A Systematic Review on Natural Language Processing and Machine Learning Approaches to Improve Requirements Specification in Software Requirements Engineering

Khaleduzzaman[1], Zarina Che Embi[2*], Ng Kok Why[3]

[1,2,3]*Faculty of Computing and Informatics, Multimedia University, 63100 Cyberjaya, Selangor, Malaysia; E-mail: zarina.embi@mmu.edu.my*

Abstracts: This systematic literature review (SLR) examines the current practices, challenges, proposed solutions, and limitations of natural language processing (NLP) and machine learning (ML) approaches in improving requirements specification in software requirements engineering. The review focuses on research conducted in the last five years and includes a selection of papers that discuss the use of NLP and ML techniques for enhancing the accuracy and clarity of requirements, particularly in the context of functional and non-functional requirements. The findings highlight the benefits and challenges associated with the integration of NLP and ML approaches, such as improved classification and identification of requirements. However, it is observed that there is a greater emphasis on non-functional requirements, with a limited representation of research on functional requirements. Comparison of this review and the recent two reviews has been done to observe the differences and highlight the novelty and contribution. The review also identifies limitations, potential bias in assuming that problems related to requirements documentation or specification can be easily resolved through simple changes as well as the need to address the functional requirements. The insights from this SLR contribute to the understanding of the current state of research in this field and provide a foundation for future research directions and practical applications in leveraging NLP and ML approaches to enhance requirements specification in software requirements engineering.

Keywords: Software Requirements Engineering, Requirements Specification, Natural Language Processing (NLP), Machine Learning (ML), Systematic Review.

1. INTRODUCTION

Software requirements engineering plays a critical role in the success of software development projects. Requirements specification serves as the basis for the entire software development process. Ensuring the accuracy, completeness, and unambiguity of the requirements specification is crucial to deliver high-quality software systems that meet stakeholder expectations. However, a significant challenge in software requirements engineering arises from the expression of requirements in natural language, which can be inherently ambiguous and lead to errors throughout the software development lifecycle. NLP and ML approaches can be used to address this challenge and enhance software requirements engineering (Dalpiaz, Ferrari, Franch & Palomares, 2018a).

In this SLR, we conducted a comprehensive analysis of relevant research papers on NLP and ML approaches for improving requirements specification in software engineering. Our aim was to identify the main areas of focus, methodologies, and findings in the selected papers. We started by compiling a diverse set of papers from reputable conferences and journals that specifically addressed the application of NLP and ML techniques in requirements specification. Each paper was carefully examined to extract key information related to the research objectives, methodologies used, and the outcomes achieved.

Our analysis revealed several common themes and research areas across the selected papers. These included the classification and identification of functional and non-functional requirements, automatic detection of ambiguous requirements, detection of requirements smells, end-to-end deep learning systems for requirements classification, and the use of ensemble learning and various hidden layers in software requirements classification.

The methodologies employed in the papers varied, with recurrent neural networks (RNNs), neural networks, deep learning, and supervised machine learning algorithms being the most commonly utilized approaches. The papers also discussed the use of different evaluation metrics to assess the effectiveness and performance of the proposed NLP and ML models. By synthesizing the findings from the selected papers, we gained valuable insights into the advancements, limitations, and challenges in the field of NLP and ML approaches for requirements specification. We identified the potential benefits of these approaches, such as improved accuracy, efficiency, and automation of the requirements classification process. However,we also noted limitations, including the bias towards non-functional requirements and the need to consider the diverse contexts of software development teams.

Overall, this SLR provides a comprehensive overview of the research conducted in the field of NLP andML approaches for requirements specification. The findings from this review contribute to the existing body of knowledge and can guide future research efforts in leveraging NLP and ML techniques to enhance the process of requirements specification in software engineering.

2. METHODS

This systematic literature review follows the guidelines proposed by Kitchenham and fellow researchers (2004) which follows a step-by-step approach outlined in the following sections.

2.1. Research Questions

RQ1: What is the current state of research on the use of NLP and machine learning approaches to improverequirements specification in software requirements engineering?

RQ2: Based on the findings of the systematic literature review in RQ1, what are the common NLP techniques and machine learning algorithms employed in the literature to enhance the accuracy and clarityof requirements specification in software requirements engineering?

RQ3: Building upon the results of RQ1, what are the identified benefits and challenges associated with the integration of NLP and machine learning approaches in requirements specification, as reported in the existing literature?

RQ4: How can the findings from the systematic literature review, as explored in RQ1, inform future research directions and practical applications in utilizing NLP and machine learning approaches to improve requirements specification in software requirements engineering?

The materials reviewed in this study for RQ1-RQ4 are recent publications from the past five years (since 2018). As the software development industry evolves quickly, it's important to consider the most up-to-date practices and tools used by Machine Learning Engineer (MLE) teams for requirements engineering. By focusing on recent sources, this review aims to provide accessible insights into the latest advancements and practices in requirements engineering, tailored to the needs of software development professionals.

2.2. Search Process

The systematic literature review conducted for this study involved a manual search of conference proceedings and journal papers published since 2018. Due to time limitations and resource constraints, the review was restricted to the preliminary search results from the first few pages of Scopus, focusing on relevant publications from reputable publishers such as IEEE, Springer, Cambridge, Elsevier and ACM. The search was performed using a combination of keywords, including "functional requirements," "non- functional requirements," "software specifications requirements," "requirement engineering," "machine learning," and

"NLP," to optimize search accuracy (Table 1 referred).

