exploratory evaluation studies are performed to evaluate the tool. The evaluation is done on two stages one which is initial on a short term basis and the second on a long term basis; this can give more credibility to the proposed approach.

3.3.3 Mobile apps that are used as front end to allow portability for stakeholders and developers

OpenProposal is a toolbar plug-in through which users can annotate screenshots of desktop software (Rashid, et al., 2008). It aims to allow end users to express their requirements or ideas for an application, for the developers it sets annotations process to simplify the requirements elicitation process. the tool process has five phases starts with specify, and discuss ideas by the end user, then prioritize ideas and decide the one to implement by the requirements analyst and finally implement the selected requirement by the software engineer, Figure 3-2 illustrates OpenProposal requirements cycle. Rashid et al. evaluated the tool through conducting case studies, one of them applied on a university intend to launch new content management system and wants to test it.

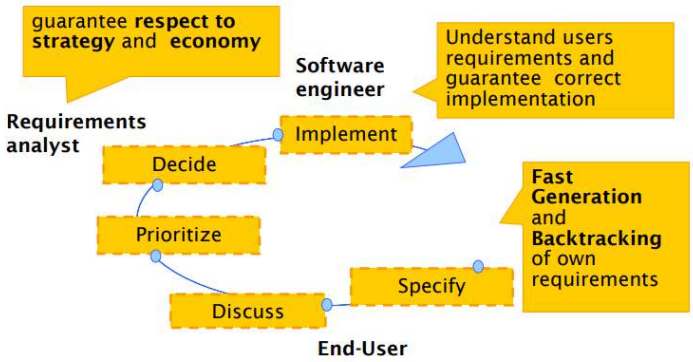


Figure 3-2. OpenProposal's requirements cycle (Rashid, et al., 2008)

3.3.4 Text mining tools

Focus on analyzing available data without actively involving stakeholders, thereby not supporting elicitation directly. Rather, they determine the relevance and importance of a sentence or statement through natural language algorithms, usually based on app store reviews (Breaux & Schaub, 2014) (Guzman & Maalej, 2014). Guzman et al. (Guzman & Maalej, 2014) studies App Stores and users submitted feedback for downloaded apps. They use natural language processing techniques to identify fine-grained app features in the reviews, through analyzing the user sentiments about the identified features and give them a general score across all reviews. Topic modeling techniques is used to group fine-grained features into more meaningful high-level features. The approach is evaluated with 7 apps from the Apple App Store and Google Play Store and compared its results with a manually, peer-conducted analysis of the reviews with *precision* up to 91% (59% average) and a *recall* up to 73% (51% average). The extracted features were coherent and relevant to requirements evolution tasks. The study is well written and very clear evaluation for the proposed approach is presented; however, the results is below average if we compared it to other publications. On the other hands we cannot accurately compare the results because we are not certain about the dataset used.

Breaux et al. (Breaux & Schaub, 2014) study the use of NLP algorithms in a crowdsourcing-based approach to extract users' requirements and compare it with the manual extraction of requirements to compare between the cost and time for both. The

study researchers rely on a small number of trained experts to perform a labor-intensive manual analysis of the text; they conducted three experiments to evaluate crowdsourcing a manual requirements extraction task to a larger number of untrained workers. In these experiments, they balance worker payment and overall cost, as well as worker training and data quality to study the feasibility of distributing requirements extraction to the crowd. The task consists of extracting descriptions of data collection, sharing and usage requirements from privacy policies. The study includes the task decomposition workflow and three metrics for measuring worker performance. The final evaluation shows a 60% reduction in the cost of manual extraction with a 16% increase in extraction coverage. The research limitations and threats to validity are not mentioned. Also, there is no clear mentioning about the privacy and quality criteria in the study.

Hosseini et al. have proposed CRAFT (Hosseini, et al., 2017), it is a technique that utilizes the crowd power to enrich text mining by allowing the crowd to categorize and annotate feedback through a context menu. This, in turn, helps in better identifying user requirements within forums feedback (Hosseini, et al., 2017). In CRAFT crowd members can annotate any piece of feedback they want at any given time in context, and a piece of feedback can be annotated several times by several crowd members. The outcome is a list of statements that may represent a requirement expressed in user feedback. It is evaluated by 12 randomly selected postgraduate computer science students who responded to an open call. The participants were asked to use the CRAFT technique to annotate eight feedback statements on a mobile application on Google Play. The study advocates that there is a huge potential of crowdsourcing for requirements elicitation and observed that there is not a significant amount of literature investigating it.

Buchan et al. (Buchan, et al., 2018) have investigated machine learning techniques to automatically identify text that represents users' ideas for new features from their online reviews. A binary classification approach to categorize extracted text as either a feature or non-feature was evaluated experimentally. Three machine learning algorithms were evaluated in the experiments: Naïve Bayes, Support vector machines and logistic

regression. Variations on the configurations of k-fold cross validation, the use of n-grams and review sentiment were also experimentally evaluated. The results have confirmed the feasibility and accuracy of semi-automated extraction of candidate requirements from a large volume of unstructured and noisy online user reviews. Results of accuracy (average precision, recall and F1 values typically between 87% and 91%) in a variety of experimental contexts.

Nayebi et al. (Nayebi, et al., 2017) investigated a method to suggest features that are useful for emergency apps called MAPFEAT. It combines various machine learning techniques to analyze tweets in conjunction with crowdsourcing and guides an extended search in app stores to find currently missing features in emergency apps based on the needs stated in social media. MAPFEAT is evaluated by a real-world case study of the Fort McMurray wildfire, where 69,680 unique tweets recorded over a certain period were analyzed. A range of features were extracted but without determining being essential feature or not. Also, a range of needs in tweets can be mapped to features. MAPFEAT looks beyond the current functionality of apps in the same domain and extracts features using variety of crowdsourced data.

Vliet et al. (Vliet, et al., 2020) have presented a novel method Kyōryoku for engaging a crowd to elicit user requirements from online user feedback. Kyōryoku is a crowdsourcing method for filtering out irrelevant app store reviews and for identifying features and qualities. The dataset contains user reviews from 2011–2015.A validation study has shown positive results in terms of feasibility, accuracy, and cost. The crowd workers achieved precision rates of 93% and 88% and recall rates of 84% and 81%, respectively in the outcome of two phases, however, Kyōryoku has not been tested against automated classifiers yet.

Ahmed et al. (Ahmad, et al., 2019) have investigated how the topic modeling algorithm LDA is used to identify NFRs in StackOverflow posts for iOS application development. Findings reveal that iOS developers focus mostly on usability, reliability, and functionality,

however they are found comparatively to be less focused on efficiency and portability, while maintainability is almost neglected.

3.3.5 Wiki-based tools

Wikis are a lightweight approach to produce documentation more powerful than plain office suites or collaborative tools, and easier to use and tailor than proprietary RE tools. Moreover, wikis are regarded as promising tools for requirements elicitation/negotiation in distributed environments. The adoption of a wiki in RE enables the various members of the project to contribute by adding, modifying, or deleting contents. In addition, a wiki platform natively supports the versioning of the handled documents. In this sense, contributors can always access to the history a requirement had, and they can trace its evolution (De Angelis, et al., 2016). G. De Angelis, et al. used the KJ method and wikis to analyze requirements in requirements elicitation process in a European research project. Table 3-2 provides a summary that gives the of name of the crowd-based solution, the supportive tool of Crowdsourcing used and the scope of research along with the limitations in RE previously discussed.

Table 3-2. List of the crowd-based solutions in RE

Reference	Solution Name	Supportive tool	Scope
(Adepetu, et al., 2012)	CrowdREquire	Crowdsourcing platform	CrowdREquire Platform in the requirements specification phase **Limitation:** Evaluation method is not mentioned clearly.
(Srivastava & Sharma, 2015)	_____	SNA	Used SNA in Requirements elicitation **Limitation:** Results are not justified by clear method of evaluation.

(Lim, et al., 2010)	StakeSource	SNA	StakeSource uses SNA for the stakeholders' analysis in UCL admission system.
(Lim & Finkelstein, 2012)	StakeRare	SNA	StakeRare uses SNA to recommend stakeholders for large –size software.
(Renzel & Klamma, 2014)	Requirements Bazaar	Social SW	Requirements Bazaar social software for requirements elicitation and prioritization. **Limitation:** Detailed evaluation is not mentioned
(Rashid, et al., 2008)	OpenProposal	User Involvement	OpenProposal a user involvement SW for requirements management
(Breaux & Schaub, 2014; Guzman & Maalej, 2014)	----------	NLP	Used NLP for requirements extraction on mobile app reviews. **Limitations:** Results comparison, privacy & quality factors.
(Hosseini, et al., 2017)	CRAFT	Expert Survey	CRAFT Expert survey for the requirements elicitation.
(Snijders, et al., 2015)	CrowdCentric RE	Gamification	CrowdCentricRE a gamified requirements elicitation SW.
(Buchan, et al., 2018)		Machine Learning	Performed an empirical study on users' reviews to automatically recognize text that represents users' needs, thoughts or requirements using machine learning techniques e.g. Naïve Bayes, N-gram, K-fold for new features. **Limitations:** The extracted text is classified as either a feature or non-feature. The study doesn't provide the users' requirements and the application domain is general.

(Nayebi, et al., 2017)		Machine Learning	Investigated a method called MAPFEAT that used crowdsourcing and machine learning techniques to analyze tweets and map it as a mobile application feature. **Limitations** The study scope is only for Twitter tweets, and mapping it to as only mobile apps features.
(Vliet, et al., 2020)	Kyōryoku	Crowd workers	Crowdsourcing method for filtering out irrelevant app store reviews and for identifying features and qualities using crowd workers. **Limitations** -Mobile apps dataset -Compare with other algorithms. - No. of reviews 1,000
(Ahmad, et al., 2019)		Machine Learning	LDA is used for analyze NFRs in StackOverflow for iOS software development

3.4 Crowdsourcing Platforms to Support RE

Sharma and Sureka (Sharma & Sureka, 2017) have proposed CRUISE; the platform for crowdsourcing requirements elicitation and evolution. There are separate modules for managing users and their roles. Similarly, there is a different module for project management. Any user logging in to CRUISE first gets to view his/her dashboard where the projects owned by the user, projects to which user is contributing, and other projects in CRUISE are listed. From dashboard, the user can browse to selected project, its modules and their respective requirements. Each requirements statement has a scope, priority, dependency and type associated to it. The contributing users can follow, score and comment on the requirements. The moderators only can commit a requirement, i.e. can

finalize a requirement to be promoted for design and development. Figure 3-3 shows the schematic diagram for CRUISE

Sharma and Saureka proceed to development, testing and validation of CRUISE. Developing the first version of CRUISE and conducting validation study, reveals that the effort spent in planning the tool as well as the preliminary study are of help to mitigate the associated challenges and risks. The validation study with CRUISE reveals that crowdsourcing could be successfully used for RE, however crowd formation needs special attention from the project owners and moderators. The role of moderators is very important in facilitating the discussions over requirements and finalizing the requirements to be developed. This responsibility cannot be left to the crowd alone.

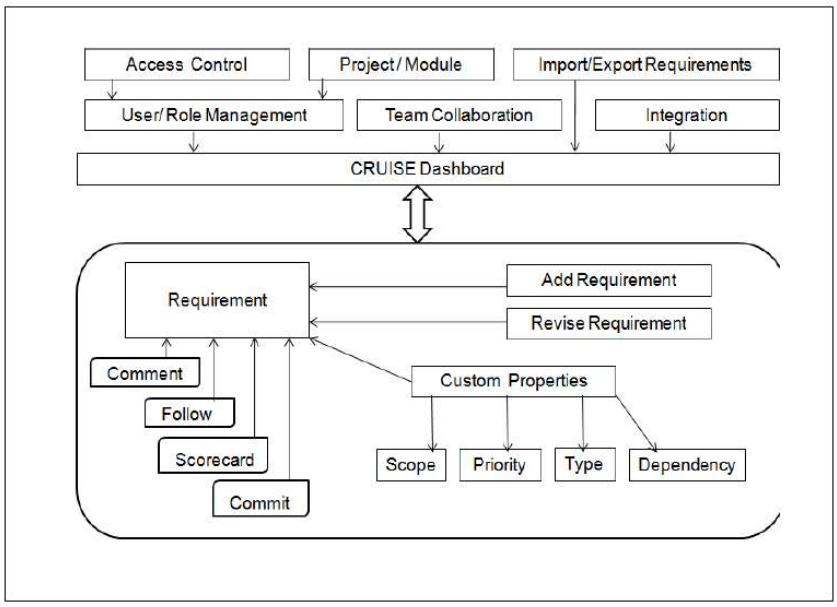


Figure 3-3. Schematic diagram of CRUISE (Sharma & Sureka, 2017)

Snijders et al. (Snijders, et al., 2015) have proposed REfine, a gamified online platform for requirements elicitation and refinement by involving a crowd of stakeholders: users, developers, analysts. REfine tool provides participation incentives via gamification; functional architecture of Refine is depicted in Figure 3-4. The tool aims to promote the long-term, sustainable collaboration among stakeholders, clarifying of the identified needs,

ease the software engineer's job of turning them into system requirements. A case study is conducted to evaluate the tool; it shows the potential of the approach for improving RE in software production. Refine limitations; it is difficult to attract a large crowd that is a good sample of the active users, software engineers need to be transparent and open to discussion, also they find it hard to have long-term incentives.

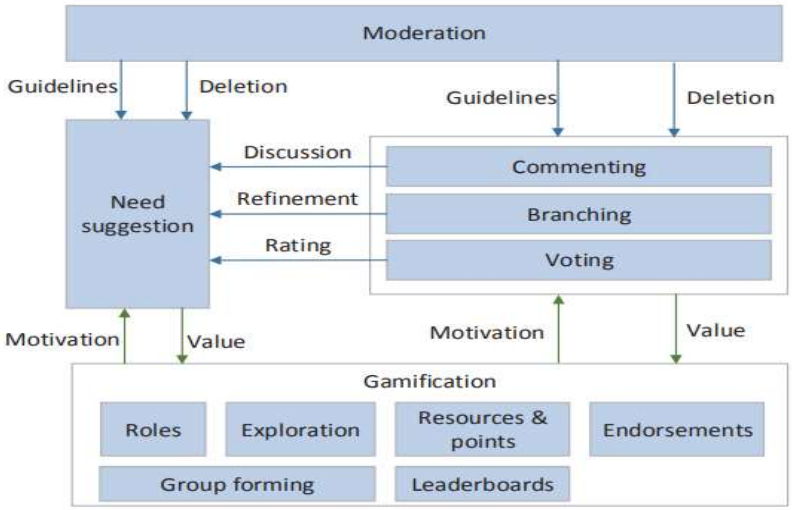


Figure 3-4. Functional architecture of Refine (Snijders, et al., 2015)

Snijders et al. (Snijders, et al., 2015) have presented CrowdCentric Requirements Engineering (CCRE) it is a method that guides software product companies in effectively applying crowdsourcing throughout RE processes. CCRE relies on crowdsourcing to support a broader user involvement, and on gamification to motivate that voluntary involvement. CCRE has 8 phases, depicted in Figure 3-5. It starts with determining the applicability of CCRE for the specific situation through the feasibility study phase. Then analyzing the context by defining scope and intended outcome, and the stakeholders that can be involved in crowdsourcing are identified. Then, crowd has to be formed, evaluated and prepared in crowd preparation phase. Then the phase in which the crowdsourcing is conducted which is the crowd involvement phase; Inviting stakeholders provide their input on the interactive platform and feedback on the other channels is collected. Then, in the requirement identification phase the needs that were suggested, discussed and voted upon

will be analyzed as requirements. Then, focus groups are organized to develop the requirements in focus group preparation phase. Then, after preparation, the actual focus group can take place, where this is the time that it should be turned into a requirement definition. Finally, the development sprint; this phase is generic and has to be instantiated in a way that is favorable to the company., this phase shows the result of the crowdsourced input. A case study-based evaluation was conducted on Qubus 7. A beta version of CCRE was released to a select group of customers and users, who were subsequently invited to be involved in the improvement of the software. All the phases of CCRE are instantiated on Qubus. Results we evaluated in four ways, Observation; a questionnaire for participants; an interview with the product management of Qubus; expert interviews in the product software industry.

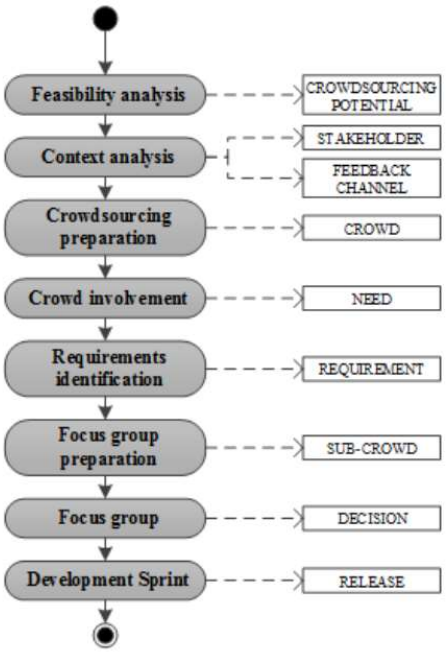


Figure 3-5. CCRE phases (Snijders, et al., 2015).

3.5 The Use of Crowdsourcing in eLearning

The use of crowdsourcing in eLS is concerned with providing the service with a large number of users e.g. Coursera that have classes with thousands of students. Following some publications in the use of crowdsourcing in eLS are presented.

Barbosa et al. 2013 (Barbosa, et al., 2013) have studied crowdsourcing tools for eLearning, It focuses on Massive Online Open Coursees (MOOCs). Barabosa built framework and set eleven dimensions to classify the types of crowdsourcing tools for eLearning. The publication discusses and classifies 22 crowdsourcing tools found on the Internet e.g. Coursera, Udacity and MIT OpenCourseW. The tools are varying from online universities to marketplaces for online courses. The publication discusses crowdsourcing in eLearning as the use of collaboration is the key aspect, as in a class of thousands of students, it is virtually impossible for a teacher to give attention to every single student. The crowd must help itself to enable this approach; moreover, it offers learning in a natural way. The job of the teacher becomes more explanatory, and less evaluative. (D. S. Weld, 2012). Table 3-3 provides a list of the publications mentioned earlier about the use of crowdsourcing in eLearning publications.

Punjabi et.al. (Punjabi, et al., 2013) have presented CrowdSMILE is a system that addresses anytime anywhere learning content access that is organized and presented using a location-based context. The system also addresses content creation and publishing while providing a familiar Facebook interface for the social aspect of learning. Its cloud-based architecture and standard web-based inter-component communications allow it be a very scalable system and extendable system. Users of the system found the system to be useful and showed positive attitude towards the system. Given the features it provides, CrowdSMILE can be considered to be a system that supports Lifelong Learning. The experiments conducted showed that users accepted the system and actually liked using it as they found it an easy way to learn.

Tarasowa et al. (Tarasowa, et al., 2015) have presented Crowd-Learn which is the use of learning objects e.g. Shareable Content Object Reference Model (SCORM) for structured eLearning systems to support the system and to manage the learning objects (files, presentations), it uses from Slide Wiki Application. The system was evaluated by case study applied on an information system lecture at Chemnitz Technical University. The Wiki slides were structured within the lecture and added questions for student self-

assessment before the final exam. Tarasowa found that eLearning material when combined with crowd-sourcing and collaborative social approaches can help to cultivate innovation by collecting and expressing different individual's ideas.

Paulin & Haythornthwaite (Paulina & Haythornthwaiteb, 2016) have addressed how the evolution of the Web changes how, where and with whom people learn, and the opportunities and challenges this rises for the future of educational practice. The publication focuses on taking advantage of crowdsourcing to create and manage large-scale learning enterprises. MOOCs are the principal point for large-scale online learning. The power of the crowd is being leveraged to address many of the scale-related issues that arise in MOOCs. Elements suggested to be addressed by crowdsourcing are Content, discussion, evaluation, behavior, practices, learning analytics, and assessment and feedback.

Karataev and Zadorozhny (Karataev & Zadorozhny, 2017) have studied the crowdsourcing of learning content to anyone. They introduce SALT; a novel framework for social learning that allows any person to author educational content as mini-lessons, learn lessons by use adaptive learning pathways, and interact with their peers. The system is evaluated through a number of classroom studies. The results show that adaptive social learning can be utilized by collective learning experiences also they found that students with very high similarity tend to arrange groups.

Suhonjic et al. (Suhonjić, et al., 2019) have proposed a crowdsourcing model that combines the collaborative learning and crowdsourcing mechanisms to implement it on learner-centered approach. The study aims to enhance the participation and collaboration of learners as learning creators. The study is evaluated by case study applied on 74 students on Blgrade University; it shows an enhancement in user participation, and good quality of learning contents.

We can conclude that crowdsourcing gives some opportunities to eLearning and some threats.

3.5.1 Possible Crowdsourcing Opportunities to eLearning

3.5.1.1 Learning and Collaboration

Participatory sites such as question-answering platforms provide opportunities for learning and collaboration.

3.5.1.2 Harnessing Collective Intelligence

Participatory sites are often considered as powerful venues of crowdsourcing. Users often turn to Q&A sites to obtain opinions and perspectives about particular tasks.

