Conquering the challenge of reliability: text mining to map trends in reliability engineering literature

by

Charles K. Brown

B.A. Physics and Mathematics, Knox College (2007)
M.S. Mechanical Engineering, Northwestern University (2012)

Submitted to the System Design and Management Program
in partial fulfillment of the requirements for the degree of

Master of Science in Engineering and Management

at the

MASSACHUSETTS INSTITUTE OF TECHNOLOGY

June 2023

Authored by: Charles K. Brown
System Design and Management Program
May 12, 2023

Certified by: Bruce G. Cameron
Director, System Architecture Group
Senior Lecturer in System Design and Management
Thesis Supervisor

Accepted by: Joan S. Rubin
Executive Director, System Design and Management Program

Conquering the challenge of reliability: text mining to map trends in reliability engineering literature

by

Charles K. Brown

Submitted to the System Design and Management Program
on May 12, 2023, in partial fulfillment of the
requirements for the degree of
Master of Science in Engineering and Management

Abstract

Reliability engineering faces many of the same challenges in 2023 that it did at its inception in the 1950s. The fundamental issue remains uncertainty in system representation, specifically related to performance model structure and parametrization. Details of a design are unavailable early in the development process and therefore performance models must either account for the range of possibilities or be wrong. Increasing system complexity has compounded this uncertainty. In this work, we seek to understand how reliability engineering literature has changed over time with the assumption that the focus of literature shifts in part due to challenges in the field. Illuminating this change provides reliability practitioners guidance for what they can do in the face of growing complexity. We build this understanding by executing a systematic literature review of 30,543 reliability engineering papers. Topic modeling was performed on the abstracts of those papers to identify 279 topics. Hierarchical topic reduction resulted in the identification of 8 top-level method topics (prognostics, statistics, maintenance, quality control, management, physics of failure, modeling, and risk assessment) as well as 3 domain-specific topics (nuclear, infrastructure, and software). We found that topics more associated with later phases in the development process (such as prognostics, maintenance, and quality control) have increased in popularity over time relative to other topics. We propose that this is a response to the challenges posed by the previously-discussed model uncertainty and increasing complexity. Through zero-shot classification by a large language model, we also found that papers are including more practical examples or case studies and that those topics associated with later phases typically include more practical examples. Thus, while reliability remains fundamentally difficult to predict early in the development process, the field has shifted focus to later-stage and more applicable activities.

Thesis Supervisor: Bruce G. Cameron
Title: Director, System Architecture Group
Senior Lecturer in System Design and Management

Acknowledgments

I would like to thank my wife for her patience and support not just over the course of the writing of this thesis, but over my last two years in the System Design and Management program. Without her sacrifice, none of this would have been possible. I will continue to do what I can to balance the scale.

Similarly, I would like to thank my friends and family for not losing hope during my period of seclusion. Between this research, classes, starting a new job, and having a second child I have been in wholly overwhelmed for at least the last year and a half. I look forward to catching up and enjoying time spent together doing nothing in particular.

Dedicated to Aaron Swartz

Contents

1 Introduction **13**

1.1 Defining reliability 15

1.2 Why reliability is hard 16

1.3 What the literature suggests we do 17

1.4 What this thesis does 18

2 Methods **20**

2.1 Overview 20

2.2 Systematic literature review 22

2.2.1 Research questions 22

2.2.2 Data sources and query 23

2.2.3 Inclusion/exclusion criteria 25

2.2.4 Aggregation methods 26

2.3 Unsupervised topic modeling 26

2.3.1 Overview of techniques 26

2.3.2 Topic modeling execution plan 30

2.4 Text classification 30

2.4.1 Overview of techniques 31

2.4.2 Classification execution plan 32

2.5 Assessment dimensions 32

2.5.1 Proactive/reactive 33

2.5.2 Practicality 34

2.6 Limitations 36

3 Results **37**

3.1 Document population . . . 37

3.1.1 Validation of document population . . . 38

3.2 Topic modeling results . . . 39

3.2.1 Hierarchical topic reduction . . . 39

3.2.2 Domain sub-topics . . . 45

3.2.3 Selected reduced topic hierarchy . . . 45

3.3 Trends in topics . . . 48

3.4 Zero-shot classification results . . . 49

3.4.1 Validation . . . 50

3.5 Trends in classifications (practicality) . . . 50

3.6 Trends in proactive/reactive . . . 51

4 Discussion **55**

4.1 Analysis of topics . . . 55

4.2 Interpretation of trends . . . 56

4.3 Answering the research questions . . . 58

4.4 Reflection on text mining academic literature . . . 60

5 Conclusion **62**

5.1 Reliability outlook . . . 62

5.2 Lessons learned . . . 62

5.3 Future work . . . 63

A Full topic listing **64**

B Domain subtopic visualizations **82**

B.1 Software domain . . . 82

B.2 Infrastructure domain . . . 86

B.3 Nuclear domain . . . 90

C Analysis scripts **94**

C.1 Data import and wrangling 94

C.2 Topic modeling 99

C.3 Classification 102

C.4 Domain topics 103

C.5 Document pool validation 104

C.6 Visualizations 105

D References **110**

List of Figures

2-1 Overview of research procedure/pipeline. The process input is the entire body of academic literature while the core outputs are aggregated topics along with reactive/proactive and practicality scores. Secondary outputs are the full topic list and validation accuracy from both the topic modeling and classification processes. 21

2-2 Traditional product development process including proactive/reactive scores and example topics. This process is adapted from Ulrich 2020. The main distinction is that we include all post-development activities (such as service) in the final category. Additionally, we note that most management activities are captured by the "Planning" phase, as described by Thornton 2021. 33

3-1 Top-level topic counts in reliability engineering papers (cumulative). All documents are included except for those in domain-topics. 41

3-2 Two-dimensional representation of top-level reliability topic clustering using the UMAP algorithm (McInnes, Healy, and Melville 2018), the same used to identify topics using `top2vec`. Each dot represents a publication. There are 22,275 publications represented in this visualization. Note the visible sub-clusters among each topic which hint at the full 297 topics. We can also see that similar topics (like risk analysis and management) tend to appear geometrically closer than those which we might expect to be more dissimilar (like quality control and physics of failure). 45

3-3 Top-level topic count in reliability engineering papers over time. Topics are stacked according to count of publications in 2022. Note the overall exponential growth. Prognostics had the largest count of publications in 2022. 48

3-4 Top-level topic proportion in reliability engineering papers over time. Topics are stacked according to count of publications in 2022. Note that prognostics represented the largest fraction of publications in 2022. 49

3-5 Year-over-year growth rate of top-level topics in reliability engineering literature. Traces have been smoothed with locally estimated scatter plot smoothing (LOESS) to extract trends. The inset highlights trends between 2000 and 2023. Note that prognostics consistently the highest growth rate which continues to increase. Early growth rates were highly volatile due to low absolute paper counts. This volatility coupled with the smoothing algorithm produces artifacts such as the extremely negative growth rate of modeling. 52

3-6 Mean example content for top-level topics in reliability engineering literature over time. Publications' abstracts were assessed to determine whether no example, an illustrative example, or a case study was present. These categories were assigned values and those values were averaged for each year and each topic. Traces have been smoothed with locally estimated scatter plot smoothing (LOESS) to extract trends. The inset highlights trends between 2000 and 2023. Note that in general, publications have been increasing in practical content. 53

3-7 Mean timing period (product development phase) associated with the main topic of reliability engineering literature over time. Topics are correlated with phases and a weighted average is used to summarize the topic mix of papers for each year. The trace has been smoothed with locally estimated scatterplot smoothing (LOESS) to extract trends. The shaded region represents a 95% confidence interval for the smoothing. Note that reliability publications increasingly focus on activities which occur later in the product development process. 54

B-1 Subtopic count in software papers over time. Topics are stacked according to count of publications in 2022. Labels are based on the top three terms in the subtopic. 82
B-2 Subtopic proportion in software papers over time. Topics are stacked according to count of publications in 2022. Labels are based on the top three terms in the subtopic. 83
B-3 Subtopic growth in software papers over time. Labels are based on the top three terms in the subtopic. 84
B-4 Subtopic practicality in software papers over time. Labels are based on the top three terms in the subtopic. 85
B-5 Subtopic count in infrastructure papers over time. Topics are stacked according to count of publications in 2022. Labels are based on the top three terms in the subtopic. 86
B-6 Subtopic proportion in infrastructure papers over time. Topics are stacked according to count of publications in 2022. Labels are based on the top three terms in the subtopic. 87
B-7 Subtopic growth in infrastructure papers over time. Labels are based on the top three terms in the subtopic. 88
B-8 Subtopic practicality in infrastructure papers over time. Labels are based on the top three terms in the subtopic. 89

B-9 Subtopic count in nuclear papers over time. Topics are stacked according to count of publications in 2022. Labels are based on the top three terms in the subtopic. 90
B-10 Subtopic proportion in nuclear papers over time. Topics are stacked according to count of publications in 2022. Labels are based on the top three terms in the subtopic. 91
B-11 Subtopic growth in nuclear papers over time. Labels are based on the top three terms in the subtopic. 92
B-12 Subtopic practicality in nuclear papers over time. Labels are based on the top three terms in the subtopic. 93

List of Tables

2.1 Example applications of the practicality classification scoring system. The abstracts selected were labeled with the respective scores by both a human and a large language model and are pulled from the document population. Abstracts are copyrighted by their publishers as indicated and reproduced here for fair use. Citations: Lisnianski, Frenkel, and Khvatskin 2021, Carot and Sanz 2000, Mancuso et al. 2016. 35

3.1 The reference lists from six survey papers were compared to the document population to estimate coverage. Titles were matched with a simple character ratio (number matching/total number), with values above 0.9 considered matching. Note that this result indicates a significant lack of coverage across the reliability engineering literature. . 38

3.2 Topic modeling results. The model automatically identified 297 topics; these 11 were generated using hierarchical topic reduction. Target reduced topic numbers between 5 and 20 were surveyed and 11 was qualitatively determined to be the optimal amount of aggregation to elucidate the maximum number of relevant top-level topics. The full list of topics can be seen in Appendix A. 40

3.3 Software topic sub-topics as identified with a target 5 topic hierarchical reduction. We note that these could be mapped onto a timing dimension like the top-level topics, though the dimension would be unique to software development. 46

3.4 Infrastructure topic sub-topics as identified with a target of 8 topic hierarchical reduction. These appear to include numerous further domain-specific topics, so assignment of a timing score would be difficult without probing at a lower level. 46

3.5 Nuclear topic sub-topics as identified with a target of 8 topic hierarchical reduction. We note that these could be mapped onto a timing dimension like the top-level topics, though the dimension would be unique for nuclear plant development. 46

3.6 These are the timing or product development phase dimension assignments for each topic. Each was assigned based on discussions of best practices as described in Raheja and Gullo 2012. Numeric scores were associated with each phase (1 through 6) as described in Section 2.5.1. The scores are linear and used only for calculation of mean product phase in visualizations. 51

A.1 This table contains the native 297 topics identified by `top2vec` in the population corpus. For each topic, the top three terms are included as well as the count of documents associated with that topic and the title of the most representative paper associated with that topic. 81

Chapter 1

Introduction

Product reliability weighs heavily on total product cost. Unreliable products drive warranty cost (Hussain and Murthy 2003) while trying to attain unrealistic reliability on new designs adds significant cost to the development process (Quigley and Walls 2003). Traditionally, the strategy has been to attempt prediction of reliability early to inform the design (Economou 2004). The field of reliability engineering was born with this initial purpose, though it has changed focus to align with the products it addresses.

Reliability engineering faces many of the same problems in 2023 it did in its founding in the 1950s. Most of these problems stem from the challenge of representing a system with minimal uncertainty early in the development process. Details of a design are unavailable early in the development process and therefore performance models must either account for the range of possibilities or be wrong. Increasing system complexity has highlighted how poor our predictions can be, since more complex systems compound uncertainties. Said another way, there exists a large set of possible designs that satisfy basic functional requirements. Each design will have unique properties, including reliability. Reliability model uncertainty stems from a combination of failing to identify the entire set of possible designs and from failing to attain sufficient model resolution on known designs.

As an example, consider a designing the first screwdriver. The basic functional requirements are that it must impart a certain torque on a fastener with a certain head geometry. Early in the development process, the set of possible designs would include everything from a traditional manual twist screwdriver to an electric screwdriver. The level of complexity and reliability between these two is vastly different. As the development process continues, requirements are refined, concepts are eliminated, and more design detail is added to the front-runner (or front-runners for set-based design). In this example, the company lands on a manual screwdriver but with a ratcheting mechanism. More complex than a two-piece manual screwdriver, the reliability can only be predicted once the ratcheting mechanism has been designed. After prototypes are made, the decision is made to use a sintering process for a component rather than machining. This change in process for scale manufacturing weakens the mechanism and changes the ultimate reliability. In the end, it is extremely unlikely the topology of the model (ratcheting screwdriver) and the parameters (sintered components) would have been accounted for at the beginning of the development process.

The field of reliability engineering was born to predict performance over time of what were at the time the most complex products yet created: vacuum tube electronic systems (Fazlollahtabar and Niaki 2017). While more complex than a manual screwdriver, these systems pale in comparison to the complexity of their modern replacements. In general, the constituent parts (components) of new systems are more reliable, but their count is greater therefore so is the number of possible interactions. Thus, reliability methods may have met the needs when the field started, but that is no longer the case with modern products.

We see two possible paths to address this: 1. devise more advanced system representations which can reduce and/or characterize uncertainty or 2. identify activities which improve reliability without the need to predict it. The former has the disadvantage of adding complexity to the project while the latter significantly extends the scope of reliability engineering. Over the course of this thesis we will answer which path is being pursued by reliability engineers.

1.1 Defining reliability

Reliability requirements are hard to generate in part because reliability is most evident in deficiency. In a vacuum, most stakeholders' utopia is perfect reliability, but this must be traded against cost and performance. Articulating these needs or expectations and appropriately assigning value to them them (while including uncertainty) is the tough part. Establishing clear definitions for reliability and reliability engineering helps, and provides a context for this research.

Bradley 2022 cites Robert Lussar's 1950s definition of reliability as "the probability that a system will continue to work, for a stated period of time, given defined operating conditions." Indeed, this is the most common formal definition of reliability, specifically calling out 1. probability, 2. time, and 3. operating conditions. For example, Raheja and Gullo 2012 describe reliability as "the probability of performing all the functions (including safety functions) satisfactorily for a specified time and specified use conditions."

Aven 2017 specifically tackles the field of "reliability engineering," noting that while there is no single definition, the following casts a wide enough net to get most of them: "reliability engineering is all activities carried out to obtain the right reliability of a technical system, through the various life cycle phases of the system." This is potentially a more appealing definition to the present work since it encompasses activities other than probabilistic modeling.

The certification body American Society for Quality (ASQ 2023) mirrors the formal definition of reliability, stating "Reliability is defined as the probability that a product, system, or service will perform its intended function adequately for a specified period of time, or will operate in a defined environment without failure." They offer a "certified reliability engineer" credential which identifies professionals "who understands the principles of performance evaluation and prediction to improve product/ systems safety, reliability and maintainability." The full body of knowledge includes "design review and control; prediction, estimation, and apportionment methodology; failure mode effects and analysis; the planning, operation and analysis of reliability testing and field failures, including mathematical modeling; understanding human factors in reliability; and the ability to develop and administer reliability information systems for failure analysis, design and performance improvement and reliability program management over the entire product life cycle."

What of seemingly related terms like durability, performance, maintenance, changeability, robustness, and quality? At a high level, reliability can be thought of as the addition of a time dimension to each of these. For example, with durability and robustness, we are interested in the resilience of a product in edge conditions. Reliability is concerned with how that resilience deteriorates over the life of the product, ultimately resulting in failure. Similarly, quality is a broad field which focuses on ensuring a product meets stakeholders' needs. Reliability is concerned with how specified properties change over time, often especially when they stop meeting those needs (Goel 1998).

1.2 Why reliability is hard

At a high level, we believe that the fundamental challenge of reliability is the need to predict emergent system behavior over long timescales with limited information. Reliability prediction is most valuable early in the development process, where changing course is less costly (Ulrich 2020). Conversely, it is much less impactful to know your product is unreliable after it's in customers' hands (Raheja and Gullo 2012). This is at odds with the reality that little is known about the performance of a system early in the development process. The outcome is great uncertainty at a time when reliability predictions would be most valuable.