The selection process involved reviewing the title, abstract, and conclusion of each publication to assessits relevance to the study. Specifically, publications that discussed the intersection of functional and non-functional requirements, the current state of practice in machine learning development and the challenges and consequences faced by machine learning teams were considered for inclusion in the review.

It is important to note that the review process was limited in scope, and only a subset of available literature was analyzed. Additionally, the search results may not encompass the entirety of relevant publications in the field. Future studies could benefit from broader search strategies and more extensive review capabilities to ensure a comprehensive analysis of the literature.

Table 1. Search Queries (SQ) on Scopus

No	Search queries
1	“Functional requirements” software specifications requirements
2	“Non-functional requirements” software specifications requirements
3	Approach “Machine learning” requirement engineering
4	“NLP” requirement engineering

2.3. Inclusion and exclusion criteria

The search results obtained from the initial search were further refined based on specific inclusion and exclusion criteria, as outlined in Table 2. The inclusion criteria were as follows:

I1. The literature should be relevant to software requirement specifications.

I2. The paper should discuss functional and non-functional requirements in the context of NLP and Machine Learning Approach.

I3. The discussion should be within the ML algorithm and methodology.

I4. The literature should describe current practices and challenges related to functional and non-functional requirements based on NLP and ML approaches.

On the other hand, the exclusion criteria were applied to filter out irrelevant publications. The exclusion criteria were as follows:

E1. Publications that do not discuss functional or non-functional or software requirement engineering.

E2. Publications that do not consider requirements engineering in the context of the ML algorithm and methodology.

E3. Publications published before the year 2018.

E4. Search results not published in IEEE, Springer, Cambridge, Elsevier or ACM.

E5. Duplicate search results.

E6. Non-English literature.

E7. Exclusive content.

Table 2. Inclusion and exclusion criteria filter result

Publisher	Result count	Inclusion filtered	Exclusion filtered
IEEE	68	27	16
Springer	43	16	6
ACM	41	14	3
Cambridge	20	7	2
Elsevier	15	8	2
Others	113	62	3

2.4. Quality Assessment

The selected literature obtained after applying the search result filters is subjected to a detailed review, and its quality is assessed based on a set of quality assessment criteria. The quality assessment criteria include:

QA1. Does the literature have relevance to software requirements specifications?

QA2. Does the literature specifically investigate software requirements specifications within the context of NLP and ML approaches?

QA3. Does the literature provide insights into the current practices of functional and non-functional requirements in NLP and ML algorithms?

QA4. Does the literature address the challenges associated with FR and NFR based on ML approaches andpropose potential solutions?

Each criterion is assigned a score to rate the quality of the literature. The scores used are Y=1.0 (Yes), P=0.5 (Partial), and N=0.0 (No).

2.5. Data Collection

Various types of research papers related to the topic are focused in this SLR. The review collects and analyzes the following data and metadata for the selected papers:

- Full reference of the literature material.
- Type of research conducted.
- Research domain or field of study.
- Authors of the papers.
- Summary of the research, including research questions and corresponding answers.
- Evaluation of the quality of each paper.

- Assessment of whether the research is relevant to software requirements specifications andspecifically addresses FR and NFR based on ML approaches.

2.6. Data Analysis

The collected results are analyzed and categorized to address the research questions posed in the study:

- Current practices of NLP and ML approaches in software requirements specification (RQ1).

- Common NLP and ML algorithms employed in the literature to enhance the accuracy and clarityof requirements specification (RQ2).

- Impact of insufficient investment in functional and non-functional requirements and the associatedchallenges and consequences (RQ3).

- Available strategies to address the challenges or gap and future research directions and practicalapplications (RQ4).

3. RESULTS

This section presents the outcomes of the literature search conducted for the purpose of conducting the systematic literature review. These results serve as the foundation for the subsequent analysis and synthesis.

3.1. Search Result

A total of thirty-two literature sources that focused on requirements documentation in software engineering have been obtained. These sources primarily discussed the NLP and ML approaches to improve software requirement specifications. Table 3 presents an overview of the search results, including the publication year and the main content topic relevant to this review. Figure 1 shows the distribution of article types in the search results.

Table 3. Search Results

Author	Year	Main Area(s)
Ahmad et al.	2020	Requirements Identification
AlDhafer et al.	2022	Requirements Classification
Amasaki et al.	2018	Non-functional Requirements Classification
Asif, M et al.	2019	Nonfunctional Requirements
Baker et al.	2019	Non-functional Requirements Classification
Binkhonain et al.	2019	Non-functional Requirements Classification
Cheligeer et al.	2022	Requirements Elicitation
Cheema, S. M. et al.	2020	Aid Requirements Reuse
Dave et al.	2022	Requirements Identification