3.5.1.3 Rewarding Knowledge Sharing

Users who provide answers to questions on Q&A sites often can be upvoted or liked by fellow users, and often are thus rewarded in ways that encourage them toward further participation. Their contributions to Q&A sites are considered a part of knowledge sharing.

3.5.1.4 Defining a Scholarly Identity on Social Spaces

Recently, some scholars have begun to disseminate their research and other works with fellow academics via social networking sites such as ResearchGate or Academia. This type of activity adds new potential dimensions to traditional forms of scholarly communication.

3.5.2 Possible Threats for Crowdsourced eLearning

There are some possible threats for crowdsourced eLearning as we will discuss next.

3.5.2.1 Content Quality

Online participatory sites essentially provide users with a platform to create and consume content. content is a major concern for educators and researchers. This uncertainty for users significantly compromises the potential benefits of crowdsourced learning.

3.5.2.2 Intellectual Property

Online participatory sites present numerous questions and issues related to intellectual property and ownership of content, ranging from brief answers and explanations. Some Internet users who are not familiar with copyright violation rules often get away with breaking the rules.

3.5.2.4 Privacy

Posting content to online participatory sites, our identity is often revealed to other members of that site, which leads to privacy concerns.

Table 3-3. List of the use of crowdsourcing in eLearning publications

Model Name	Context	Purpose	Publication
CrowdSmile	LifeLong learning	Crowdsourcing organizing learning contents to SNS users	(Punjabi, et al., 2013)
CrowdLearn	LCMS	Crowdsourcing the creation of LOs	(Tarasowa, et al., 2015)
_____	MOOCs	Crowdsourcing the curriculum	(Paulina & Haythornthwaiteb, 2016)
SALT Framework	Adaptive social learning	Crowdsourcing of learning content to anyone	(Karataev & Zadorozhny, 2017)
Collaborative learning and crowdsourcing	Learner centered approach	Enhance learners participation as learning creators	(Suhonjić, et al., 2019)

3.6 Literature Findings

We can conclude that the related work presented chapter 3 ensures that there is a gap in the requirements elicitation activity for the eLS; therefore, there is a need for a new or enhanced approach to fill in this gap. Moreover, there is a need for newer and up to date activities that take benefits from the existing concepts of sharing in Web 2.0 and social networking to enhance the elicitation phase, and to find better ways for software evolution. The literature survey gives a motivation and recommendation to use crowdsourcing in RE as an emerging approach. Moreover, from recognizing the features of traditional and

crowdsourcing-based approaches in requirements elicitation, we can conclude that crowdsourcing-based approach greatly matches eLS special characteristics, so we believe that the use of a new approach based on crowdsourcing can help in the requirements elicitation process for eLS. Therefore, we present Crowdsourcing based requirements elicitation for eLS (CREeLS), the following chapter presents detailed overview of our proposed method CREeLS.

3.7 Summary

In this chapter a range of surveyed related work publications were discussed to spot the gap in literature and determine the importance of the research. The related work areas are the requirements elicitation for eLS, the use of crowdsourcing as approach in RE in general and requirements elicitation in particular. Also, and for the completeness of coverage of the literature we studied some of the publications in crowdsourcing for eLS. The findings from surveying the literature helped us understand the current research achievements, the areas of concentration, and how requirements related activities can be enhanced.

4 The Proposed Crowd-based Requirements Elicitation Method for eLearning Systems (CREeLS)

4.1 Introduction

This chapter presents the proposed method (CREeLS); the Crowdsourcing based requirements elicitation for eLearning Systems. CREeLS is made up of a framework, and phases on how to apply the proposed method on eLS. The framework presents the broad lines and the basic concepts of the use of crowdsourcing as a requirement elicitation helping hand for eLS. Also, the implementation of CREeLS will be presented.

4.2 CREeLS Framework

After reviewing the literature, we wanted to set broadlines and general tools that can help in eliciting requirements for eLS based on crowdsourcing. We came up with recommended sources of crowd that can be useful within the eLS context, and some of the supported tools that can utilize the crowd interactions or users' opinions needed in the requirements elicitation activity. Based on these crowd sources and tools we could be able to propose CREeLS framework. The recommended crowd sources are interactivity, the use of the power of social networking and social collaboration, text mining tools and the ability to extract information from users' text, and the use of the users' received feedback. The framework gives broadlines and general not specific selections for achieving crowd-based requirements elicitation for eLS and affecting its several functions e.g. assignments, quizzes, posting materials. The use of each suggested tool in crowdsourcing for requirements elicitation has been separately evaluated in the literature. The proposed framework is illustrated in Figure 4-1.

4.2.1 eLearning Systems

The proposed framework is operating in the context of eLS which can be LMS or LCMS. Within this eLS context, there are lots of interactions, social collaboration, text by users and feedback options and functions which are considered as parts of the eLearning process.

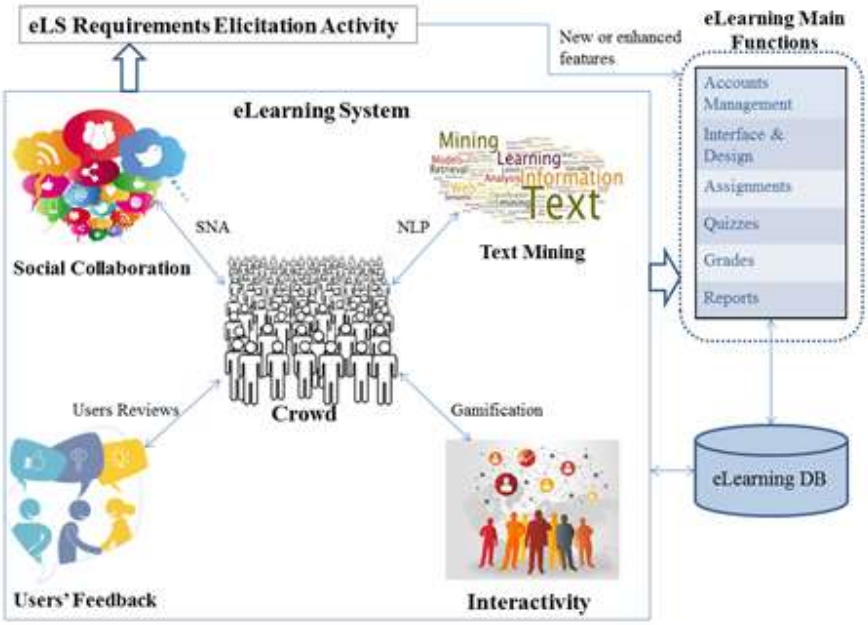


Figure 4-1. CREeLS framework

These different options and functions can be used in the crowdsourcing method of requirements elicitation for eLS as we will explain later in the next sub-sections. They also considered as inputs for the eLS requirements elicitation activity, the results from the analysis of those elements can reveal a number of eLS features which can be new features that users wants to add to their favorite eLS of enhancement for the existing features they already use.

4.2.2 The Crowd

The crowd in crowdsourcing context is the group of people who engage and participate in the crowdsourcing activity. According to Hosseini et al. (Hosseini, 2014) crowd is characterized by: diversity, suitability, anonymity, largeness, and undefined-ness. Crowd in our framework are the eLS users who can be; learners, instructors, administrators, management, or parents. The framework can be applied in the context of crowdsourcing platforms that has been explained earlier. Crowdsourcing platform connects requesters with online workers ((Zanatta, et al., 2016).

4.2.3 Users' Feedback

The target of the proposed framework is to minimize the gap between the development team and the eLS stakeholders. Feedback is one source of getting what in users' mind without intentionally get in interaction with them. There must be different feedback methods from the interaction between eLearning participants in the different forms of social collaborations that support eLS. Feedbacks can be on the eLS itself, course and its material, or the instructor and management.

Users' comments or reviews are methods of feedback that can be found in the eLS in different modules. Users' comments and reviews can contain useful information for developers; they include good, bad, or recommended features. So the analysis of these reviews is important for the RE activities (Wang, et al., 2019) (Rizk, et al., 2015). Pagano et al. (Pagano & Maalej, 2013) have performed an empirical study on users' feedback in mobile stores. One part of the study is to investigate the feedback content. They found that there are feedbacks that suggest new features and they are strongly justified. We suggest the use of feedback analysis methods to gain the benefits from users' comments and reviews while developing eLS.

4.2.4 Interactivity

Interactivity between the eLearning participants (Instructor, learner, course, and management) is a must. Interactivity leads to better course results for learners (Palazuelos, et al., 2013), to know the participants' opinions in the different modules of the eLS, e.g. course material, quizzes, assignments, scheduling, and eLearning process, and participation between the participants. Improving social interaction in eLS can improve user satisfaction. Social interactions involve more collaborative activities.

Crowdsourcing in eLS not only increases the amount of educational content but also improves its quality (Tarasowa, et al., 2015). As Paulo Freire wrote in his 1968 book Pedagogy of the Oppressed, "education must begin with the solution of the teacher-student

contradiction, by reconciling the poles of the contradiction so that both are simultaneously teachers and students".

The collaboration of a person's contribution with a larger, shared, cooperative work is a type of interaction, thus the act of crowdsourcing can be considered naturally interactive. Jinnifer and Brigid (Proctor & Maher, 2013) identified three possible categories of interactive, crowdsourced works:

Category 1: Linear/single-channel works created from multiple user contributions.

Category 2: Interactive works created from unique individual contributions.

Category 3: Interactive works created from multiple user contributions.

In the proposed method we are concerned with category 3 as many stakeholders are involved in the requirements elicitation process. Under this category we suggest the use of gamification in the interactivity element of the framework. Gamification seeks of integrating the game process and techniques in a non-gaming process to be more attractive. Gamification seeks out for improvement of the user's performance, commitment, and motivation (Pedreira , et al., 2015). The use of gamification technique is promising in the RE field (Snijders, et al., 2015). The use of incentives, collected points, and badges are different techniques of gamification.

4.2.5 Text Mining Tools

Text mining is the method of analyzing unstructured text using data mining techniques. In the framework we use text mining in order to analyze eLearning participants' written interactions, which exist in the different modules in the eLearning system. It can be found in social networking applications attached to the eLS, discussion forums, comments spaces below blogs posts, or any other different posts. Using text mining techniques will extract the hidden requirements. One way of text mining analysis is Natural Language Processing (NLP).

4.2.6 Social Collaboration

Social Network Sites (SNS) are example of social software, they are used for communicating and connecting with others – anytime and anywhere. SNS allow the creation of social groups, where many people with similar interests are connected together and communicate in different forms. Studying the patterns of social collaborations in SNS is a technique of getting the users requirements and knowing their behavior. SNA is a technique of understanding the relationships among the nodes of interactions and studies the patterns and effects of the relationships.

4.3 CREeLS Phases

A proposed new Crowdsourcing requirements elicitation method for eLS (CREeLS) was given because of the following factors; eLS characteristics, the crowd-based RE definitions mentioned earlier, the limitations and recommendations findings from the literature review, in addition to, the proposed framework discussed in the previous section. There is a need for an automated method that combines both the social and technical aspects of software engineering in general and requirements elicitation in particular (Baxter & Sommerville, 2011).

CREeLS method has five phases, (1) Creating a channel for users to post their feedback, or show their interactions. (2) Extracting users' interactions or feedback. (3) Analyze users' interactions or feedback. (4) Evolve fine software requirements. (5) Categorize and consolidate the requirements. eLS stakeholders are CREeLS's crowd.

CREeLS method has five phases depicted in Figure 4-2; whereas the details of the roles and responsibilities performed by the eLS stakeholders can be found in Table 4-1: (1) Creating a channel for users to post their feedback, or show their interactions; (2) Extracting users' interactions or feedback; (3) Analyze users' interactions or feedback (it can be analyzed through text mining tools, social network analysis, or the use of gamification); (4) Evolve fine software requirements, and (5) Consolidate and categorize the requirements.

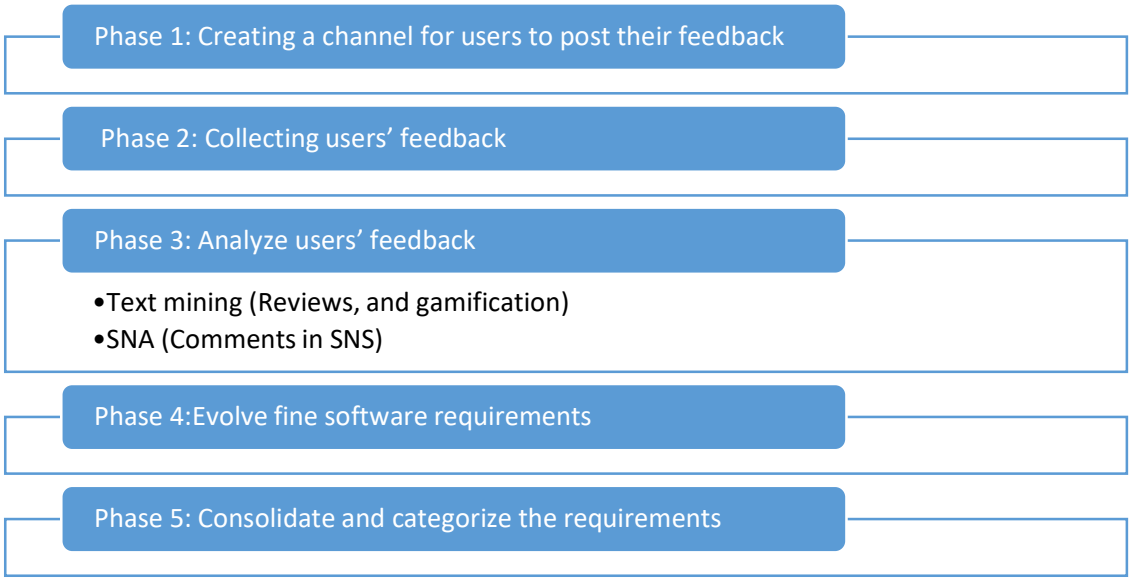


Figure 4-2. CREeLS phases

CREeLS include all of the eLS users allowing all of them to participate in the evolution of the eLS through posting feedback. CREeLS analyze the users' feedback within a timeframe to enable the evolution of the eLS. In the context of eLS malicious participation is at minimum because most of the eLS users are identified participants to the institution which operates the eLS.

Table 4-1. eLS's stakeholders and their roles in CREeLS phases

Stakeholders	**Role**
Students, Instructors and admin staff	Post feedback
LMS Admin and Data Analysts	Phase 2, 3 and 4
Requirements Engineer	Phase 1, 4 and 5
Database Administrators	Phase 2

Phase 1: Creating a channel for users to post their feedback. The phases start by creating a facility for eLS users to post their comments or feedback. The requirements engineer decides which channel to be created or it can be more than one channel to be created. The suggested channels are:

1- Enabling eLS users to post their feedback through posting reviews on their use of eLS.
2- Enabling the use of social networking sites (SNS), or the eLS itself has its own social network application.
3- Creating a gamified way to encourage users to post their experience of using the eLS.

The channel number one is considered as a direct channel where users know and intend to post their feedback. However, channel two and three are considered indirect way; as in the use of SNS users can post their feedback through a thread of discussion among different members or administrators. Channel three can be considered as both direct and indirect way; it can be direct because users may know the game intention of collecting reviews, on the other hand for users who like to be involved in games, they intend to do it for the fun more than intentionally posting their reviews.

Phase 2: Collect Users' feedback. Users' feedback should be collected in phase two: users' feedback are stored in the eLS databases so the database administrators are involved in this phase to collect the stored feedbacks and send it to the analysts. The requirements engineer decides the data needed and ask the database administrator to prepare it and then send it to the analyst to analyze it.

Phase 3: Analyze users' interactions or feedback. Analysis of users' feedback can be accomplished using different ways. The suggested analysis tools are: text mining analysis or SNA; we will give next a brief explanation on the different measurements in SNA for the completeness of coverage in the thesis; however the text mining analysis will be under experimentation and will be described later in this chapter.

SNA is concerned with the systems' stakeholders, and how they are visualized as a social network. It enables analysts and stakeholders to study a stakeholder's position in the social network, their details, priority in the project, the stakeholders who recommended them, the stakeholders they recommend, and comments from other stakeholders (Lim & Finkelstein, 2012).

SNA often relies on well-defined measures to provide an important overview of network characteristics. Power: is a fundamental property of networks; generally, actors with more connections enjoy greater power in a relationship network and therefore see a greater proportion of the information flowing through the network. SNA attempts to measure power through the composite measure of centrality, which comprises variables such as degree, closeness, and betweenness. Centrality degree: is to some extent a power measure, because it shows the proportion of nodes that are adjacent to each node. The higher a node's centrality degree, the greater its access to information resources or peers in the network, i.e. the greater its power and popularity. Closeness: is a centrality measure of how quickly one actor can access another. It is defined as the sum of geodesic distances from one node to all others. Closeness varies inversely with centrality: small closeness values indicate greater proximity to other nodes, whereas larger values indicate greater distances from other nodes. Betweenness indicates how actors mediate the communication among themselves. Actors that are positioned between powerful actors can enjoy more privileges in a network. Density: which indicates the number of relationships actually observed in a network divided by the total number of possible relationships. Density is a quantitative way to capture important sociological characteristics such as cohesion, solidarity and membership. Block modeling: uses blocks to represent the relationships among nodes, thereby reducing the complexity of the network representation and simplifying the analysis.

Phase 4: Extract fine software requirements. An enhancement function for NLP algorithms are applied to extract fine software requirements. enhancement function intend to find the

best values that achieve the highest coherence value and best collection of keywords and topics that represent the eLS requirements.

Phase 5: Consolidate and categorize the requirements. This phase is accomplished manually by the requirement engineer. Fine software requirements output from phase four are considered as input for this phase, requirements engineer classifies and categorizes software requirements then merge similar requirements together to have the final form of the eLS users' requirements. A brief description of CREeLS phases is presented in Table 4-2 illustrating each phase and its input, process and output.

Table 4-2. CREeLS phases brief description

Phase #	Input	Process	Output
1	LMS	Add interactivity channel to LMS	LMS with interactivity channel
2	LMS interactivity channel and LMS users	Encourage LMS users to use the Interactivity channels	Users' feedback and interactions
3	Users' feedback and interactions	Feedback Analysis	Features' keywords
4	Extracted features keywords	Enhanced feedback analysis	Enhanced Features keywords
5	Enhanced Features keywords	Turn features keywords to requirements statements	eLS users' Requirements

4.4 CREeLS Feature Extraction steps

We proposed six steps for eLS feature extraction that should be accomplished in phase 3 and 4 in CREeLS method. The steps are illustrated in Figure 4-3. Below is a complete explanation for each step. It's important to mention that it is an evolutionary approach in which the results are at a certain point in the requirements elicitation activity, because there

should be continuous stream of information from an LMS. CREeLS feature extraction steps are:

1- **Classification**: First, if the corpus is already classified into features which users like, features that users dislike, we would select reviews about features which users dislike, because this will imply the users' needs and their requirements for the LMS. Otherwise, if the reviews are not classified, we should apply sentiment analysis to classify the reviews into negative, positive and neutral reviews. Second, Select the role of the respondents, e.g. administrator, student, or instructor.

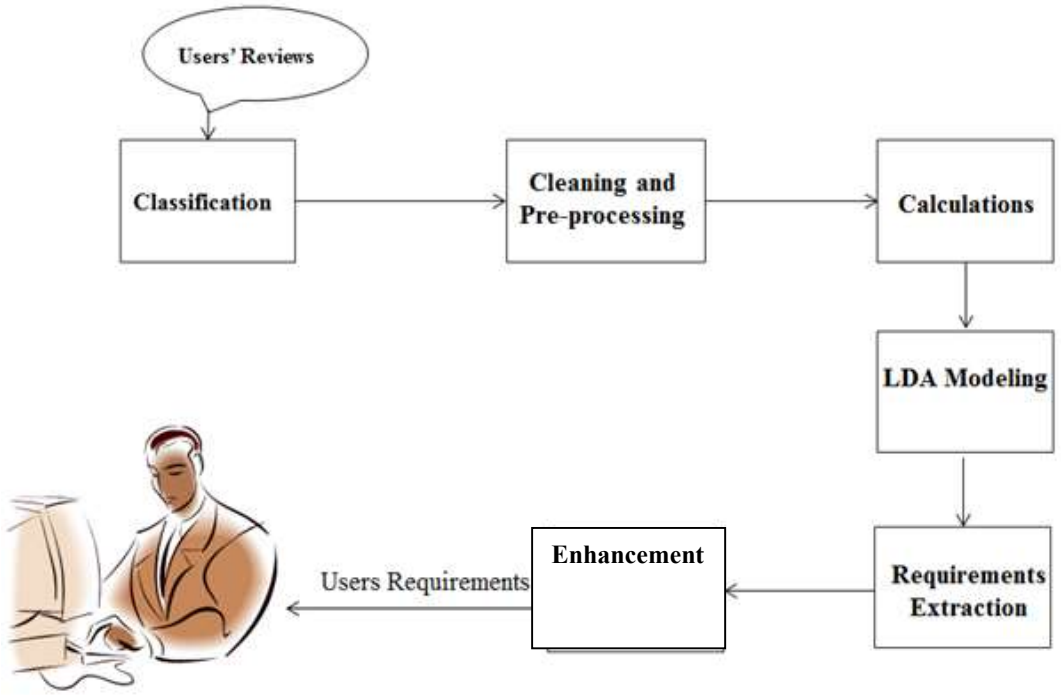


Figure 4-3. Requirements extraction steps

2- **Cleaning and pre-processing**: To perform the feature extraction, raw data should be prepared so that it can be easily analyzed and also to get better results. Preparation of data should be performed in terms of: 1- cleaning or removing the special characters from the text e.g. commas, dashes, or semi colons, because these special characters were not needed in the extraction process. 2- Tokenization which is splitting the text into words, so that we can treat each word separately. The processes of cleaning and tokenizing text were performed together in one step using one

function. 3- Stopwords removal: in computing, stop words are common words that has little value in the text (Manning, et al., 2019) e.g. and, is, are. We used the standard list of stopwords provided by Gensim library because it has the greater number of stopwords (Singh, 2020) and we added words that we found common in users' reviews, but weren't describing features e.g. "like", "dislike", "there", "easy".