The reliability community recognizes some of the challenges it faces. Enrico Zio is perhaps the most active researcher in this area. With E. Zio 2009, he aptly captured the authors' feelings regarding reliability. That is, we are still dealing with many fundamental problems in reliability while adding new complexity. Some of the key challenges discussed deal with soft failures associated with multi-state systems, network effects that are difficult to model, and software reliability. He proposes system health monitoring (prognostics) and other “dynamic modeling” techniques as a way to manage the increasing complexity of systems. He followed up that 2009 paper with another (Enrico Zio 2016) which further sharpens the focus on dynamic modeling as savior.

Plessis et al. 2014 notes many challenges which relate to the reliability discipline as a subset of systems engineering. Specifically, emphasis on stakeholder requirements, analysis of the system against those requirements, and general engineering management pervade his list.

Uniquely, Freeman, Medlin, and Johnson 2019 focuses on the challenges associated with experimental design. Two of the fundamental challenges align with those discussed by Zio, namely that complex systems are designed to operate in a wide range of conditions and may continue to operate in a degraded state without total failure. The former significantly expands the test variable space, while the later makes clear requirements mandatory.

Blanks 1998 demonstrates with specific examples the challenges associated with quantitative reliability prediction. They specifically target the conventional MIL-HDBK-217 prediction scheme (series-parallel models with failure rates from a standard table) with several examples of how off it can be. To counteract this, they suggest several approaches including the use of physics of failure. They also discuss many of the same issues with testing reliability raised by Freeman.

Aven 2017 frames the core problems in reliability engineering as 1. understanding system reliability and 2. determining the right reliability level. The former deals with issues of system representation and uncertainty while the later comes down to requirements.

1.3 What the literature suggests we do

Meeker and Hong 2014 discusses reliability in the context of big data, calling high-resolution operational information “the next generation of reliability data.” Specifically, they suggest application of dynamic modeling and prognostics. This same theme is discussed by Farsi and Enrico Zio 2019, where it's given the buzzword "industry 4.0."

Mentioned previously, literature on "design for reliability" advocates for perhaps the most radical changes from the status quo. Raheja and Gullo 2012 summarizes their recommended changes nicely with the following eight paradigms. They are paraphrased and reproduced here due to their importance.

1. Promise a minimum life, never use averages
2. Spend a lot of time on requirements
3. Measure all life cycle costs
4. Design for twice the promised life
5. Safety-critical components should be designed for four lives
6. Consider the full life cycle when making design trades
7. Design to avoid latent manufacturing flaws
8. Design for prognostics and health monitoring

1.4 What this thesis does

This work seeks to map the focus of reliability engineering literature over time. The purpose is similar to some of the challenges/future direction review papers previously discussed, but we add rigor through the use of the systematic literature review framework as well as unsupervised topic modeling to collect and map articles over time. Carnerud 2017 executed a very similar piece of research (unsupervised clustering to model quality and reliability topics over time). In their case, a single journal was studied with a small number (1,475) of articles. No secondary analysis or discussion was executed beyond the identification of topic trends, and the intermingling of quality and reliability topics clouds the results. In this thesis, we go deeper to provide a window into how the field has changed and where it might be heading, along with proposed rationale. After identifying trends, we map them to two dimensions: timing (proactive/reactive) and practicality (theoretical/applied). This higher-level analysis suggests where the field may be going beyond which topics are growing in popularity. The structure is as follows:

1. Introduction: context and motivation for this work.
2. Methods: definition of the research strategy and tools
3. Results: outcome and artifacts from execution of the research strategy
4. Discussion: placing those results in context
5. Conclusion: reflections on the research process

In the course of this work, we will address these core research questions:

RQ1. What topics (areas of common subject matter) comprise the body of reliability engineering academic literature?

RQ2. How has the volume of work in these topics changed over time?

RQ3. Are reliability engineering publications becoming more or less geared towards proactive versus reactive interventions?

RQ4. Are reliability engineering publications becoming more or less practical?

Chapter 2

Methods

While this thesis' primary goal is to address research questions pertaining to reliability engineering, exploration and development of a portable literature analysis procedure was a necessary secondary goal achieved along the way. This chapter describes this procedure as well as considerations from its development. The scripts used for analysis are included in the appendices with the intent that this process could be pointed at other similar problems and yield insight with minimal modification.

2.1 Overview

At a high level, our research questions (enumerated in 1.4) hinge on grouping papers. Because we intend to include a large number of papers, manual grouping is not feasible and may not be reliable. Thus, we rely on machine learning techniques. Figure 2-1 below schematically shows the overall analysis flow from research questions through analysis and result. Briefly, we consider the totality of academic literature to be our input. This is filtered through our query, which results in a corpus of article abstracts. From here, the corpus is analyzed via two parallel paths.

First, topic modeling identifies latent commonalities amongst abstracts and places them into groups. These groups are reduced to result in a more manageable number of aggregated topics, one of the main outputs. We also samples labeled abstracts and validate the modeling by blindly labeling the samples and calculating accuracy. We can also ascribe a timing or proactive/reactive score to the topics through a mapping to a standardized product development process, resulting in the second main output, a timing score.

The second path begins by classifying abstracts to assign a practicality score based on the type of examples mentioned (or not). As before, we sample a set of classified abstracts and perform a blind validation to assess classifier performance. In this case, we do not need an additional mapping step as the classification directly applies the practicality score.

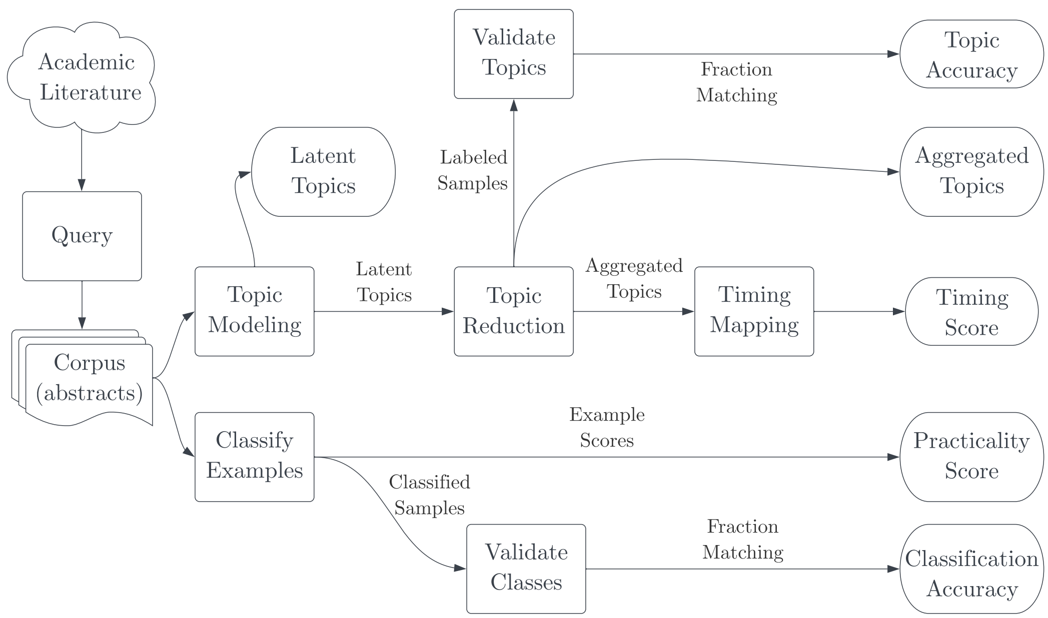

Figure 2-1: Overview of research procedure/pipeline. The process input is the entire body of academic literature while the core outputs are aggregated topics along with reactive/proactive and practicality scores. Secondary outputs are the full topic list and validation accuracy from both the topic modeling and classification processes.

2.2 Systematic literature review

The systematic literature review is a technique for synthesizing a body of scientific literature. It originates from medicine, where often many studies are combined to produce a more powerful secondary analysis. The field recognized the importance of rigor in this work (Tranfield, Denyer, and Smart 2003), ultimately resulting in the publication and acceptance of standards like the Cochrane Handbook (Higgins et al. 2019).

Fundamentally, a systematic literature review consists of 12 stages as discussed in Sundaram and Berleant 2022. They are:

1. Commissioning a review
2. Defining the research questions
3. Determining a protocol
4. Evaluating the protocol
5. Identification of research (corpus)
6. Selection of studies
7. Assessing study quality
8. Data extraction and monitoring
9. Data synthesis
10. Specify dissemination
11. Formatting the main report
12. Evaluating the report

2.2.1 Research questions

In accordance with the scientific process, a systematic literature review is structured to precisely articulate research questions prior to collecting documents. It is not necessary for the research questions to propose an outcome (e.g., a hypothesis), but clear success criteria should be baked in to shape the overall analysis.

As described in the introduction, the top-level purpose of this thesis is to understand how the reliability engineering discipline is maturing as it responds to challenges. Thus, we define three research questions (RQ):

RQ1. What topics (areas of common subject matter) comprise the body of reliability engineering academic literature?

RQ2. How has the volume of work in these topics changed over time?

RQ3. Are reliability engineering publications becoming more or less geared towards proactive versus reactive interventions?

RQ4. Are reliability engineering publications becoming more or less practical?

2.2.2 Data sources and query

The research questions above cast a wide net across all reliability publications. As a result, the primary metric for data sources is the count of applicable papers. Query authoring is inextricably linked with data source identification since data sources cannot be assessed without a pool of results.

Review of previous queries and related studies

It is worth learning from data sources and queries used by previous systematic literature reviews in the reliability engineering space. Though the research questions of these studies differ from the present, they still inform our strategy.

Forcina et al. 2020 performed a review of reliability allocation methods. Theirs was a more traditional systematic literature review wherein papers were manually gathered and read for analysis. They pulled data from Elsevier Scopus[1], ResearchGate[2], and Google Scholar[3] using the base query `"reliability allocation"` (in quotes). Their initial document population included 1,670 papers of which 93 were included in the final analysis. They further added subtopics with Boolean `AND` statements to group papers by specific allocation techniques. Their research question was limited to reliability allocation, specifically identifying which methods were most used and which industries were most discussed. They found that most reliability allocation takes place in the electronics industry, closely followed by the machinery industry.

[1]https://www.scopus.com/home.uri
[2]https://www.researchgate.net
[3]https://scholar.google.com

They did not attempt to map trends over time.

Righi, Saurin, and Wachs 2015 performed a review of resilience engineering, a field which could be considered a subset of reliability engineering. In their case, they used all available sources to them at their institution (16 journals and databases). Their query was `‘‘resilience engineering’’` (in quotes) and they noted this was selected as not using the quotes or using only “resilience” yielded papers outside their scope. Their document population included 637 articles, or 237 after de-duplication. They focused on categorization of papers, dividing them into two dimensions: domain and research area. They also did not attempt to map trends over time.

Ahmed, Raza, and Al-Anazi 2021 performed a review of fault analysis models, specifically those that rely on machine learning. They leveraged Scopus and Web of Science[4]. Their paper lists their query as `(‘‘fault’’ AND ‘‘reliability’’) OR (‘‘machine learning’’ AND ‘‘artificial intelligence)`. It is not clear how this query would produce the intended results, so we assume their actual query was `(‘‘fault’’ OR ‘‘reliability’’) AND (‘‘machine learning’’ OR ‘‘artificial intelligence’’)`.

Their initial document population included 552 articles, of which 243 were analyzed. They performed focused on citations, not only to identify the most influential authors but also to create clusters. That is, rather than clustering based on text analysis, a network diagram was created purely from papers citing each other. They labeled their clusters and found that detection and diagnosis represented the largest fraction of papers.

Selected query

Query authoring is an iterative process. Additional iteration is also required since different data sources provide different query fields and features. For example, many

[4] https://www.webofscience.com/wos/

database providers (e.g., Elsevier Engineering Village[5] or JSTOR Constellate[6]) include some degree of labeling or tagging. This in itself could be used to query an area and identify topics without additional analysis.

More concretely, Engineering Village calls its topics "controlled vocabulary." Most applicable to this thesis is the vocabulary term "reliability." Though there is a volume of papers resulting from the use of this term, subsequent execution of topic modeling shows that it casts too wide a net, including papers which mention or relate to reliability but do not focus on it. This is consistent with the fact that these vocabulary terms do not partition papers, they label them.

Ultimately, the search query of `"reliability engineering"` was selected (with quotes). All surveyed data sources interpret this as a literal string of "reliability" followed by a space, followed by "engineering." This query returns results which recognize reliability engineering as a discipline rather than only discussing reliability as a quality. Since this thesis is focused on mapping reliability engineering as a discipline, this produces a more targeted document pool. For this study, Elsevier's Engineering Village is used as the data source via its API.

2.2.3 Inclusion/exclusion criteria

In addition to data sources and search queries (effectively primary and secondary filters, respectively), systematic literature reviews explicitly define inclusion and exclusion criteria to permit further refinement of the document pool. For the purpose of this thesis, we attempt to limit these tertiary filters to the extent possible, relying mostly on data source selection and search query to provide desired results.

The criteria that are applied are necessary for successful application of aggregation and analysis tools. Namely,

[5] https://www.engineeringvillage.com/home.url
[6] https://constellate.org

1. Documents must be written in English to permit topic modeling. If documents were present in multiple languages, they would appear as separate topics.
2. Documents must include a publication year. This enables mapping topic trends over time.
3. Documents must include an abstract. These abstracts comprise the corpus for subsequent text mining.

2.2.4 Aggregation methods

Having defined data sources, search queries, and inclusion/exclusion criteria we are now left with a set of documents. Part of the magic of systematic literature reviews is that they distill these documents, providing insight greater than the sum of their parts. This thesis uses two machine learning techniques to perform this aggregation: topic modeling and zero-shot text classification. These are described in Sections 2.3 and 2.4, respectively.

2.3 Unsupervised topic modeling

2.3.1 Overview of techniques

This is not the first work to employ topic modeling in the context of a systematic literature review. Sundaram and Berleant 2022 notes that some degree of clustering provides value when considering which papers to include and exclude. More to the point, Carnerud 2017 applies clustering to map topics in quality and reliability literature over time, though with a narrower scope and less resolution than we will achieve. In their case, they used k-means clustering and identified 8 top-level clusters (quality management, quality functional deployment, process capability, quality, reliability, ISO standards, service, and six sigma). They performed an additional analysis of the reliability cluster, breaking it into 8 subtopics (fuzzy methods, reliability systems, sampling and inspection, software, maintenance, failure, warranty/repairs, and models). We consider the strengths of techniques used in those articles as well as broader trends in the NLP field to select techniques for this work. A full performance comparison between different techniques for topic modeling is beyond the scope of this thesis.

Latent Dirichlet allocation

First described in Blei, Ng, and Jordan 2003, latent Dirichlet allocation (LDA) is generally considered to be the baseline of topic modeling at the time of writing. It provides generally good performance with minimal overhead (e.g., training).

At a high level, LDA works by assuming documents can be described by a mixture of latent topics. Topics in documents and words in topics are iteratively allocated using Bayesian inference. Briefly, the probability of each document belonging to each topic is calculated, given the other documents already associated with that topic. The process continues until documents are no longer reassigned to new topics.

Practically, LDA can be executed with a number of Python libraries such as Gensim (described in Rehurek and Sojka 2010). Purpose-built analysis and visualization libraries such as LDAvis (Sievert and Shirley 2014) provide nearly turnkey analysis. Though execution of analysis is straightforward, it still requires nontrivial programming abilities as pre-processing (preparing the raw corpus to a form rife for analysis) is not automatic or built-in. Typically this will include removal of stop words (common words which won't be related to topics), lemmatization (collecting different conjugations of words such as "model", "modelling", and "modeller"), and generation of n-grams (grouping common words together that will have a different meaning, like "reliability engineering").

LDA was tested early in the course of this research. One of the main advantages was that standard analyses such as the aforementioned LDAvis rely on principal component analysis for dimensionality reduction. One possible framework for analysis would be to accept whatever principal components fall out of the model and attempt to ascribe meaning to them. For example, if one component had “management” on one side and “testing” on the other we could ascribe some concept of “closeness to the product” to that component.

Ultimately, the downfall of LDA for this study was that the number of topics must be prescribed. One could iterate through all possible numbers of topics, but the technique does not natively understand hierarchy. Practically, this means that if topics of interest appear at multiple levels (e.g., with different target numbers of topics) we cannot tell which subtopics exist for them. For example, if the corpus includes a mix of domain-agnostic articles along with a large number of articles that describe reliability tools in the context of a single domain (construction, for example), it’s possible the algorithm would identify “construction” as a topic rather than grouping those domain-specific papers based on tools discussed. The importance of this is made clear in Section 3.2.3.

