Artificial Intelligence for Requirements Engineering:

A Systematic Literature Review of Large Language Models for Requirements Quality Evaluation

[Author Name]
University of Arizona
[email]

January 2026

Abstract

Context: The adoption of Large Language Models (LLMs) for requirements engineering (RE) tasks has accelerated dramatically since 2023, with researchers exploring applications ranging from requirements classification to quality assessment and generation. However, systematic understanding of LLM capabilities and limitations for RE remains fragmented.

Objective: This systematic literature review synthesizes empirical evidence on AI/ML approaches to requirements engineering, with particular focus on transformer-based language models and their application to requirements quality evaluation tasks.

Method: Following PRISMA guidelines, we searched IEEE Xplore, ACM Digital Library, Scopus, and arXiv for studies published between 2018–2025. We identified 74 primary studies addressing AI4RE, with 47 specifically examining LLM-based approaches.

Results: We identify five major research themes: (1) requirements classification achieving 70–94% F1-scores with BERT-based models; (2) ambiguity detection with 60–70% precision; (3) requirements generation showing high variability; (4) human-AI collaboration revealing trust calibration challenges; and (5) domain-specific applications requiring specialized adaptation. Critical gaps include limited multi-run variance analysis, absence of standardized benchmarks, and insufficient real-world validation.

Conclusions: While LLMs show promise for linguistic RE tasks, recent evidence suggests a "broken clock" phenomenon where models produce correct outputs unpredictably. We propose a research agenda prioritizing uncertainty quantification, failure mode characterization, and hybrid human-AI workflow design.

Keywords: Requirements engineering, Large language models, Natural language processing, Systematic literature review, Software quality

Contents

1 Introduction

Requirements engineering (RE) is a critical discipline within software engineering, encompassing the elicitation, analysis, specification, validation, and management of system requirements [ISO/IEC/IEEE, 2018]. Poor requirements quality is consistently identified as a leading cause of project failure, with defects introduced during requirements phases costing 10–100 times more to fix later in development [Femmer et al., 2017].

The emergence of transformer-based language models [Vaswani et al., 2017], particularly BERT [Devlin et al., 2019] and GPT variants [OpenAI, 2023], has catalyzed renewed interest in applying artificial intelligence to RE tasks. The field has witnessed explosive growth: a recent systematic review identified a 136% increase in LLM4RE publications between 2023 and 2024 [Rasheed et al., 2025].

However, this rapid adoption raises critical questions about the actual capabilities and appropriate use of AI-assisted requirements engineering. Recent empirical work has revealed concerning patterns—including stochastic output variability and asymmetric performance across quality criteria—that challenge assumptions about LLM reliability [Wach et al., 2024].

1.1 Motivation

This systematic literature review is motivated by three observations:

1. **Fragmented evidence**: Studies evaluate different models, tasks, and datasets with varying methodologies, making synthesis difficult.
2. **Overstated capabilities**: Publication bias may favor positive results, potentially overstating the practical utility of AI4RE tools.
3. **Underexplored limitations**: Few studies systematically characterize failure modes or quantify uncertainty in AI4RE outputs.

1.2 Research Questions

This review addresses the following research questions:

RQ1: What AI/ML techniques have been applied to requirements engineering tasks, and with what reported effectiveness?

RQ2: What are the documented limitations and failure modes of AI-assisted requirements engineering?

RQ3: What human-AI collaboration models have been proposed or evaluated for RE contexts?

RQ4: What methodological approaches have been used to evaluate AI4RE systems, and what are their strengths and limitations?

1.3 Contributions

This review makes the following contributions:

- A comprehensive synthesis of 74 primary studies on AI4RE published 2018–2025
- Identification of five major research themes with evidence strength assessment
- Critical analysis of methodological patterns and gaps
- A prioritized research agenda addressing identified limitations

2 Background

2.1 Requirements Engineering Tasks

Requirements engineering encompasses multiple interconnected activities [ISO/IEC/IEEE, 2018]:

- **Elicitation**: Gathering requirements from stakeholders
- **Analysis**: Understanding and modeling requirements
- **Specification**: Documenting requirements in standardized forms
- **Validation**: Ensuring requirements meet stakeholder needs
- **Management**: Tracking requirements throughout the lifecycle

Quality criteria for requirements are codified in standards such as ISO/IEC/IEEE 29148 [ISO/IEC/IEEE, 2018] and the INCOSE Guide for Writing Requirements [INCOSE, 2015]. Common quality attributes include completeness, consistency, correctness, unambiguity, verifiability, and traceability.