3- **Part of Speech Tagging (POS):** it is tagging each word in the text as its corresponding grammatical part of speech (SketchEngine Team, 2019). We used the POS of wordnet in Natural Language ToolKit (NLTK) Package, in this step we defined and extracted the nouns, verbs, adjectives, and adverbs in the text. Because, it was important to understand what a text is about. 5- Stemming: which refers to a process that removes the ends of words to reduce it to its base or root form. 6- Lemmatization: aims to remove inflectional endings (are letters at the end of a word that change its meaning) and to return the word to its base or dictionary form.

4- **Words Calculations**: A dictionary for each word used in the text is created. The output from this function was the minimum number each word has appeared in the text, and the maximum ratio of each word appeared in the text. The dictionary is based on TFIDF weight measurement. According to (Manning, et al., 2008) TF-IDF stands for Term Frequency-Inverse Document Frequency, the TF-IDF weight is a statistical measure used to evaluate how important a word is to a document in a collection or corpus. The importance increased proportionally to the number of times a word appeared in the document but was offset by the frequency of the word in the corpus. Variations of the TF-IDF weighting scheme are often used by search engines as a central tool in scoring and ranking a document's relevance given a user query (TFIDF). Typically, the tf-idf weight is composed by two terms. The first term computes the normalized Term Frequency (TF); the number of times a word appears in a document, divided by the total number of words in that document. The second term is the Inverse Document Frequency (IDF), computed as the logarithm of the number of the documents in the corpus divided by the number of documents where

the specific term appears. (Stecanella, 2019). To put it in more formal mathematical terms, to produce a composite weight for each term in each document. The TF-IDF score for the word *t* in the document *d* from the document set *D* is calculated as follows and given by equations (1), (2) and (3):

$$tf\ idf\ (t, d, D) = tf\ (t, d)\ .\ idf\ (t, D) \quad (1)$$

Where

$$tf\ (t, d) = log\ (1 + freq\ (t, d)) \quad (2)$$

$$idf\ (t, D) = log\ \left(\frac{N}{count\ (d \in D : t \in d)} \right) \quad (3)$$

5- **Latent Drichlet Allocation (LDA) modeling:** LDA is one of the most common topic modeling algorithms; it is a generative probabilistic algorithm for data collection. The basic idea is that the documents are represented as random mixtures over latent topics, where a topic is characterized by a distribution over words; in our case it is LMS users' reviews. LDA is chosen in CREeLs because it is a popular method for fitting a topic model. It treats each document as a mixture of topics, and each topic as a mixture of words. This allows documents to "overlap" each other in terms of content, rather than being separated into discrete groups, in a way that mirrors typical use of natural language (Silge & Robinson, 2017). The other option is Latent Semantic Analysis (LSA), LSA focuses on reducing dimension of classification while LDA solves topic modeling problems (Ma, 2018); the latter is what we focus on. LDA represents topics by word probabilities. The words with highest probabilities in each topic usually give a good idea of what the topic is can word probabilities from LDA. In the context of topic modeling, each topic is considered as a group of topics (Blei, et al., 2003). This means that each user review can have more than one feature (topic) associated; also, each requirement (topic) can have more than one keyword associated to it. In this step we used LDA algorithm to

extract the top keywords in the text based on the calculations done in the step 3, which was calculating the weight of each word in each review and in the whole text. The result of applying LDA was the top feature keywords in the whole text. More detailed explanation for the algorithm will take place in Appendix B.

6- **Requirements extraction**: Based on the top keywords in the text, we applied function that categorized these keywords into number of topics; these topics represented the users' requirements, and composed of number of the top keywords and its percentage of relevance to this requirement.

7- **Enhancement:** an enhancement process is applied in the algorithm, it specifies the optimum number of topics (group of keywords) in the dataset. This function calculates the coherence value of the top keywords in text that represent the requirements and the distance between them to reduce the overlapping between the topics. Topic coherence score is a measure of how good a topic model is in generating coherent topics. A coherent topic should be semantically interpretable and not an artifact of statistical inference. A higher coherence score indicates a better topic model (Sharma, 2020). In order to calculate this optimum coherence value, it goes into many iterations where the user specifies the minimum and maximum values of and the function perform iterations between these values (number of topics, minimum number each word appeared in the text, and the maximum ratio of this word appeared in the text). In each iteration the function calculates these values and at the end it shows the optimum value for keywords in terms of the minimum number each word appeared in the text, and the maximum ratio of this word appeared in the text. These values can be used to show the list of keywords in each topic.

In order to accomplish the previous steps, we recommend to develop a tool using python programming language that takes the eLS users' feedback or interactions as input and perform the previously mentioned steps for features extraction and then the output of this

tool should be keywords that represent the eLS features which needs to be enhanced or created.

4.5 CREeLS Implementation

After discussing CREeLS feature extraction steps in the previous section and in order to put the proposed method into action and use it, an implementation of this part of CREeLS method is conducted. The implementation is developed using python programming language version 3.7 using Jupyter Notebook; the open-source web-based application for writing python documents. The application was developed because there were no free, open–source application released for this type of programs. Figure 4-4 illustrated the pseudocode for CREeLS implementation presenting its different functions that were explained in the previous section.

```
Reading corpus (excel file in English language)
      Define file path
      Check for the LMS category
      Check for the dislike column
Preprocessing steps
      Tokenization 'nltk'
      Part of Speech Tagging (POS) 'Wordnet nltk'
      Lemmatization and stemming 'nltk'
      Removing stop words 'nltk'
Feature Extraction
      Create a dictionary for the corpus based on tf-idf (weight not
count)
Applying LDA to the result to group the keywords into topics
Enhancement function for best results
      Get the minimum count
      Get the max ratio
      Determine the number of topics
Predicting from each topic the best matching topic
```

Figure 4-4: Pseudocode for CREeLS implementation

is assigned a topic as illustrated in Figure 4-5. To view the output in a more clear and understandable view a function is created to view the visualize the output in a graphical view as illustrated in Figure 4-6.

	Document_No	Dominant_Topic	Topic_Perc_Contrib	Keywords	Text	Review
0	0	1.0	0.6991999745368958	assignment, time, post, class, scroll_post, gr...	['great', 'grade', 'form', 'quiz']	There isn't too much I don't like about Google...
1	1	1.0	0.7315000295639038	assignment, time, post, class, scroll_post, gr...	['incapatibilty', 'program']	I dislike the incapatibilty with other programs.
2	2	1.0	0.7793999910354614	assignment, time, post, class, scroll_post, gr...	['grade', 'book', 'sink', 'current', 'tool']	The grade book doesn't sink with the current t...
3	3	0.0	0.7508000135421753	grade, teacher, time, assignment, forum_discus...	['learn', 'curve', 'class', 'somewhat', 'uncle...	I dislike that there is learning curve setting...
4	4	1.0	0.855400025844574	assignment, time, post, class, scroll_post, gr...	['concern', 'difficulty', 'begin', 'pilot', 'y...	All of the concerns I had have been fixed. The...
5	5	1.0	0.8253999948501587	assignment, time, post, class, scroll_post, gr...	['share', 'folder', 'step', 'case', 'bother', ...	When u go to share a folder or pp I have to go...
6	6	0.0	0.8719000220298767	grade, teacher, time, assignment, forum_discus...	['lack', 'ability', 'forum', 'natively', 'orde...	I really lack the ability to tie in a forum na...
7	7	0.0	0.9120000004768372	grade, teacher, time, assignment, forum_discus...	['huge', 'fact', 'individual', 'instructor', '...	I'm not a huge fan of the fact that I can't ma...
8	8	0.0	0.7975000143051147	grade, teacher, time, assignment, forum_discus...	['teacher', 'complain', 'transfer', 'state', '...	Teachers complain it doesn‰Ûªt transfer to sta...
9	9	1.0	0.802299976348877	assignment, time, post, class, scroll_post, gr...	['fact', 'wallpaper', 'choose', 'individual', ...	I dislike the fact that there are not enough w...

Figure 4-5: Sample of keywords and topics of each user review.

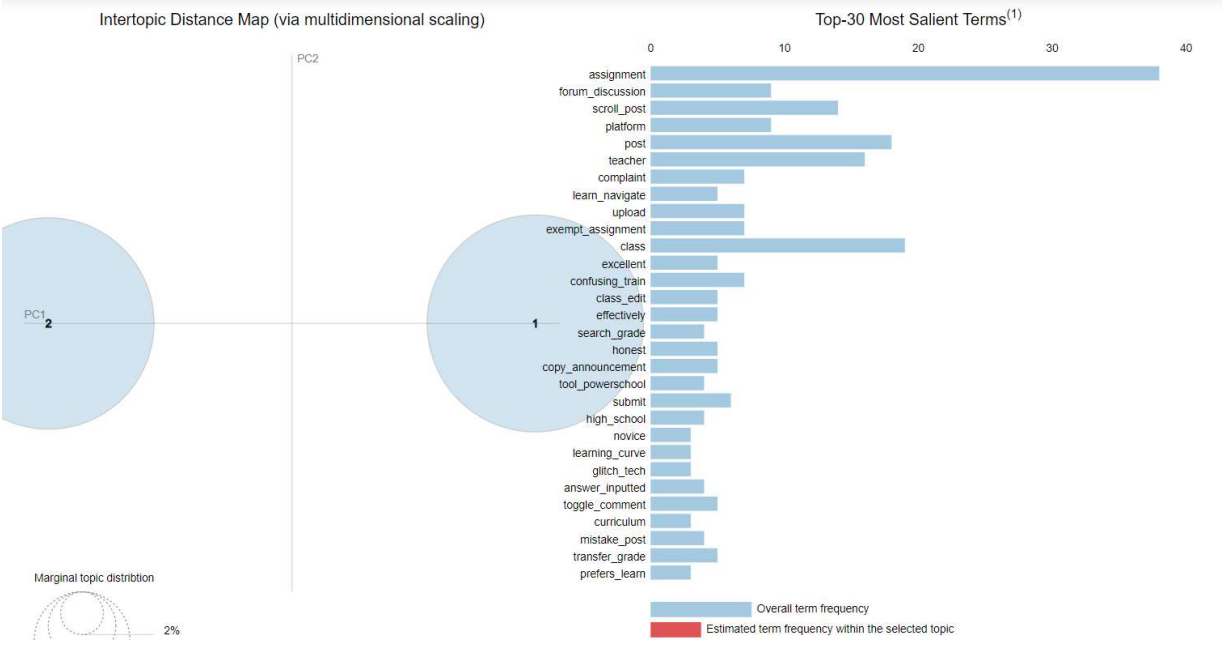


Figure 4-6: Graphical view of the extracted topics and their keywords.

In an attempt to enhance the the results and output of the program, an enhancement function is developed. It contains two parts, the first part starts by entering the range of values for variables of (the number of topics, minimum count, and maximum ratiothe result from this part is just the count of iterations as illustrated in Figure 4-7 to make sure that the function is running. After the first part is completed the second part will begin, it just responsible

for selecting the best values out of the range of the possible values entered in the first part. The best values when entered to the TF-IDF and LDA function it produces the best number of topics and keyords for each topic with the best value for coherence. The output of this part should be as illustrated in Figure 4-8.

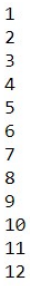


Figure 4-8: The output of the first part of the enhancement function.

min_count	12.000000
max_ratio	0.100000
n_topics	5.000000
Coh_Values	0.578034

Figure 4-7: The output of the second part of the enhancement function.

4.6 Summary

This chapter presented the proposed method CREeLS; the Crowdsourcing based Requirements Elicitation for eLS. It is made up of a framework, and phases. The framework presents the broadlines and the basic concepts of the use of crowdsourcing as a requirement elicitation helping hand for eLS. CREeLS phases are composed of five phases, which will be applied in the requirements elicitation activity to extract eLS users' requirements. Implementation of CREeLS was also presented.

5 Experimental Studies and Evaluation

5.1 Introduction

This Chapter discusses in details the experimental study conducted to evaluate the proposed method CREeLS. A tool was developed based on the topic modeling algorithms to extract the eLS requirements from eLS users' reviews.

5.2 CREeLS Experimental Studies

According Wohlin et al. (Wohlin, et al., 2012), an experiment gives more control over the situation; it allows to compare the results when one variable is changing and the others are fixed. We wanted to analyze eLS users' feedback and evaluate whether this feedback was truly representing eLS requirements, then we will check whether the extracted requirements were similar and coherent as the manually extracted users' requirements. The experimental study reflects phase 3 in CREeLS, because we wanted to test whether CREeLS will succeed to extract keywords that reflect LMS users' requirements in less time than the manual extraction of requirements. The fixed variables are the topic modeling algorithm, the tool used to extract the requirement, the nature of the eLS users' feedback, and the eLS category (LMS). The only changing variable is the LMS product so we can compare the results of different LMS products. Phases 1, 2, 4 and 5 will be skipped because of the following. Phase 1 requires software development into the LMS to attach one of the suggested facilities tools, and this is out of the current research scope as we are focusing on LMS requirements extraction from users' reviews. Phase 2 is responsible for collecting and extracting the required reviews from the LMS database; we substituted this phase with the dataset we acquired from G2Crowd Company. Phases 4 and 5 will be left for future work when we enhance CREeLS. For this research there are two experimental studies were conducted to evaluate CREeLS approach and then a comparison between both of them is performed.

5.2.1 Experimental Study # 1

The experimental study started by data collection and went through the requirements extractions steps as detailed earlier in chapter 4. The only data we could be able to collect is LMS users' reviews from G2Crowd Company; the email in which the company is sending the dataset is presented in Appendix A. The dataset is on an Excel sheet. The reviews are for more than 20 products for both LMS for education purpose, e.g. universities and schools, and LMS for corporate purpose. The reviews are classified into reviews on features that users like, and reviews on features that users dislike. Table 5-1 depicts a sample of the row data under study. We used the reviews as it is with no pre cleaning or preparing for the analysis except for minimizing the number of columns, and leave only Date, "What do you like and what do you dislike?" We used three products users' reviews from the category of LMS for educational purpose; the three LMS products are 'Blackboard', 'Canvas', and 'Google Classroom'. Table 5-2 illustrates the three LMS products under the current study and the corresponding number of eLS users' reviews for each one. The reviews are between 2012 and 2018 with total number of reviews 11886 review for the three LMSs under study.

Table 5-1. Dataset sample

	A	B	C	D	E	F	G	H	I
1	survey_response	submitted at	star rating	product	Category	review title	what do you like be	what do you dislike?	business problems solved
2	473	14-Sep-2012	3.5	Saba Cloud	Corporate LMS	Best of a mediocre bunc	Good learning path de	Clunky admin and some	
3	20837	26-Mar-2014	3	SilkRoad Technology	Corporate LMS	Bumpy integration succ	I like how the Greenlig	Administrative procedur	Consolidating all Learning & D
4	21563	3-Apr-2014	1.5	SumTotal Learning Ma	Corporate LMS	Non-validated SumTotal	SumTotal is easy to us	The course versioning is	Would like to use one LMS syst
5	21653	8-Apr-2014	4	Saba TalentSpace	Corporate LMS	Halogen Software Revie	I love that you have th	The software does requ	We used it at my past compar
6	21655	8-Apr-2014	4.5	Saba TalentSpace	Corporate LMS	I have used Halogen to	I like how intuitive Hal	My one dislike is it is nc	Centralizing our job descriptio
7	21658	8-Apr-2014	1.5	SumTotal Learning Ma	Corporate LMS	SumTotal LMS	The system is reliable.	Lack of of technical sup	was able to track employee tr
8	21701	9-Apr-2014	1.5	SumTotal Learning Ma	Corporate LMS	Mixed Experience with S	The integrated platfor	EXECUTION! My experi	Streamlining processes and pr
9	21917	18-Apr-2014	4.5	Brightspace	Corporate LMS	Desire2Learn	I have used Desire2Le	Having used Desire2Lea	The business problem being s
10	21984	22-Apr-2014	3.5	Lumesse ETWeb	Corporate LMS	Lumesse TalentLink	Quarterly user forum n	Customer support functi	I use Lumesse's system for re
11	22732	2-May-2014	3.5	Saba Cloud	Corporate LMS	Saba is a quality produc	Saba has top-of-the-in	There are two things tha	We are managing the learning
12	23339	17-May-2014	0.5	Cornerstone Learning	Corporate LMS	If You are Looking Just f	I have been a sys adm	I have been a sys admin	Training and assessment of cc
13	23802	6-Jun-2014	5	SAP SuccessFactors	Corporate LMS	I love it because it's use	I like getting feedback	I haven't found anything	It helps you set goals and sho
14	24185	25-Jun-2014	2.5	SumTotal Learning Ma	Corporate LMS	Good LMS for end users	Good product for an er	Not a great product for t	We needed a system to track
15	24412	1-Jul-2014	4.5	Saba Cloud	Corporate LMS	Robust and highly flexib	SEC is a highly robust :	Content launch and tem	We've solved matrix requirem
16	24542	3-Jul-2014	4.5	SilkRoad Technology	Corporate LMS	Training solution	We were able to strea	The role out of each pro	Working mainly in Greenlight,
17	24568	3-Jul-2014	4	SAP SuccessFactors	Corporate LMS	Right Sized Learning	When installing Succe	Our experience with Suc	The ability to link training to r
18	24795	14-Jul-2014	4.5	Saba TalentSpace	Corporate LMS	As the HR Manager, I le	I love that it centralize	It is not as intuitive as I	There are almost too many to
19	26161	9-Sep-2014	4.5	Lessonly	Corporate LMS	One of the easiest SaaS	Lessonly just works. M	I wish the billing wasnt	With a distributed workforce a
20	34553	27-Jan-2015	4	Lumesse Learning Ga	Corporate LMS	Great online authoring t	Not having to install a	Although online is good,	The need to move face to face
21	34569	27-Jan-2015	4	SAP SuccessFactors	Corporate LMS	SuccessFactors is a succ	The User Interface is i	Downtime, as with in Cl	I have worked on EC, Goals, Pe
22	34749	29-Jan-2015	4	Oracle Talent Manage	Corporate LMS	Heavy Taleo Learn Use	Taleo Learn offers us t	While Taleo Learn offers	Increase communication of HF
23	8720	12-Dec-2012	1	SAP SuccessFactors	Corporate LMS	SucksFactors	OK tool for performanc	Our HR systems person	
24	21763	11-Apr-2014	3.5	Saba Cloud	Corporate LMS	I'm implementing it as v	Localization...lots of la	Implementation of new	Our first LMS...we need to trac

Table 5-2. Overview of LMS no. of reviews under study

Educational LMS	No. of Analyzed Reviews
Blackboard	5004
Canvas	3953
GoogleClassroom	2929
Total	11886

We followed all of six steps for eLS feature extraction. It's important to mention that it is an evolutionary approach in which the results are at a certain point in the requirements elicitation activity, because there should be continuous stream of information from an LMS. CREeLS requirements extraction steps are:

1- Classification: The dataset was classified according to the LMS type; features which users like, features that users dislike, and the role of the respondent, e.g. administrator, student, or instructor. we selected first LMS products for educational purpose, because they have a larger number of reviews than corporate LMS. Second, we selected reviews about features which users dislike, because this will imply the users' needs and their requirements for the LMS. Third, role of users, this criterion we couldn't consider as most of the users were students and only few reviews were for administrators or instructor or management. Because text analysis needs a large amount of text for better results, we decided to include all the reviews with no classification according to users' roles. Table 5-3 shows organized sample of the dataset.

Table 5-3. Organized dataset sample

Product	Category	What do you like best?	What do you dislike
Schoology	Learning M	and parents. Everyone will find a large number of pluses for themselves. Students like quality interesting lessons. You can continue chatting after classes and ask questions at any time. Students are captivated by the work and communication in Schoology, as it is like communicating in the social networks that are familiar to them. Teachers also like this service for high-quality work, modern and easy interface, a large number of interesting and important tools for work and for an ability to work with different	Indeed, there are functions that need to be improved, for example, the mobile application is slightly behind the desktop version. Also chatting sometimes hangs, but despite all this, I like to use. Schoology has all I need to work.
Showbie	Learning M	-The online version and the app are basically identical, so students don't have any trouble switching back and forth. This makes it easy for students to complete smaller tasks at home, even without a tablet.-Evaluate student work right in the app (onlin	I can't use the pen/markup tool in the online version. That's it...everything else is amazing.
		Easy to use for younger students in 1st grade and above; excellent way to share assignments, documents, videos, links, and pictures with students in the classroom; Super easy for students to share their completed assignments and get quick feedback from the teacher; the free version	There is nothing I dislike about Showbie. It's a great app and they are always making improvements and getting teacher

2- Cleaning and Pre-processing: To perform the feature extraction, raw data should be prepared so that it can be easily analyzed and also to get better results. Preparation of data should be performed in terms of: 1- cleaning or removing the special characters from the text e.g. commas, dashes, or semi colons, because these special characters were not needed in the extraction process. 2- Tokenization: The processes of cleaning and tokenizing text were performed together in one step using one function. 3- Stopwords removal: We used the standard list of stopwords provided by Gensim library because it has the greater number of stopwords (Singh, 2020) and we added words that

3- Words calculations: We created a dictionary for each word used in the text. The output from this function was the minimum number each word has appeared in the text, and the maximum ratio of each word appeared in the text. We used the Gensim dictionary function in this step. TFIDF weight measurement was used for extracting features from the users' reviews.

