

## AUTOMATIC CLASSIFICATION AND EXTRACTION OF NON-FUNCTIONAL REQUIREMENTS FROM TEXT FILES: A SUPERVISED LEARNING APPROACH

Vatchala. S<sup>a</sup>, Bingi Manorama Devi<sup>b</sup>, M. Sharmila Devi<sup>c</sup>, Sathish. A<sup>d</sup>

<sup>a</sup>Research Scholar, Anna University, Chennai, India

<sup>b</sup>Assistant Professor, Department of CSE, KSRM College of Engineering, Kadapa, AP, India.

<sup>c</sup>Assistant Professor, Department of CSE, Santhiram Engineering College, Nandyal, AP, India

<sup>d</sup>Research Scholar, Anna University, Chennai, India

### Address for Correspondence

**Vatchala. S**

Research Scholar, Anna University, Chennai, India

**Abstract:** Non-functional requirements play a critical role in choosing various alternative model and ultimate implementation criteria. It is extremely significant in the earlier stages of software development that requirement engineering produces successful technology and eliminates system failure. The recent work has shown that the automated extraction and classification of quality attributes from text files have been demonstrated by artificial intelligence approaches including machine learning and text mining. In the automated extraction and classification of non-functional specifications, we suggest a supervised categorization approach. To test our approach to obtain interesting outcomes, a very well-known dataset is used. In terms of security and performance, we obtained a specific range of 85% to 98% and obtained a best result together for security, performance and usability.

**Keywords:** *Non Functional requirement, Machine Learning, Artificial intelligence.*

## 1 Introduction

The non-functional requirements describe significant software development and software system behavior. They offer a variety of quality characteristics for developed software. Such characteristics play an important part in architectural design and should therefore be taken into account as early as possible during the review process. A basic question is how to find out what users really need in Requirements Engineering (RE). A vital aspect of software development projects is the achievement of the required specifications. The most important of the RE-recommended activities that resulted are the successful elicitation of requirements. The application of software engineering artificial intelligence techniques (AI&SE) has provided some promising results. Over the last years, research has shown that artificial intelligence automates the extraction and classification through techniques, including machine learning and text mining, of quality attributes from text documents. The most widely cited data collection has currently

been derived from the 15 specifications of master's degree (M) projects for requirement engineering at DePaul University and first appears in 2006 literature for automated classification of non-functional criteria (NFR) with supervised learning techniques. In the course of writing the specifications, NFR analysis was carried out by the students themselves and the method of identifying the data set missed logic.

## 2. Related work

The literature survey is mainly done by taking deep insight into different literature sources and the following literature review is focus on different parameters and methods that are used extraction and classification of NFR from text files: a supervised learning approach. Zhang, Wen, et al [2] Throughout this analysis, vector space and machine learning techniques were implemented to automatically classify NFR, to convert NFRs into numerical vectors of specific index words and to analyse their output in the automatic classification of NFRs. The machine learning classification used in this journal is Support vector machine with a linear kernel, a classification that is strongly advisable for text mining. Additional information is provided for choosing apps and Support vector machines for automated sorting. Experiments have been carried out using the PROMISE data set obtained.

Jane Cleland-Huang et al [3] A modern methodology has been implemented focused on knowledge recovery approaches to identify and recognize non-functional criteria, both in formal specifications and free text formats. Even though an observer is always expected to determine the consistency of the nominee NFR, it may involve less work as for example the Theme / Doc approach than before semi-automated classification techniques. It's helpful for software developers to plan, develop and evaluate a software framework to be able to distinguish NFRs in this manner, as this will allow them to consider stakeholders' desires and to see consistency between different device restrictions. The NFRs may be identified by and marked as intrusiveness in architectural design meetings or collected from a security expert by evaluating particular security issues, while the applicant can receive interviews or minutes from stakeholders. Aspects of applicants should be determined early in the selection phase, thereby raising the expense of recovery measures which might otherwise have existed. For example, an emerging method built at the University of DePaul integrates NFR identification into a shared modeling framework for a broad variety of non-functional objectives. The NFR method perspectives provide important insight into the modeling phase.