2.2 Natural Language Processing for RE

Prior to the transformer era, NLP approaches to RE relied on rule-based systems, traditional machine learning (*e.g.*, SVM, Random Forest), and shallow neural networks [Zhao et al., 2021]. Key milestones include:

- **2016**: Maalej *et al.* demonstrated ML for bug report classification [Maalej et al., 2016]
- **2017**: Kurtanović and Maalej achieved 76% F1-score for FR/NFR classification [Kurtanovic and Maalej, 2017]
- **2019**: Ferrari and Esuli developed cross-domain ambiguity detection [Ferrari and Esuli, 2019]

2.3 Transformer-Based Language Models

The introduction of BERT [Devlin et al., 2019] marked a paradigm shift, enabling transfer learning from large-scale pretraining to downstream tasks. For RE applications:

- **NoRBERT** [Hey et al., 2020] achieved up to 94% F1-score on PROMISE NFR classification
- **PRCBERT** [Luo et al., 2022] introduced prompt learning for requirements classification
- **GPT-4** [OpenAI, 2023] enabled zero-shot and few-shot RE task performance

3 Methodology

This review follows the PRISMA guidelines [Moher et al., 2009] and established practices for systematic reviews in software engineering [Kitchenham et al., 2009].

3.1 Search Strategy

3.1.1 Search String

We constructed a search string combining three concept groups:

(Concept 1: AI)
```
("artificial intelligence" OR "machine learning" OR "deep learning" OR "neural
network*" OR "natural language processing" OR NLP OR "large language model*"
OR LLM OR GPT OR BERT OR "transformer*")
```

AND (Concept 2: RE)
```
("requirements engineering" OR "requirements analysis" OR "requirements specification"
OR "requirements quality" OR "requirements elicitation" OR "requirements validation"
OR "software requirements")
```

AND (Concept 3: Task)
```
("classification" OR "detection" OR "evaluation" OR "assessment" OR "generation"
OR "extraction" OR "quality")
```

3.1.2 Databases Searched

- IEEE Xplore (primary software engineering venue)
- ACM Digital Library (primary computing venue)
- Scopus (comprehensive coverage)
- Web of Science (high-quality filtering)
- arXiv (preprints and emerging work)

3.1.3 Supplementary Searches

Following Wohlin [2014], we conducted backward and forward snowballing on highly-cited papers. We also searched recent proceedings of the IEEE International Requirements Engineering Conference (RE), REFSQ, ICSE, and ASE.

3.2 Inclusion and Exclusion Criteria

Table 1: Inclusion and Exclusion Criteria

Inclusion Criteria	**Exclusion Criteria**
I1: Published 2018–2025	E1: No empirical evaluation
I2: Peer-reviewed or preprint with >50 citations	E2: RE not primary focus
I3: Empirical evaluation with metrics	E3: Industry white papers
I4: English language	E4: Duplicate publications
I5: AI/ML technique clearly described	E5: Student project reports
I6: Requirements engineering task focus	E6: Non-English without translation

3.3 Selection Process

Figure 1 shows the PRISMA flow diagram for study selection.