BERTopic (sentence-transformers, UMAP, and HDBSCAN)

One of the perceived limitations of older techniques like LDA is that they do not consider word order. They are commonly called “bag of words” embeddings (embeddings are numeric encodings of words) since the words could be jumbled up and produce the same results. At the time of writing, the modern solution for this is called the transformer. First described in Vaswani et al. 2017, this is the concept that enabled large language models such as OpenAI’s GPT (Brown et al. 2020, Google’s PaLM (Chowdhery et al. 2022, and Meta’s LLaMA (Touvron et al. 2023). BERT is one of the earliest implementations of the transformer concept, first described in Devlin et al. 2019.

At a high level, transformers (and therefore BERT) extend the bag of words concept by weighting the importance of each word in a sentence and each sentence in a paragraph, et cetera. This is accomplished by iteratively assigning importance scores between each pair of words in the corpus. This enables the type of context-awareness that humans rely on to parse text.

BERTopic (Grootendorst 2022) combines the powerful embeddings of transformers with concepts from `top2vec`, discussed in the next section. Embeddings are generated using BERT, creating context-aware representations of each document. At this point, each document is represented by a high-dimensional vector. Dimension reduction is accomplished by UMAP (McInnes, Healy, and Melville 2018) to aggregate less unique dimensions and clustering with HDBSCAN (Rahman et al. 2016) identifies areas of high density in the population, or topics. Term frequency inverse document frequency (TF-IDF) is used to label the topics with the most unique words.

BERTopic has many qualities needed for this research, namely automatic topic identification and topic hierarchy. Ultimately it was not selected because the general purpose `sentence-transformer` model may not be sensitive to the jargon used in academic abstracts about reliability engineering.

Top2Vec (doc2vec, UMAP, and HDBSCAN)

Another modular topic modeling framework, `top2vec`, was first described in Angelov 2020. Though it initially used the `doc2vec` model for embeddings (Le and Mikolov 2014), it now supports newer alternatives including BERT. Also like BERTopic, dimension reduction is accomplished by UMAP and clustering with HDBSCAN. Also similar to BERTopic, hierarchical topic reduction is possible and the resulting hierarchy can be used to consider topics at multiple levels. Finally, unlike LDA, no preprocessing is required so the analysis process is streamlined. With all topics identified, the distance to each centroid for each document is calculated to establish relative similarity to each topic. Also like BERTopic, top terms in each topic are determined with TF-IDF and used to label the topics.

Since `doc2vec` is the main difference between BERTopic and `top2vec`, it's worth explaining the possible advantages for the present application. At the most basic level, `doc2vec` is an algorithm that extends the traditional bag of words technique by making it distributed. That is, rather than only calculating the probability of a word based on all of the words in a document, it calculates the probability of that word being near other words in a sentence in that document. Thus, the vector representation describes the entire document including its structure. Conversely, BERT (or transformers in general) represent a much more complex neural network architecture that can result in similar context-awareness like `doc2vec`, though with more limited scope and with significantly more overhead.

2.3.2 Topic modeling execution plan

We choose to use `top2vec` for this study as the `doc2vec` embedding should perform well with a moderate corpus of domain-specific terminology. If the corpus was very large and more varied, BERTopic or an alternative pre-trained embedding model (such as BERT) may be preferable. The anticipated outputs will be a list of topics identified amongst the documents, top words unique to each topic, documents associated with each topic, and for each document the distance to every other topic.

2.4 Text classification

The topic modeling discussed previously addresses the problem of identifying topics from a group of documents and assigning documents to those topics. Classification differs from this in that the topics (or classes in this case) are defined *before* any analysis begins. One canonical instance is sentiment analysis, for example determining whether product reviews are negative, neutral, or positive.

2.4.1 Overview of techniques

Text classification is a field essentially as old as natural language processing and as such we do not attempt to consider all options for this thesis. Rather, we bookend the space with the oldest and newest techniques to understand the field's evolution.

Bag of words methods

The most basic technique for classification is to create a list of terms for each possible class and check which list is most represented in a given document. If this sounds familiar it's because this is the essence of TF-IDF, described in Section 2.3.1. More advanced versions of this technique would generate the lists automatically, also known as training the model.

This requirement for training is the biggest downfall, particularly when your classifications are unique. Domains with significant interest (like sentiment analysis) have robust pre-trained models. For our application, we want the flexibility of classifying documents more loosely and without extensive training.

Large language models

Large language models (LLMs) are deep learning language models trained on very large datasets with the intent of understanding and generating natural language. When asked, OpenAI's GPT-4 (a LLM) explains, "A large language model is a highly complex, deep learning-based neural network trained on massive amounts of text data to understand and generate human language with impressive accuracy across various natural language processing tasks." At the time of writing, the significance of large language models (LLMs) continues to ripple through many domains. It should not be surprising that one domain is NLP itself, including classification. OpenAI recognized this application early on and initially provided a dedicated endpoint for classification tasks, later incorporating it into the more general fine-tuning API[7].

The power of LLMs is that they are capable of zero- or few-shot training for classification tasks. That is, you can feed a corpus into an LLM and ask it to assign a classification with no specific training beyond the base model. If needed, you can also provide some examples to shape its output (few-shot training), or many samples to optimize the model for your application (fine-tuning).

2.4.2 Classification execution plan

In the course of this research, both fine-tuning (using the OpenAI `text-davinci-003 model` and zero-shot prompting (using the OpenAI `gpt-3.5-turbo` model) were assessed for classification tasks. Validation of the selected model (comparing machine and human calculation) is discussed in Section 3.4.1. Ultimately, the fine-tuning model could not match the accuracy of the zero-shot prompt, so the latter was selected for this research.

As more powerful LLMs become available (e.g., `gpt-4`), the classification analysis can be rerun to gain better performance. Alternatively, the prompt (instructions provided to the LLM to execute the classification task) can be tuned as part of the analysis process to increase accuracy relative to a human-coded data set.

2.5 Assessment dimensions

Two research questions, **RQ3** and **RQ4**, do not immediately follow from topic modeling or text classification. Answering these questions requires an additional layer of inference beyond those topics/classifications. We can think of this additional layer as a mapping along a specified dimension. We define a unique dimension for each of **RQ3** and **RQ4**.

[7]https://platform.openai.com/docs/guides/fine-tuning/classification

2.5.1 Proactive/reactive

RQ3 can be interpreted as asking how many papers are published in reactive topics versus proactive topics. Through this lens, we can extend the results of topic modeling by assigning a proactive/reactive score to each topic. Topics that are purely reactive may have a score of 1 while topics that are proactive may have a score of 6. The primary limitation of this strategy is that it lacks grounding since scores are arbitrarily determined and scaled.

For this thesis, we establish an intermediate mapping from topic to traditional product development phase. Since traditional product development occurs in a linear fashion, this establishes a time dimension. Additionally, topics that occur prior to product launch can be considered increasingly proactive while those that occur subsequent to product launch are reactive.

The product development phases and their corresponding scores are shown in Figure 2-2. Example topics are included for reference. Actual topics and justification for their scoring is described are Section 3.6.

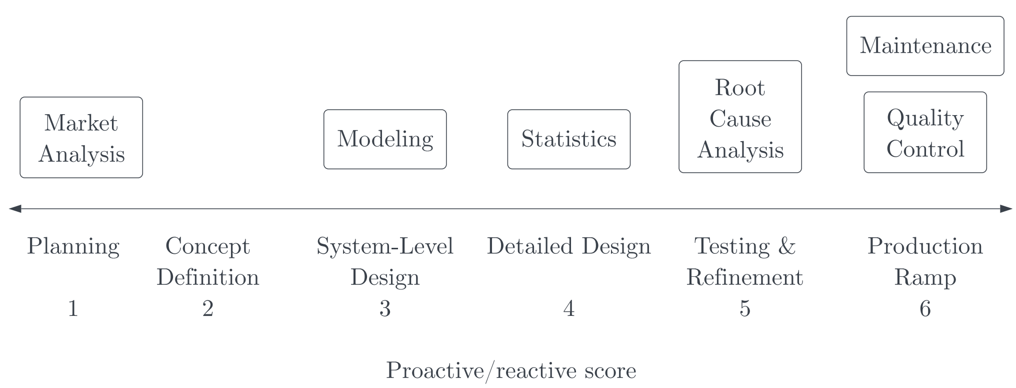

Figure 2-2: Traditional product development process including proactive/reactive scores and example topics. This process is adapted from Ulrich 2020. The main distinction is that we include all post-development activities (such as service) in the final category. Additionally, we note that most management activities are captured by the "Planning" phase, as described by Thornton 2021.

Note that some topics occur at different intensities throughout the process. For example, management permeates all phases. We choose to classify each topic in the earliest phase in which it provides significant impact. We justify these classifications in Section 3.6 through illustrative examples and the literature. Validation of this dimension occurs through validation of the topic modeling, since topics are transparently mapped to development phases.

2.5.2 Practicality

RQ4 has more latitude for interpretation. In this case, we choose to define practicality as "actionable" or "applicable." We establish the practicality dimension from purely theoretical to purely applied. Documents with abstracts that mention case studies or real examples are considered to be most actionable or applied with a score of 2. Documents with abstracts that mention any type of example are scored as moderately actionable with a score of 1. All other documents are scored as 0 and are assumed to be theoretical. This is rigorously defined by the system prompt described in Section 3.4. To make this more concrete, examples of each classification are shown in Table 2.1. In mapping categorical attributes like "has example" to a continuous spectrum (0 to 2), we are implying an equal impact going from no example (0) to illustrative example (1) as from illustrative example (1) to case study (2). This is a subjective scale. In the interest of transparency, we will display results with categorical rather than numerical labels.

We note that there are almost certainly documents which include examples but do not mention so in their abstract. At the same time, an abstract could mention a case study but execute it too poorly to be actionable. As we operate only on abstract text, we must accept this level of uncertainty. Future works could parse the full text of documents to better align dimensions like this.

To validate our classifications, we sampled 186 abstracts from the document population and assigned scores according to the above criteria. These scores are then

Classification	Score	Sample abstract
No example	0	During last decades, a specific approach called Universal Generating Function (UGF) technique was widely applied to MSS reliability analysis. The universal generating function technique allows one to find the entire MSS performance distribution based on the performance distributions of its elements by using simple algebraic procedures. © 2021, Springer Nature Switzerland AG.
Illustrative example	1	In the design of complex systems there is a great interest to know the relative importance of each of their elements. In this paper, we define a new method for measuring the relative importance of each element of the system. We have to specify that this paper concerns only non-repairable systems and components. We present a way of calculating the criticality of each component for a complex system no matter what the random distribution of the life of the component is. The paper also demonstrates a simple way of calculating how the system life improves when the life of a component is improved.
Case study	2	In this paper, we present a novel risk-based methodology for optimizing the inspections of large underground infrastructure networks in the presence of incomplete information about the network features and parameters. The methodology employs Multi Attribute Value Theory to assess the risk of each pipe in the network, whereafter the optimal inspection campaign is built with Portfolio Decision Analysis (PDA). Specifically, Robust Portfolio Modeling (RPM) is employed to identify Pareto-optimal portfolios of pipe inspections. The proposed methodology is illustrated by reporting a real case study on the large-scale maintenance optimization of the sewerage network in Espoo, Finland. [All rights reserved Elsevier].

Table 2.1: Example applications of the practicality classification scoring system. The abstracts selected were labeled with the respective scores by both a human and a large language model and are pulled from the document population. Abstracts are copyrighted by their publishers as indicated and reproduced here for fair use. Citations: Lisnianski, Frenkel, and Khvatskin 2021, Carot and Sanz 2000, Mancuso et al. 2016.

compared to the output of the text classifier and an accuracy metric is established in Section 3.4.1.

2.6 Limitations

While we believe this to be the most comprehensive survey of reliability engineering literature to date, we acknowledge several limitations of the methods described here. Due to the structure of the academic literature industry, we are not including all published works in our survey. That is, we include only documents to which we have access which may not be a representative sample of the whole. This is discussed more in Section 4.4.

The next impactful limitation is that we rely on paper abstracts to contain sufficient information to cluster and classify their associated documents. Implicitly then, we are relying on the documents' authors to accurately represent their contents. This is likely the case regarding topic modeling (and therefore the timing or proactive/reactive dimension), but perhaps more tenuous when considering classification based on whether an example is mentioned in the abstract.

Finally, our mapping between topics and timing score are subjective, though based on the literature. Our decision to represent topics which span multiple phases in their first phase is an arbitrary decision, one could just as easily pick the weighted average, weighting by perceived importance.

Chapter 3

Results

We now execute the methods described in the previous chapter and consider their output. Specifically, we describe the documents collected through the systematic literature review process, perform an analysis of those documents through topic modeling and classification, and apply of second-order scoring (e.g., timing and practicality). Interpretation and ascription of meaning to these results is reserved for Section 4, discussion.

3.1 Document population

The overall document population described in this thesis includes 30,543 papers. Of these, 20,634 are journal articles, 7,764 are conference papers, with the balance being book chapters and miscellaneous reports. These papers span publication years from 1955 through 2023. The most papers were published in 2022, with a total of 2,141. A plurality (11,789) of these papers come from the Reliability Engineering and System Safety journal. The next most popular source is Quality and Reliability Engineering International with 5,829 papers. Each of the other sources contribute fewer than 1,000 papers.

Paper	References	In population	Coverage
E. Zio 2009	170	24	14%
Forcina et al. 2020	119	11	9%
An, Kim, and Choi 2015	132	5	4%
Emura and Michimae 2022	81	4	5%
Nor, Pedapati, and Muhammad 2021	160	41	26%
Maurya and Kumar 2020	96	22	23%

Table 3.1: The reference lists from six survey papers were compared to the document population to estimate coverage. Titles were matched with a simple character ratio (number matching/total number), with values above 0.9 considered matching. Note that this result indicates a significant lack of coverage across the reliability engineering literature.

3.1.1 Validation of document population

To validate our document population, we manually select six recent review papers from the reliability engineering field. These papers were selected as they represent different specialties in the field and include robust reference lists. We then search the document pool for their references and determine what fraction of references are present in the document pool. These results are shown in Table 3.1.

All things being equal, higher fractions of represented references are seen as a validation of this thesis' document pool. That the highest fraction of represented references is only 26% is disappointing, though not unexpected given the challenges of securing a data source. As we discuss in Section 5.3, this aspect represents the greatest opportunity for future work as piping in more robust data sources is nearly turnkey.

Another note is that all of these papers cite references outside of the reliability engineering field, so we should not expect 100% coverage. Particularly in fields which are more mathematical (e.g., modeling and statistics as in the case of Forcina et al. 2020 Emura and Michimae 2022, respectively), there is a high chance of referencing mathematical or computer science publications which would not be included in the reliability engineering document population.

3.2 Topic modeling results

With a document pool we are now able to begin the second phase of analysis, topic modeling. As discussed in Section 2.3.1, we elect to use the `top2vec` library for this purpose. Our corpus is the set of abstracts from the 30,543 papers. This corpus is fed into the topic modeling function, with identified topics along with how close each paper is to each topic as output. The algorithm identified 297 topics amongst these papers which are listed in Appendix A. These 297 topics were identified with default `top2vec` parameters, namely `topic_merge_delta=0.1`, equivalent to the epsilon parameter of HDBSCAN. The merges topics which have a cosine distance of less than 0.1. Another key parameter left default is `min_count=50`, which filters out infrequent words.

3.2.1 Hierarchical topic reduction

Following topic modeling, hierarchical topic reduction is performed to aggregate topics into larger and potentially more meaningful groups. The reduced topics and paper counts are enumerated in Table 3.2 and shown graphically in Figure 3-1. The three terms included in each topic are the most frequent words unique to that topic (TF-IDF), while the label is an human interpretation of the documents in that topic. This list of topics addresses **RQ1**, "what topics (areas of common subject matter) comprise the body of reliability engineering academic literature?"

We note that while we spend the remainder of this work discussing documents as if they belong to a single topic, we anticipate they are mixtures of topics. The mixture of a given paper can be described by the distance of it to the centroid of each topic in the dimension-reduced vector space. We choose to label it with the topic it is closest to. For example, consider Pascual et al. 2017's "Optimal repairable spare-parts procurement policy under total business volume discount environment."