4- Latent Drichlet Allocation (LDA) modeling: LDA represents topics by word probabilities. The words with highest probabilities in each topic usually give a good idea of what the topic is can word probabilities from LDA. In the context of topic modeling, each topic is considered as a group of topics (Blei, et al., 2003). This means that each user review can have more than one requirement (topic) associated; also, each requirement (topic) can have more than one keyword associated to it. In this step we used LDA algorithm to extract the top keywords in the text based on the calculations done in the step 3, which was calculating the weight of each word in each review and in the whole text. The result of applying LDA was the top feature keywords in the whole text.

5- Requirements extraction: Based on the top keywords in the text, we applied function that categorized these keywords into number of topics; these topics represented the users' requirements, and composed of number of the top keywords and its percentage of relevance to this requirement.

6- Enhancement: an enhancement function is used to specify the optimum number of requirements in the text. This function calculates the coherence value of the top keywords in text the minimum number each word appeared in the text, and the maximum ratio of this word appeared in the text. Next Table 5-4 shows the optimum values for the three LMS products under study.

Table 5-4. Enhancement function results

LMS	Min Count	Max Ratio	No. of Topics	Coherence Value
Blackboard	14	0.45	7	0.47
Canvas	21.5	0.3	6	0.50
GoogleClassroom	12	0.1	5	0.58

The results from the enhancement are entered as an input for another function, we call it visualize the topics, to produce the output in a visualized format. The topics distributions results for the three LMSs according to the function of visualizing the topics are illustrated in the next figures 5-1, 5-2, and 5-3.

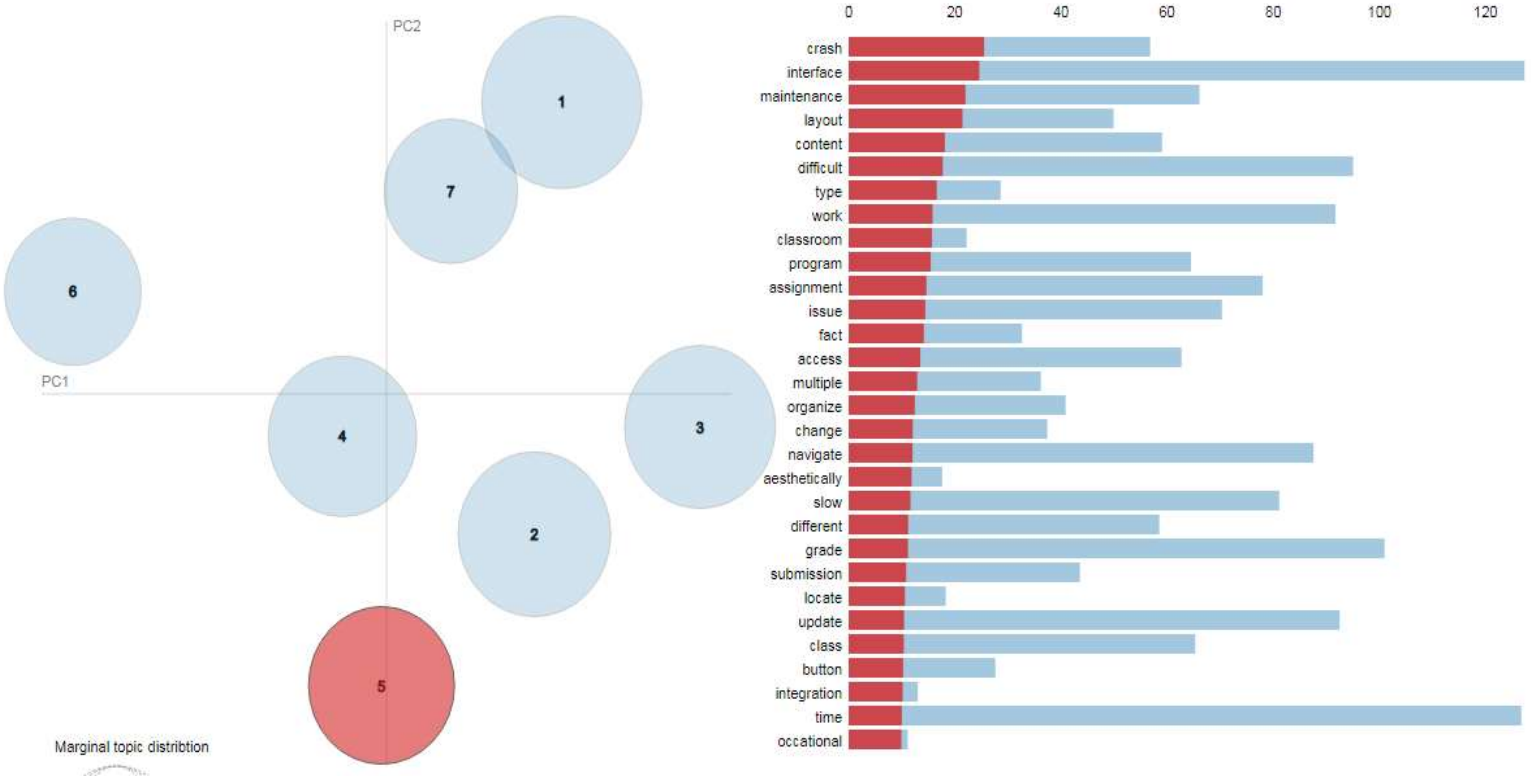


Figure 5-1. Visualizing Blackboard topics results

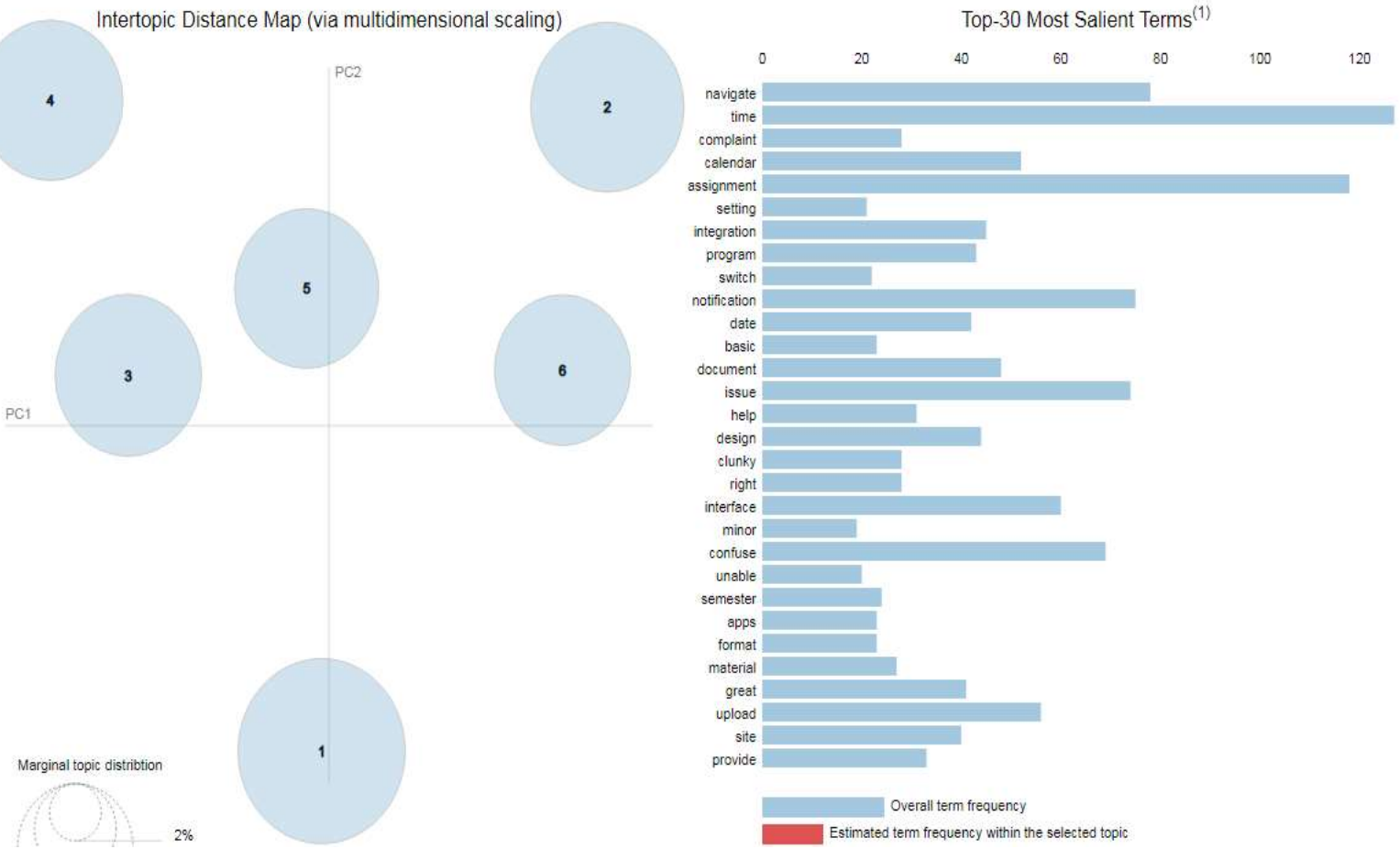


Figure 5-2. Visualizing Canvas topics results

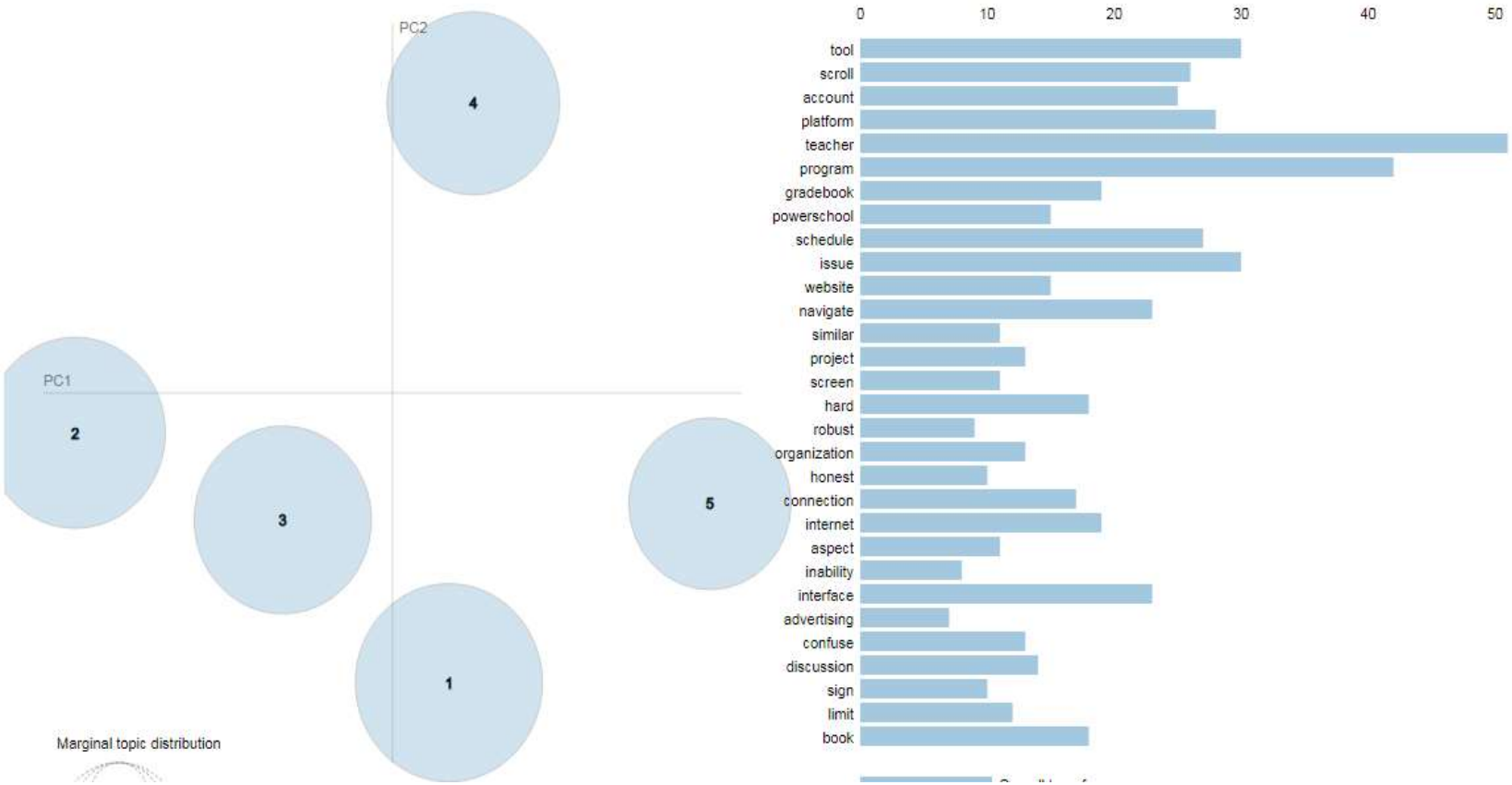


Figure 5-3. Visualizing Google Classroom topics results

We noticed that the topics circles were mostly not overlapped which means that the topics keywords were not overlapped, each topic represented its own keywords and topic. Circles with numbers represented the topics, the size of the circle indicated the dominance of this topic within the text, the keywords of the topic appeared in the text, and the highest frequency keyword appeared at the first. Tables 5-5, 5-6 and 5-7 illustrate the dominant keywords in each topic (first 10 keywords) for Blackboard, Canvas and Google Classroom respectively. We made some of these keywords bold to emphasize them, as we believe that they have higher tendency on indicating some of the eLS users' requirements.

Table 5-5. Blackboard topics' keywords

Topic No.	**Keywords**
1	**Time**, work, document, issue, **confuse**, work, experience, , **content glitch, load**
2	**Interface, difficult, navigate, confuse, time, grade, clunky, link, access, design**
3	Class, **grade, different, discussion**, administrator, learn, professor, **post, difficult, time**
4	**Buggy, mobile, website, time**, instructor, notifi**cation, scroll, navigate, site, discussion**
5	**Crash, interface, maintenance, layout, content, difficult**, type, work, program, **assignment**
6	**Update, slow, assignment, design, functionality, hard, time, improve, crash, outdated**
7	**Glitch, grade, update, clunky, file, interface, website, confuse, time, assignment**

Table 5-6. Canvas topics' keywords

Topic No.	Keywords
1	Issue, **notification, module, assignment**, class, date**, confuse**, page, **email, mobile**
2	**Grade, integration, design, file**, help, change, site, certain, class, **confusing**
3	**Page, notification, confuse**, setting, **material**, semester, limited, access, device, document
4	**Time**, great, different, **clunky**, professor, function, **quiz**, instructor, **navigation**,
5	**Navigate, assignment**, tool, post, **time, upload, file, grade, image**, test
6	**Assignment, calendar,** program, complaint, **interface**, allow, basic, right, feedback, **format**

Table 5-7. Google Classroom topics' keywords

Topic No.	Keywords
1	Teacher, **account, email, Gradebook, parent**, connect, allow, **discussion, forum, platform**
2	Program, **schedule**, issue, connection, drive, **document, notification**, similar, **confuse, interface**
3	**Website**, create, project, **organize**, limit, **share, document, access, update, email**
4	Multiple**, page, edit, organization, interface**, screen, **access, delete, functionality, online**
5	Tool, **scroll**, **platform, Powerschool, navigate, submit, internet**, book, hard, **upload**

5.2.2 Experimental Study #1 Evaluation

There is no better way to accurately evaluate the topics generated from the proposed method, rather than manually, using human brain, to revise the available users' reviews. We manually evaluated the approach by analyzing and reviewing each user's review for each LMS product to extract the user requirements for each review. Then we counted the number of reviews relevant to each requirement, and then we got the percentage of frequency for each requirement by dividing the number of reviews related to one requirement by the total number of reviews. This calculation was repeated for each manually extracted user requirement. The number of extracted requirements for each of the three LMSs under study is shown in Table 5-8. The topics were sorted by importance from the highest percentage to lowest percentage. Finally, we compared it with the extracted topics. The process of manually extracting the users' requirements from the LMS reviews was accomplished by one person familiar with RE and extracting user requirements.

Table 5-8. The number of extracted requirements for each LMS

LMS	Number of requirements
Blackboard	17
Canvas	22
Google Classroom	28

We evaluated the results of our automatic topic modeling by using *precision, recall, and F-measure*. According to Brownlee (Brownlee, 2014), *Precision* is how many selected items are relevant; it is computed by dividing the number of true positives by the sum of true positives and false positives as given in equation (4).

$$Precision = \frac{True\ Positive}{True\ Positive + False\ positive} \qquad (4)$$

Recall is how many relevant items are selected; it is computed by dividing the number of true positives by the sum of true positives and false negatives as given in equation (5).

$$Recall = \frac{True\ Positive}{True\ positive + False\ negative} \qquad (5)$$

F-measure tests the experiment's accuracy; it's approximately the average of both *recall* and *precision*, the best value is 1 and the worst is 0. *F-measure* is calculated as follows as given in equation (6):

$$F - measure = 2 * \frac{Precesion * Recall}{Precision + Recall} \quad (6)$$

True positive is the requirement that is both manually and automatically identified, false positive is the requirement that is automatically but not manually identified, and finally false negative is the requirement that is manually but not automatically identified from the approach. while working on the experimental study we considered a feature as true positive, if it was automatically extracted from a review and was also manually identified in that review. False positives are features that were automatically associated to a review in one of the topics, but were not identified manually in that review. Finally, false negative features were manually identified in a review but were not present in any of the extracted topics associated to the review. The results were as follows in Table 5-9; those results were at certain points when running the approach; any changes in text reviews, or their number could affect the final results.

Table 5-9. Precision, recall, and f-measure results

LMS	*Precision*	*Recall*	*F-Measure*
Blackboard	0.83	0.53	0.65
Canvas	0.79	0.46	0.58
Google Classroom	0.76	0.32	0.45
Average	**0.79**	**0.44**	**0.56**

5.2.3 Experimental Study #2

Experimental study # 2 aims to perform further analysis on the method and apply different techniques to improve the results. Four steps will be added from Bansal (Bansal, 2015) to enhance the performance of topic modeling results; the four steps that we will add to the requirements extraction steps are:

1- Domain specific features of the corpus. We applied further classification over LMS reviews under evaluation; we analyzed reviews for the role ‘user’ for better coherency and relevance of the results.

2- Noise free corpus. We performed a cleaning function to eliminate the special characters.

3- Use an exhaustive stopword list. Beside the use of language stopwords provided by Gensim library, we used supporting words, which are in less importance in the study e.g. LMS name, easy, little, user, like.

4- Use complex features such as Bi-gram. Considering a feature as a combination of two words provides better understanding and understanding than considering feature as a single word.

In general, Figure 5-4 illustrates the enhanced requirements extraction steps by adding some steps emphasized in bold to enhance the performance of topic modeling results.

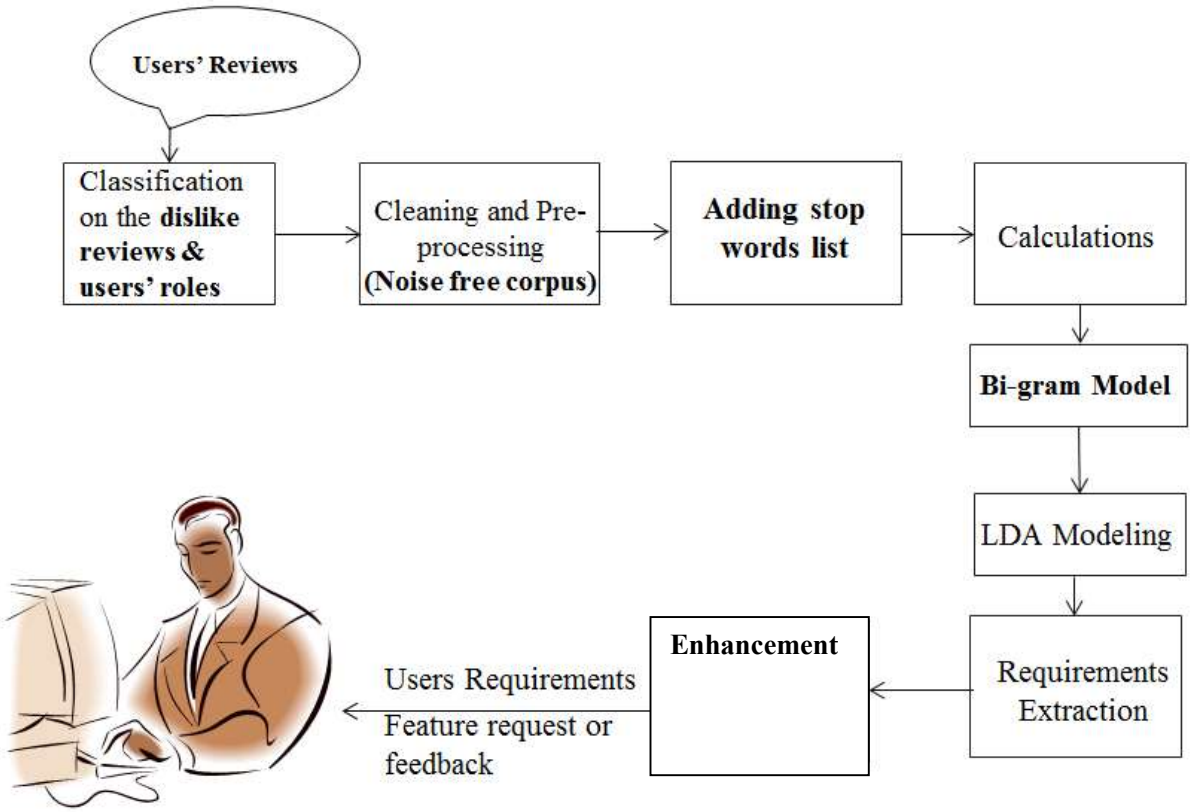


Figure 5-4. Enhanced requirements' extraction steps

1- **Classification**: the user's reviews were classified into the three LMS products under study with the highest number of reviews; we selected the 'dislike' category for features that users don't like; and the 'user' as a role in the LMS.