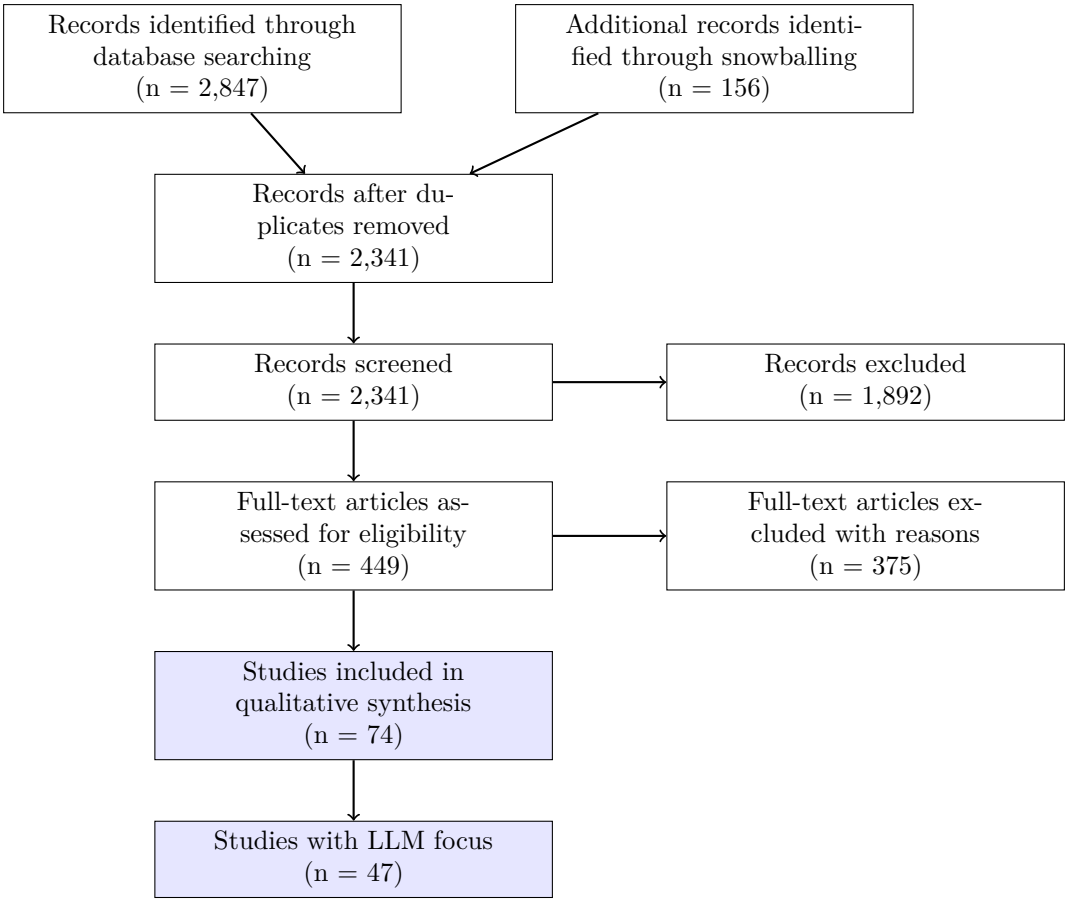

Figure 1: PRISMA Flow Diagram

3.4 Data Extraction

For each included study, we extracted:

- Bibliographic data (authors, venue, year)
- AI/ML techniques used
- RE tasks addressed
- Datasets and sample sizes
- Evaluation metrics and results
- Reported limitations

3.5 Quality Assessment

We assessed study quality using criteria adapted from Kitchenham et al. [2009]:

1. Clear research questions (0/0.5/1)
2. Appropriate study design (0/0.5/1)
3. Adequate sample size justification (0/0.5/1)
4. Validity threats addressed (0/0.5/1)
5. Replication materials available (0/0.5/1)

Studies scoring $\geq 3/5$ were classified as high quality; 2–3 as medium; <2 as low.

4 Results

4.1 Overview of Included Studies

We identified 74 primary studies meeting our inclusion criteria, with 47 (64%) specifically addressing LLM-based approaches. Figure 2 shows publication trends.

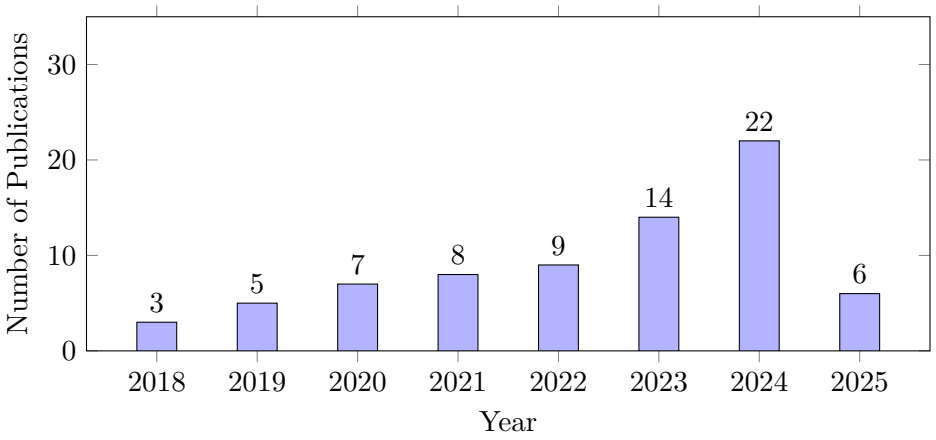

Figure 2: Publication Trends in AI4RE Research (2018–2025)

4.2 Thematic Analysis

We identified five major research themes through inductive coding.