Label	Top terms	Paper count
Software	bug, developers, software	4,211
Management	engineering, organizations, development	4,064
Statistics	weibull, estimation, estimators	3,737
Modeling	cut, minimal, binary	2,841
Physics of Failure	electron, silicon, oxide	2,542
Risk Analysis	human, experts, hra	2,479
Maintenance	preventive, replacement, maintenance	2,414
Quality Control	charts, chart, shewhart	2,333
Infrastructure	infrastructure, transportation, disruptions	2,204
Prognostics	rul, prediction, prognostic	1,865
Nuclear	nuclear, plants, reactor	1,853

Table 3.2: Topic modeling results. The model automatically identified 297 topics; these 11 were generated using hierarchical topic reduction. Target reduced topic numbers between 5 and 20 were surveyed and 11 was qualitatively determined to be the optimal amount of aggregation to elucidate the maximum number of relevant top-level topics. The full list of topics can be seen in Appendix A.

This paper is a 38% match to the Maintenance category and so it is counted amongst those ranks. However, it is also a 22 % match to the Management category and a 20 % match to the Modeling category. Indeed, the paper appears roughly between the middle of these three clusters in the point cluster plot (Figure 3-2).

We note that three of the topics in this group are domain specific: nuclear, infrastructure, and software. Since we are unable to assign a clear timing element to these papers (and indeed, subtopics among each of these likely have their own timing elements), we elect to perform a separate analysis on each of them, described in Section 3.2.2. For the first pass, we will only consider the eight remaining topics. Conveniently, this places us within the magic range of 7 ± 2 described by Miller 1956.

These 8 remaining topics are prognostics, statistics, maintenance, quality control, management, physics of failure, modeling, and risk assessment. There are 22,275 documents in these topics. As part of the `top2vec` work flow, we can visualize the clustering of documents shaded by these topics using uniform manifold approximation and projection (UMAP) as described in McInnes, Healy, and Melville 2018. This is

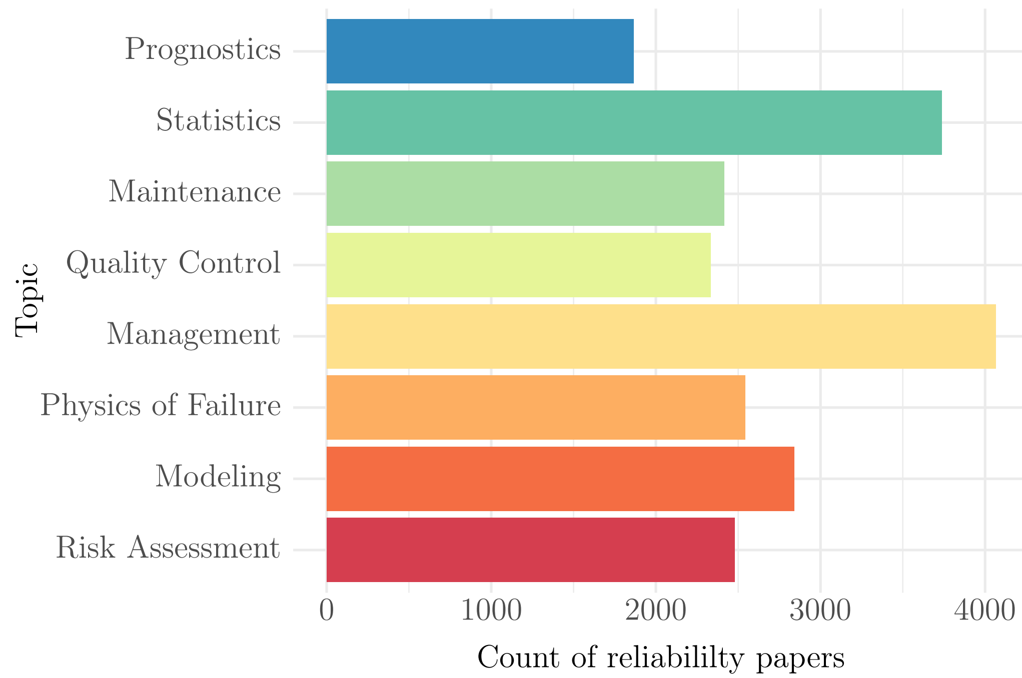

Figure 3-1: Top-level topic counts in reliability engineering papers (cumulative). All documents are included except for those in domain-topics.

shown in Figure 3-2. In this case, we are showing all documents over all time. Time evolution could be shown in this form, though we choose to show it in summarized charts (by count) for clarity.

Software

This topic focuses on the related but unique field of software reliability, also including site reliability engineering. Subject matter ranges from development and testing (hence the "bug" and "developers" terms) to deployment of software. Subtopics for this topic are shown in Table 3.3. The most representative paper (closest to the cluster centroid) is Luo et al. 2021's "A Runtime Monitoring Based Fuzzing Framework for Temporal Properties."

Management

The second most populous topic included papers which describe engineering management practices which impact reliability. We note that the relatively broad term of "engineering" is represented as the most frequent term. Since this term was likely present in other areas, that it showed up here indicates it must have been truly over-represented. The most representative paper is Leech 1985's "An introduction to quality assurance in the information processing industry."

Statistics

The authors expected this topic (with modeling) to have the largest document population, but it ranked third. This topic included papers discussing reliability from a probabilistic standpoint, focused on prediction and estimation. That the oft-used Weibull distribution appears as the most frequent term is not a surprise. The most representative paper is Bhattacharya and Pradhan 2018's "Bayesian design of life testing plans under hybrid censoring scheme."

Modeling

Statistics are often applied to modeling, which is the next topic identified. These papers discussed representation of complex systems in an effort to model and predict reliability. (Minimal) cut sets and binary operations are frequently used in reliability modeling, so they appear in the term list. The most representative paper is Alkaff 2021's "Discrete time dynamic reliability modeling for systems with multistate components."

Physics of failure

In terms of trends, the authors expected papers related to physics of failure to show the highest growth. These papers focus on descriptions of specific failure mechanisms and their effect on reliability. Because these are domain- and application-specific it is somewhat surprising they were aggregated. The top terms show that most of the papers related to electronic component physics of failure, though other papers in the topic discussed plastics and metallurgical crack propagation. The most representative paper is Liu et al. 1999's "The properties of 2.7 eV cathodoluminescence from SiO2 film on Si substrate."

Risk analysis

This topic seems closely related to the management topic, but is unique in its focus on safety and risk. The top terms include "hra" (hazard and risk assessment) and "human" indicating its focus on the role of operators rather than on hardware failure as is typically the focus of reliability engineers. The most representative paper is Ralph L. and Winterfeldt 1988's "Probabilities are useful to quantify expert judgments."

Maintenance

This topic deals with service or maintenance of equipment, related to reliability engineering by way of the fact that failures necessitate service and by the relatively new field of reliability-centered maintenance. Specifically, we note that the most common term is "preventative," so the papers are focused on increasing reliability through actions which prevent systems from failing. The most representative paper is Sheu, Lin, and Liao 2006's "Optimum policies for a system with general imperfect maintenance."

Quality control

As discussed in the introduction, the line between quality and reliability can be blurry so it is expected that some papers discussing quality would also discuss reliability engineering. Those included in this topic are limited to process quality, specifically techniques which measure and track process performance. The most representative paper is Yang and Cheng 2011's "A new non-parametric CUSUM mean chart."

Infrastructure

This is a domain topic that primarily includes papers discussing transportation and defense infrastructure reliability. Subtopics for this topic are shown in Table 3.4. The most representative paper is Adachi and Ellingwood 2008's "Serviceability of earthquake-damaged water systems: Effects of electrical power availability and power backup systems on system vulnerability."

Prognostics

Prognostics and health management (PHM) is a topic that the authors expected would be small but growing, as is represented here. Papers in this topic focus on on-line assessment of systems for probability of failure as well as prediction of remaining useful life (RUL). The most representative paper is Li, Zhang, and Ding 2019's "Deep learning-based remaining useful life estimation of bearings using multi-scale feature extraction."

Nuclear

Another domain topic, this one focuses on reliability in nuclear power plants. Subtopics for this topic are shown in Table 3.5. The most representative paper is Kelly 1992's "Probabilistic analysis of flow control as an alternative to level control for BWR ATWS."

Beyond being a pretty picture, this visualization does show clear subtopic clusters within each of the eight reduced topics. This aligns well with the fact that the algorithm identified 297 topics as discussed previously.

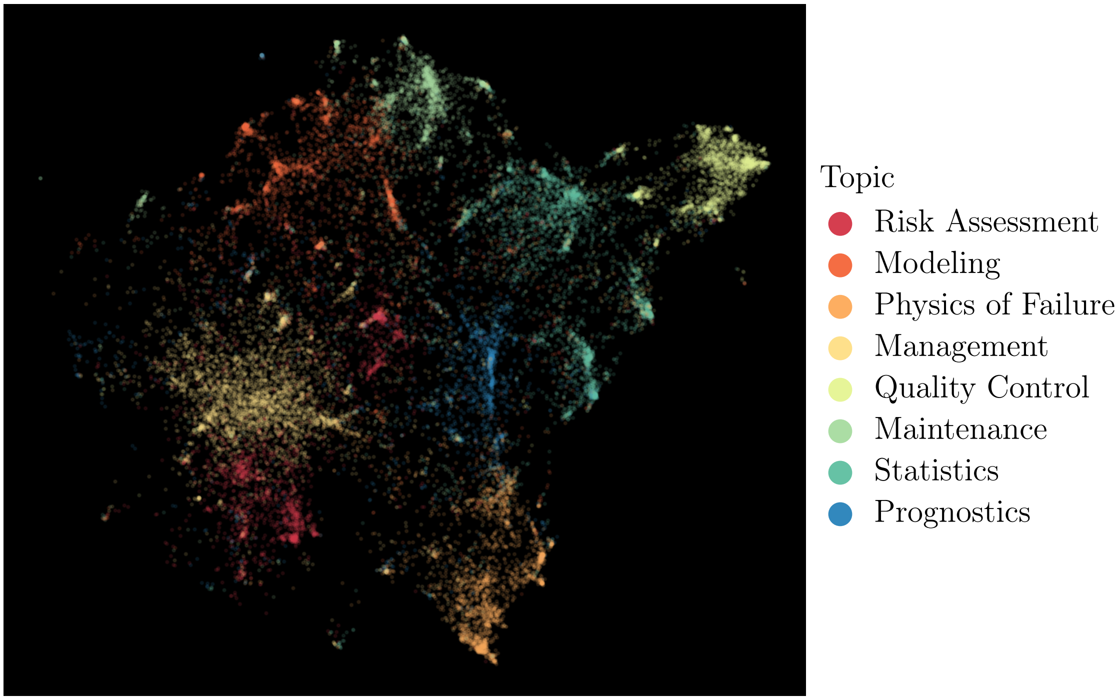

Figure 3-2: Two-dimensional representation of top-level reliability topic clustering using the UMAP algorithm (McInnes, Healy, and Melville 2018), the same used to identify topics using `top2vec`. Each dot represents a publication. There are 22,275 publications represented in this visualization. Note the visible sub-clusters among each topic which hint at the full 297 topics. We can also see that similar topics (like risk analysis and management) tend to appear geometrically closer than those which we might expect to be more dissimilar (like quality control and physics of failure).

3.2.2 Domain sub-topics

We can repeat the same `top2vec` work flow for each of the three domain topics, identifying all topics within each and subsequently aggregating them into meaningful reduced topics. These are shown in Tables 3.3 (software), 3.4 (infrastructure), and 3.5 (nuclear). Visualizations for these domain sub-topics are included in Appendix B.

3.2.3 Selected reduced topic hierarchy

As noted previously, hierarchical topic reduction requires some degree of subjectivity to assign a target number of topics. To understand how we arrived at 11 topics, we

Top terms	Paper count
formal, safety, elsevier	1,102
apps, false, positives	803
test, suite, coverage	782
releases, defect, predictions	772
service, rejuvenation, cloud	752

Table 3.3: Software topic sub-topics as identified with a target 5 topic hierarchical reduction. We note that these could be mapped onto a timing dimension like the top-level topics, though the dimension would be unique to software development.

Top terms	Paper count
concrete, corrosion, reinforced	439
ieee, problem, scheme	285
resilience, recovery, functionality	285
sea, accident, accidents	274
power, energy, outage	272
risk, pipeline, safety	240
cascading, node, network	236
attacker, defender, contest	173

Table 3.4: Infrastructure topic sub-topics as identified with a target of 8 topic hierarchical reduction. These appear to include numerous further domain-specific topics, so assignment of a timing score would be difficult without probing at a lower level.

Top terms	Paper count
maintenance, paper, is	431
limitations, risk, ensure	281
simulation, monte, carlo	257
escalation, domino, industrial	226
waste, geologic, repository	179
real, tree, synthesis,	173
operator, errors, operators	171
bwr, core, pwr	135

Table 3.5: Nuclear topic sub-topics as identified with a target of 8 topic hierarchical reduction. We note that these could be mapped onto a timing dimension like the top-level topics, though the dimension would be unique for nuclear plant development.

will describe more extreme cases of 3 and 20 topics.

Constraining the topic reduction to 3 topics, the topics are "charts, estimators, multivariate", "repairable, repair, preventive", and "software, virtualization, developers". The main issue here is that the large volume of software papers has resulted in less fidelity of other topics. The first topic appears to combine statistics, quality, and modeling, while the second deals with maintenance. Clearly there is cause for more than 3 topics.

With 20 topics, the topics are:

1. bug, developers, bugs
2. charts, chart, shewhart
3. companies, customer, market
4. disruptions, disruption, infrastructure
5. electron, oxide, silicon
6. engineering, book, topic
7. experts, linguistic, opinions
8. human, hra, cognitive
9. nuclear, plants, reactor
10. pipelines, ship, corrosion
11. preventive, replacement, maintenance
12. redundancy, rrap, solve
13. repairable, markov, repair
14. rul, prediction, prognostics
15. surrogate, sobol, kriging
16. trees, tree, boolean
17. uml, language, checking
18. vibration, rotor, rotating
19. virtualization, virtualized, vm
20. weibull, censoring, censored

Looking at this list, there are several topics which would be readily aggregated when looking at our timing dimension. For example, "trees, tree, boolean" and "uml, language, checking" would both relate to modeling. At the same time, topics like "engineering, book, topic" and "experts, linguistic, opinions" cut across many domains and tools and are therefore impossible to place in the proactive/reactive (timing) dimension. We do see many of the same topics in our 11-topic reduction, which gives credence to the importance of those topics.

3.3 Trends in topics

With the topics identified and assigned to papers, we can answer **RQ2** (How has the volume of work in these topics changed over time?) by looking at how the volume of reliability engineering literature in these topics has changed over time. This result is shown in Figure 3-3. This addresses **RQ2.**, showing that the volume of reliability papers is rapidly growing in nearly all areas.

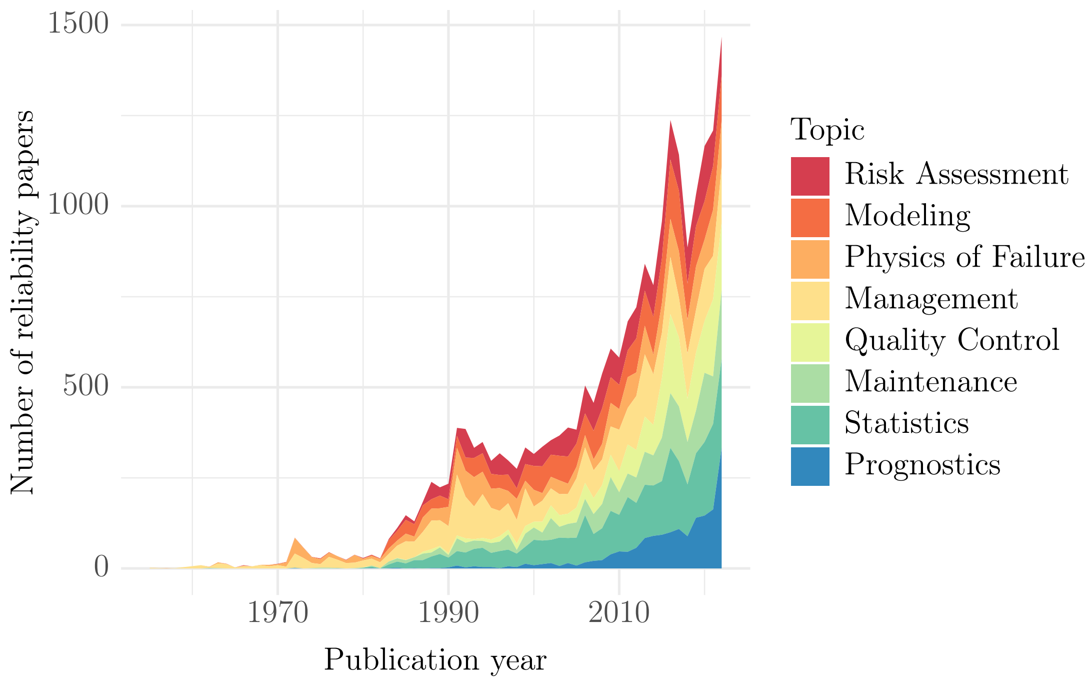

Figure 3-3: Top-level topic count in reliability engineering papers over time. Topics are stacked according to count of publications in 2022. Note the overall exponential growth. Prognostics had the largest count of publications in 2022.