2- **Cleaning and preprocessing (Noise free corpus)**: two functions are created; one for cleaning the corpus (dataset) from the special characters that are not needed in the requirements extraction process e.g. commas, dashes, or semi colons. The second function is the preprocessing of the users' reviews; the preprocessing steps selected are tokenization, stopwords removal, part of speech tagging (POS) of wordnet in NLTK Package, stemming and finally lemmatization.

3- **Calculations**: this function created a dictionary for each word used in the users' reviews text with the assistance of Gensim dictionary function, to calculate the minimum number each word has appeared in the text, and the maximum ratio of

each word appeared in the text. In this function, Term Frequency-Inverse Document Frequency (TF-IDF) weight measurement was used to assess the importance of each word to a document in the corpus.

4- **Bi-gram Model**: this function is created to find number of sequences of two words in the corpus (Jurafsky & Martin, 2019); we believe it would enhance the results and make it more meaningful instead of search for feature-based keywords, which composed of only one word.
5- **LDA Modeling**: a function is created to apply LDA algorithm. Based on the results of words calculation function performed in step 3, LDA aims to extract the top keywords in the text. The result of this function was the top keywords representing LMS features in the corpus.
6- **Requirements' extraction**: a function is created to classify the top keywords results from step 5 into number of topic. Each topic represents a user requirement.
7- **Enhancement**: an enhancement function is created to calculate the optimum number of topics to be generated, to achieve this, it has to specify the minimum number the selected keywords appeared in the text, and the maximum ratio of these keywords appeared in the text. Next Table 5-10 shows the optimum values for the three LMS products under study.

Table 5-10. Enhancement function results

LMS	Min Count	Max Ratio	No. of Topics
Blackboard	9	0.4	2
Canvas	5	0.3	2
GoogleClassroom	7	0.2	2

The results from the enhancement are entered as an input for another function 'visualize the topics' to produce the output in a visualized format. The topics distribution according to the function 'visualize the topics' is illustrated in the next Figures 5-5, 5-6 and 5-7, for each LMS under study.

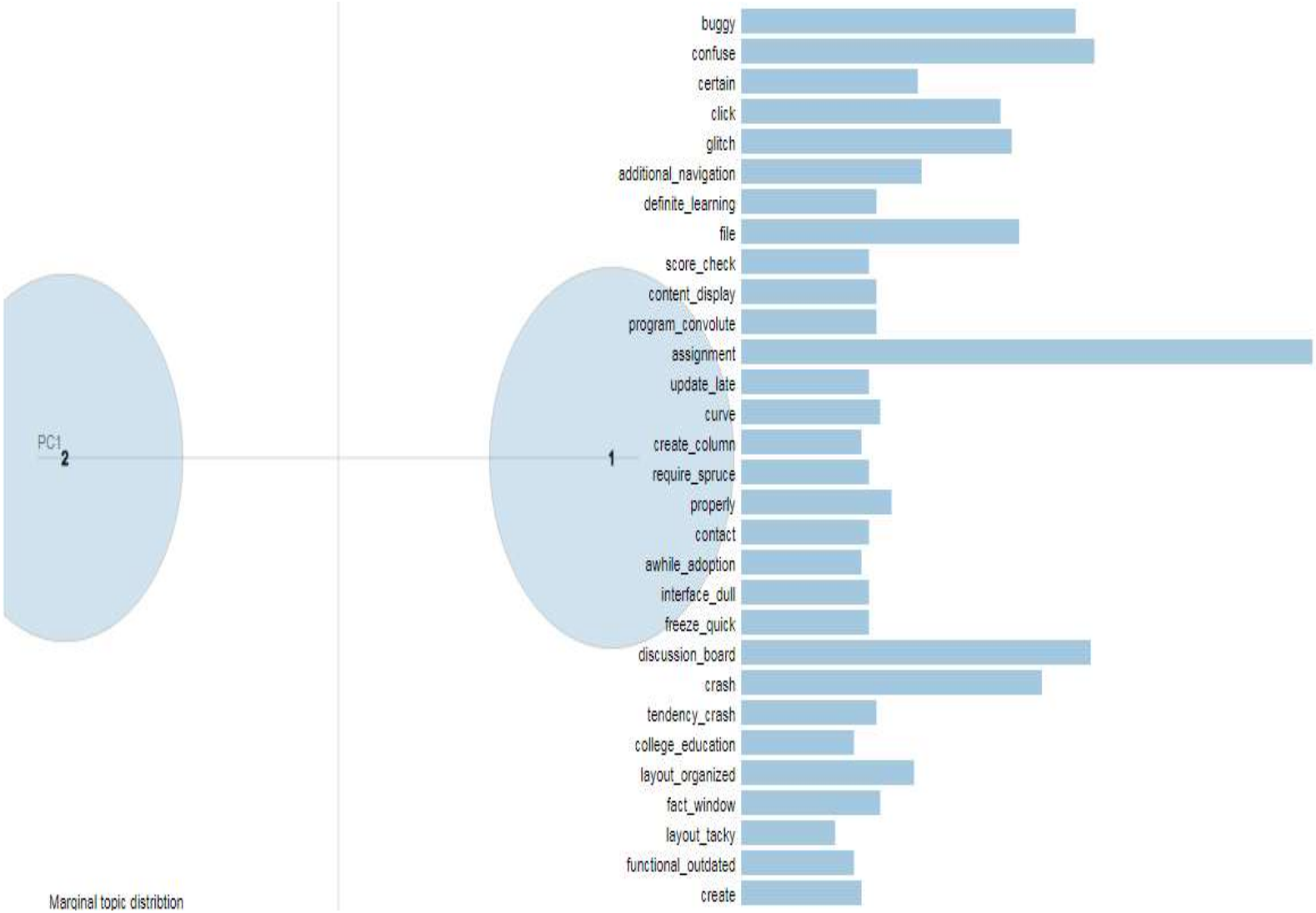


Figure 5-5. Visualizing Blackboard bi-gram topics results

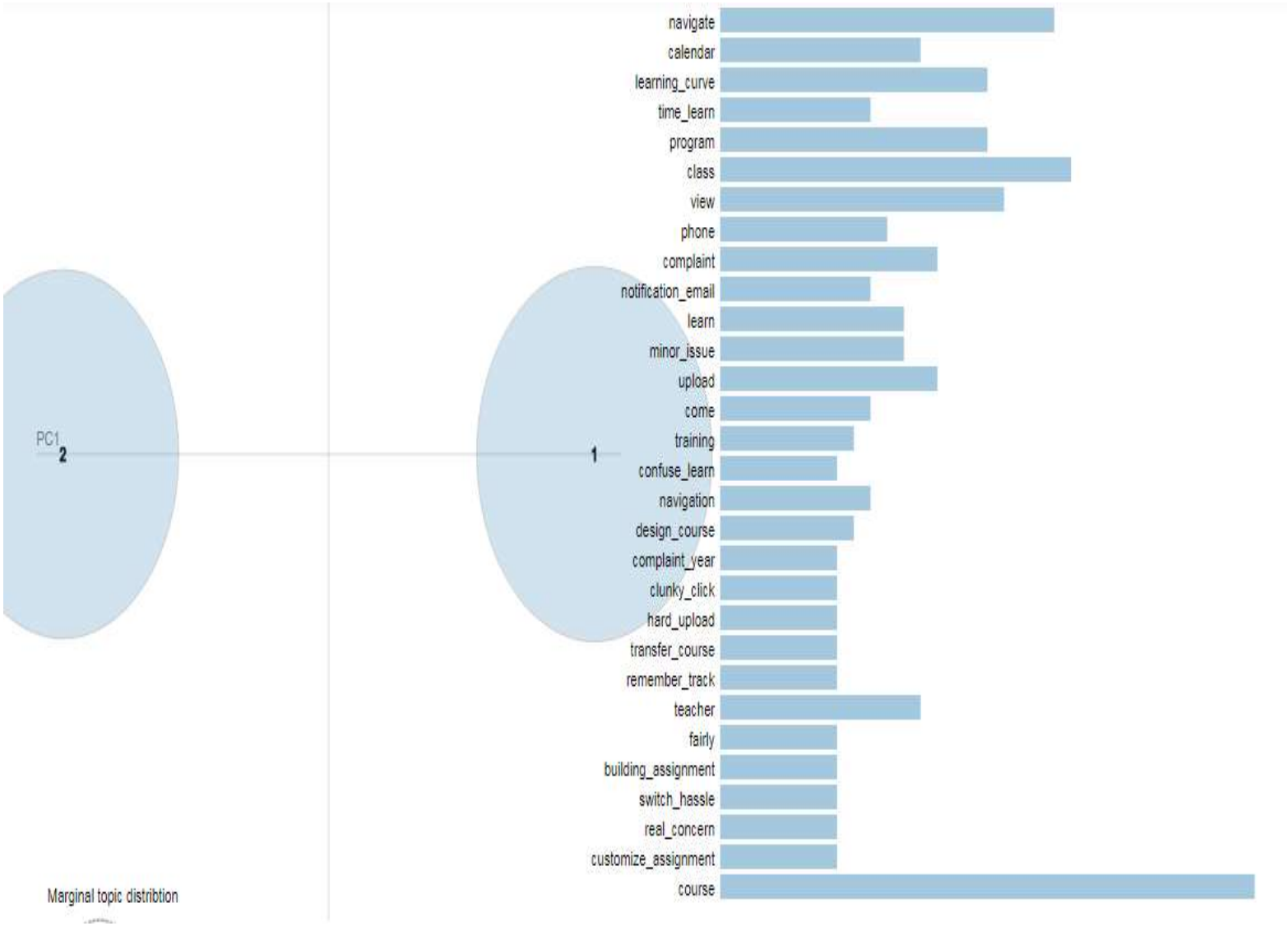


Figure 5-6. Visualizing Canvas bi-gram topics results

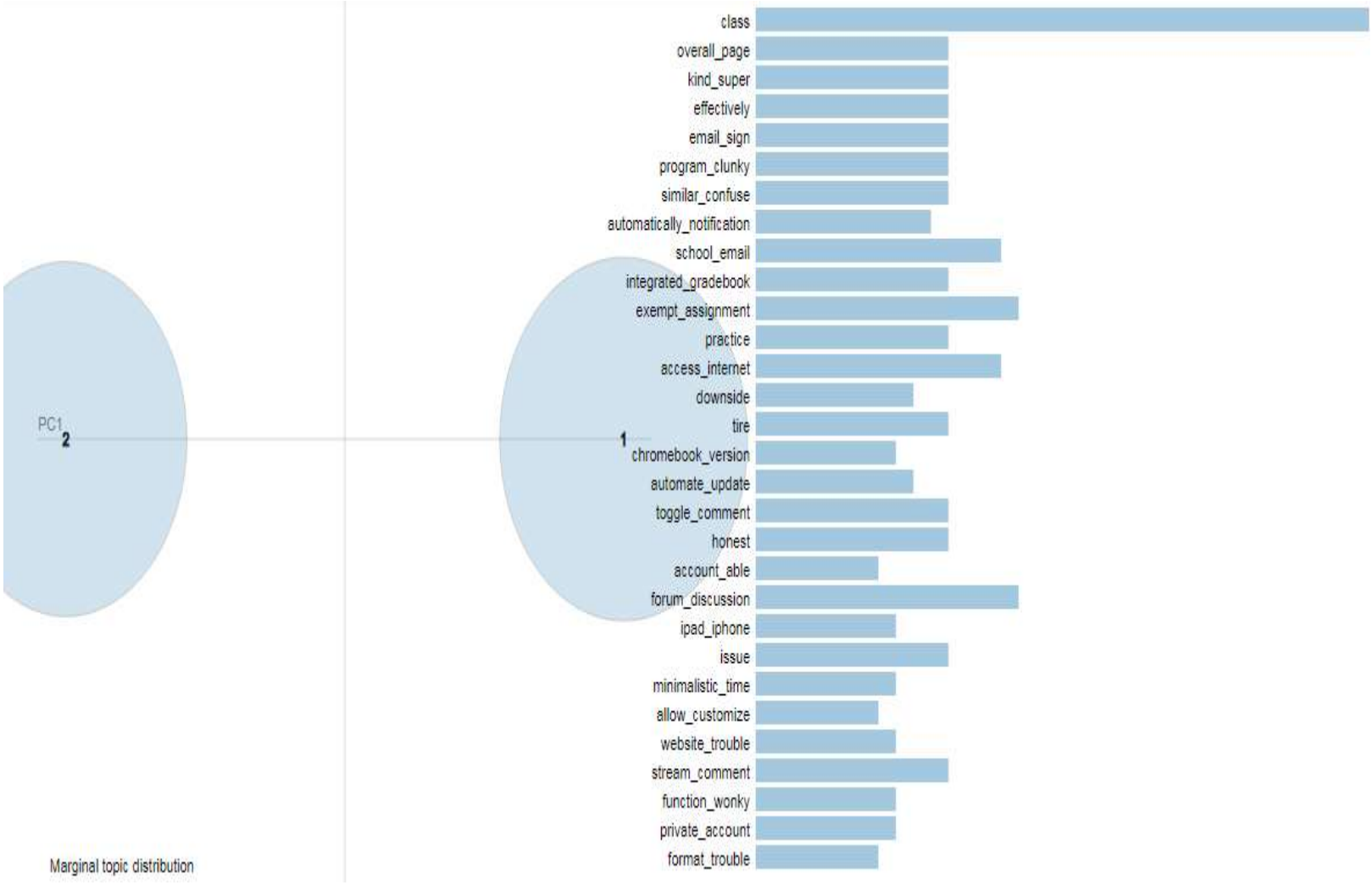


Figure 5-7. Visualizing Google Classroom bi-gram topics results

We noticed that the topics circles were mostly not overlapped which means that the topics keywords were not overlapped, each topic represented its own keywords. Circles with numbers represented the topics, the size of the circle indicated the dominance of this topic within the text, the keywords of the topic appeared in the text, and the highest frequency keyword appeared at the first. Table 5-11 illustrates the dominant bigram keywords composed of two words.

Table 5-11. LMS' Bigram topics' keywords

Topic No.	Keywords
Blackboard	Update_late, content_display, additional_navigation, discussion_board, layout_organized, functional_outdated, mobile_device, mobile_device, confuse_text, interface_dull
Canvas	Notification_email, clunky_click, building_assignment, transfer_course, hard_upload, learning_curve, customize_assignment, hard_navigate, design_course, time_consume, discussion_board
Google Classroom	Website_trouble, allow_customize, integrated_gradebook, automate_update, grading_powerschool, toggle_comment, automatically_notification, transfer_grade, formatting_option

5.2.4 Experimental Study #2 Evaluation

The same evaluation method of experimental study #1 was followed, and the results of *precision*, *recall*, and *f-measure* of the bigram method for the three LMSs under study were as follows, illustrated in Table 5-12.

Table 5-12. Bigram evaluation results

LMS	Precision	Recall	F-Measure
Blackboard	0.77	0.61	0.68
Canvas	0.75	0.53	0.63
Google Classroom	0.78	0.68	0.73
Average	**0.76**	**0.61**	**0.68**

5.3 Results Interpretations

Results interpretations section is giving a discussion for the achieved results of both studies and a benchmarking with other related publications results.

5.3.1 Results Discussion

The quantitative results and the qualitative evaluation of the proposed method, for both studies, were positive. CREeLS succeeded to extract an adequate number and good representation of topics, which are users' requirements of the LMS in a small-time frame compared to the manual process. The qualitative evaluation showed coherent topics; most of the keywords represent LMS features, but not noise, and the topics' keywords were relevant to the LMS requirements. If we compared the bi-gram evaluation results (study # 2) with the unigram LDA evaluation (study # 1), we found that unigram is higher with only 0.03 in *precision*, however the *recall* in bi-gram is higher by 0.17; in addition to *f- measure* which is higher in bi-gram by 0.12 than LDA. Bi-gram model achieved an adequate and more understandable number of the extracted LMS features. Having Synonym keywords and duplicate topics in the results indicate the importance of this keyword as an LMS product requirement, e.g. interface, design, look, clunky, and layout, reflect the need for better and modern design for the LMS. The keyword 'time' appeared in five topics but in different contexts; in topic 2 with the keyword navigation, it means that the navigation process takes time from the user. Also, in topic 4, the keyword time appeared with mobile, phone, and confuse means that the LMS version on the mobile takes more time. Finally, in topic 6 the keyword 'time' appeared with update, slow, functionality and crash, which means that some functions are slow and takes time, and this is not adequate with users. CREeLS method can overcome some of the challenges in traditional requirements elicitation techniques; CREeLS is based on eLS users, so it solved the limitation of inadequate involvement of users. CREeLS also solved the problem of scope and volatility of requirements because it used the interactions of eLS users in the eLS or their feedback on the eLS. All of these interpretations for the different contexts of using the word time were supported by the manual evaluation of the results.

5.3.2 Results Benchmarking

In our attempt to assess the final results, we benchmarked the experimental study evaluation results with some of the related work publications results. Below are some of these results.

Table 5-13 shows the results of Buchan et al. (Buchan, et al., 2018). The average measurements results listed in the table are comparable to the evaluation results for experimental study #2.

Table 5-13. Results of Buchan et al. (Buchan, Bano, Zowghi, & Volabouth, 2018)

Classifier	Class	Precision	Recall	F1
Linear SVM, n-gram (1,4), sentence + sentiment, K-fold 10	Class 0	81.7%	95.5%	88.1%
	Class 1	94.6%	78.6%	85.9%
	Average	88.2%	87.1%	87.0%

Table 5-14 depicts the results of Guzman and Maalej (Guzman & Maalej, 2014) and Table 5-15 shows the results of Galvis Carreno and Winbladh (Galvis Carreño & Winbladh, 2013). We can conclude that the current results of *precision, recall, and f- measure* are comparable and even better.

Table 5-14. Results of Guzman and Maalej (Guzman & Maalej, 2014)

App	Precision	Recall	F-measure
	F_S		
AngryBirds	0.335	0.332	0.334
Dropbox	0.608	0.475	0.533
Evernote	0.474	0.416	0.443
TripAdvisor	0.421	0.399	0.410
PicsArt	0.750	0.669	0.707
Pinterest	0.644	0.623	0.634
Whatsapp	0.843	0.728	0.781
F_S Average	**0.582**	**0.520**	**0.549**
	F_{NS}		
AngryBirds	0.368	0.321	0.343
Dropbox	0.603	0.473	0.531
Evernote	0.451	0.389	0.418
TripAdvisor	0.403	0.370	0.386
PicsArt	0.815	0.661	0.730
Pinterest	0.658	0.592	0.623
Whatsapp	0.910	0.734	0.813
F_{NS} Average	**0.601**	**0.506**	**0.549**

Table 5-15. Results of Galvis Carreno and Winbladh (Galvis Carreño & Winbladh, 2013)

Classification	Precision	Recall	F-Measure
Classification 1	70%	68%	68.98%
Classification 2	71%	48%	57.27%
Classification 3	75%	19%	30.32%

Figure 5-8 provides a summary for all the compared results. We can conclude that the current results of *precision, recall, and f- measure* are in average, comparable and even better.