Dave, D. et al.	2021	Requirements Classification, Automation
Dalpiaz, F. et al.	2018a	Ambiguity Detection
Dalpiaz, V. et al.	2018b	Natural Language Processing
Gondal et al.	2020	Non-functional Requirements Integration
Habib et al.	2021	Requirements Smells Detection
Habibullah et al.	2021	Non-functional Requirements
Handa et al.	2021	Non-functional Requirements Engineering, Questionnaire
Kaur, H. et al.	2019	Significance Testing, Non-functional Requirements
Munoz, D. J.	2021	Variability Models
Pasquariello, A. et al.	2022	Functional Requirements Specification
Rahman et al.	2019	Non-functional Requirements Classification
Riaz et al.	2019	Ambiguous Requirements Detection
Saleem, S. et al.	2019	Requirement Engineering
Shreda & Hanani	2021	Non-functional Requirements Identification
Siddharth et al.	2022	Natural Language Processing in Design Research
Sonbol et al.	2022	NLP based text representation
Tóth & Vidács.	2018	Non-functional Requirements Identification and Classification
Tariq et al.	2021	Non-functional Requirements Modeling
Viggiato et al.	2022	Natural Language Processing in Test Case Descriptions
Vijayvargiya et al.	2022a	Functional Requirements Classification
Vijayvargiya et al.	2022b	Software Requirements Classification
W., Seppänen et al.	2020	Quality Requirements Documentation
Younas, M. et al.	2020	Requirements Elicitation

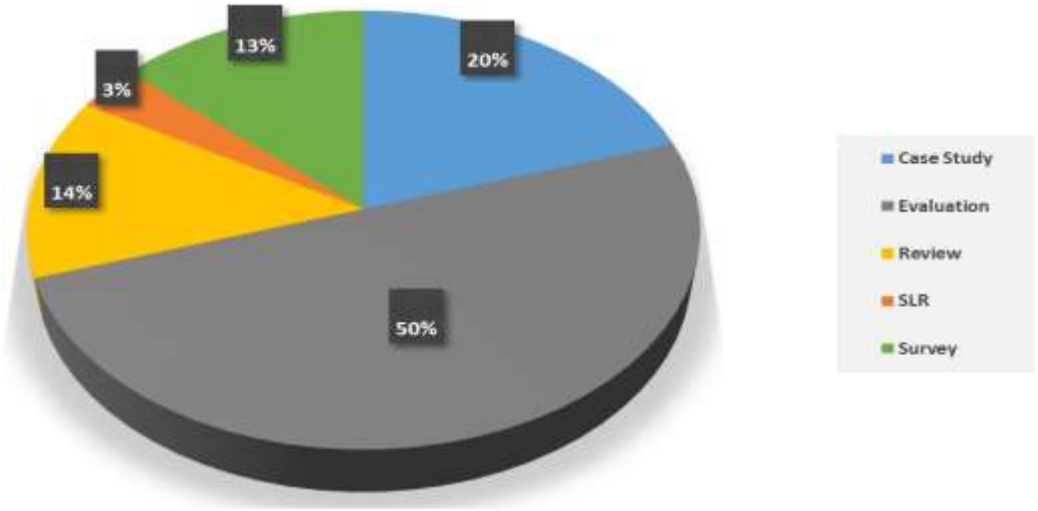

Fig. 1: Overview of research approach

3.2. Quality Factors

The literature contents have been thoroughly reviewed and their relevance has been cross-referenced using the quality assessment outlined in section 2.4. The quality of the literature obtained from the search resultshas been assessed and evaluated, as summarized in Table 4 and Figure 2.

The assessment includes a range of scores, with the highest score being 4.0 and the lowest score being 1.0. Among the papers reviewed, four of them received the highest score of 4.0, two papers scored 3.5, fourteen papers scored 3.0, six papers scored 2.5, five papers scored 2.0, and one paper scored 1.0. It is noteworthy that all papers achieved a minimum score of 1.0, and the majority of them scored 3.0 or higher. Based on this evaluation, it can be inferred that the selected papers meet the overall quality criteria for the review.

4. DISCUSSION

This section presents a comprehensive examination and evaluation of existing research studies aimed at identifying the current utilization of natural language processing (NLP) and machine learning (ML) approaches in enhancing requirements specification (RQ1) within the field of software requirements engineering. Additionally, the section investigates the impact of insufficient investment in functional and non-functional requirements and explores common ML techniques employed, while identifying the associated challenges and consequences (RQ2). Furthermore, available strategies to address this gap are discussed (RQ3). How the findings from this review, as explored in RQ1 will inform future research directions and practical applications (RQ4).

Finally, the section concludes with a critical assessment of the limitations inherent in this literature review and previous research efforts, providing a thoughtful critique of their respective shortcomings.

Table 4. Quality evaluation of literatures from search results

ID	Authors	Article type	QA1	QA2	QA3	QA4	Total Score
1	Ahmad et al.	SLR	Y	P	P	Y	3.0
2	AlDhafer et al.	Evaluation	Y	P	P	P	2.5