One thing that stands out from looking at document counts over time is the significant growth seen in all topics. For the 8 topics considered, Modeling showed a 53 % average annual growth, Maintenance 42 %, Prognostics 40%, Management 36%, Risk Assessment 29%, Physics of Failure 28%, Quality Control 27%, and Statistics 19%. Overall, this is a 35% average annual growth rate. To remove this dominant factor and tease out the signal of how topic popularity is changing, we can consider the fraction of each topic represented in each year. This is visualized in Figure 3-4.

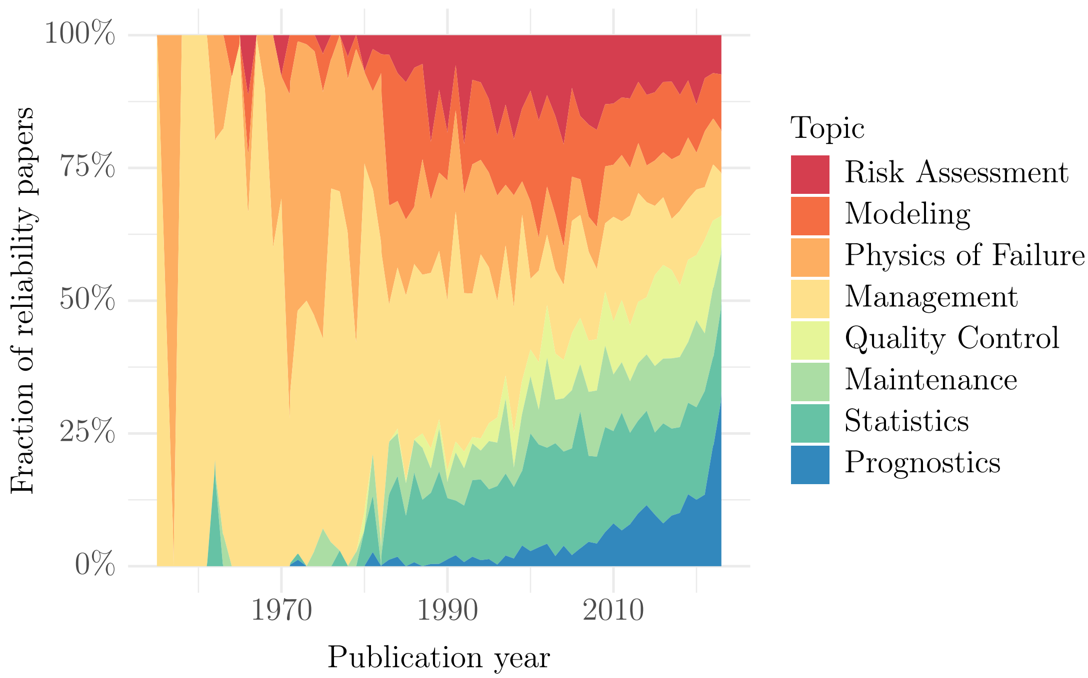

Figure 3-4: Top-level topic proportion in reliability engineering papers over time. Topics are stacked according to count of publications in 2022. Note that prognostics represented the largest fraction of publications in 2022.

From this plot, we can see clear differences in growth for each topic, represented by their differing slopes. We can directly visualize this growth by plotting the year-over-year growth in the number of publications for each topic. Since this number varies significantly and we are interested in trends, we only plot a smoothed version of this growth rate in Figure 3-5.

3.4 Zero-shot classification results

To identify practicality of papers, we elected to use zero-shot classification. Recall from Section 2.4 that this technique involves prompting a large language model (LLM) with a system prompt followed by a query prompt. For this thesis, we use the following system prompt:

```
Provided academic abstracts as prompts, classify them as one of the
following:
    0: no explicit mention of an example
    1: mentions an illustrative example or demonstration, or
    2: explicitly mentions a case study.
If an abstract mentions a case study and an example, classify it as 2.
Respond only with the classification ID number (0, 1, or 2).
```

This provides the LLM with enough instructions to understand the context of the query prompt and ensures that it only responds with a classification ID to limit the number of response tokens.

For this thesis, we use OpenAI's `gpt-3.5-turbo` model via the OpenAI API. To confirm the prompt is appropriate, we perform manual validation as described in Section 3.4.1.

3.4.1 Validation

We validated the LLM's zero-shot classification performance by manually scoring a set of 186 randomly sampled abstracts from the document population. These scores were all assigned by a single human (the author). Next, the documents were scored using the LLM. The LLM's classification matched that of the human for 70% of the sampled papers. This accuracy is on par or better than other published examples (Contreras et al. 2022 referenced a 63% accuracy in simple positive/negative sentiment analysis and Balkus and Yan 2022 achieved 73% accuracy with 10,000 training samples using a similar LLM-based approach, though much more structured).

3.5 Trends in classifications (practicality)

Having validated the LLM's classification capability, we can put it to work classifying the full set of documents based on practicality. Recall that the practicality dimension has three defined levels based on the type of examples discussed (or not) in document

Topic	Phase	Rationale
Risk Assessment	Concept	Requires CONOPs
Modeling	Sys. Design	Requires notional design
Physics of Failure	Sys. Design	Requires notional design
Management	Planning	Permeates development process
Quality Control	Production	Pertains to production of units
Maintenance	Production	Pertains to fielded units
Statistics	Detailed Design	Requires data from prototypes
Prognostics	Production	Pertains to failure in fielded units

Table 3.6: These are the timing or product development phase dimension assignments for each topic. Each was assigned based on discussions of best practices as described in Raheja and Gullo 2012. Numeric scores were associated with each phase (1 through 6) as described in Section 2.5.1. The scores are linear and used only for calculation of mean product phase in visualizations.

abstracts: 0: no mention of an example, 1: illustrative example, or 2: case study or practical example. We find that 49% are classified as illustrative example, 38 % are classified as case study, while the remainder do not mention an example. Using these numeric classifications, document publication dates, and their topics, we can visualize the trend in practicality over time. This is shown in Figure 3-6. This addresses **RQ4.**, indicating that reliability publications are becoming more practical over time.

3.6 Trends in proactive/reactive

In order to leverage the scale developed in Section 2.5.1, we must place the 8 identified topics on the timing dimension. Recall that a topic should be mapped to the earliest phase in which it makes a significant contribution to product reliability. Through this lens, we can assign the topics as shown in Table 3.6. Finally, we can plot the trend over time as shown in Figure 3-7. This addresses **RQ3.**, indicating that reliability publications are becoming more reactive.

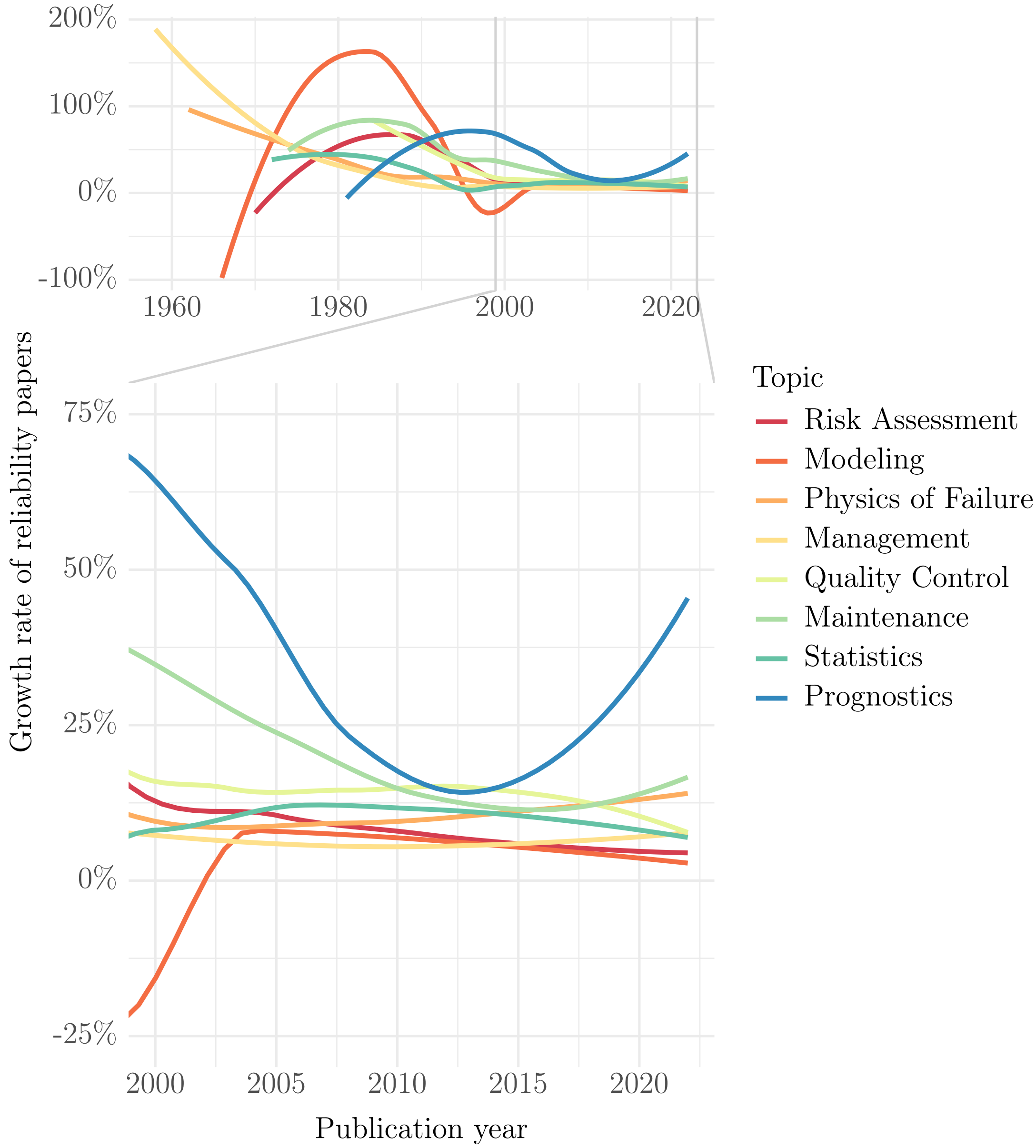

Figure 3-5: Year-over-year growth rate of top-level topics in reliability engineering literature. Traces have been smoothed with locally estimated scatter plot smoothing (LOESS) to extract trends. The inset highlights trends between 2000 and 2023. Note that prognostics consistently the highest growth rate which continues to increase. Early growth rates were highly volatile due to low absolute paper counts. This volatility coupled with the smoothing algorithm produces artifacts such as the extremely negative growth rate of modeling.

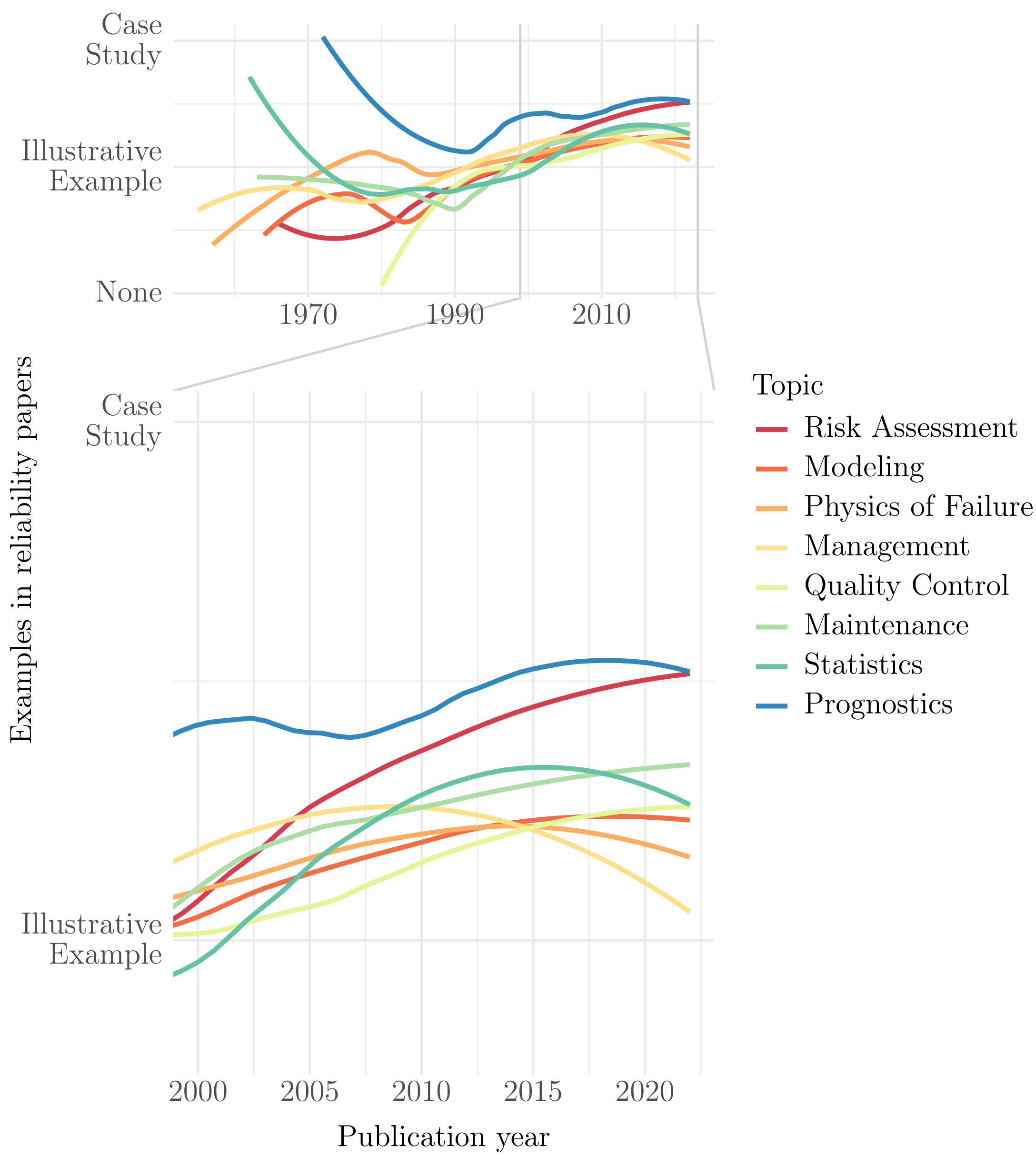

Figure 3-6: Mean example content for top-level topics in reliability engineering literature over time. Publications' abstracts were assessed to determine whether no example, an illustrative example, or a case study was present. These categories were assigned values and those values were averaged for each year and each topic. Traces have been smoothed with locally estimated scatter plot smoothing (LOESS) to extract trends. The inset highlights trends between 2000 and 2023. Note that in general, publications have been increasing in practical content.