Slankas et al [1] Presented the latest NFR Locator method and a resource to support researchers retrieve specific non-functional requirements easily from a broad range of records. A collection of system-related records in the health care sector for NFR detection and extraction have been compiled to test the device.

### 3. Methodology

In an actual world scenario, the research team in the field of software development will review all necessary documents collected by an application elicitation team in order to decide whether a software development project is prioritized and mapped with the most important category or classes of NFR (e.g. security availability, scalability, etc. This method takes time and analysts need to collaborate as each application document needs to be interpreted and classified manually. For detection and classification of non-functional requirements in a series of defined categories or groups a supervised text- approach is suggested.

We establish a structural and light-weight mapping method to support the development of a SIG catalog, following the guidance and guidelines from Petersen, Vakkalanka and Kuzney. The feasibility of classification of non-functional requirements within the proposed dataset was evaluated and categorized. The processing of the data is facilitated by the data set. The data set is established. There are three tasks involved in this systematic mapping: i)Planning of a search string ii) defining the filtration criteria and iii) Analyzing papers. It is as follows:Search string planning, the purpose of this project is to develop check list for indexing research document sites used in NFR catalog articles. The selected portals are the IEEE and Science Direct. We use search string specification keywords for NFR catalogues like "NFR Framework" and "Chung" and Paper review we integrate this with certain types of non-functional criteria such as "non-function parameters" and "non-functional," in order to achieve a search string version 1 (see Table 1).

When we first looked at portals, we noticed articles that contained NFR catalogs and used index words on the web chain to create a new edition of the application channel 2 (see Table I). The number of papers found after search definition has increased, because the search chain has increased. In the NFR Catalog division, we introduced terms expanding the spectrum of scope, including: SIG and catalogues.

**Table 1.** Search string evolution

<b>Search String V1</b>	<b>portal</b>	<b>Papers</b>
("nfr" or "nfr framework" or "chung") or ("non-functional Requirements" or "non-functional")	IEEE	1553
	Science direct	1045
	Total	2598
<b>Search String V2</b>		

("nfr" or "nfrframework" or "chung" or "sig" or "catalogue") and ("non- functional requirements" or "non- Functional" or "quality attribute"	IEEE	2882
	Science direct	3754
	Total	6636

The of this

purpose

experiment is to finding filters for indexing portals to enhance content in identified papers and eventually to create standards for inclusion and exclusion after search strings have been drawn up for review. We used two filters in the mapping test; the first one was year round. Between 2000 to 2018 we only chose papers because in 2000 the NFR catalogs were introduced. The second filter applies to Appendix A's conventions and papers.

**Table 2.** Result after filters

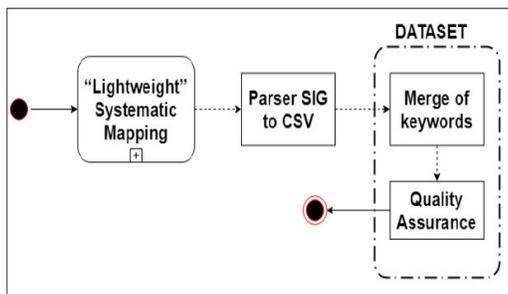
Filter	Portals	Papers
Year+	IEEE	193
Conferences	Science Direct	392
And Journals	Total papers	585

The number of publications dropped from 6636 to 585 after filters were added. (see Table 2). The next move is to introduce the criterion for incorporation and elimination. The minimum requirement was to include the document as NFR catalogs to be included and thus the rejection condition is omission. The object of the inclusion and exclusion criterion is to choose the documents to be included during the paper review stage.

The last phase of the analysis is the paper. This research aims to analyze the results of the dependent study, which discuss the consistency of the NFR catalogs in every document. We have chosen a selection of 20 articles for this lightweight systematic mapping. The documents were picked because the NFR catalogs represent a number of soft targets (in terms of security, performance and usability). In terms of the table, we prefer exceptionally informative SIG catalogues. This makes a broader variety of words in our dataset.