3	Amasaki et al.	Case Study	Y	P	N	Y	2.5
4	Asif, M. et al.	Evaluation	P	P	Y	P	2.5
5	Baker, C. et al.	Case study	Y	P	P	N	2.0
6	Binkhonain, et al.	SLR	P	P	N	N	1.0
7	Cheligeer, et al.	Survey	Y	P	P	N	2.0
8	Cheema, S. M. et al.	Evaluation	P	Y	N	Y	2.5
9	Dave, et al.	Case study	Y	Y	Y	Y	4.0
10	Dave, D. et al.	Evaluation	Y	Y	Y	Y	4.0
11	Dalpiaz, F. et al.	Evaluation	Y	Y	Y	P	3.5
12	Dalpiaz, V. et al.	Review	Y	Y	P	P	3.0
13	Gondal et al.	Case Study	Y	P	P	Y	3.0
14	Habib et al.	Case study	Y	P	P	N	2.0
15	Habibullah et al.	Survey	Y	Y	Y	P	3.5
16	Handa et al.	Review	Y	P	Y	P	3.0
17	Kaur, H. et al.	Evaluation	P	Y	Y	P	3.0
18	Munoz, D. J. et al.	Evaluation	P	P	P	P	2.0
19	Pasquariello, A. et al.	Case Study	Y	Y	P	P	3.0
20	Rahman, et al.	Evaluation	Y	P	Y	P	3.0
21	Riaz, M. Q., Butt, et al.	Evaluation	Y	P	P	P	2.5
22	Saleem, S. et al.	Survey	Y	P	P	Y	3.0
23	Shreda, et al.	Evaluation	Y	P	P	Y	3.0
24	Siddharth, et al.	Review	Y	P	Y	P	3.0
25	Sonbol et al.	SLR	P	P	Y	Y	3.0
26	Tóth, L., et al.	Evaluation	Y	P	Y	P	3.0
27	Tariq et al.	Survey	Y	P	Y	P	3.0
28	Viggiato, et al.	Evaluation	Y	Y	Y	Y	4.0
29	Vijayvargiya, et al.	Evaluation	Y	Y	Y	Y	4.0

30	Vijayvargiya, S. et al.	Evaluation	P	Y	P	P	2.5
31	W. Behutiye, et al.	Evaluation	Y	N	P	P	2.0
32	Younas, M. et al.	Case Study	Y	P	P	Y	3.0

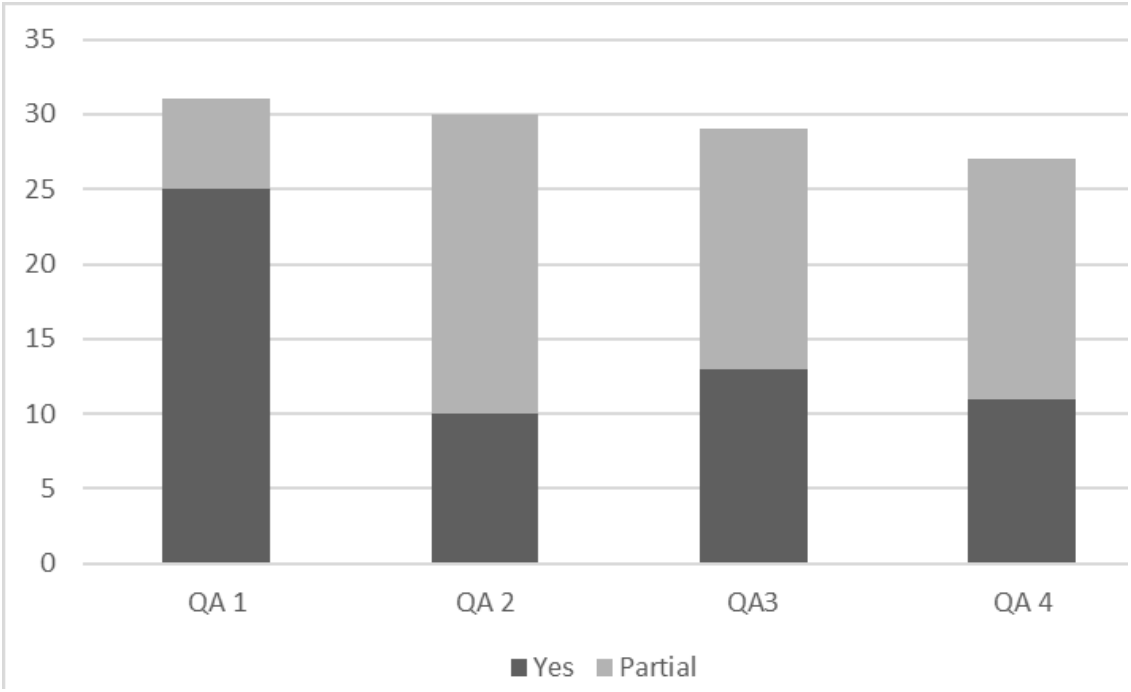

Fig. 2: Quality assessment coverage distribution

4.1 What research topics are being addressed?

All the research papers chosen for this literature review focus on using ML or NLP to improve the process of specifying requirements in software development. Discussion is based on ID number in Table 4.

32 papers in this SLR cover a diverse range of research topics in software requirements engineering. The majority of the papers (20 out of 32) focus on functional requirements classification and extraction. These papers investigate various approaches and techniques for identifying and classifying functional requirements (ID number 29, 8, 9, 26, 30, 28,1, 19, 7, 21, 17,13, 15, 18, 11, 4, 24, 14, 25 and 27). Another significant area of research is non-functional requirements identification and classification, which is addressed by 11 papers (ID number 2, 6, 20, 24, 29, 14, 3, 5, 21, 13 and 32). These papers propose various techniques and algorithms for effectively handling non-functional requirements in software development.

In addition to functional and non-functional requirements, the review includes papers that explore the use of natural language processing (NLP) techniques in requirements engineering (ID number 11, 9, 25 and 24), visualization and detection of requirements ambiguity (ID number 12, 14 and 21), integration of non-functional requirements in software development processes (ID number 13 and 19), and the application of machine learning algorithms in requirements engineering (ID number 1 and 6). Moreover, several papers address related topics such as requirements documentation (ID number 31), quality assurance (ID number 14), recommendation systems (ID number 8), agile development (ID number 32), and design research (ID number 24). Overall, these papers collectively contribute to advancing our understanding of various aspects of software requirements engineering and provide insights into effective techniques and approaches for handling different types of requirements.