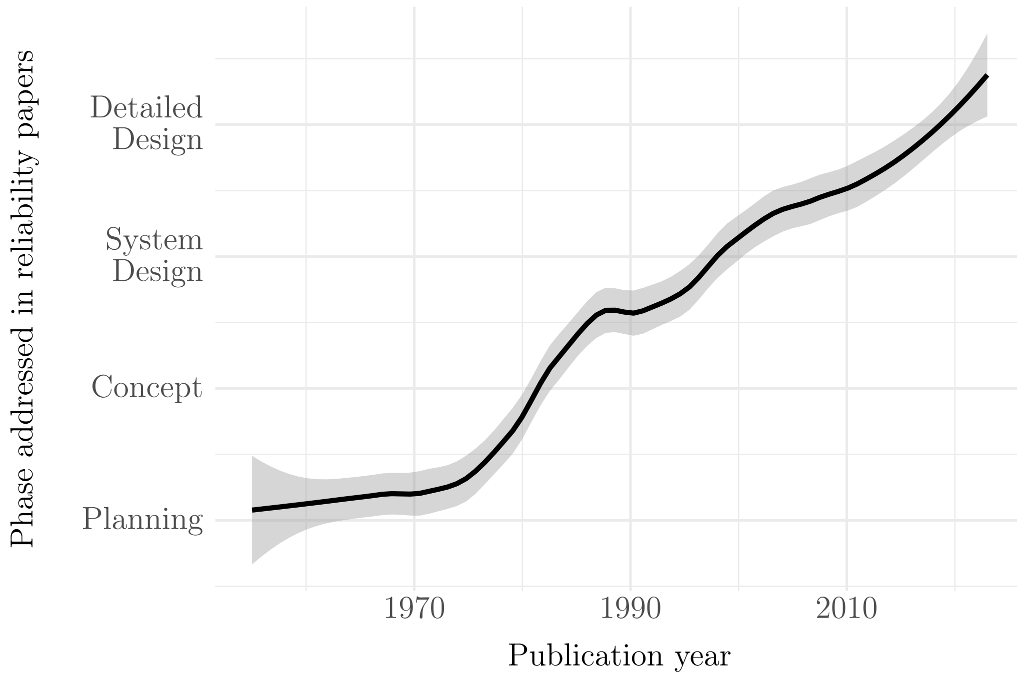

Figure 3-7: Mean timing period (product development phase) associated with the main topic of reliability engineering literature over time. Topics are correlated with phases and a weighted average is used to summarize the topic mix of papers for each year. The trace has been smoothed with locally estimated scatterplot smoothing (LOESS) to extract trends. The shaded region represents a 95% confidence interval for the smoothing. Note that reliability publications increasingly focus on activities which occur later in the product development process.

Chapter 4

Discussion

We can now leverage the results and analysis to identify patterns and ultimately answer our research questions. In addition, we reflect on the execution of the study and specifically challenges posed by the state of academic literature.

4.1 Analysis of topics

The 8 non-domain-specific topics identified in this study can be compared with those used elsewhere to partition the reliability engineering field. Previously mentioned, Carnerud 2017 is the most similar study to the present since it employs unsupervised clustering. The key difference is that their document population included quality papers, hence their topics were skewed in that direction. As mentioned in Section 2.3.1, they found the reliability cluster had the following topics: fuzzy methods, reliability systems, sampling and inspection, software, maintenance, failure, warranty/repairs, and models. There is actually quite a lot of overlap between their findings and those of the present study. For the topics which are not clear matches, sampling/inspection maps to our quality topic and fuzzy methods maps to statistics. The remainder translate directly into our topic list. The primary gap appears to be higher-level management topics (including risk assessment). This may be because those types of papers aren't published in the studied journal or more likely because management topics were included in one of their top-level clusters.

Another possible angle for checking for agreement is to look at the ASQ reliability body of knowledge[1]. There we find the following sections: management, statistics, design and development, modeling and predictions, testing, maintainability, and data collection. Again, these align well with the topics found in the present study. Testing and data collection are the most unique areas here and it would be difficult to say where they might fit into our topics. In terms of omissions, the lack of a prognostics section aligns with the idea that industry lags academia regarding the forefront of the field.

Finally, we can also look to reliability text books to see how the topic is introduced to students and new practitioners. Breneman, Sahay, and Lewis 2022 includes the following chapters: 4 on probability and statistics, testing, failure modes and effects analysis, loads and capacity, maintenance, failure interaction, and safety. There is a clear bias in this text towards probability and statistics, with many topics uncovered in this study not represented.

4.2 Interpretation of trends

At a high level, the most noticeable trends are that reliability engineering literature is becoming both more reactive and more practical. As discussed in Section 3.6, the topics associated with late product development phases have been increasing in popularity since 1955 (the earliest year in our document population). In fact, other than a brief period around 1990, papers have consistently shifted towards later development phases.

What follows is our hypothesis for why this is. First, we recognize that while it is important to have strong management early on and leverage modeling to assess

[1] https://asq.org/cert/reliability-engineer

concepts, there is a large amount of uncertainty which makes reliability prediction (and therefore distinct action) challenging early in a development project. That is, there are too many unknown unknowns early in a project to adequately predict reliability. It is challenging enough to predict a product's performance during the concept stage, so extrapolating that performance out over potentially many years in uncontrolled contexts is even more difficult.

There is a distinction here that these early practices are not necessarily considered less important, just that they were foundational to the field and do not necessitate literature to be improved upon. As discussed in Raheja and Gullo 2012, the rather recent "design for reliability" movement is not just about the design of the product, but rather doing what you can *at every phase of the development process* to improve reliability.

From a project risk perspective, it is clearly preferable to build confidence in the product's reliability as early in the development project as possible. Thus, we would expect that most effort in the reliability engineering field would be aimed at the early phases. At some point researchers realized that the metaphorical well was drying up and thus they moved onto the next phases to understand what could be done. This process repeated until the recent advent of reliability-centered maintenance and prognostics and health management began addressing the post-sale portion of the product life cycle. Thus, we can hypothesize that although intuition and general interest would prioritize early-stage reliability work, the field recognizes the fruitless nature of this and instead focuses on more successful late-stage methods.

It seems unlikely that in 20 years the only new reliability literature will be on prognostics and maintenance, thus we can't expect this trend to continue. We imagine the curve in Figure 3-7 will level off soon, and perhaps decrease as latter-phase areas of research mature.

The trend in practicality is more straightforward and we believe linked to the aforementioned trend towards later development phases. Simply stated, as products enter later phases in the development process they are more formed. This means more details about the product and execution of a reliability tool matter and therefore are likely to be shared in the literature. Said another way, it is much easier to write a robust paper on statistical techniques with concepts discussed via mathematical proof versus discussing a maintenance program without any context for the product specifics.

We can also consider growth trends for individual topics. Prognostics is of interest, showing a significant swell in the late 1990s. We believe this relates to the proliferation of the internet and greater ability for systems to report on their condition. The current upswing may be related to new capabilities afforded by deep learning (the most representative paper for this topic happens to be about deep learning, see Li, Zhang, and Ding 2019).

Another interesting trend is that of modeling, which saw an extreme peak in the 1980s before leveling out. As with Prognostics, we believe this is due to technological enabling. Computers became much more useful for semi-complex system modeling around this time so it makes sense that reliability modeling would show growth. We do note that the large initial negative growth is more of an artifact of the very small number of papers in those early years and do not ascribe any specific meaning to this trend.

4.3 Answering the research questions

We can also reflect and discuss the research questions posed in Section 2.2.1.

RQ1. What topics comprise the body of reliability engineering academic literature?

We found that reliability engineering literature is comprised of at least 279 topics as enumerated in Appendix A. From these granular topics, we find there to be 11 aggregated topics:

1. Software*
2. Management
3. Statistics
4. Modeling
5. Physics of failure
6. Risk analysis
7. Maintenance
8. Quality control
9. Infrastructure*
10. Prognostics
11. Nuclear*

Topics with an asterisk were determined to be domain-specific artifacts of the aggregation process and not associated with specific reliability tools or techniques.

RQ2. How has the volume of work in these topics changed over time?

We found that all topics show consistent positive growth over the last 20 years (35% on average). We note that this is far in excess of the overall scientific and engineering publication growth rate (approximately 4% per Larsen and Ins 2010). Currently, the prognostics topic shows the highest level of growth (over 30%) while modeling shows the lowest (below 5%). These trends can be seen in Figure 3-5.

RQ3. Are reliability engineering publications becoming more or less geared towards proactive versus reactive interventions?

Using our assignments of topics to development phases discussed in Section 3.6, we see that publications are increasingly geared towards tools and techniques that occur later in the development process. This can be considered "more reactive," though it is arguable since often the activities are planned near the beginning of the project. They only occur near the end since they require a more mature product. This overall trend is visualized in Figure 3-7.

RQ4. Are reliability engineering publications becoming more or less practical?

We found that nearly all topic areas are increasing in practicality, assessed by whether their abstracts indicate case studies versus not mentioning an example. The exceptions are the management and statistics topics which showed decreasing practicality over the last 5 to 10 years. Risk assessment showed the greatest increase. These trends are visualized in Figure 3-6.

4.4 Reflection on text mining academic literature

One of the main limitations and therefore disappointments of our results was the lack of coverage highlighted in Section 3.1.1. We achieved between 4 and 26% coverage of references in those review papers in part due to the fact that some of those references would be outside the reliability engineering field, but also due to the state of structured data access of academic literature. The former could be addressed by "snowballing" references (adding referenced papers to the corpus), but that requires structured data access to parse those references. Thus, we see that the state of structured data in academic literature is truly the problem.

The first issue is one of open access. As noted in Swartz 2008, although large portions of the academic community are embracing open access by openly publishing their works, historical documents remain controlled by a handful of publishers. This is problematic for a study such as the present as we need full (or at least representative) access to those past articles to establish trends over time. Institutions may not subscribe to every publisher's platform or may only subscribe to certain date ranges or journals. Without complete access, coverage will be negatively impacted.

The other issue is that even when one does have access, publishers of academic literature provide different and often inconsistent access to their own collections. In the course of this research, APIs for Elsevier, JSTOR, Web of Science, ProQuest, and CrossRef were explored but none satisfied the research needs entirely. Some provided the needed fields but lacked coverage while others suffered the converse. Platforms which cross publisher boundaries, such as Elsevier Scopus, include only metadata, since the full text is seen as the publishers' core asset.

The standard unit of literature continues to be a styled PDF, a document which is not conducive to machine analysis. Publishers maintain metadata databases in parallel to the full text documents, but often this is not the case for historic documents which is an issue for studies like this one. Since those documents are controlled by those publishers, third parties cannot freely create rich metadata databases.

Some effort has been expended to address these issues and provides a possible avenue for future work as discussed in Section 5.3. Carl Malamud's Public Resource created the General Index (Malamud 2021) as a response to the difficulty he perceived in text mining academic literature. That it took a non-publisher exercising potentially extra-legal methods to produce the necessary database to conduct modern analysis of scientific literature demonstrates the limitations of the status quo.

Chapter 5

Conclusion

5.1 Reliability outlook

We started this research with a decidedly negative view of the reliability engineering field. Outside of academia, much of the reliability engineering profession feels stagnant, clinging to the same methods developed in the 1950s. This research indicates a certain level of self-awareness within the published literature to the limitations of these methods and demonstrates a consistent move toward more effective tools.

Heavily regulated and/or prescribed sectors like defense and aerospace will likely lag the rest of the industry in adoption of new reliability tools, but commercial industries where the primary concern is meeting customers' reliability expectations can move much faster. Indeed, most of the examples in *Design for Reliability* (Raheja and Gullo 2012) are from the automotive and electronics industries.

5.2 Lessons learned

The core lesson from our results is that reliability engineering is increasingly seen as an outcome measured by customer experience rather than a specific set of tools. Reliability managers should be accountable for customer experience rather than specific deliverable like system models and predictions.

An example of this change in mentality is demonstrated by the author's experience as a reliability manager at a robotics company. While business goals were stated in terms of specific metrics, the reliability strategy that was implemented focused less on those metrics and more on improving reliability throughout the development process. This manifested as a heavy focus on system-level testing, a decidedly late-stage activity. The next focus was on development of a prognostics program, including hiring of dedicated resources to build that functionality into products.

5.3 Future work

As discussed in Section 4.4, the primary limitation of this research is related to the lack of document coverage. Increasing the size of the corpus through utilization of different database(s) would be a straightforward extension to the present work.

This could be accomplished either with access to commercial databases like Elsevier Scopus or by leveraging third-party databases like the General Index (Malamud 2021). The latter is of particular interest since it provides n-grams for document full texts which may provide even more robust topic modeling compared to the present study which was restricted to abstracts. The trade off would be a loss of context since these n-grams would only enable bag-of-words analysis. It would therefore be appropriate to leverage traditional latent Dirichlet analysis to perform topic analysis.

Appendix A

Full topic listing

ID	Top 3 terms	Papers	Representative paper title
0	weibull, censoring, censored	824	E-Bayesian estimation of reliability characteristics of two-parameter bathtub-shaped lifetime distribution with application
1	engineering, assurance, disciplines	806	Reliability engineering
2	replacement, preventive, maintenance	777	A condition-based maintenance policy for stochastically deteriorating systems
3	repair, repairable, markov	439	Transient analysis of reliability with and without repair for K-out-of-N:G systems with two failure modes
4	political, risk, perceptions	434	Meaning and contextualisation in risk assessment
5	shifts, charts, ewma	416	Improved Fast Initial Response Features for Exponentially Weighted Moving Average and Cumulative Sum Control Charts
6	oxide, drain, transistors	415	Reliability issues of offset drain transistors after different modes of electrical stress
7	uml, checking, checker	403	Methods of checking general safety criteria in UML statechart specifications
8	spacecraft, satellites, orbit	397	NASA Product Assurance in the 1990s
9	sobol, polynomial, gsa	391	Computing derivative-based global sensitivity measures using polynomial chaos expansions
10	certification, assurance, iso	355	Arguing software compliance with ISO 26262

ID	Top 3 terms	Papers	Representative paper title
11	srgms, srgm, nhpp	325	Towards comprehensive software reliability evaluation in open source software
12	coolant, reactor, pwr	323	Probabilistic analysis of flow control as an alternative to level control for BWR ATWS
13	licensing, pras, nuclear	310	Risk-informed evaluations of nuclear power plant digital upgrades technical and regulatory issues
14	kriging, surrogate, ak	309	An adaptive parallel learning dependent Kriging model for small failure probability problems
15	gaas, interconnects, electron	304	Degradation of ion implanted GaAs MESFETs
16	arl, chart, limits	291	Steady-state ARL analysis of ARL-unbiased EWMA-RZ control chart monitoring the ratio of two normal variables
17	disruptions, disruptive, resilience	279	A new resilience-based component importance measure for multi-state networks
18	fatigue, crack, fracture	276	DETERMINATION OF CRACK SIZE DISTRIBUTION FROM INCOMPLETE DATA SETS FOR THE CALCULATION OF FAILURE PROBABILITIES.
19	cognitive, human, psychological	271	Cognitive modeling and dynamic probabilistic simulation of operating crew response to complex system accidents. Part 1: Overview of the IDAC Model
20	mps, minimal, node	256	A simple algorithm to search for all d-MPs with unreliable nodes
21	bug, reports, bugs	229	CoLUA: Automatically Predicting Configuration Bug Reports and Extracting Configuration Options
22	radionuclide, repository, waste	219	Uncertainty and sensitivity analysis for the nominal scenario class in the 2008 performance assessment for the proposed high-level radioactive waste repository at Yucca Mountain, Nevada
23	experts, judgments, expert	216	Taxonomy of issues related to the use of expert judgments in probabilistic safety studies
24	releases, prone, discriminant	216	Exploring defect data from development and customer usage on software modules over multiple releases
25	sis, instrumented, iec	211	Demand rate and risk reduction for safety instrumented systems

ID	Top 3 terms	Papers	Representative paper title
26	attacker, defender, game	204	Protection vs. redundancy in homogeneous parallel systems
27	accelerated, adt, alt	202	Optimal design of hybrid accelerated test based on the Inverse Gaussian process model
28	rpn, fmea, linguistic	197	An Evidential Failure Mode and Effects Analysis Using Linguistic Terms
29	friction, tribological, wear	196	Investigating the water lubrication characteristics of sisal fiber reinforced ultrahigh-molecular-weight polyethylene material
30	vm, virtualized, virtualization	195	Toward optimal virtual machine placement and rejuvenation scheduling in a virtualized data center
31	exclusive, license, chapter	193	AMP Task Effectiveness
32	rul, lstm, prediction	190	Remaining useful life prediction based on intentional noise injection and feature reconstruction
33	warranty, sold, manufacturer	190	Aggregate discounted warranty cost forecasting considering the failed-but-not-reported events
34	maritime, collision, ship	188	Analysis of the marine traffic safety in the Gulf of Finland
35	rap, rrap, redundancy	187	An efficient simulated annealing algorithm for the redundancy allocation problem with a choice of redundancy strategies
36	bounds, upper, approximation	186	Geotechnical system reliability of slopes
37	chapter, chapters, book	186	The success factors – discussion
38	market, competition, products	186	Reliability in a time-driven product development process
39	suite, suites, prioritization	180	An Empirical Study of JUnit Test-suite Reduction
40	students, graduate, course	178	A structured problem-solving course for graduate students: exposing students to six sigma as part of their university training
41	app, apps, android	177	Taming Exceptions in Android Applications
42	petri, nets, net	176	Failure analysis for an airbag inflator by Petri nets
43	birnbaum, importance, vesely	166	Importance analysis of a multi-state system based on multiple-valued logic methods