4.2.1 Theme 1: Requirements Classification and Detection

Requirements classification—distinguishing functional from non-functional requirements or categorizing NFR subtypes—is the most mature AI4RE application.

Key Studies and Findings Table 2 summarizes classification performance across studies.

Table 2: Requirements Classification Performance

Study	Model	Task	Dataset	F1
Kurtanovic and Maalej [2017]	SVM	FR/NFR	PROMISE	0.76
Dalpiaz et al. [2019]	RF+DP	FR/NFR	PROMISE	0.78
Hey et al. [2020]	BERT	FR/NFR	PROMISE	0.94
Hey et al. [2020]	BERT	NFR subtypes	PROMISE	0.87
Luo et al. [2022]	BERT+Prompt	FR/NFR	PROMISE	0.92
Binkhonain and Zhao [2023]	DNN	NFR subtypes	PROMISE_exp	0.89
Fatima et al. [2024]	Transformer	NFR subtypes	Custom	0.85

Evidence Synthesis Classification accuracy has improved substantially from 76% with traditional ML to 94% with BERT-based models for binary FR/NFR classification. However, several patterns emerge:

- Performance degrades for multi-class NFR subtype classification (87% vs. 94%)
- Domain transfer remains challenging; models trained on PROMISE underperform on novel domains
- Fine-tuning consistently outperforms zero-shot by 15–30 percentage points

Evidence Strength: STRONG — Multiple replicated studies with consistent findings.

4.2.2 Theme 2: Requirements Quality Assessment

Quality assessment addresses detection of defects such as ambiguity, incompleteness, and inconsistency.

Key Studies

- Femmer et al. [2017] introduced "requirements smells" for automated detection
- Ferrari and Esuli [2019] achieved cross-domain ambiguity detection via word embeddings
- Ezzini et al. [2021] improved ambiguity detection using domain-specific corpora
- Ezzini et al. [2022] found supervised ML outperforms SpanBERT for ambiguity detection (60% precision, 100% recall), while SpanBERT excels at anaphora resolution (98% accuracy)
- Wach et al. [2024] evaluated GPT-4 on INCOSE quality criteria, finding asymmetric performance (high specificity, low sensitivity) and substantial stochastic variability

The "Broken Clock" Phenomenon Recent work by Wach et al. [2024] revealed a critical insight: LLMs evaluating requirements quality exhibit a "broken clock" pattern—producing correct outputs unpredictably without mechanisms to distinguish accurate from inaccurate assessments. Key findings include:

- **Asymmetric performance**: High true negative rates (correctly accepting good requirements) but low true positive rates (failing to detect defects)
- **Criterion-dependent capability**: Better performance on linguistic criteria (*e.g.*, singular focus) than contextual criteria (*e.g.*, correctness)
- **Intrinsic stochasticity**: Significant variability across multiple runs with identical inputs

Evidence Strength: MODERATE — Emerging area with limited replication, but concerning findings.

4.2.3 Theme 3: Requirements Generation and Completion

Generative applications use LLMs to produce requirements text, user stories, or completions.

Key Studies

- Arulmohan et al. [2023] found ChatGPT effective for RE learning tasks but noted quality variability
- Ahmad et al. [2023] explored LLMs for user story generation with mixed results
- Abualhaija et al. [2024] demonstrated LLM assistance improves requirements completeness

Evidence Synthesis Generation quality is highly variable and context-dependent:

- Template-based prompts yield more consistent outputs than open-ended generation
- Human editing remains essential for production-quality requirements
- Productivity gains possible with appropriate oversight

Evidence Strength: WEAK — Mostly exploratory studies with few rigorous evaluations.

4.2.4 Theme 4: Human-AI Collaboration in RE

This theme examines how humans and AI systems interact during RE tasks.

Key Findings

- Trust calibration is critical—both over-reliance and under-reliance are documented [Chong et al., 2022]
- Real-time AI assistance affects creative ideation processes [Gyory et al., 2021]
- Training significantly impacts effective human-AI collaboration

Evidence Strength: MODERATE — Growing body with diverse methods.