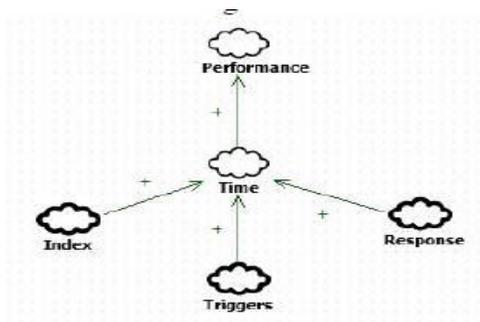
Once the comprehensive "lightweight" mapping has been done, we build a data set that collects keywords in 31 catalogs. In the purpose of creating a data collection there were two activities. The first step is to categorize keywords. The category of database specified each keyword in the NFR catalogs.

For eg, keywords were connected in many protection catalogs to the exclusion of repetitive keywords: the keyword is sensitive so a keyword for our data set was only entered once. For example, a keyword was added for several of our data catalogs. The final dataset comprises 77 words divided into: usability of 28 words, security of 24 words and performance of 25 words see Table 6. For RQ2, a systemic process has been built for defining the dataset consisting essentially of four activities according to Figure 2.



**Fig 1.**The Dataset Definition Process.

This type of method has four operations: i) Systematic "lightweight" process visualization, (ii) Converting the sig parser into csv format (iii) merging keywords (iv) Level of quality assurance. The first element of the hierarchical approach was previously described in which we showed in the systematic mapping review the "lightweight" of the three operations. The next move was to create a CSV parser SIG, which should be translated into a CSV format as the principal objective of this project.

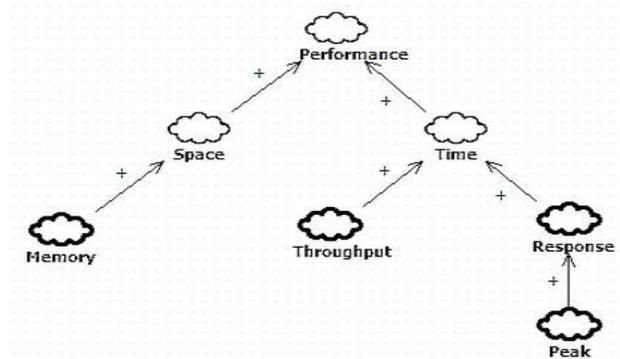


**Fig 3.**SIG Catalogue Performance

**Table 3.** Performance keywords

NFR Type	Keywords
Performance	Performance, time, index, triggers, response

These SIG catalogs were produced using StarUML3 and RE-Tools. For each SIG catalog a parser was developed and implemented. The method is configured in three phases: (i) determine the SIG root objective of the NFR category that the catalog contains. (ii) After that the NFR was classified by all keywords found in its branches. And (iii) Creating a CSV file with keywords is the final step. Table 3 and Table 4 shows the collection of keywords produced during this procedure and the two product catalogs, as seen in Figures 3 and 4.



**Fig 4.** SIG Catalogue Performance

**Table 4.** Performance keywords

NFR Type	Keywords
Performance	Performance, space, time, memory, Throughput, response, peak

Keyword combine is the third process. This method mostly includes generating a CSV file containing a collection of numerous keywords for each form of NFR from the previous level. Keywords: answer and time won't be included in the new set of keywords, as they have been introduced before. Table 5 shows the final collection of keywords after the merge phase.

**Table 5.** Performance keywords merge

NFR Type	Keywords
Performance	Performance, space, memory, time, throughput, reponse, Triggers, index, peak

Table 6 shows the collection of keywords contained after operations ii and iii in all the SIG catalogs. Following the fusion phase in the 24 security catalogues, Fig5 displays the actual SIG catalog.

**Table 6.** Keywords extracted from sig catalogues

NFR Type	Keywords
performance	performance, space, time, throughput, response, memory, consumption, fast, index, triggers, storage, low, run, runtime, perform, execute, mean, peak, compress, dynamic, offset, reduce, fixing, early, processing
Security	security, confidentiality, integrity, availability, accuracy, completeness, secure, access, registration, , authorization, identification, authentication, validation, transaction, user, password, control, encryption, key, spoofing, attack, policy, logging, permission.
Usability	Utility, Homogeneity, Easiness, Operationability, intuitively, adaptable, Understandability, accessible, Configuration, Integrity, Administration, Conformity, cognizance, applicable, Linguistics, supportive, tutorials, trainable, helping, flexible, easy, usable, Graphics, timing.