ID	Top 3 terms	Papers	Representative paper title
44	shock, shocks, competing	162	Reliability modeling for systems subject to dependent competing failure processes with set of random shocks affect specific components
45	concrete, reinforced, rc	161	Probabilistic capacity models and seismic fragility estimates for RC columns subject to corrosion
46	psfs, psf, shaping	158	Estimating the quantitative relation between PSFs and HEPs from full-scope simulator data
47	solder, joints, cycling	156	High cycle cyclic torsion fatigue of PBGA Pb-free solder joints
48	reserve, adequacy, outage	156	Probabilistic assessment of spinning reserve in isolated and interconnected generation systems
49	factorial, designs, fractional	154	The general balance metric for mixed-level fractional factorial designs
50	portfolio, investments, insurance	154	Carbon footprinting of information technology products based on ISO standards: Fujitsu case study
51	cut, minimal, mcss	154	Fault tree reduction and quantification - an overview of IRRAS algorithms
52	producer, lot, plan	152	Variables Sampling Plan for Resubmitted Lots in a Process with Linear Profiles
53	solar, photovoltaic, pv	150	A New Numerical Modelling Method for System Energy Efficiency Calculation
54	iot, internet, authentication	150	Detection of IoT Devices That Mine Cryptocurrency
55	spare, inventory, spares	149	Multiobjective spare part allocation by means of genetic algorithms and Monte Carlo simulation
56	pareto, nsga, dominated	148	Fuzzy multiobjective system reliability optimization by genetic algorithms and clustering analysis
57	occupational, injury, accidents	146	Multi-hazard multi-person quantitative occupational risk model and risk management
58	cpm, cpk, cp	145	Computer program for calculating the p-value in testing process capability index C_{pmk}
59	vss, vsi, vssi	144	VSSI median control chart with estimated parameters and measurement errors

ID	Top 3 terms	Papers	Representative paper title
60	wiener, rul, degradation	143	Degradation modeling and RUL prediction using Wiener process subject to multiple change points and unit heterogeneity
61	pipe, pipes, buried	142	Time-dependent finite element reliability assessment of cast-iron water pipes subjected to spatio-temporal correlated corrosion process
62	cox, proportional, covariates	140	Maintainability analysis considering time-dependent and time-independent covariates
63	sre, organizations, software	139	Guiding reengineering with the operational profile
64	topics, dealt, proceedings	136	Proceedings of the 2010 IEEE 21st International Symposium on Software Reliability Engineering (ISSRE 2010)
65	autocorrelated, autoregressive, autocorrelation	129	Performance of cusum charts from the viewpoint of change-point estimation in the presence of autocorrelation
66	vibration, gearbox, signals	129	Blind vibration component separation and nonlinear feature extraction applied to the nonstationary vibration signals for the gearbox multi-fault diagnosis
67	taguchi, controllable, noise	128	Mixed resolution designs as alternatives to Taguchi inner/outer array designs for robust design problems
68	travel, metro, transit	128	Time-varied accessibility and vulnerability analysis of integrated metro and high-speed rail systems
69	ugf, universal, mss	128	Reliability evaluation of non-reparable three-state systems using Markov model and its comparison with the UGF and the recursive methods
70	sreqom, lulea, india	126	A framework for improvement of production plant performance using production assurance programs
71	domino, escalation, fire	125	Application of dynamic Bayesian network to performance assessment of fire protection systems during domino effects
72	logs, log, anomaly	123	A survey on automated log analysis for reliability engineering
73	vlsi, circuits, cmos	123	Evolution of VLSI reliability engineering
74	culture, organizational, cultural	122	Safety culture: a survey of the state-of-the-art

ID	Top 3 terms	Papers	Representative paper title
75	calibration, validation, pooling	117	New validation metrics for models with multiple correlated responses
76	patients, patient, hospital	116	Identifying Oncological Patient Information Needs to Improve e-Health Communication: a Preliminary Text-Mining Analysis
77	processors, processor, reconfiguration	116	Development and Analysis of the Software Implemented Fault-Tolerance (SIFT) Computer
78	batteries, battery, lithium	114	Prognostics of Lithium-Ion batteries using knowledge-constrained machine learning and Kalman filtering
79	javascript, web, client	113	JavaScript Errors in the Wild: An Empirical Study
80	localization, bug, debugging	112	FTMES: A Failed-Test-Oriented Mutant Execution Strategy for Mutation-Based Fault Localization
81	profiles, profile, explanatory	112	Monitoring nonlinear profile data using support vector regression method
82	coherent, signature, lifetimes	111	Computation of survival signatures for multi-state consecutive-k systems
83	robot, robots, robotic	111	Modular approach to kinematic reliability analysis of industrial robots
84	algebra, boolean, calculus	110	Analysis of system reliability by logical differential calculus and decision diagrams
85	linguistic, fuzzy, membership	109	Safety analysis and synthesis using fuzzy sets and evidential reasoning
86	multivariate, charts, hotelling	107	An assessment of the kernel-distance-based multivariate control chart through an industrial application
87	aleatory, epistemic, uncertainties	107	Mixed aleatory-epistemic uncertainty quantification with stochastic expansions and optimization-based interval estimation
88	prognostics, prognostic, phm	106	A general model for life-cycle cost analysis of Condition-Based Maintenance enabled by PHM capabilities
89	conforming, nonconforming, ccc	104	CCC-<i>r</i>charts' performance with estimated parameter for high-quality process
90	rejuvenation, aging, restart	104	A New Software Rejuvenation Model for Android
91	standby, warm, cold	103	Reliability evaluation of power systems with multi-state warm standby and multi-state performance sharing mechanism

ID	Top 3 terms	Papers	Representative paper title
92	inspectors, developers, taxonomy	103	Using a Cognitive Psychology Perspective on Errors to Improve Requirements Quality: an Empirical Investigation
93	dft, gates, dfts	100	Is Cut Sequence Necessary in Dynamic Fault Trees?
94	sigma, six, lean	99	Lean Six Sigma Implementation in a Food Processing SME: A Case Study
95	turbines, wind, farm	97	On the theoretical distribution of the wind farm power when there is a correlation between wind speed and wind turbine availability
96	grids, grid, cascading	96	An integrated modeling framework for cascading failure study and robustness assessment of cyber-coupled power grids
97	dod, army, department	96	TQM applied to US DOD electronics acquisition
98	pms, pmss, phased	95	Efficient reliability analysis of dynamic k-out-of-n heterogeneous phased-mission systems
99	diagnosis, diagnose, symptom	95	Outline of COPILOT, and expert system for reactor operational assistance, using a Bayesian diagnostic module
100	synthesis, interactive, trees	95	PROPAGATION OF FAULTS IN PROCESS PLANTS: 3. AN INTERACTIVE, COMPUTER-BASED FACILITY.
101	maturity, business, innovation	94	Research on evaluation method of electronic product maturity
102	antifouling, adhesion, coatings	92	Construction of robust slippery lubricant-infused epoxy-nanocomposite coatings for marine antifouling application
103	proceedings, conference, papers	92	Proceedings of the 1995 ESREL Conference
104	skewness, kurtosis, moments	92	A flexible distribution and its application in reliability engineering
105	machining, cnc, cutting	91	Probability distribution of machining center failures
106	heterogeneity, frailty, unobserved	85	Unobserved heterogeneity in the power law nonhomogeneous Poisson process
107	lhs, latin, hypercube	84	Efficient Monte Carlo methods for estimating failure probabilities
108	pso, swarm, particle	83	Reliability Algorithm Based on Adaptive Hybrid Particle Swarm Optimization and Simulated Annealing Algorithm

ID	Top 3 terms	Papers	Representative paper title
109	journals, journal, articles	82	Output distributions and topic maps of safety related journals
110	copulas, copula, pcs	81	Multivariate Degradation Modeling of Smart Electricity Meter with Multiple Performance Characteristics via Vine Copulas
111	taylor, francis, llc	80	Some notes on probabilities and non-probabilistic reliability measures
112	ccf, ccfs, common	80	A pragmatic approach to modeling common cause failures in multi-unit PSA for nuclear power plant sites with a large number of units
113	infant, burn, mortality	80	Optimal burn-in time under cumulative free replacement warranty
114	split, randomization, plot	79	Experimentation with randomization restrictions: Targeting practical implementation
115	tensorflow, ml, fi	78	TensorFI: A Configurable Fault Injector for TensorFlow Applications
116	shapes, weibull, plot	78	Parametric study of multiplicative model involving two Weibull distributions
117	earthquake, earthquakes, seismic	78	Serviceability of earthquake-damaged water systems: effects of electrical power availability and power backup systems on system vulnerability
118	modal, excitation, fem	78	Experimental investigation into amplitude-dependent modal properties of an eleven-span motorway bridge
119	client, server, servers	78	Increasing the reliability of three-tier applications
120	epc, assignable, spc	77	Economic design of the integrated multivariate EPC and multivariate SPC charts
121	tolerant, hardware, masking	77	System reliability analysis of an N-version programming application
122	bdd, ordering, bdds	76	Efficient basic event ordering schemes for fault tree analysis
123	shafer, dempster, ignorance	76	Application of the Dempster-Shafer theory of evidence for accident probability estimates
124	induction, motors, motor	75	Effectiveness of vibration and current monitoring in detecting broken rotor bar and bearing faults in an induction motor
125	kalman, filter, filtering	75	Remaining useful life estimation in aeronautics: Combining data-driven and Kalman filtering
126	pipeline, pipelines, gas	74	A systematic framework of vulnerability analysis of a natural gas pipeline network

ID	Top 3 terms	Papers	Representative paper title
127	natech, tanks, floods	73	Release of hazardous substances in flood events: damage model for atmospheric storage tanks
128	adversarial, adaptation, domain	73	Intelligent fault diagnosis of rotating machinery using a multi-source domain adaptation network with adversarial discrepancy matching
129	abstract, satellites, satellite	73	What are emergent properties and how do they affect the engineering of complex systems?
130	covariance, outliers, hotelling	73	Robust T²control chart using median-based estimators
131	discovery, oss, vulnerabilities	73	Modeling the vulnerability discovery process
132	shaft, hull, propulsion	73	Numerical and experimental analysis of coupled transverse and longitudinal vibration of a marine propulsion shaft
133	surveillance, ageing, ts	72	Evaluation of risk impact of changes to surveillance requirements addressing model and parameter uncertainties
134	count, poisson, counts	72	A Control Chart for COM–Poisson Distribution Using Resampling and Exponentially Weighted Moving Average
135	edges, edge, nodes	72	Analysis of network cascading failure based on the cluster aggregation in cyber-physical systems
136	rejuvenation, aging, completion	72	Availability optimization in operational software system with aperiodic time-based software rejuvenation scheme
137	hazop, operability, digraph	72	Digraph-based models for automated HAZOP analysis
138	injected, linux, os	71	Do Injected Faults Cause Real Failures? A Case Study of Linux
139	inheritance, object, oo	71	Inter-class mutation operators for Java
140	tunnel, dbn, metro	71	A dynamic Bayesian network based approach to safety decision support in tunnel construction
141	traceability, artifacts, assurance	70	Towards Automated Evidence Generation for Rapid and Continuous Software Certification
142	pilots, landing, mental	70	A systems perspective on the unstable approach in commercial aviation

ID	Top 3 terms	Papers	Representative paper title
143	psa, pc, super	70	PC-based probabilistic safety assessment in Japan [of nuclear power stations]
144	braking, brake, steering	70	Fault-tolerant automobile steering based on diversity of steer-by-wire, braking and acceleration
145	arima, svr, forecasting	69	Forecasting systems reliability based on support vector regression with genetic algorithms
146	seismic, earthquake, fragility	68	Seismic risk evaluation for high voltage air insulated substations
147	qra, qras, precursor	67	Quantitative risk analysis offshore - human and organizational factors
148	pomdp, voi, drl	67	Value of information analysis in non-stationary stochastic decision environments: A reliability-assisted POMDP approach
149	flood, flooding, surge	67	Impact of including interdependencies between multiple riverine flood defences on the economically optimal flood safety levels
150	pcis, pci, cpm	67	Yield-based capability index for evaluating the performance of multivariate manufacturing process
151	arrhenius, accelerated, temperature	66	Limitations and extended applications of Arrhenius equation in reliability engineering
152	rcm, centered, maintenance	66	Reliability centered maintenance
153	imbalance, classifiers, label	65	Cross-Project Aging-Related Bug Prediction Based on Joint Distribution Adaptation and Improved Subclass Discriminant Analysis
154	abort, rescue, mission	65	Optimal mission abort policy for systems in a random environment with variable shock rate
155	functionally, versions, version	65	Reliability and performance analysis for fault-tolerant programs consisting of versions with different characteristics
156	wsn, wireless, routing	64	Fuzzy based optimized routing protocol for wireless sensor networks
157	bns, bn, inference	64	Non-parametric Bayesian networks: Improving theory and reviewing applications
158	manet, hoc, ad	63	Performance reliability evaluation for mobile ad hoc networks
159	ahp, hierarchy, analytic	63	ABC Inventory Classification Using AHP and Ranking Methods via DEA

ID	Top 3 terms	Papers	Representative paper title
160	desirability, responses, designs	63	Balancing the Subjective and Objective Weights for Correlated Multiresponse Optimization
161	berlin, heidelberg, verlag	62	Fundamental of Reliability
162	diode, manufactured, semiconductor	61	Diode Step Stress Program for JANTZ1N5417
163	sustainability, sustainable, businesses	61	Toward an Improved Strategy for Functional Product Development by Predicting Environmental and Economic Sustainability
164	creep, specimens, strain	60	Axial creep-rupture time of boron-aluminum composites
165	electromagnetic, shielding, se	60	Design and optimization of multilayered electromagnetic shield using a real-coded genetic algorithm
166	ann, pso, neural	60	Predicting cumulative number of failures in software using an ANN-PSO based approach
167	blades, blade, turbine	60	Fuzzy finite element model updating of a laboratory wind turbine blade for structural modification detection
168	rsm, experimentation, experiments	60	ASRSM: a sequential experimental design for response surface optimization
169	mttr, mtbf, mttf	59	Random effects model for the reliability management of modules of a fighter aircraft
170	fts, ft, trees	59	Fault tree conditioning methods to trace system configuration changes for the application to low-power/shutdown PSA
171	profile, usage, profiles	59	An extended operational profile model
172	plastic, encapsulated, humidity	58	Recent humidity accelerations, a base for testing standards
173	clpr, liner, cylinder	58	Effects of thread groove width in cylinder liner surface on performances of diesel engine
174	track, railway, geometry	58	A stochastic model for railway track asset management
175	evacuation, route, passengers	58	Multi-objective evacuation routing optimization for toxic cloud releases
176	bbn, bbns, belief	58	Bayesian belief networks for human reliability analysis: A review of applications and gaps
177	testability, radar, software	57	Radar System Testability Design and Demonstration based on Fault Modes and Software Control

ID	Top 3 terms	Papers	Representative paper title
178	lubrication, lubricated, bearing	57	Film-Thickness Identification Method and Lubrication Characteristic Experiment of Full-Size Water-Lubricated Stern Bearing under Offset Load
179	capacitors, resistors, capacitor	56	Electrical characterization and reliability evaluation of capacitors by means of in situ leakage current measurements
180	gauge, repeatability, reproducibility	56	Confidence intervals for unbalanced two-factor gauge R&R studies
181	consecutive, window, linearly	56	Consecutive sliding window systems
182	tacom, eops, appropriateness	56	A study on the validity of a task complexity measure for emergency operating procedures of nuclear power plants-Comparing task complexity scores with two sets of operator response time data obtained under a simulated SGTR
183	bridges, bridge, highway	56	A bridge network maintenance framework for Pareto optimization of stakeholders/users costs
184	selective, missions, breaks	56	Multi-mission selective maintenance and repairpersons assignment problem with stochastic durations
185	plots, proportions, plot	55	Fraction of design space plots for evaluating ridge estimators in mixture experiments
186	uav, uavs, aerial	55	Network approach for resilience evaluation of a UAV swarm by considering communication limits
187	trans, publications, tech	55	New methods of reliability engineering
188	signed, sign, gwma	55	An efficient nonparametric EWMA Wilcoxon signed-rank chart for monitoring location
189	synchronous, execution, specification	55	Using CLP to automatically generate test sequences for synchronous programs with numeric inputs and outputs
190	hurricane, hurricanes, outages	55	Estimating the spatial distribution of power outages during hurricanes in the Gulf coast region
191	actuators, actuator, controller	54	Optimal reliability design for over-actuated systems based on the MIT rule: Application to an octocopter helicopter testbed