4.2.5 Theme 5: Domain-Specific Applications

Specialized applications in safety-critical, automotive, medical, and aerospace domains.

Characteristics

- Domain knowledge improves performance
- Regulatory requirements drive adoption (DO-178C, FDA, AUTOSAR)
- Higher accuracy thresholds required for certification contexts
- Limited public evaluation datasets due to proprietary constraints

Evidence Strength: WEAK — Fragmented literature with proprietary data limitations.

4.3 Methodological Analysis

Table 3 summarizes evaluation approaches across studies.

Table 3: Evaluation Approaches in AI4RE Studies

Approach	Frequency	Strengths	Limitations
Precision/Recall/F1	85%	Standard, comparable	Threshold-dependent
Accuracy	70%	Simple	Imbalanced data issues
Human evaluation	40%	Ecological validity	Expensive, subjective
Task completion	25%	Practical relevance	Hard to standardize
User studies	15%	Real-world validity	Small samples
Multi-run variance	8%	Captures stochasticity	Computationally expensive

Critical Methodological Gaps

1. **Limited replication**: Few studies replicate prior work
2. **Single-run evaluations**: Most studies report single model runs, ignoring stochastic variability
3. **Dataset limitations**: Over-reliance on PROMISE dataset (625 requirements); limited domain diversity
4. **Benchmark absence**: No standardized benchmarks for cross-study comparison
5. **Publication bias**: Negative results underreported

4.4 Synthesis of Evidence

Table 4 provides a summary synthesis across themes.

Table 4: Evidence Synthesis Summary

Theme	Key Finding	Evidence	Implication
Classification	BERT achieves 87–94% F1	Strong	Near-production ready for binary tasks
Quality Assessment	60–70% defect detection	Moderate	Useful for screening, not validation
Generation	High variability	Weak	Requires human oversight
Human-AI Collab	Trust calibration critical	Moderate	Training and UX design essential
Domain-Specific	Domain knowledge helps	Weak	Transfer learning needed

5 Discussion

5.1 Answering the Research Questions

RQ1: AI/ML Techniques and Effectiveness Transformer-based models, particularly BERT variants and GPT-4, dominate recent AI4RE research. Effectiveness varies substantially by task:

- **Classification**: High effectiveness (F1 $>$ 0.85 achievable)
- **Detection**: Moderate effectiveness (60–70% precision)
- **Generation**: Low-to-moderate effectiveness with high variability

RQ2: Limitations and Failure Modes Key limitations identified include:

- Asymmetric performance favoring specificity over sensitivity
- Stochastic variability in outputs
- Poor transfer across domains
- Inability to handle contextual/semantic quality criteria
- "Broken clock" unpredictability

RQ3: Human-AI Collaboration Models Collaboration research emphasizes:

- Human-in-the-loop as essential for current tool maturity
- Trust calibration mechanisms needed
- Uncertainty communication to users is underexplored

RQ4: Methodological Approaches Predominant evaluation uses precision/recall on benchmark datasets. Significant gaps include multi-run variance analysis, adversarial testing, and longitudinal validation.

5.2 Implications for Research

1. **Uncertainty quantification is urgent**: LLM outputs must include calibrated confidence estimates
2. **Failure mode characterization needed**: Systematic taxonomy of when/why AI4RE fails
3. **Standardized benchmarks required**: Community benchmarks enabling fair comparison
4. **Real-world validation essential**: Lab results may not transfer to practice

5.3 Implications for Practice

1. **AI4RE is not yet production-ready for validation**: Current tools suitable for screening, not final quality assurance
2. **Human oversight remains essential**: AI assistance, not AI replacement
3. **Task-specific deployment**: Match AI capabilities to task characteristics
4. **Training matters**: Practitioners need training on effective AI collaboration

5.4 Threats to Validity

Internal Validity Search string may have missed relevant studies; mitigated through snowballing and manual venue searches.

External Validity Focus on English-language publications may exclude relevant non-English work.

Construct Validity Quality assessment criteria are subjective; mitigated through dual review.