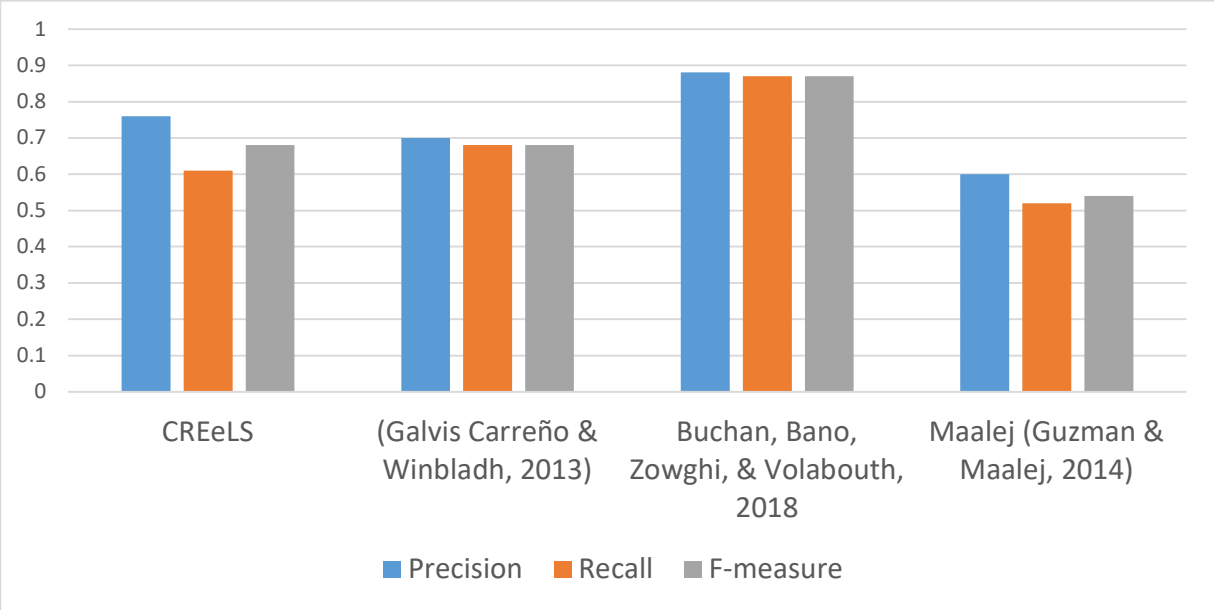


Figure 5-8. Comparing CREeLs results.

5.4 Research Limitations

This section gives some limitations of the research:

- The approach only applied on the requirements elicitation activity from the RE phase.
- The extracted requirements are not classified into functional or non-functional requirements.
- Infrequently mentioned users' requirements can't be detected by the approach.
- The LDA approach used in the study needs large number of reviews to perform better.

5.5 Threats to Validity

The results of the study may be influenced by the coverage of study search, bias on study selection and personal judgment in study. Therefore, according to the guidelines in Wohlin et al. (Wohlin, et al., 2012) four types of threats to validity of the review results are discussed below.

5.5.1 Conclusion Validity

Thesis experimental studies can be applied by other researchers. We briefly explained each function performed, and the method of evaluation. However, concerning the manual interpretation of the reviews is a human act, that maybe subjective. Different researchers may have different understanding on the manual requirements extraction of the user's reviews, and in turn, might bring different results in the evaluative measures (*precision, recall, and f-measure*) of the study. However; this threat is not only applicable in our study but also in real life requirements extraction process, because different requirements engineers can have different understanding for users' requirements. To reduce this threat, requirements extraction process can be performed by more than one researcher or requirement engineer and discussed together to assure common understanding on the final result.

5.5.2 Internal Validity

Internal validity is affected by flaws within the study itself such as not controlling some of the major variables. In the thesis the number of analyzed LMS users' reviews can affect the output, because LDA algorithm works better on large corpus (dataset), the greater the reviews number, the better the results.

5.5.3 External Validity

External validity is the extent to which the study findings can be generalized to a larger group or other contexts. The study is only valid for the topic of crowdsourced requirements elicitation, however larger group of users' reviews can be analyzed, and different LMSs categories are applicable to use the same study.

5.5.4 Construct Validity

Construct validity is used to determine how well a test measures what it is supposed to measure. In this thesis we repeat the extraction process on three different LMSs to ensure the results validity. The construct validity in the research is minimal.

5.6 Summary

This chapter presented an evaluation for CREeLS. A tool was developed based on topic modeling algorithms. Two techniques were used; unigram, and bigram LDA. The results were very promising with an adequate number of keywords that represent LMS features. Finally, the results are benchmarked with number of related publications to assess their reliability.

6 Conclusions and Future Work

6.1 Conclusions

Recently crowdsourcing was investigated as an opportunity in the requirements elicitation activity. Crowdsourcing in requirements elicitation is a growing model for helping organizations to gather accurate and useful requirements. Although there are many platforms and models for the use of crowdsourcing in RE, it is hardly to find one study applied on eLearning. Crowdsourcing is used in requirements elicitation to help RE. It assists in finding, detecting and involvement of different stakeholders who can outline software requirements. Crowdsourcing increases and improves the range of elicited requirements and, as a result, helps getting a whole idea of users' and other stakeholders' expectations from a software. The research presented an overview and a comparison between current requirements elicitation approaches and crowdsourcing based approaches as we envision that it can be useful and offers solutions for current problems of the eLS domain. We also considered current studies that address crowdsourcing in RE. It's common to find some supported tools used with crowdsourcing concept. Social network analysis tools are used in stakeholders' analysis, requirements elicitation analysis and requirements prioritization. The use of NLP and text mining algorithms are used for extracting requirements from users' reviews within the context of mobile applications stores to help in requirements elicitation process. Also, gamification is used for developing a crowdsourcing-based tool and method for requirements elicitation activity.

The thesis has presented CREeLS, the crowdsourcing-based requirements elicitation method for eLS. It is made up of a framework for the necessary elements of crowdsourcing, suggesting specific tools for each element and phases to implement the framework. CREeLS phases suggested the way of how to benefit from CREeLS in the requirements elicitation for eLS domain, using SNA and topic modeling techniques to extract requirements from eLS users' feedback. An experimental study was conducted to extract users' requirements from real-life LMS users' reviews. The experimental study was applied on three LMS products following specific steps to reach for the results. The steps

are; classification, cleaning and preprocessing, words calculations, Latent Drichlet Allocation (LDA) algorithm modeling, requirements extraction, and enhancement. The results were evaluated by manually revising the user's reviews which showed coherence of topics up to 0.52, *recall* value 0.44, *precision* 0.79 and *f-measure* of 0.56. The results were very promising, but we wanted to enhance the results and added the bi-gram modeling as a way for enhancing the results of LDA topic modeling algorithm. Bi-gram is considering a feature as a combination of two words; it provides better understanding than considering feature as a single word. Hence, we modified the extraction steps to be; Classification (we applied more classification criteria over this step in comparison with first experimental study, cleaning and preprocessing (Noise free corpus), calculations, bi-gram model, LDA algorithm modeling, requirements' extraction and enhancement). Another experimental study was conducted on the same three LMS products user's reviews. The resulted keywords were very promising and succeeded to be more understandable to the requirements engineer in the requirements elicitation phase. Manual analysis for the LMS users' reviews was conducted to evaluate the results, the same three measures were used and their results were very encouraging while compared to the normal LDA algorithm results. The bi-gram model average results were 0.76 *precision*, 0.61 *recall* and 0.68 *f-measure*. By comparing the results of both experimental studies, we found that:

- Uni-gram is higher with only 0.03 in *precision.*
- *Recall* in bi-gram is higher by 0.17.
- *F- measure* is higher in bi-gram by 0.12 than LDA.
- Bi-gram model achieved an adequate and more understandable number of the extracted LMS features.
- Most of the keywords represented LMS features.

The extracted requirements were relevant and can help in requirements evolution of the eLS. CREeLS method can overcome some of the challenges in traditional requirements elicitation techniques; CREeLS is based on eLS users, so it solved the limitation of

inadequate involvement of users. CREeLS also solved the problem of scope and volatility of requirements because it used the interactions of eLS users in the eLS or their feedback on the eLS. We can contend that CREeLS can help requirements engineers for eLS to analyze users' opinions and identify the most common users' requirements for better software evolution.

6.2 Future Work

While working on the research and after analyzing the results of the studies performed, more than one future work area has been identified as follows, and we expect this will lead to better results and use of the method:

- The experimental study conducted in the research covers only phase number three in CREeLS approach; our future work is intended to test and evaluate the approach for all its phases.
- Tri-gram model or dynamic N-gram model could be conducted with greater number of LMS users' reviews.
- Adding the time frame of the user's reviews analyzed, gives more awareness and exactness to users' needs in a specific time frame.
- Including the field domain of the LMSs' users (Computer Science university students -high school students – accountants) into consideration while classifying the LMS users' reviews.
- Including the non-functional requirements alongside the functional requirements.
- Aiming to address more of the rest of the challenges of traditional requirements elicitation approaches e.g. problems in scope understanding and the bias of requirements engineers.
- Another interesting improvement is the use of deep learning techniques Such as Recurrent Neural Networks (RNN) and auto-encoders which can give high level features for better results and for more understanding for the users' requirements.

References

Adepetu, A., Ahmed, . K. A. & Al Ab, Y., 2012. *CrowdREquire: A Requirements Engineering Crowdsourcing Platform,Technical Report,* Association for the Advancement of Artificial Intelligence.

(DoD), Department of Defense, 1991. *Software Measurement for DoD Systems: Recommendations for Initial Core Measures, Technical Report,* (DoD), Department of Defense.

Abdul Rahman, N. & Sahibuddin, S., 2010. *E-Learning Technology for Problem-Based Learning Technique: An Introduction of Requirements Elicitation Model.* Kuching, Sarawak, pp. 217-221.

Abdul Rahman, N. & Sahibuddin, S., 2011. *Challenges in E-learning: an Requirements Engineering Perespective. In The Proceedings of The 3rd International Conference on Computing and Informatics.* Bandung, Indonesia, World Academy of Science, Engineering, and Technology, pp. 167-173.

Ahmad, A. et al., 2019. Toward Empirically Investigating Non-Functional Requirements of iOS Developers on Stack Overflow. *IEEE Access, IEEE,* Issue 7, pp. 61145-61169.

Alharthi, A. D., Spichkova, M. & Hamilton, M., 2015. *Requirements Engineering Aspects of eLearning Systems. In The Proceedings of The 24th Australian Software Engineering Conference,* Adelaide, SA, Australia, ACM, pp. 132-133.

Alharthi, A. D., Spichkova, M. & Hamilton, M., 2019. Sustainability requirements for eLearning systems: a systematic. *Requirements Engineering,Springer,* December, 24(4), p. 523–543.

Ali, N. & Lai, R., 2017. A Method of Requirements Elicitation and Analysis for Global Software Development. *Journal of Software Evolution and Process,Wiley,* 29(4), pp. 1-27.

Ali, R., Dalpiaz, F., Giorgini, P. & Souza, V. E. S., 2011. *Requirements Evolution: From Assumptions to Reality. In The Proceedings of International Workshop on Business*

Process Modeling, Development and Support, International Conference on Exploring Modeling Methods for Systems Analysis and Design, Tallinn, Estonia, Springer, pp. 372-382.

AlKhuder, S. B. & AlAli, F. H., 2017. E-Learning Systems Requirements Elicitation: Perspectives and Considerations. *International Journal of Virtual and Personal Learning Environments (IJVPLE),IGI Global,* 7(1), pp. 44-55.

Allen, W., 2016. *Learning for Sustainability.* [Online] Available at: http://learningforsustainability.net/social-learning/ [Accessed 10 February 2019].

Ambreen, T., 2019. *Handling Socio-Technical Barriers Involved in Crowd-Based Requirements Elicitation.In The Proceedings of The 27th International Requirements Engineering Conference (RE),* Jeju Island, South Korea, IEEE, pp. 476-481.

Australian Flexible Learning framework, 2007. *Practical Guide to E-learning for Industry. White paper.*

Bansal, S., 2015. *6 Practices to Enhance The Performance of a Text Classification Model.* [Online] Available at: https://www.analyticsvidhya.com [Accessed 5 February 2019].

Barbosa, C. E. et al., 2013. *Crowdsourcing Environments in e-Learning Scenario: A Classification Based on Educational and Collaboration Criteria. In The Proceeding of The International Conference on Systems, Man, and Cybernetics*, Manchester, United Kingdom, IEEE, pp. 687-692.

Batanero, C. et al., 2017. Accessible Platforms for E-learning: A Case Study. *Accessible platforms for e-learning: A case study,Wiley,* Volume 25, pp. 1018-1027.

Baxter, G. & Sommerville, I., 2011. Socio-Technical Systems: From Design Methods to Systems Engineering. *Interacting with Computers,* Oxford Academic, 23(1), pp. 4-17.

Bird, S., Klein, E. & Loper, E., 2009. *Natural Language Processing with Python.* [Online] Available at: http://www.nltk.org/book_1ed/ [Accessed 20 Decmeber 2020].

Blei, D. M., NG, A. Y. & Jordan , M. I., 2003. Latent Dirichlet Allocation. *Journal of Machine Learning Research, ACM,* Volume 3, p. 993–1022.

Breaux, T. D. D. & Schaub, F., 2014. *Scaling Requirements Extraction to the Crowd: Experiments with Privacy Policies. In The Proceedings of The 22nd International Requirements Engineering Conference,* Karlskrona, Sweden, IEEE, pp. 163-172.

Brinkkemper, S., 1996. Method Engineering: Engineering of Information Systems Development Method and Tools. *Information and Software Technology,ElSevier,* 38(4), pp. 275-280.

Brownlee, J., 2014. *Classification Accuracy is Not Enough: More Performance Measures You Can Use.* [Online] Available at: www.machinelearningmastery.com [Accessed 1 May 2019].

Buchan, J., Bano, M., Zowghi, D. & Volabouth, P., 2018. *Semi-Automated Extraction of New Requirements from Online Reviews for Software Product Evolution. In The Proceedings of The 25th Australaian Software Engineering Conference (ASWEC)*. Adelaide, SA, Australia, ACM, pp. 31-40.

Capers, J., 1996. *Applied Software Measurement: Assuring Productivity and Quality.* Second ed. New York: McGraw-Hill.

Chernev, B., 2019. *21 Astonishing E-learning Statistics for 2019.* [Online Article] Available at: https://techjury.net/stats-about/elearning/#gref [Accessed 20 November 2019].

Coulin , C., Zowghi , D. & Sahraoui, . A., 2006. A Situational Method Engineering Approach to Requirements Elicitation Workshops in The Software Development Process. *Software Process: Improvement and Practice,Wiley,* 11(5), pp. 451-464.

De Angelis, G., Ferrari, A., Gnesi, S. & Polini, A., 2016. *Collaborative Requirements Elicitation in European Research Project. In The Proceedings of The 31st Annual ACM Symposium on Applied Computing*, New York, ACM, pp. 1282-1289.

Ernst, N., Borgida, A., Jureta, I. J. & Mylopoulos, J., 2014. An Overview of Requirements Evolution. In: S. A. C. A. Mens T., ed. *Evolving Software Systems.* Springer, Berlin, Heidelberg, pp. 3-32.

Franco, A. J., 2015. *Requirements elicitation approaches: A systematic review.* In The *Proceeding of The 9th International Conference on Research Challenges in Information Science*.Athens, Greece, IEEE, pp. 520-521.

Fuentes-Fernández, R., Gómez-Sanz, J. J. & Pavón, J., 2010. Understanding The Human Context In Requirements Elicitation. *Requirements Engineering,* ElSevier, 15(3), pp. 267-283.

Galvis Carreño, L. V. V. & Winbladh, K., 2013. *Analysis of User Comments: An Approach for Software Requirements Revolution. In The Proceedings of The 35th International Conference on Software Engineering (ICSE).* San Francisco, CA, USA, IEEE.

Garrison, R. D. & Anderson, T., 2003. *E-learning in The 21st Century: A Framework For Research and Practice.2nd Ed.,* New York: Routledge Falmer.

Gheorghiu, . G., 2017. *The Content in a Learning Content Management System (LCMS).* [Online] Available at: https://www.selecthub.com/learning-management/learning-content-management-system-lcms/ [Accessed 5 May 2018].

Goldsworthy , K. & Rankine, L., 2009. *Identifying the Characteristics of eLearning Environments Used to Support Large Units. In the Proceedings of The Australasian Society for Computers in Learning in Tertiary Education* (Ascilite) Auckland pp. 338-345.

Groen, E. C., Doerr, J. & Adam, . S., 2015. Towards Crowd-Based Requirements Engineering A Research Preview. In: S. A. Fricker & K. Schneider, eds. *Requirements Engineering: Foundation for Software Quality.* Essen, Germany: Springer International Publishing, pp. 247-253.

Groen, E. C. et al., 2018. *Is there Really a Need for Using NLP to Elicit Requirements? A Benchmarking Study to Assess Scalability of Manual Analysis, In The Proceedings of The International Working Conference on Requirements Engineering: Foundation for Software Quality (Workshop),* Utrecht, Netherlands.

Groen, E. C. et al., 2017. The Crowd in Requirements Engineering: The Landscape and Challenges. *IEEE Software,* March, 34(2), pp. 44-52.

Groen, E. & Koch, M., 2016. *How Requirements Engineering can Benefit from Crowd.* [Online] Available at: https://re-magazine.ireb.org/articles/how-requirements-engineering-can-benefit-from-crowds [Accessed 9 May 2017].

Guzman, E. & Maalej, W., 2014. *How Do Users Like This Feature? A Fine Grained Sentiment Analysis of App Reviews. In The Proceeding of The Requirements Engineering Conference (RE),* Karlskrona, Sweden, IEEE, pp. 153-162.

Hammad, R. & Khan, Z., 2013. *Towards A Generic Requirements Model for Hybrid and Cloud-based e-Learning Systems.* Bristol, IEEE, pp. 106-111.

Hammad, R., Odeh, M. & Khanz, Z., 2013. *Towards A Generic Requirements Model for Hybrid and Cloud-based e-Learning Systems. In The Proceeding of The Cloud Computing Technology and Science, IEEE 5th International Conference On Volume 2*, BristolUK, IEEE, pp. 106-111.

Hickey, A. M. & Davis, A. M., 2004. A Unified Model of Requirements Elicitation. *Journal of Management Information Systems Taylor & Francis,* 20(4), pp. 65-84.

Holmes, B. & Gardner, J., 2006. *E-learning: Concepts and Practice,.* 1st ed. :Sage Publications.

Horton, W. & Horton, K., 2003. *E-learning Tools and Technologies.*Wiley.

Hosseini, M., Groen, E. C., Shahri, A. m. & Ali, R., 2017. *CRAFT: A Crowd-Annotated Feedback Technique. In The Proceedings of The 25th International Requirements Engineering Conference Workshops* ,Lisbon, Portugal, IEEE. pp. 170-175.

Hosseini, M., Phalp, K., Taylor, J. & Ali, R., 2014. Toward Crowdsourcing for Requirements Elicitation: Results from Expert Survey. *Management,IEEE,* 2(33), pp. 9-44.

Hosseini, M. P. K. T. T. J. a. A. R., 2014. *Towards Crowdsourcing for Requirements Engineering.* In The *Proceeding of The 20th International Working Conference on Requirements Engineering: Foundation for Software Qualit,* Essen,Germany, Springer.

Howe, J., 2006. *Crowdsourcing: A Definition.* [Online] Available at: https://crowdsourcing.typepad.com/cs/2006/06/crowdsourcing_a.html [Accessed 7 January 2017].

Howe, J., 2006. *Wired.* [Online] Available at: http://archive.wired.com/wired/archive/14.06/crowds.html?pg=4&topic=crowds&topic_set= [Accessed 31 07 2015].

Hull, E., Jackson, K. & Dick, J., 2011. *Requirements Engineering.* 3rd ed. Springer-Verlag London.

Huuhka, H., 2003. *Evolution of Requirements,* University of Helsinki, Department of Computer Science: Seminar Report.

Irlbeck , S. & Mowat, J., 2007. *Learning Content Management System (LCMS).* [Online] Available at: http://www.herridgegroup.com/pdfs/LCMS_chptr_Aug29%202005_%20Final%20Sept%206%2005.pdf [Accessed 6 January 2018].

Jurafsky , D. & Martin, J. H., 2019. *Speech and Language Processing..* [Online] Available at: https://web.stanford.edu/~jurafsky/slp3/3.pdf [Accessed 9 November 2019].

Karataev, E. & Zadorozhny, V., 2017. Adaptive Social Learning Based on Crowdsourcing. *IEEE Transactions on Learning Technologies, IEEE,* 10(2), pp. 128 - 139.

Kramer, J., Ng, K., Potts, C. & Whitehead, K., 1988. Tool Support for Requirements Analysis. *Software Engineering Journal, Springer,* 3(3), pp. 86-96.

Li, C. & Lalani, F., 2020. *The COVID-19 Pandemic has Changed Education Forever. This is How.* [Online] Available at: https://www.weforum.org/agenda/2020/04/coronavirus-education-global-covid19-online-digital-learning/ [Accessed 20 August 2020].

Lim, S. L. & Finkelstein, A., 2012. StakeRare: Using Social Networks and Collaborative Filtering for Large-Scale Requirements Elicitation. *IEEE Transactions on Software Engineering,IEEE,* 38(3), pp. 707-735.

Lim, S. L., Quercia, D. & Finkelstein, A., 2010. *StakeSource: Harnessing the Power of Crowdsourcing and Social Networks in Stakeholder Analysis. In The Proceedings of The 32nd International Conference on Software Engineering,* New York, United States, IEEE.

Ma, E., 2018. *Towards Data Science.* [Online] Available at: https://towardsdatascience.com/2-latent-methods-for-dimension-reduction-and-topic-modeling-20ff6d7d547#:~:text=Both%20LSA%20and%20LDA%20have,LDA%20solves%20topic%20modeling%20problems. [Accessed 7 January 2020].