4.2. What are the trends for the research methodology?

In analyzing the research methodology employed in the selected 32 papers, several trends can be observed. Discussion is also based on ID number in Table 4. The majority of the papers utilize an empirical approach, conducting experiments, case studies, or surveys to gather data and validate their proposed methods or frameworks. This empirical approach allows researchers to collect real-world data and evaluate the effectiveness of their techniques in practical settings (ID number 1, 3, 4, 6, 7, 8, 9, 12, 13, 14, 15, 16, 17,18, 19, 20, 21, 22, 23, 24, 26, 28, 29, 30 and 32). By employing systematic literature review methodologies,some papers aim to summarize and analyze the state of the art in a particular research area (ID number 2, 25 and 31). This approach involves systematically identifying, evaluating, and synthesizing existing research on a specific topic.

Moreover, a significant number of papers leverage machine learning techniques to train and test their proposed systems. Various classification algorithms, deep learning models, and ensemble learning are employed to automate the process of requirement identification, classification, and analysis (ID number 1, 3, 4, 6, 7, 8, 10, 11, 13, 14, 15, 16, 17, 18, 19, 20, 21, 22, 23, 24, 26, 27, 28, 29, 30 and 32). These machine learning-based approaches enable researchers to handle the complexity and volume of software requirements data. Additionally, some papers make use of natural language processing (NLP) techniques to extract and analyze requirements from textual sources. Text mining, information extraction, and sentiment analysis are among the NLP methods employed, enhancing the understanding and interpretation of software requirements (ID number 1, 5, 9, 12, 13, 15, 17, 19, 20, 22, 24, 26, 27, 30 and 32). By leveraging these methodologies, researchers strive to address various challenges in software requirements engineering and improve the overall process of specifying requirements in software development.

4.3. What are the current practices of NLP and ML Approaches to Improve requirementsspecification?

Several papers in the list address the current practices of using NLP and ML approaches to improve requirements specification. AlDhafer et al. (2022) developed a deep learning system for requirements classification. Baker et al. (2019) focused on automatic classification of non-functional software requirements. Binkhonain and Zhao (2019) conducted a review of machine learning algorithms for identifying and classifying non-functional requirements. Cheligeer et al. (2022) conducted a literature review on machine learning in requirements elicitation. Dalpiaz et al. (2018) explores the use of information visualization and natural language processing (NLP) techniques to identify and address ambiguity and incompleteness in requirements engineering.

All of these papers contribute valuable insights into the application of NLP and ML in enhancing requirements specification practices in software engineering.

4.4. What are the challenges of using NLP and ML approaches?

The application of NLP and ML approaches to improve requirements specification in software engineering faces several challenges. These include the ambiguity and variability in natural language understanding, the scalability and generalization of models to handle diverse requirements, the availability and quality of labeled training data, the interpretability and explainability of models, the handling of non-functional requirements, and the domain-specific challenges associated with different industries. Addressing these challenges requires ongoing research and development to enhance the robustness, scalability, interpretability, and effectiveness of NLP and ML techniques in the context of requirements specification.

4.5. What are the proposed solutions?

The papers in the provided list offer various proposed solutions to address the challenges associated with using NLP and ML approaches to improve requirements specification. Some of the proposed solutions include:

- **Enhanced NLP Techniques:** Researchers propose the development and refinement of NLP techniques to improve the understanding and interpretation of natural language requirements. This includes leveraging

advanced algorithms, semantic analysis, and language modeling to capture the meaning and context of requirements more accurately (Cheligeer et al., 2022; Siddharth et al., 2022).

- **Feature Engineering and Representation:** Several papers suggest exploring effective feature engineering and representation methods specific to requirements specification. This involves identifying relevant features and representations that can capture both functional and non-functional aspects of requirements in a structured and meaningful way (Pasquariello, A. et al., 2022; Binkhonain& Zhao, 2019).

- **Transfer Learning and Pretrained Models:** The utilization of transfer learning and pretrained models from large-scale language models such as BERT (Bidirectional Encoder Representations from Transformers) is proposed. Pretrained models can be fine-tuned on requirements-specific datasets to leverage their knowledge and improve the performance of requirements classification tasks (Vijayvargiya et al., 2022a; AlDhafer et al., 2022).

- **Hybrid Approaches:** Hybrid approaches that combine both rule-based and machine learning techniques are suggested. This entails leveraging domain-specific rules and knowledge to augment thecapabilities of ML models, improving their accuracy and interpretability (Rahman et al., 2019; Vijayvargiya et al., 2022b).

- **Human-in-the-Loop Approaches:** Some papers advocate for the involvement of domain experts and stakeholders in the requirements specification process. Human-in-the-loop approaches enable the collaboration between automated techniques and human expertise, ensuring a more comprehensive and accurate analysis of requirements (Habib et al., 2021; Dave & Anu, 2022).

- **Evaluation and Metrics:** Proposed solutions emphasize the importance of evaluating the performanceof NLP and ML models in requirements specification tasks. The development of appropriate evaluation metrics and benchmark datasets allows for a standardized assessment of different approaches, enabling comparisons and advancements in the field (Baker et al., 2019; Tóth & Vidács, 2018).