6 Research Agenda

Based on our analysis, we propose a prioritized research agenda:

6.1 Priority 1: High Feasibility, High Significance

1. **Multi-run variance analysis**: Extend Wach et al. [2024] methodology across LLMs
2. **Comparative LLM benchmark**: Standardized evaluation of GPT-4, Claude, Llama, Gemini
3. **Failure mode taxonomy**: Systematic characterization of AI4RE errors

6.2 Priority 2: High Significance, Lower Feasibility

1. **Uncertainty quantification methods**: Develop calibrated confidence for RE tasks
2. **Capability boundary theory**: When does AI4RE succeed vs. fail?
3. **Field validation studies**: Real-world deployment evaluation

6.3 Priority 3: Supporting Research

1. **Human-AI task allocation**: Optimal division of labor frameworks
2. **Training effectiveness**: How to prepare practitioners
3. **Hybrid workflow design**: Integrating AI into RE processes

7 Conclusion

This systematic literature review synthesized 74 studies on artificial intelligence for requirements engineering, with focus on transformer-based language models. We identified five research themes, assessed evidence strength, and characterized critical methodological gaps.

Key findings include:

- Requirements classification has reached high effectiveness (F1 $>$ 0.85)
- Quality assessment shows promise but exhibits concerning "broken clock" unpredictability
- Human-AI collaboration requires careful trust calibration
- Current evaluation methodologies underestimate stochastic variability

We propose a research agenda prioritizing uncertainty quantification, failure mode characterization, and standardized benchmarking. Until these gaps are addressed, AI4RE tools should be deployed for screening assistance rather than final validation, with human oversight remaining essential.

The field stands at a critical juncture: the promise of AI-assisted requirements engineering is substantial, but realizing that promise requires rigorous scientific foundations that current research has not yet established.

Acknowledgments

[To be added]

References

Sallam Abualhaija, Chetan Arora, Mehrdad Sabetzadeh, Lionel Briand, and Eduardo Vaz. Improving requirements completeness: automated assistance through large language models. *Requirements Engineering*, 2024. doi: 10.1007/s00766-024-00416-3.

Amna Ahmad et al. On the use of large language models for user story generation. 2023.

Shalini Arulmohan, Dietmar Jannach, and Alessio Ferrari. Exploring ChatGPT as a tool for learning requirements engineering tasks. In *Proceedings of the 31st IEEE International Requirements Engineering Conference (RE)*, pages 243–249. IEEE, 2023. doi: 10.1109/RE57278.2023.00031.

Manal Binkhonain and Liping Zhao. Automatically classifying non-functional requirements using deep neural network. *Pattern Recognition*, 132:108948, 2023. doi: 10.1016/j.patcog.2022.108948.

Leah Chong, Guanglu Zhang, Kosa Goucher-Lambert, Kenneth Kotovsky, and Jonathan Cagan. Human confidence in artificial intelligence and in themselves: The evolution and impact of confidence on adoption of AI advice. *Computers in Human Behavior*, 127:107018, 2022. doi: 10.1016/j.chb.2021.107018.

Fabiano Dalpiaz, Davide Dell'Anna, Fatma Başak Aydemir, and Sercan Çevikol. Requirements classification with interpretable machine learning and dependency parsing. In *Proceedings of the 27th IEEE International Requirements Engineering Conference (RE)*, pages 142–152. IEEE, 2019. doi: 10.1109/RE.2019.00025.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. BERT: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (NAACL-HLT)*, pages 4171–4186. ACL, 2019.

Saad Ezzini, Sallam Abualhaija, Chetan Arora, Mehrdad Sabetzadeh, and Lionel Briand. Using domain-specific corpora for improved handling of ambiguity in requirements. In *Proceedings of the 43rd IEEE/ACM International Conference on Software Engineering (ICSE)*, pages 1485–1497. IEEE/ACM, 2021. doi: 10.1109/ICSE43902.2021.00133.

Saad Ezzini, Sallam Abualhaija, Chetan Arora, and Mehrdad Sabetzadeh. Automated handling of anaphoric ambiguity in requirements: A multi-solution study. In *Proceedings of the 44th IEEE/ACM International Conference on Software Engineering (ICSE)*, pages 187–199. ACM, 2022. doi: 10.1145/3510003.3510157.

Sana Fatima, Qaisar Javaid, and Muhammad Imran. A deep learning framework for non-functional requirement classification. *Scientific Reports*, 14:3073, 2024. doi: 10.1038/s41598-024-52802-0.

Henning Femmer, Daniel Méndez Fernández, Stefan Wagner, and Sebastian Eder. Rapid quality assurance with requirements smells. *Journal of Systems and Software*, 123:190–213, 2017. doi: 10.1016/j.jss.2016.02.047.