Manning, C. D., Raghavan , P. &. H. & Schütze, H., 2008. TF-IDF Weighting. In: *Introduction to Information Retrieval.* s.l.:Cambridge University Press.

Manning, C. D., Raghavan , P. & Schütze, H., 2019. *Stanford.* [Online] Available at: https://nlp.stanford.edu/IR-book/html/htmledition/dropping-common-terms-stop-words-1.html [Accessed 3 June 2019].

Mao, K., Capra, L., Harman , . M. & Jia, Y., 2015. *A Survey of the Use of Crowdsourcing in Software Engineering,* London: University College London, Department of Computer Science.

Mao, K., Capra, L., Harman, M. & Jia, Y., 2016. A Survey of the Use of Crowdsourcing in Software Engineering. *Journal of Systems and Software,Elsevier,* Volume 126, pp. 57-84.

Michael, C. & Kyo, K., 1992. *Issues in requirements elicitation,* Pittsburgh, Pennsylvania: Technical Report, CMU/SEI-92-TR-012, Carnegie Mellon University.

Nayebi, M. et al., 2017. *Crowdsourced Exploration of Mobile App Features: A Case Study of the Fort McMurray Wildfire. In The Proceedings of The 39th Intern ational Conference on Software Engineering: Software Engineering in Society Track (ICSE-SEIS),* Buenos Aires, Argentina, IEEE/ACM.

Nichani, M., 2001. *LCMS = LMS + CMS.* [Online] Available at: http://www.elearningpost.com/articles/%20archives/lcms_lms_cms_rlos [Accessed 5 April 2016].

Nuseibeh, B. & Easterbrook, S., 2000. *Requirements Engineering: A Roadmap.* Limerick, In The *Proceedings of The Conference on The Future of Software Engineering,* Irland, IEEE. pp.35-46.

O'Reilly, T., 2005. *What is WEB 2.0 – design patterns and business models for the next generation of software.* [Online] Available at: www.oreillynet.com/pub/a/oreilly/tim/news/2005/09/30/what-is-web-20.html/ [Accessed 10 July 2019].

Pagano, D. & Maalej, W., 2013. *User FeedBack in Appstore an Empirical Study. In The Proceedings of The Requirements Engineering Conference*, Rio de Janeiro, IEEE, pp. 125-134.

Palazuelos, C., García-Saiz, D. & Zorril, M., 2013. Social Network Analysis and Data Mining: An Application to the E-Learning Context. In: C. Bădică, N. T. Nguyen & M. Brezovan, eds. *Computational Collective Intelligence. Technologies and Applications.* Craivo: Springer Berlin Heidelberg, pp. 651-660.

Paulina , D. & Haythornthwaiteb, C., 2016. Crowdsourcing the Curriculum: Redefining eLearning Practices Through Peer-Generated Approaches. *The Information Society, Taylor and Francis,* 32(2), pp. 130-142.

Pedreira , O., García , . F. & Brisabo, N., 2015. Gamification in Software Engineering – A Systematic Mapping. *Information and Software Technology,* Volume 57, pp. 157-168.

Peng, H., Ma , S. & Spec, J. M., 2019. Personalized Adaptive Learning: an Emerging Pedagogical Approach Enabled by a Smart Learning Environment. *Smart Learning Environemnts.*

Piccioli, V. & Moriera, I., 2015. *Docebo.* [Online] Available at: https://www.docebo.com/2015/04/23/8-types-elearning-students/ [Accessed 25 April 2016].

Proctor, J. & Maher, B., 2013. *Emotional Multiplicities in Multi-Sourced Work.* [Online] Available at: http://dnaanthology.com/anvc/dna/emotional-multiplicities-in-multi-sourced-work [Accessed 28 March 2016].

Punjabi, D. M., Tung, L.-P. & Pau, B.-S., 2013. *CrowdSMILE: A Crowdsourcing-based Social and Mobile Integrated System for Learning by Exploration. In The Proceedings of The 10th International Conference on Ubiquitous Intelligence & Computing and 2013 IEEE 10th International Conference on Autonomic & Trusted Computing*, Vietri sul Mere, IEEE, pp. 521 - 526.

Radha, R., Mahalakshmi, K., Kumar, V. & Saravanakumar, A., 2020. E-Learning during Lockdown of Covid-19 Pandemic: A Global Perspective. *International Journal of Control and Automation,* 13(4), pp. 1088-1099.

Rashid, A., Wiesenberger, J., Mede, D. & Baumann, J., 2008. *Bringing Developers and Users Closer Together:The OpenProposal Story. In The Proceedings of The conference at the Multikonferenz Wirtschaftsinformatik*, München, Germany, dblp, pp. 9-26.

Renzel, D. & Klamma, R., 2014. Requirements Bazaar: Open-Source Large Scale Social Requirements Engineering in the Long Tail. *IEEE Computer Society Special Technical Community on Social Networking E-Letter*, IEEE, 2(3).

Rizk, N. M., Ebada, A. & Nasr, E. S., 2015. *Investigating Mobile Applications' Requirements Evolution Through Sentiment Analysis of Users' Reviews. In The Proceedings of 11th International Computer Engineering Conference (ICENCO)*.Giza, IEEE, pp. 123-130.

Sharma, A. K., 2020. *Understanding Latent Dirichlet Allocation (LDA).* [Online] Available at: https://www.mygreatlearning.com/blog/understanding-latent-dirichlet-allocation/ [Accessed 21 December 2020].

Sharma, R. & Sureka, A., 2017. *CRUISE: A Platform for Crowdsourcing Requirements Elicitation and Evolution. In The Proceedings of The Tenth International Conference on Contemporary Computing (IC3)*, Noida, India, IEEE, pp. 1-7.

Silge, J. & Robinson, D., 2017. *Text Mining with R.* 1st ed. O'Reilly Media, Inc.

Singh, M., 2020. *Stop the Stopwords using Different Python Libraries.* [Online] Available at: https://medium.com/towards-artificial-intelligence/stop-the-stopwords-using-different-python-libraries-ffa6df941653#:~:text=Conclusion,disk%20space%20and%20increases%20accuracy. [Accessed 19 December 2020].

SketchEngine Team, 2019. *Sketch Engine.* [Online] Available at: https://www.sketchengine.eu/pos-tags/ [Accessed 2 June 2019].

Snijders, R. et al., 2015. *REfine: A Gamified Platform for Participatory Requirements Engineering. In The Proceedings of The 1st International Workshop on Crowd-Based Requirements Engineering (CrowdRE),* Ottawa, Canada, IEEE. pp. 1-6.

Snijders, R., Ozum, A., Brinkkemper, S. & Dalpiaz, F., 2015. *Crowd-Centric Requirements Engineering: A Method Based On Crowdsourcing And Gamification,* Utrecht, The Netherlands: University of Utrecht.

Soledade , M. P. d. P. et al., 2013. *Experimenting with Design Thinking in Requirements Refinement for a Learning Management System. Anais do Simpósio Brasileiro de Sistemas de Informação*. pp. 182-193. Available at: http://www.lbd.dcc.ufmg.br/colecoes/sbsi/2013/0016.pdf

Sommerville, I., 2015. *Software Engineering.* 10th ed., Pearson.

Souza, C. L. d. C. & Silva, C., 2015. An Experimental Study of the Use of Design Thinking as a Requirements Elicitation Approach for Mobile Learning Environments. *CLEI Electronic Journal,* April, 18(1), pp. 6-6.

Srivastava, P. K. K. & Sharma, R., 2015. *Crowdsourcing to Elicit Requirements for MyERP Application. In The Proceedings of The 1st International Workshop on Crowd-Based Requirements Engineering (CrowdRE),* Ottawa,Canada, IEEE, pp. 31-35.

Standish Group International, 2013. *CHAOS manifesto,*The Standish Group.

Stecanella, B., 2019. *MonkeyLearn.* [Online] Available at: https://monkeylearn.com/blog/what-is-tf-idf/ [Accessed 20 December 2020].

Suhonjić, A. Z. et al., 2019. Fostering Students' Participation in Creating. *Interactive Learning Environments*, March, 27(1), pp. 72-85.

Tarasowa, D., Khalili, A., Auer, S. & Unbehauen, J., 2015. CrowdLearn: Crowd-sourcing the Creation of Highly-structured E-Learning Content. *International Journal of Engineering Pedagogy,* 5(4).

Thi Tran, H. M. & Anvari, F., 2016. A Five-Dimensional Requirements Elicitation Framework for e-Learning Systems. *International Journal of Information and Electronics Engineering,* 6(3), pp. 185-191.

Tuunanen, T., 2003. A New Perspective on Requirements Elicitation Methods. *The Journal of Information Technology Theory and Application (JITTA),* 5(3), pp. 45-72.

United Nations, 2020. *Policy Brief: Education during COVID-19 and Beyond,* United Nations.

Vliet, M. v., Groen, E. . C., Dalpiaz, F. & Brinkkemper, S., 2020. *Identifying and Classifying User Requirements in Online Feedback via Crowdsourcing. In The Proceeding of The International Working Conference on Requirements Engineering: Foundation for Software Quality*, Pisa, Italy, Springer, pp. 143-159.

Vogel, P. & Grotherr, C., 2020. *Collaborating with the Crowd for Software Requirements Engineering: A Literature Review, In The Proceedings of TheAmericas Conference On Information Systems (AMCIS), Association for Information Systems* Virtual.

Wang, C., Daneva, M., van Sinderen, M. & Liang, P., 2019. A Systematic Mapping Study on Crowdsourced Requirements Engineering Using User Feedback. *Journal of Software: Evolution & Process, Wiley,* pp. 1-20.

White, B., 2007. The Implications of Web 2.0 on Web Information Systems. In: *Web Information Systems and Technologies. Lecture Notes in Business Information Processing.* Berlin: Springer, pp. 3-7.

Wohlin, C. et al., 2012. *Experimentation in Software Engineering.* Springer.

Zanatta, A. L. et al., 2016. Software crowdsourcing platforms. *IEEE Software, IEEE,*33(6), pp. 112-116.

Zhao , L. & Zhao, A., 2019. Sentiment Analysis Based Requirement Evolution Prediction. *Future Internet MDPI,* 11(52).

Zowghi , D. & Coulin, C., 2005. Requirements Elicitation: A Survey of Techniques,Approaches, and Tools. In: *Engineering and Managing Software Requirements.* Berlin: Springer, pp. 19-46.

Appendix A

Here is the Acceptance from G2 to send me their Data illustrated in Figure A-1.

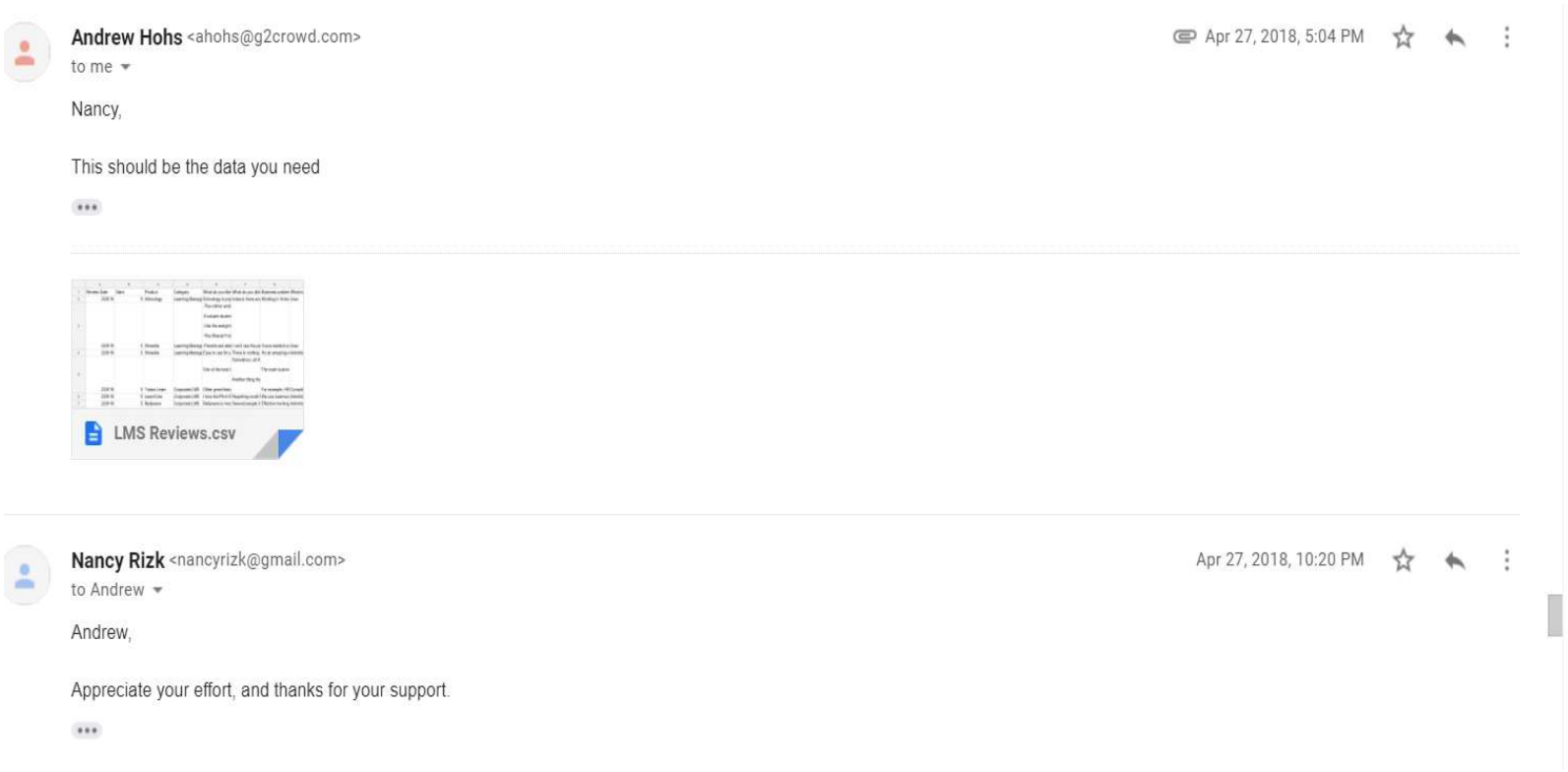


Figure A-1. G2 Acceptance email

Below is an acknowledgment email to G2 team, illustrated in Figure A-2 .

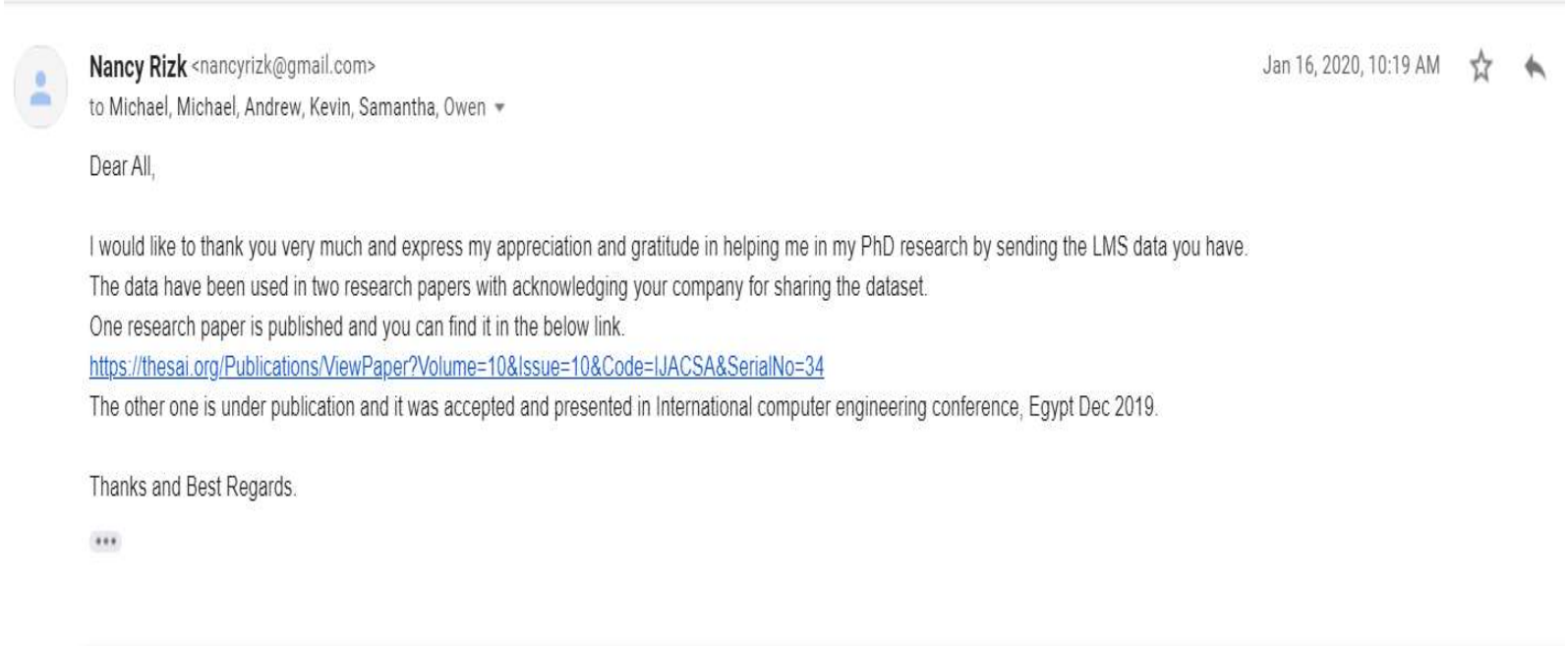


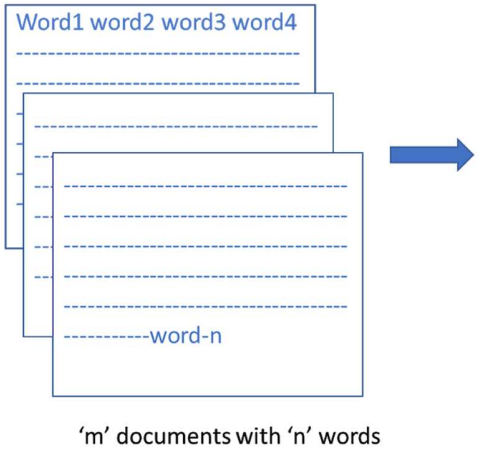
Figure A-2. Our acknowledgment to G2

Appendix B

B.1 Latent Dirichlet Allocation (LDA)

LDA is one of the most popular topic modeling methods. Each document is made up of various words, and each topic also has various words belonging to it. The aim of LDA is to find topics a document belongs to, based on the words in it (Blei, et al., 2003). The word 'Latent' indicates that the model discovers the 'yet-to-be-found' or hidden topics from the documents. 'Dirichlet' indicates LDA's assumption that the distribution of topics in a document and the distribution of words in topics are both Dirichlet distributions. 'Allocation' indicates the distribution of topics in the document.

LDA assumes that documents are composed of words that help determine the topics and maps documents to a list of topics by assigning each word in the document to different topics. The assignment is in terms of conditional probability estimates. In the figure, the value in each cell indicates the probability of a word wj belonging to topic tk. 'j' and 'k' are the word and topic indices respectively. It is important to note that LDA ignores the order of occurrence of words and the syntactic information. It treats documents just as a collection of words or a bag of words.



	word1	word2	word3				Word-n
Topic-1	0.024	0.012	0.014	-	-	-	0.086
Topic-2	0.026	0.186	0.164	-	-	-	0.194
Topic-3	0.018	0.112	0.192	-	-	-	0.028
|	-	-	-	-	-	-	-
Topic-K	0.128	0.144	0.084	-	-	-	0.036

Figure B-1. Probability estimates for topic assignment to words.

Once the probabilities are estimated (we will get to how these are estimated shortly), finding the collection of words that represent a given topic can be done either by picking top 'r' probabilities of words or by setting a threshold for probability and picking only the words whose probabilities are greater than or equal to the threshold value. For instance, if we focus on topic-1 in and pick top 4 probabilities assuming that the probabilities of the words are less than 0.012, then topic-1 can be represented as shown below using the 'r' top probabilities words approach.

In the above example, if word-k, word1, word3 and word2 are respectively trees, mountains, rivers and streams then topic-1 could correspond to 'nature'.

One of the important inputs to LDA is the number of expected topics in the documents. In the above example the expected topics is set to 3, each document can be represented as shown below.

$$D_i = w_{1_i} \times Topic - 1 + w_{2_i} \times Topic - 2 + w_{3_i} \times Tpoic - 3$$

In the above representation**,** there are the three weights for topics: topic-1, topic-2 and topic-3 respectively for a given document. Indicates the proportion of words in document that represent topic-1, indicates the proportion of words in document that represent topic-2 and so on.

B.2 LDA Algorithm

LDA assumes that each document is generated by a statistical generative process. That is, each document is a mix of topics, and each topic is a mix of words. For example, Figure B-2 shows a document with ten different words. This document could be assumed to be a mix of three topics; tourism, facilities and feedback. Each of these topics, in turn, is a mix of different collections of words. In the process of generating this document, first, a topic is selected from the document-topic distribution and later, from the selected topic, a word is selected from the multinomial topic-word distributions.