By implementing these proposed solutions, researchers aim to overcome the challenges and improve the effectiveness and applicability of NLP and ML approaches in enhancing requirements specification processes in software engineering.

4.6. What are the limitations presented in the existing research?

The existing research papers on NLP and ML approaches to improve requirements specification in software engineering highlight several limitations that should be considered. Firstly, many studies in the literature review suffered from limited dataset sizes, which may restrict the generalizability and robustness of the proposed approaches (Dalpiaz, V. et al., 2018; Viggiato et al., 2022). Additionally, the focus on specific domains, such as mobile apps or software testing, in the selected papers may limit the applicability of the proposed solutions to other domains (Dave & Anu, 2022; Riaz et al., 2019). Another limitation observed is the insufficient evaluation of the proposed approaches, with some papers lacking comprehensive performance metrics or comparisons with existing methods (Baker et al., 2019; Binkhonain et al., 2019).

Furthermore, there was an overemphasis on functional requirements, with limited attention given to non-functional requirements, which are equally crucial in software development (Binkhonain & Zhao, 2019; Viggiato et al., 2022). The lack of interpretability in some ML-based approaches poses challenges in understanding and validating the reasoning behind classification or prediction outcomes (Habib et al., 2021;Rahman et al., 2019). Lastly, the limited real-world implementation of NLP and ML approaches in industry settings indicates a gap between research and practical application (Cheligeer et al., 2022; Vijayvargiya etal., 2022a).

4.7. Comparison of Review Papers

In this section, we present a comparison between the systematic review we have done which is referred as

Paper 3 and the recent two reviews that have been done by Ahmad et al., (2020): Paper 1; and Sonbol et al., (2022): Paper 2 as depicted in Table 5. We highlight the differences in terms of research focus, research questions, methodology, scope and limitations as well as contribution and novelty, providing insights into the unique contributions to the field of software requirements engineering.

Table 5. Table of Comparison of review papers

Paper ID/ Comparison Factor	Paper 1	Paper 2	Paper 3
Research Focus	Utilize ML algorithms for software requirements identification on Stack Overflow.	Support requirement engineering tasks using NLP-based text representation techniques.	Improve requirements specification using NLP and ML approaches.
Research questions	Accuracy of classification models, types of requirements that can be identified, performance onStack Overflow data.	Applications of NLP-based text representation techniquesin requirement engineering tasks.	Effectiveness of different algorithms, impact on requirement quality, implementation challenges.
Methodology	Systematic literature review	Systematic mapping review	Systematic literature review
Scope and Limitations	Limited to ML algorithms for software requirements identification on Stack Overflow.	Application of NLP-based text representation techniques in requirement engineering tasks.	Broad range of NLP and ML approaches in requirements engineering.
Contribution& Novelty	Understanding the use of ML algorithms for software requirements identification on Stack Overflow.	Support for requirement engineering tasks through NLP-based text representation techniques.	Insights into NLP and ML techniques for improved requirements specification.

It can be seen that Paper 3 provides valuable insights into using NLP and ML techniques to enhance requirements specification in software engineering. It reviews existing studies and analyzes different algorithms, their impact on requirement quality, and implementation challenges. This contribution is important for practitioners and researchers looking for practical guidance in applying NLP and ML approaches to improve requirements specification.

Paper 3 is unique as it focuses research on requirements specification in software engineering. While Paper 1 looks at ML algorithms for software requirements identification and Paper 2 explores NLP-based text representation techniques for requirement engineering tasks, Paper 3 takes a broader approach. Its specific focus on requirements specification allows for a deeper exploration of the challenges and techniques relevant to this aspect of software development. Paper 3 offers unique insights tailored to requirements specification, providing a comprehensive understanding of the latest advancements in NLP and ML approaches for this field.

In terms of novelty, Paper 3 stands out by conducting a systematic review and analysis of the existing literature, presenting a holistic view of the advancements in NLP and ML approaches for requirements specification. By synthesizing the findings of multiple studies, Paper 3 provides a comprehensive overview and critical evaluation of the research in this domain. This novel approach allows for the identification of research gaps, challenges, and future directions for further advancements in the field. Overall, the contribution and novelty of Paper 3 lies in its focused research on requirements specification, its systematic review methodology, and its comprehensive insights into NLP and ML techniques, providing valuable guidance to practitioners and researchers and advancing the understanding of requirements engineering.

4.8. Limitations of this study

This review only includes research findings from the past five years, which is due to time constraints and limited review capabilities. Consequently, the reviewed materials and content focus on recent highlights and may not provide a comprehensive overview of software requirements specification, specifically concerning functional and non-functional requirements. It is important to note that older unresolved issues might receive less attention over time, as new pressing issues tend to dominate the research landscape. Therefore, the review's scope is limited to the current trends and may not encompass the entirety of softwarerequirements specification in a broader context.

Another limitation of this SLR is that most of the papers selected for analysis focus primarily on non-functional requirements rather than functional requirements. These papers discuss topics related to aspects of software requirements other than how the software should function. While these studies provide valuable insights into non-functional requirements, there is a lack of research representation on how the software should work. This could limit the overall scope and applicability of the findings, as the review does not fully cover both functional and non-functional requirements in software development.