The last process in the data selection system is quality assurance. (Figure 2). The first dataset collected in this operation is where the efficiency of research is checked and the classification accuracy is observed. The consequence of this method is the coherence of the terms collected in such catalogues. They use an adjective that makes sense in a collective setting, to which the word refers, for every phrase in the foundation. For examples, the term convenience generates adjectives in the category of usability: simple and quick; for security category we obtain adjectives such as privacy and secrecy use the word confidentiality.

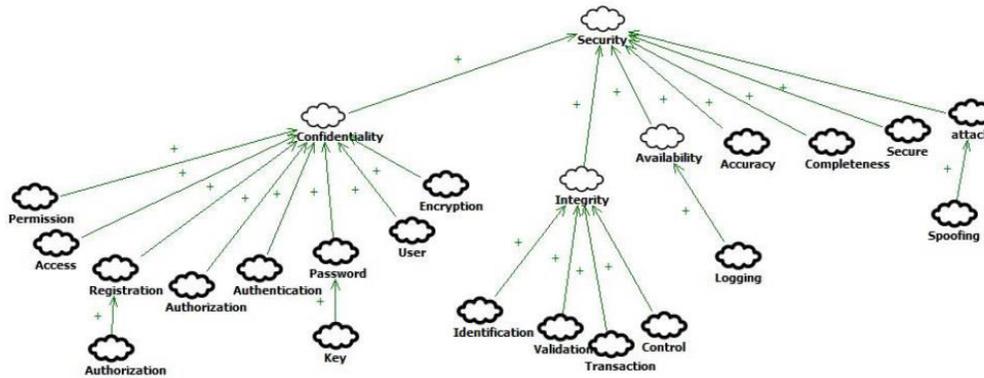


Fig 5 : The final SIG Security Catalogue.

#### 4. Dataset

For the assessment of the proposed technique a previously identified dataset with non-functional criteria is needed. We have chosen an Open Science Tera-PROMISE dataset comprising 625 specifications, either functionally or not. It is described in this paper as a promise dataset. Dividing into 11 types of non- criteria. Such parameters were originally obtained from 15 project specifications of a master degree in engineering requirements from DePaul University. An initial non-criteria selection of more phrases per group was carried out to verify the strategy: security (66 phrases), performance (54 phrases), usability (67 phrases) and operational (62 phrases). The operating class is omitted since there is no relation to an existing NFR catalog. The result was 187 phrases regarding security, performance and usability.

#### 5. Conclusion

The right requirements are considered an important and complex step in the creation of software projects. The work has demonstrated in recent years that artificial technology automatically extracts and classifies the content properties of textual records utilizing strategies including machine learning and text mining.

This paper introduces a new way of defining non-functional criteria using NFR catalogs by means of machine learning algorithms and offers a way of accessing such catalogs utilizing the structural "light weight mapping" technique.

One of the massive problems to solve this step searching is trying to define the training data set performing the criteria manual assignment, which will increase effort. Such classification distinguishes between the generated data, such that machine learning is still partial because the managed algorithms require such dataset to be categorized. Their analysis provides a way of generating data sets used to classify non-functional specifications by NFR catalogs extracted from the research of mapping in order to address this circumstance.

## References

- 1 Slankas, John, and Laurie Williams. "Automated extraction of non-functional requirements in available documentation." 2013 1st International Workshop on Natural Language Analysis in Software Engineering (NaturaLiSE). IEEE, 2013
- 2 Zhang, Wen, et al. "An empirical study on classification of non-functional requirements." The twenty-third international conference on software engineering and knowledge engineering (SEKE 2011). 2011.
- 3 J. Cleland-Huang, R. Settimi, X. Zou, and P. Solc. The detection and classification of non-functional requirements with application to early aspects. In Requirements Engineering, 14th IEEE International Conference. IEEE, 2006.