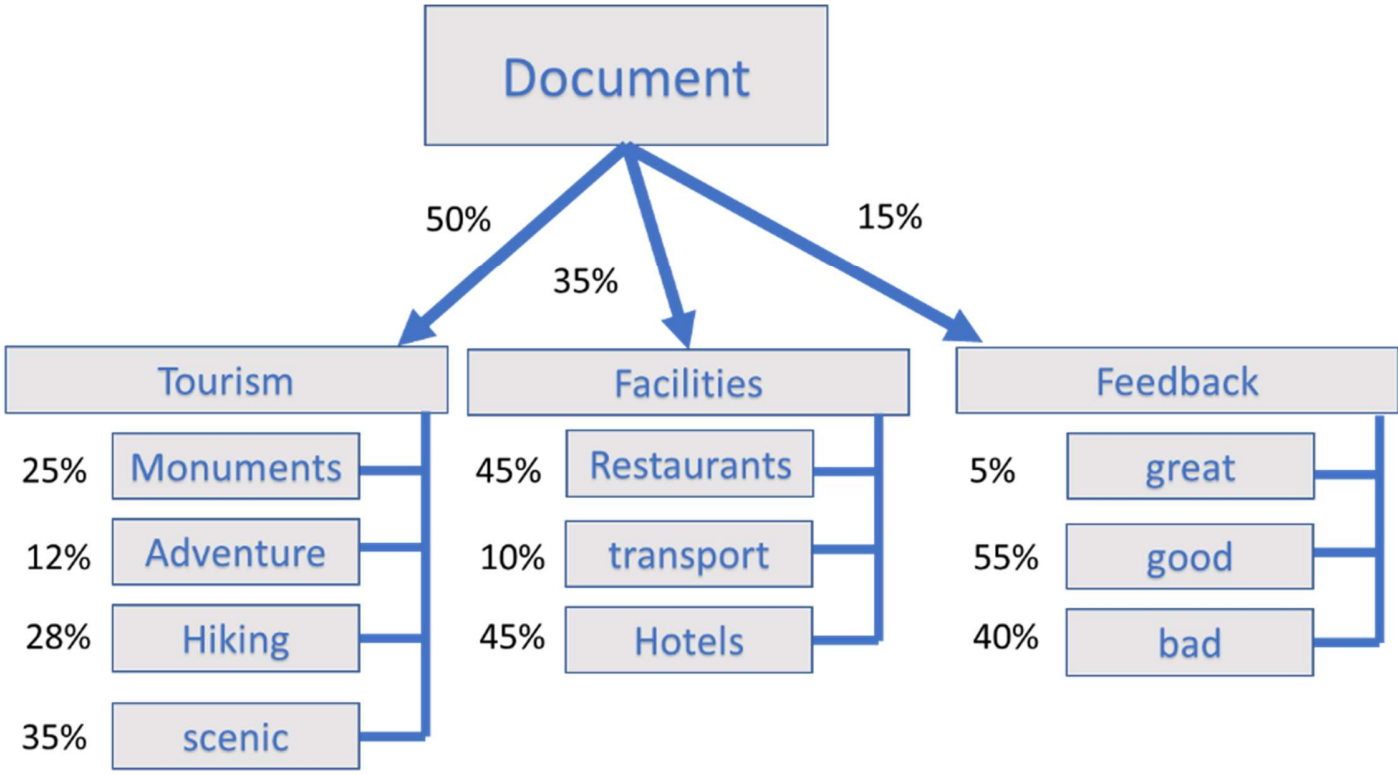


Figure B-2. Document generation assumption

While identifying the topics in the documents, LDA does the opposite of the generation process. The general steps involved in the process are shown in Figure B-3. It's important to note that LDA begins with random assignment of topics to each word and iteratively improves the assignment of topics to words through Gibbs sampling.

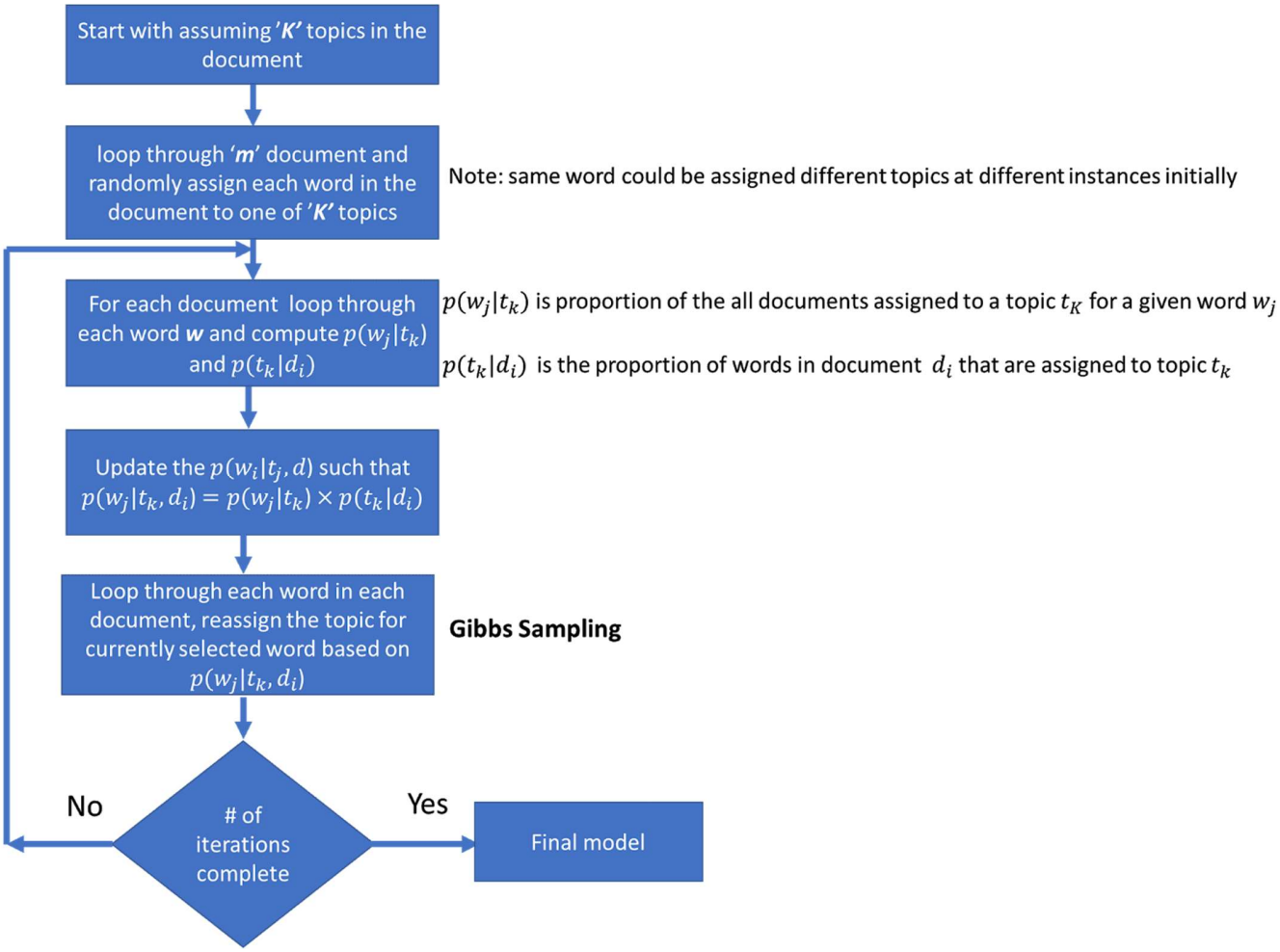


Figure B-3. General steps in LDA

Figures B-4 and B-5 are explaining LDA by considering a corpus of 'm' documents with five words vocabulary according to (Sharma, 2020).

B.3 Hyper parameters in LDA

LDA has three hyper parameters: 1) document-topic density factor 'α', shown in step 7 of Figure C-5, topic-word density factor 'β', shown in step-8 of Figure C-5) the number of topics 'K' to be considered.

The 'α' hyperparameter controls the number of topics expected in the document. Low value of 'α' is used to imply that fewer number of topics in the mix is expected and a

higher value implies that one would expect the documents to have higher number topics in the mix.

The ‘β’ hyper parameter controls the distribution of words per topic. At lower values of ‘β’, the topics will likely have fewer words and at higher values topics will likely have more words. Ideally, it is likely to see a few topics in each document and few words in each of the topics. So, α and β are typically set below one. The ‘K’ hyperparameter specifies the number of topics expected in the corpus of documents. Choosing a value for K is generally based on domain knowledge. An alternate way is to train different LDA models with different numbers of K values and compute the ‘Coherence Score’. Choose the value of K for which the coherence score is highest.

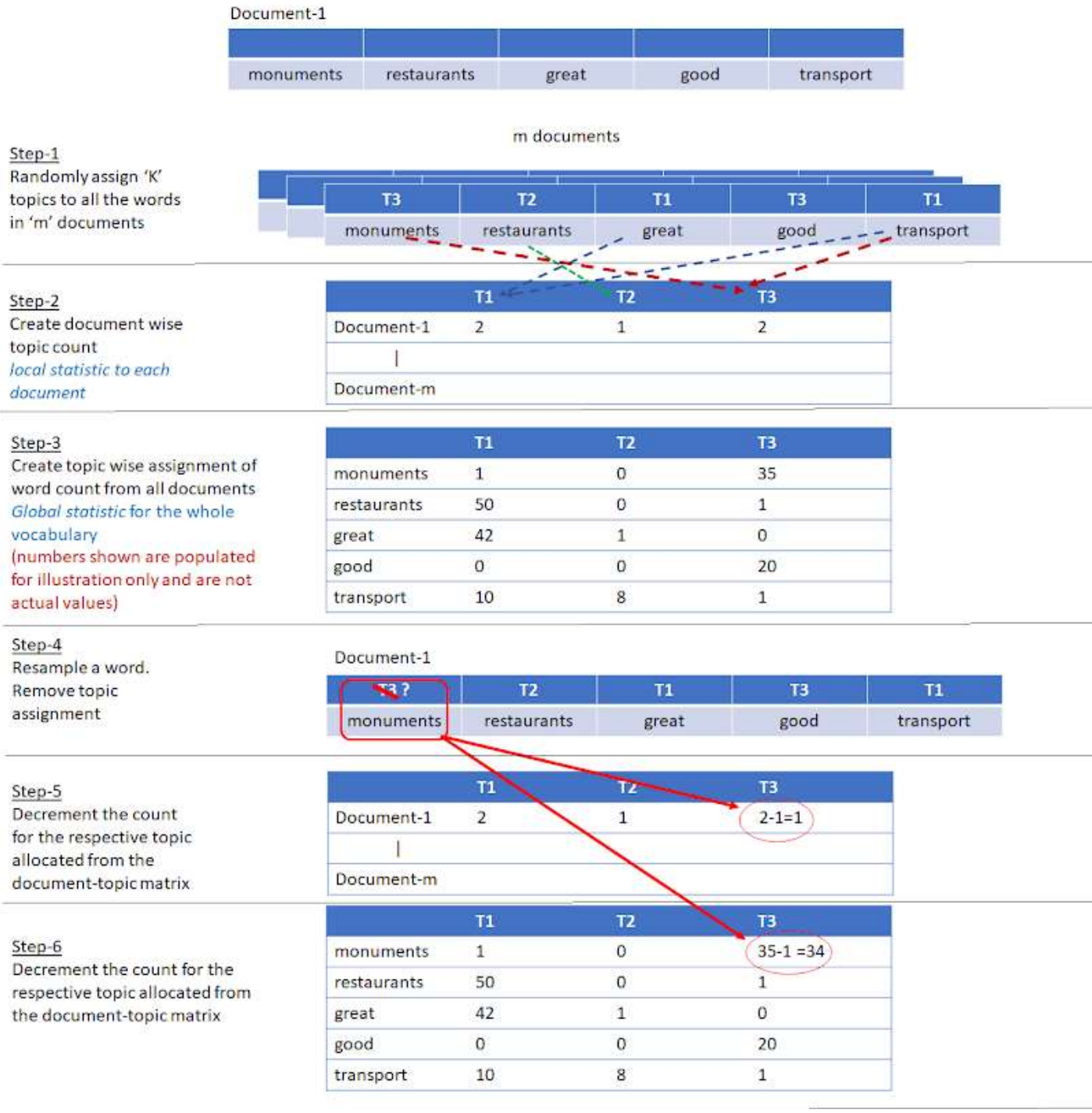


Figure B-4. LDA example (Part 1)

Step-7
calculate
$p(t_k|d_i)$
Indicates how much document d_i likes topic t_k

$$p(t_k|d_i) = \frac{n_{ik} + \alpha}{N_i - 1 + K\alpha}$$

	T1	T2	T3
Document-1	2	1	2-1=1
|			
Document-m			

Where n_{ik} is the total number of words in 'i^{th}' document in 'k^{th}' topic, N_i is the number of words in the 'i^{th}' document, K is the number of topics considered. α is a hyper parameter

For instance, for document-1 and topic-3, $p(t_k|d_i) = \frac{1+0.1}{5-1+3\times0.1} = 0.155$ assuming α = 0.1

Step-8
calculate
$p(w_j|t_k)$
Indicates how much topic t_k likes word w_j

$$p(w_j|t_k) = \frac{m_{j,k} + \beta}{\sum_{j \in V} m_{j,k} + V\beta}$$

	T1	T2	T3
monuments	1	0	35-1 =34
restaurants	50	0	1
great	42	1	0
good	0	0	20
transport	10	8	1

Where $m_{j,k}$ is the corpus wide assignment of word wj to k^{th} topic. V is the vocabulary of the corpus. β is a hyper parameter

For instance, for the word monument in topic-3, $p(w_j|t_k) = \frac{34+0.5}{56+5\times0.5} = 0.56$ assuming β = 0.5

Step-9
calculate for word w_j $p(w_j|t_k,d_i)$

$$p(w_j|t_k,d_i) = p(t_k|d_i) \times p(w_j|t_k)$$

Step-10
Resampling/ Reassign

For a given word w_j in a document d_i, find the topic 'k' for which $p(w_j|t_k,d_i)$ is maximum and reassign the word to the 'k^{th}' topic

For instance, for the word 'monument' in 'document-1' if $P(monument|T2, Document-1) = 0.4$, the highest value, then reassign the word 'monument' to 'Topic-2'.

T2	T2	T1	T3	T1
monuments	restaurants	great	good	transport

Step-11
Repeat for all

Repeat steps 4 to 10 for all the words in all the document

Step-12
Repeat process

Repeat steps 2 to 11 for a predefined number of iterations

Figure B-5. LDA example (Part 2)

الملخص

إكتسب التعليم الإلكتروني أهمية كبيرة، وأصبح أكثر شعبية في الوقت الحاضر. تفشي جائحة فيروس كورونا (كوفيد -19) في جميع أنحاء العالم أجبر المؤسسات التعليمية على الإغلاق للسيطرة على إنتشار الفيروس. هذا الحدث الزم بانتشار التعليم الإلكتروني أو التعليم عبر الإنترنت الذي يتصل فيه المعلمون والطلاب إفتراضيا. وأصبح التعليم الإلكتروني إلزاميًا لجميع المؤسسات التعليمية مثل المدارس والكليات والجامعات في جميع أنحاء العالم .لأنظمة التعليم الإلكتروني طبيعة مميزة ولها خصائص خاصة مثل العدد الكبير للمستخدمين وتنوعهم وتشتتهم جغرافيا. تحتاج أنظمة التعليم الإلكتروني باستمرار إلى التحسينات لتلبية متطلبات المستخدمين. ولذلك فإن نشاط إستخلاص المتطلبات لها أهمية كبرى، ويعد نشاط إستخلاص المتطلبات من ضمن مراحل عملية هندسة المتطلبات مرحلة هندسة المتطلبات التي تعد مرحلة مبكرة في عملية هندسة البرمجيات، التي يستمد منها متطلبات النظام وتطورها. يجب على مهندسي المتطلبات الإنتباه إلى المتطلبات الخاصة بأنظمة التعليم الإلكتروني في مراحل تطوير البرامج وخاصة مرحلة إستخلاص المتطلبات من أجل تلبية إحتياجات أصحاب المصلحة بشكل أفضل وسريع. تواجه التقنيات التقليدية لإستخلاص المتطلبات التي يستخدم فيها تقنيات مختلفة مثل المقابلات، والإستبيان، والملاحظة.بعض المشاكل مثل عدم المشاركة الكافية للمستخدمين وأصحاب المصلحة وخصوصا المنتشرين جغرافيا. لإستخلاص المتطلبات العديد من القيود ولا يمكنها تلبية الإحتياجات المتغيرة والمستمرة والسريعة لمستخدمي التعليم الإلكتروني. مفهوم مصادر الحشد من الممكن أن يكون حلا لهذه المشكلة. يعد إستخدام مصادر الحشد أحد الطرق الجديدة المستخدمة لحل مشاكل مناهج استخلاص المتطلبات الحالية. مصادر الحشد هي عملية مهمة يقوم بها الحشد؛ ويعد أحد الطرق الجديدة المستخدمة في حل المشاكل المتعلقة بالمناهج الحالية لإستخلاص المتطلبات. يتم إستخدام مصادر الحشد في مجال هندسة المتطلبات كمفهوم ناشئ للمساعدة في تبسيط أنشطتها. في حدود ما نعرفه هناك القليل من الأدلة على وجود نهج أو طريقة لاستخلاص المتطلبات بحيث يكون قائم على إستخدام مصادر الحشد أو مصمم خصيصا لأنظمة التعليم الإلكتروني ويتناول خصائصها الخاصة. في هذا البحث نحاول سد هذه الفجوة. في هذا البحث نقدم طريقة جديدة لإستخلاص متطلبات نظم التعليم الإلكتروني تعتمد على مصادر الحشد CREeLS، وتتكون من إطار ومجموعه من المراحل. تأخذ الطريقة الجديدة بعين الإعتبار الطبيعة الخاصة لنظم التعليم الالكتروني أثناء نشاط إستخلاص المتطلبات. يتكون الإطار من العناصر الضرورية للإستعانة بمصادر الحشد والعناصر اللازمة هي الحشد ، وتعليقات المستخدمين ، والتفاعل ، وأدوات التنقيب عن النصوص ، والتعاون الإجتماعي. ولكل مصدر من مصادر الحشد في إطار العمل أداة مقترحة لإستخدامها لتنفيذ مفهوم مصادر الحشد.أما

المراحل فتهدف إلى تنفيذ الإطار وتتكون من خمس مراحل وهي: 1-إنشاء قناة للمستخدمين لنشر ملاحظاتهم أو إظهار تفاعلاتهم. 2-استخراج تفاعلات المستخدمين أو تعليقاتهم. 3-تحليل تفاعلات المستخدمين أو تعليقاتهم. 4- تطوير وتحسين متطلبات البرامج. 5-تصنيف وتوحيد المتطلبات. قمنا بتقييم الطريقة المقترحة من خلال تحليل لحوالي 4000 من آراء واقعيه للمستخدمين كل من الثلاثة أنظمة لإدارة التعليم وقد تم إستخلاص الكلمات المفتاحية التي تمثل متطلبات المستخدمين باستخدام تقنيات نمذجة الموضوعات. أولا إستخدمنا خوارزمية Latent Drichlet أحادية الكلمات. تم تقييم النتائج من خلال المراجعة اليدوية لنصوص آراء المستخدمين، وقد تبين أن الصفات المستخرجة ذات علاقة. حصل هذا النهج على متوسط دقة 0.79 واستدعاء 0.44 و 0.56 قياس f. ثم قمنا بعد ذلك بتقييم الطريقة مرة أخرى لتحسين النتائج باستخدام خوارزمية ثنائية الكلمات Latent Drichlet لإستخراج متطلبات مستخدمي التعليم الإلكتروني بشكل أفضل والمساعدة في إستخلاص متطلبات وتطور نظم التعليم الإلكتروني. تم إستخدام تقييم الكلمات الثنائية على نفس أنظمة التعليم الإلكتروني المستخدمة في تقييم الكلمات الأحادية وعلى نفس عدد آراء المستخدمين وقمنا أيضا بمقارنة النتائج بما تم الحصول عليه بعد الاستخراج اليدوي للمتطلبات. وكان متوسط النتائج 0.76 دقة وإستدعاء 0.61 و 0.68 قياس f. تمت مقارنة هذه النتائج بالنتائج المنشورة لبعض الأعمال ذات الصلة التي تطبق تقريبًا نفس تقنيات وتدابير التقييم المستخدمة في CREeLS وكانت مقاربه لها. كما أن المتطلبات المستخرجة تعد أكثر قابلية للفهم وذات صلة. وبالتالي فإننا نعتبر أن هذه الطريقة يمكن أن تساعد مهندسي متطلبات نظم التعليم الإلكتروني لتحليل آراء المستخدمين وتحديد متطلباتهم الأكثر شيوعًا لتطوير أفضل للبرامج.

جامعة القاهرة

كلية الدراسات العليا للبحوث الاحصائية
قسم تكنولوجيا نظم المعلومات

طريقة جديدة لاستخدام مصادر الحشد لاستخلاص متطلبات أنظمة التعليم الإلكتروني

هذه الرسالة قدمت إلى قسم تكنولوجيا نظم المعلومات كجزء مكمل متطلب لنيل درجة الدكتوراه في تخصص نظم المعلومات

إعداد
نانسي محمد رزق

تحت إشراف
د. ميرفت غيث
د. إيمان نصر

2020