Furthermore, it is worth noting that this review primarily focuses on research papers and may not encompass unpublished research or industry insights. This reliance on published literature may introduce a bias towards more theoretical and academic perspectives, potentially overlooking practical experiences andreal-world challenges faced by software development teams. Therefore, the findings and conclusions drawn from this review should be considered within the context of the selected papers and their limitations, and further research is needed to gain a more comprehensive understanding of the use of NLP and ML approaches to improve requirements specification in software engineering.

CONCLUSION

In conclusion, this systematic literature review provides valuable insights into the current state of research on the use of NLP and ML approaches to improve requirements specification in software engineering. Thereview highlights the common practices, challenges, proposed solutions, and limitations in this domain. It is evident that NLP and ML techniques have been extensively explored to enhance the accuracy and clarity of requirements specification, particularly in the context of non-functional requirements. However, there isa need for further research to bridge the gap in addressing the functional requirements.

The review also acknowledges certain limitations, including the focus on recent research findings, which may not provide a comprehensive overview of requirements specification. Moreover, the emphasis on non-functional requirements in the selected papers may limit the broader understanding of both functional and non-functional aspects.

To advance this field, future research should explore the integration of NLP and ML approaches to address functional requirements and consider the specific needs of different software development contexts. Industry insights and practical experiences should be incorporated to provide a more holistic understanding. Overall, this literature review serves as a foundation for future research and practical applications in utilizing NLP and ML approaches to improve requirements specification in software engineering.

REFERENCES

[1] Ahmad, A., Feng, C., Khan, M., Khan, A., Ullah, A., Nazir, S., & Tahir, A. (2020). A systematic literature review on using machine learning algorithms for software requirements identification on stack overflow. Security and Communication Networks, 2020, 1-19.

[2] AlDhafer, O., Ahmad, I., & Mahmood, S. (2022). An end-to-end deep learning system for requirements classification using recurrent neural networks. Information and Software Technology, 147, 106877.

[3] Amasaki, S., & Leelaprute, P. (2018, August). The effects of vectorization methods on non-functional requirements classification. In 2018 44th Euromicro Conference on Software Engineering and Advanced Applications (SEAA) (pp. 175-182). IEEE.

[4] Asif, M., Ali, I., Malik, M. S. A., Chaudary, M. H., Tayyaba, S., & Mahmood, M. T. (2019). Annotation of software requirements specification (srs), extractions of nonfunctional requirements, and measurement of their tradeoff. IEEE Access, 7, 36164-36176.

[5] Baker, C., Deng, L., Chakraborty, S., & Dehlinger, J. (2019, July). Automatic multi-class non-functional software requirements classification using neural networks. In 2019 IEEE 43rd annual computer software and applications conference (COMPSAC) (Vol. 2, pp. 610-615). IEEE.

[6] Binkhonain, M., & Zhao, L. (2019). A review of machine learning algorithms for identification and classification of non-functional requirements. Expert Systems with Applications: X, 1, 100001.

[7] Cheligeer, C., Huang, J., Wu, G., Bhuiyan, N., Xu, Y., & Zeng, Y. (2022). Machine learning in requirements elicitation: a literature review. AI EDAM, 36, e32.

[8] Shahbaz, M., Jam, F. A., Bibi, S., & Loganathan, N. (2016). Multivariate Granger causality between CO2 emissions, energy intensity and economic growth in Portugal: evidence from cointegration and causality analysis. Technological and Economic Development of Economy, 22(1), 47-74.

[9] Cheema, S. M., Adnan, M., Baqir, A., Malik, S., & Munawar, B. A. (2020, January). A recommendation system for functional features to aid requirements reuse. In 2020 3rd International Conference on Computing, Mathematics and Engineering Technologies (iCoMET) (pp. 1-4). IEEE.

[10] Dalpiaz, F., Ferrari, A., Franch, X., & Palomares, C. (2018a). Natural language processing for requirements engineering: The best is yet to come. IEEE software, 35(5), 115-119.

[11] Dave, D., & Anu, V. (2022, June). Identifying Functional and Non-functional Software Requirements from User App Reviews. In 2022 IEEE International IOT, Electronics and Mechatronics Conference (IEMTRONICS) (pp. 1-6). IEEE.

[12] Dave, D., Anu, V., & Varde, A. S. (2021, December). Automating the Classification of Requirements Data. In 2021 IEEE International Conference on Big Data (Big Data) (pp. 5878-5880). IEEE.

[13] Dalpiaz, F., Van der Schalk, I., & Lucassen, G. (2018b). Pinpointing ambiguity and incompleteness in requirements engineering via information visualization and NLP. In Requirements Engineering: Foundation for Software Quality: 24th International Working Conference, REFSQ 2018, Utrecht, The Netherlands, March 19-22, 2018, Proceedings 24 (pp. 119-135). Springer International Publishing.

[14] Gondal, M. A., Qureshi, N. A., Mukhtar, H., & Ahmed, H. F. (2020). An Engineering Approach to Integrate Non-Functional Requirements (NFR) to Achieve High Quality Software Process. In ICEIS (2) (pp. 377- 384).

[15] Habib, M. K., Wagner, S., & Graziotin, D. (2021, September). Detecting Requirements Smells with Deep Learning: Experiences, Challenges and Future Work. In 2021 IEEE 29th International Requirements Engineering Conference Workshops (REW) (pp. 153-156). IEEE.

[16] Habibullah, Khan Mohammad, and Jennifer Horkoff. "Non-functional requirements for machine learning: understanding current use and challenges in industry." In 2021 IEEE 29th International Requirements Engineering Conference (RE), pp. 13-23. IEEE, 2021.