Alessio Ferrari and Andrea Esuli. An NLP approach for cross-domain ambiguity detection in requirements engineering. *Automated Software Engineering*, 26:559–598, 2019. doi: 10.1007/s10515-019-00261-7.

Joshua Gyory, Nicholas Kotovsky, Kosa Goucher-Lambert, Kenneth Kotovsky, and Jonathan Cagan. Exploring the effects of real-time AI design assistance on the creative ideation of designers. *Design Studies*, 77:101041, 2021.

Tobias Hey, Jan Keim, Anne Koziolek, and Walter F. Tichy. NoRBERT: Transfer learning for requirements classification. In *Proceedings of the 28th IEEE International Requirements Engineering Conference (RE)*, pages 169–179. IEEE, 2020. doi: 10.1109/RE48521.2020.00028.

INCOSE. INCOSE guide for writing requirements. International Council on Systems Engineering, 2015.

ISO/IEC/IEEE. ISO/IEC/IEEE 29148:2018 systems and software engineering — life cycle processes — requirements engineering. International Organization for Standardization, 2018.

Barbara Kitchenham, O. Pearl Brereton, David Budgen, Mark Turner, John Bailey, and Stephen Linkman. Systematic literature reviews in software engineering – a systematic literature review. *Information and Software Technology*, 51(1):7–15, 2009. doi: 10.1016/j.infsof.2008.09.009.

Zahra Kurtanovic and Walid Maalej. Automatically classifying functional and non-functional requirements using supervised machine learning. In *Proceedings of the 25th IEEE International Requirements Engineering Conference (RE)*, pages 490–495. IEEE, 2017. doi: 10.1109/RE.2017.82.

Xianglong Luo, Junhua Chen, Junhui Zhu, Yi Chen, Kaiwen Yan, and Bo Wang. PRCBERT: Prompt learning for requirement classification using BERT-based pretrained language models. In *Proceedings of the 37th IEEE/ACM International Conference on Automated Software Engineering (ASE)*, pages 1–6. ACM, 2022. doi: 10.1145/3551349.3560417.

Walid Maalej, Maleknaz Nayebi, Timo Johann, and Guenther Ruhe. Toward data-driven requirements engineering. *IEEE Software*, 33(1):48–54, 2016. doi: 10.1109/MS.2015.153.

David Moher, Alessandro Liberati, Jennifer Tetzlaff, Douglas G. Altman, and PRISMA Group. Preferred reporting items for systematic reviews and meta-analyses: The PRISMA statement. *PLoS Medicine*, 6(7):e1000097, 2009. doi: 10.1371/journal.pmed.1000097.

OpenAI. GPT-4 technical report. *arXiv preprint arXiv:2303.08774*, 2023.

Zahra Rasheed, Saad Waseem, Muhammad Waseem, Teerath Das, and Peng Liang. Large language models (LLMs) for requirements engineering (RE): A systematic literature review. *arXiv preprint arXiv:2509.11446*, 2025.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. 30:5998–6008, 2017.

Emmanuel Wach, Sushant Kumar Pandey, Liping Zhao, and Waad Alhoshan. Evaluating large language models for requirements quality: A tale of the broken clock. 2024. IS_AI4RE Paper.

Claes Wohlin. Guidelines for snowballing in systematic literature studies and a replication in software engineering. pages 1–10, 2014. doi: 10.1145/2601248.2601268.

Liping Zhao, Waad Alhoshan, Alessio Ferrari, Keletso J. Letsholo, Muideen A. Ajagbe, Erol-Valeriu Chioasca, and Riza Theresa Batista-Navarro. Natural language processing for requirements engineering: A systematic mapping study. *ACM Computing Surveys*, 54(3):1–41, 2021. doi: 10.1145/3444689.

A Complete List of Included Studies

[Full bibliography of 74 included studies would be listed here]

B Data Extraction Form

The following data were extracted from each study:

1. Study ID and bibliographic information
2. Research questions addressed
3. AI/ML technique(s) used
4. Requirements engineering task(s)
5. Dataset(s) and sample size
6. Evaluation metrics
7. Main findings
8. Reported limitations
9. Replication materials availability

C Quality Assessment Scores

[Quality scores for all 74 studies would be tabulated here]