[17] Setyawarno, D. ., Rosana, D., Widodo, E. ., & ., M. (2023). Development of A Computer-Based Diagnostic Assessment to Measure Junior High School Students' Readiness to Participate AKM and PISA with Automatic Feedback. International Journal of Membrane Science and Technology, 10(2), 1050-1065. https://doi.org/10.15379/ijmst.v10i2.1411

[18] Handa, N., Sharma, A., & Gupta, A. (2021). Non-functional Requirements Engineering Questionnaire: Novel Visions and Review of Literature. Recent Innovations in Computing: Proceedings of ICRIC 2020, 799-808.

[19] Kaur, H., & Sharma, A. (2019). ANOVA Based Significance Testing of Non-functional Requirements in Software Engineering. International Journal of Information Technology Project Management (IJITPM), 10(4), 100-117.

[20] Kitchenham, B. A., Dyba, T., & Jorgensen, M. (2004, May). Evidence-based software engineering. In Proceedings. 26th International Conference on Software Engineering (pp. 273-281). IEEE.

[21] Munoz, D. J., Gurov, D., Pinto, M., & Fuentes, L. (2021, June). Category theory framework for variability models with non-functional requirements. In Advanced Information Systems Engineering: 33rd International Conference, CAiSE 2021, Melbourne, VIC, Australia, June 28–July 2, 2021, Proceedings (pp. 397-413). Cham: Springer International Publishing.

[22] Pasquariello, A., Vitolo, F., & Patalano, S. (2022, September). Systems and Requirements Engineering: An Approach and a Software Tool for the Interactive and Consistent Functional Requirement Specification. In Advances on Mechanics, Design Engineering and Manufacturing IV: Proceedings of the International Joint Conference on Mechanics, Design Engineering & Advanced Manufacturing, JCM 2022, June 1-3, 2022, Ischia, Italy (pp. 491-502). Cham: Springer International Publishing.

[23] Rahman, M. A., Haque, M. A., Tawhid, M. N. A., & Siddik, M. S. (2019, August). Classifying non- functional requirements using RNN variants for quality software development. In Proceedings of the 3rd ACM SIGSOFT International Workshop on Machine Learning Techniques for Software Quality Evaluation (pp. 25-30).

[24] Riaz, M. Q., Butt, W. H., & Rehman, S. (2019, March). Automatic detection of ambiguous software requirements: An insight. In 2019 5th International Conference on Information Management (ICIM) (pp. 1-6). IEEE.

[25] Saleem, S., Cheema, S. M., Zafar, R., Ali, S. Y., & Shah, S. Z. H. (2019, March). An empirical study on practice of requirement engineering activities in software industries in pakistan industry. In 2019 International Conference on Information Science and Communication Technology (ICISCT) (pp. 1-6). IEEE.

[26] Shreda, Q. A., & Hanani, A. A. (2021). Identifying non-functional requirements from unconstrained documents using natural language processing and machine learning approaches. IEEE Access.

[27] Siddharth, L., Blessing, L., & Luo, J. (2022). Natural language processing in-and-for design research. Design Science, 8, e21.

[28] Sonbol, R., Rebdawi, G., & Ghneim, N. (2022). The use of nlp-based text representation techniques to support requirement engineering tasks: A systematic mapping review. IEEE Access.

[29] Tóth, L., & Vidács, L. (2018). Study of various classifiers for identification and classification of non- functional requirements. In Computational Science and Its Applications–ICCSA 2018: 18th International Conference, Melbourne, VIC, Australia, July 2-5, 2018, Proceedings, Part V 18 (pp. 492-503). Springer International Publishing.

[30] Tariq, S., & Cheema, S. M. (2021, November). Approaches for Non-Functional Requirement Modeling: A Literature Survey. In 2021 4th

International Conference on Computing & Information Sciences (ICCIS) (pp. 1-6). IEEE.

[31] Viggiato, M., Paas, D., Buzon, C., & Bezemer, C. P. (2022, May). Using natural language processing techniques to improve manual test case descriptions. In Proceedings of the 44th International Conference on Software Engineering: Software Engineering in Practice (pp. 311-320).

[32] Vijayvargiya, S., Kumar, L., Malapati, A., Murthy, L. B., & Misra, S. (2022a, July). Software Functional Requirements Classification Using Ensemble Learning. In Computational Science and Its Applications– ICCSA 2022 Workshops: Malaga, Spain, July 4–7, 2022, Proceedings, Part V (pp. 678-691). Cham: Springer International Publishing.

[33] Vijayvargiya, S., Kumar, L., Murthy, L. B., & Misra, S. (2022b, September). Software Requirements Classification using Deep-learning Approach with Various Hidden Layers. In 2022 17th Conference on Computer Science and Intelligence Systems (FedCSIS) (pp. 895-904). IEEE.

[34] W., Seppänen, P., Rodríguez, P., & Oivo, M. (2020). Documentation of quality requirements in agile software development. In Proceedings of the Evaluation and Assessment in Software Engineering (pp. 250- 259).

[35] Younas, M., Jawawi, D. N. A., Shah, M. A., Mustafa, A., Awais, M., Ishfaq, M. K., & Wakil, K. (2020). Elicitation of nonfunctional requirements in agile development using cloud computing environment. IEEE access, 8, 209153-209162.

DOI: https://doi.org/10.15379/ijmst.v10i2.1828