ID	Top 3 terms	Papers	Representative paper title
192	microservice, microservices, services	54	Automatic performance monitoring and regression testing during the transition from monolith to microservices
193	usability, website, user	53	Website user experience (UX) testing tool development using open source software (OSS)
194	equivalence, lifetimes, mle	53	Reliability equivalence factors of a general series-parallel system
195	white, box, black	53	Black-Box and White-Box Test Case Generation for RESTful APIs: Enemies or Allies?
196	agile, development, practices	52	Entropy metrics for agile development processes
197	propagated, trigger, combinatorial	52	Reliability analysis of multi-trigger binary systems subject to competing failures
198	throughput, buffer, performability	52	Automatic Petri net simulation model generation for a continuous flow transfer line with unreliable machines
199	sizing, horizon, preventive	51	Integrating noncyclical preventive maintenance scheduling and production planning for multi-state systems
200	gui, ui, automated	51	Generating test cases for GUI responsibilities using complete interaction sequences
201	mutants, mutation, suites	51	Prioritizing mutation operators based on importance sampling
202	contracts, contract, smart	51	Safety contract based design of software components
203	intrusion, norm, chi	50	An anomaly detection technique based on a chi-square statistic for detecting intrusions into information systems
204	mewma, mcusum, multivariate	50	Memory-type multivariate charts with fixed and variable sampling intervals for process mean when covariance matrix is unknown
205	increments, gamma, wiener	50	Improved inverse Gaussian process and bootstrap: degradation and reliability metrics
206	precursor, precursors, accident	50	German precursor study - methodology and insights
207	mine, coal, underground	50	Gas-related, fire, and blasting accidents in mines and methods for determining mine atmosphere status
208	supplier, suppliers, designated	49	The difference test statistic for two suppliers with linear profiles
209	voting, consensus, units	49	Automating the analysis of voting systems

ID	Top 3 terms	Papers	Representative paper title
210	warranty, replacement, policy	49	Warranty cost analysis for second-hand products under a two-stage repair-or-full refund policy
211	rail, originality, railway	49	Monitoring Safety in Rail Transport by Means of Digital Economy Solutions
212	restart, rare, scales	48	Asymptotic optimality of RESTART estimators in highly dependable systems
213	unlabeled, labeled, supervised	47	Outliers detection using an iterative strategy for semi-supervised learning
214	fatality, fatalities, cancer	47	Uncertainty and sensitivity analysis of early exposure results with the MACCS reactor accident consequence model
215	rbi, lcc, lock	47	Risk-based inspection as a reliability-engineering tool for fixed equipment decisions
216	rss, orss, srs	47	New Synthetic Control Charts for Monitoring Process Mean and Process Dispersion
217	networked, controller, controllers	46	Distributed controller design for systems interconnected over chordal graphs
218	glr, bernoulli, sustained	46	The design of geometric generalized likelihood ratio control chart
219	msp, mission, abort	46	Optimal loading of repairable system with perfect product storage
220	leak, leaks, memory	45	MemDefender: An Allocation Monitoring and Memory Leak Injection Tool for Java
221	arrays, orthogonal, designs	45	Graphical methods for evaluating covering arrays
222	aib, maxewma, auxiliary	45	Memory-type control charts with multiple auxiliary information for process mean
223	arithmetic, bounds, fuzzy	44	Computer arithmetic for probability distribution variables
224	titanium, alloys, alloy	44	Effect of surface modification on surface properties and tribological behaviours of titanium alloys
225	svm, vector, machine	44	Support vector machine based estimation of remaining useful life: current research status and future trends
226	detectors, detector, anomaly	44	How Far Have We Come in Detecting Anomalies in Distributed Systems? An Empirical Study with a Statement-level Fault Injection Method
227	partition, communities, topology	43	Robustness in network community detection under links weights uncertainties

ID	Top 3 terms	Papers	Representative paper title
228	port, ports, container	43	Green port oriented resilience improvement for traffic-power coupled networks
229	converter, supplies, electronic	43	Reliability modeling and analysis for a novel design of modular converter system of wind turbines
230	wafer, maps, map	43	Defect pattern recognition on wafers using convolutional neural networks
231	inflated, zip, zero	42	CUSUM Control Charts for the Monitoring of Zero-inflated Binomial Processes
232	wheel, wheels, train	42	Reliability analysis for degradation of locomotive wheels using parametric bayesian approach
233	digraph, tracing, directed	42	Development of maintainability index for mechanical systems
234	ordinal, categorical, attribute	41	Latent change-point detection in ordinal categorical data
235	cots, shelf, commercial	41	The design of complete systems: Providing human factors guidance for COTS acquisition
236	sql, queries, query	40	Automated fix generator for SQL injection attacks
237	cis, scada, ci	40	Adopting HLA standard for interdependency study
238	qmu, margins, epistemic	40	Quantification of margins and uncertainties: alternative representations of epistemic uncertainty
239	universal, genetic, generating	40	Structure optimization of multi-state system with two failure modes
240	qos, services, cloud	39	An Adaptive PID Control for QoS Management in Cloud Computing System
241	image, images, vision	39	A multi-image monitoring framework for statistical process control to improve manufacturing systems
242	cream, hep, hra	39	A Bayesian Network to Ease Knowledge Acquisition of Causal Dependence in CREAM: Application of Recursive Noisy-OR Gates
243	ultrasonic, cast, iron	38	Ultrasonic Control and Inspection of Cast Steel Parts for Gear Machining
244	halt, accelerated, alt	38	Ten things you should know about HALT & HASS
245	mins, interconnection, sen	38	Multi-source multi-terminal reliability evaluation of interconnection networks

ID	Top 3 terms	Papers	Representative paper title
246	airport, screening, passengers	37	Airport safety management system for the future
247	fuzzing, mutation, mutants	37	Towards Effective Performance Fuzzing
248	shot, lifetimes, accelerated	37	Constant-Stress Accelerated Life-Test Models and Data Analysis for One-Shot Devices
249	piping, pipe, erosion	37	Markov models for evaluating risk-informed in-service inspection strategies for nuclear power plant piping systems
250	ant, colony, multiobjective	37	Coupling ant colony and the degraded ceiling algorithm for the redundancy allocation problem of series-parallel systems
251	assertions, executable, fuzzing	36	Putting assertions in their place
252	sysml, diagrams, block	36	Automated generation of failure modes and effects analysis from SysML models
253	blast, explosive, fatality	36	Security risks and probabilistic risk assessment of glazing subject to explosive blast loading
254	metamorphic, oracle, relations	36	Using machine learning techniques to detect metamorphic relations for programs without test oracles
255	balanced, sojourn, balance	35	Reliability analysis for balanced engine systems with m sectors by considering start-up probability
256	malware, av, android	35	Frequent Subgraph Based Familial Classification of Android Malware
257	tbe, shifts, charts	35	Monitoring of time between events with a double generally weighted moving average control chart
258	qfd, deployment, customer	35	Robust QFD: framework and a case study
259	fram, resonance, systemic	35	Comparing a multi-linear (STEP) and systemic (FRAM) method for accident analysis
260	rbdo, kriging, loop	34	An integrated reliability-based design optimization of offshore towers
261	spreadsheets, localization, faulty	34	Mutation-based spreadsheet debugging
262	site, unit, inter	34	Evaluation of inter-unit dependency effect on site core damage frequency: Internal and seismic event
263	cuscore, disturbance, autocorrelated	33	Robustness properties of Cuscore statistics for monitoring a nonstationary system

ID	Top 3 terms	Papers	Representative paper title
264	scan, anomalous, surveillance	33	A comparison of CUSUM, EWMA, and temporal scan statistics for detection of increases in Poisson rates
265	categorical, contingency, table	33	Phase-I monitoring of log-linear model-based processes (a case study in health care: Kidney patients)
266	sdn, plane, networking	33	Programming the Network: Application Software Faults in Software-Defined Networks
267	redundancies, redundancy, allocation	32	On allocation of redundancies in two-component series systems
268	ice, arctic, sea	32	An operational risk analysis tool to analyze marine transportation in Arctic waters
269	fpga, programmable, processors	32	A practical application of NUREG/CR-6430 software safety hazard analysis to FPGA software
270	weld, welding, welded	32	Study on the mechanical properties and defect detection of low alloy steel weldments for large cruise ships
271	ontology, pss, architecture	32	Ontology-Based Reuse of Failure Modes in Existing Databases for FMEA: Methodology and Tool
272	forest, ensemble, classifiers	31	Forest fire induced Natech risk assessment: A survey of geospatial technologies
273	survivability, destroyed, protected	30	Survivability of series-parallel systems with multilevel protection
274	gp, surrogate, gaussian	30	Life-cycle reliability-based robust design optimization for GP model with response uncertainty
275	images, encryption, image	30	A map-based image steganography scheme for RGB images
276	hmm, hidden, healthy	30	Hidden Markov model with auto-correlated observations for remaining useful life prediction and optimal maintenance policy
277	rpd, controllable, settings	29	Simultaneous optimization of robust parameter and tolerance design based on generalized linear models
278	filling, designs, stratified	29	Fast flexible space-filling designs with nominal factors for nonrectangular regions
279	cv, mcv, charts	29	Side-sensitive modified group runs charts with and without measurement errors for monitoring the coefficient of variation

ID	Top 3 terms	Papers	Representative paper title
280	sl, wl, assured	29	Probability of loss of assured safety in systems with multiple time-dependent failure modes: Representations with aleatory and epistemic uncertainty
281	imbalance, rotating, rotor	29	Vibration response based reliability modeling for rotor systems with imbalance
282	atm, air, emergent	28	A methodology used for the development of an Air Traffic Management functional system architecture
283	ess, screening, stresses	28	ESS profiles with step stress level
284	benchmarking, inferential, companies	26	Benchmarking barometers for products and processes
285	gear, gears, torque	26	Reliability analysis of gear transmission with considering failure correlation
286	dl, adversarial, training	25	DeepMutation: mutation testing of deep learning systems
287	sos, constituent, behavioral	25	Modeling Governance and Management in Socio-Technical SoS
288	mbt, executable, api	25	An initial evaluation of model-based testing
289	tier, classifications, center	24	Reliability of Example Mechanical Systems for Data Center Cooling Selected by Tier Classification
290	checkpoint, warm, encoding	24	Experience report: An application-specific checkpointing technique for minimizing checkpoint corruption
291	grey, relational, fuzzy	24	The Prediction of Cellphones’ Fault Rates with Grey Models
292	go, repairable, logic	24	A new reliability analysis method for repairable systems with multifunction modes based on goal-oriented methodology
293	dsfs, detectability, blade	23	Sequential projection pursuit for optimised vibration-based damage detection in an experimental wind turbine blade
294	sr, recommended, signed	21	IEEE Recommended Practice on Software Reliability
295	cpn, timed, petri	21	Backward reachability of Colored Petri Nets for systems diagnosis
296	ds, macr, synthetic	21	The double sampling range chart

Table A.1: This table contains the native 297 topics identified by `top2vec` in the population corpus. For each topic, the top three terms are included as well as the count of documents associated with that topic and the title of the most representative paper associated with that topic.

Appendix B

Domain subtopic visualizations

B.1 Software domain

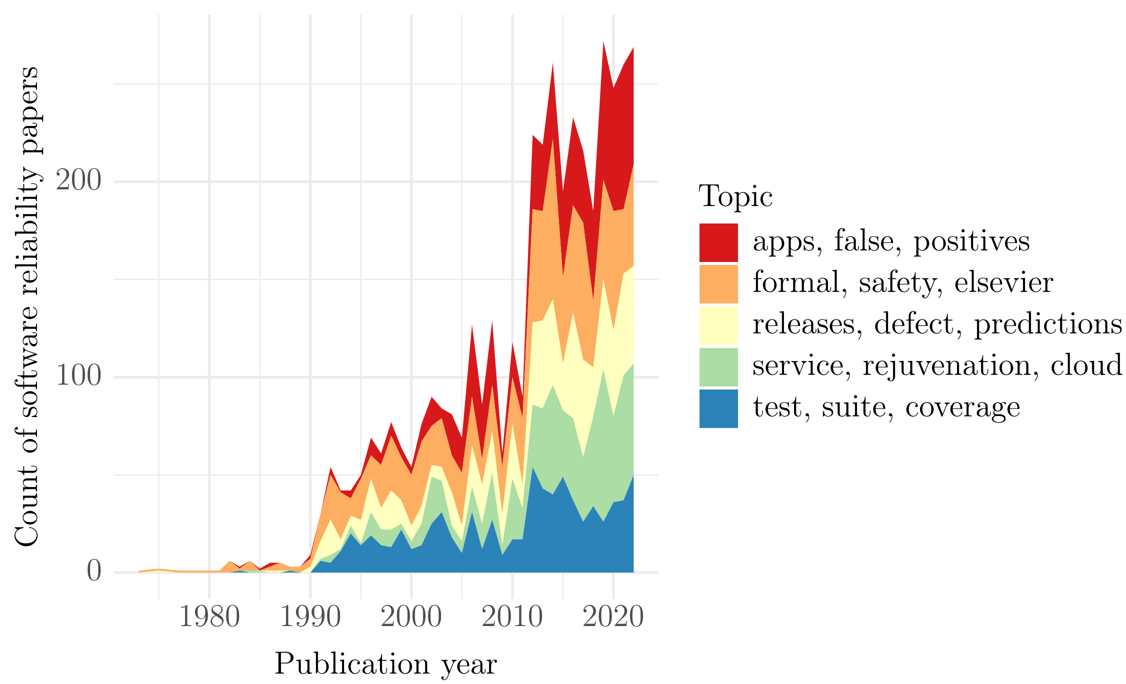

Figure B-1: Subtopic count in software papers over time. Topics are stacked according to count of publications in 2022. Labels are based on the top three terms in the subtopic.

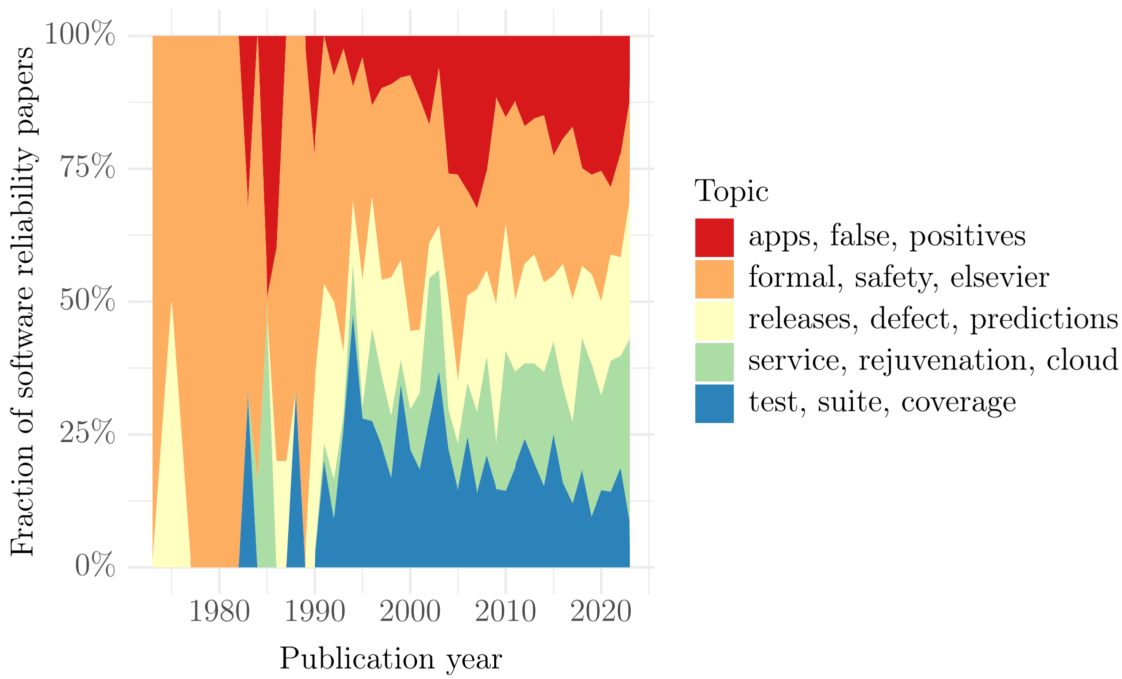

Figure B-2: Subtopic proportion in software papers over time. Topics are stacked according to count of publications in 2022. Labels are based on the top three terms in the subtopic.

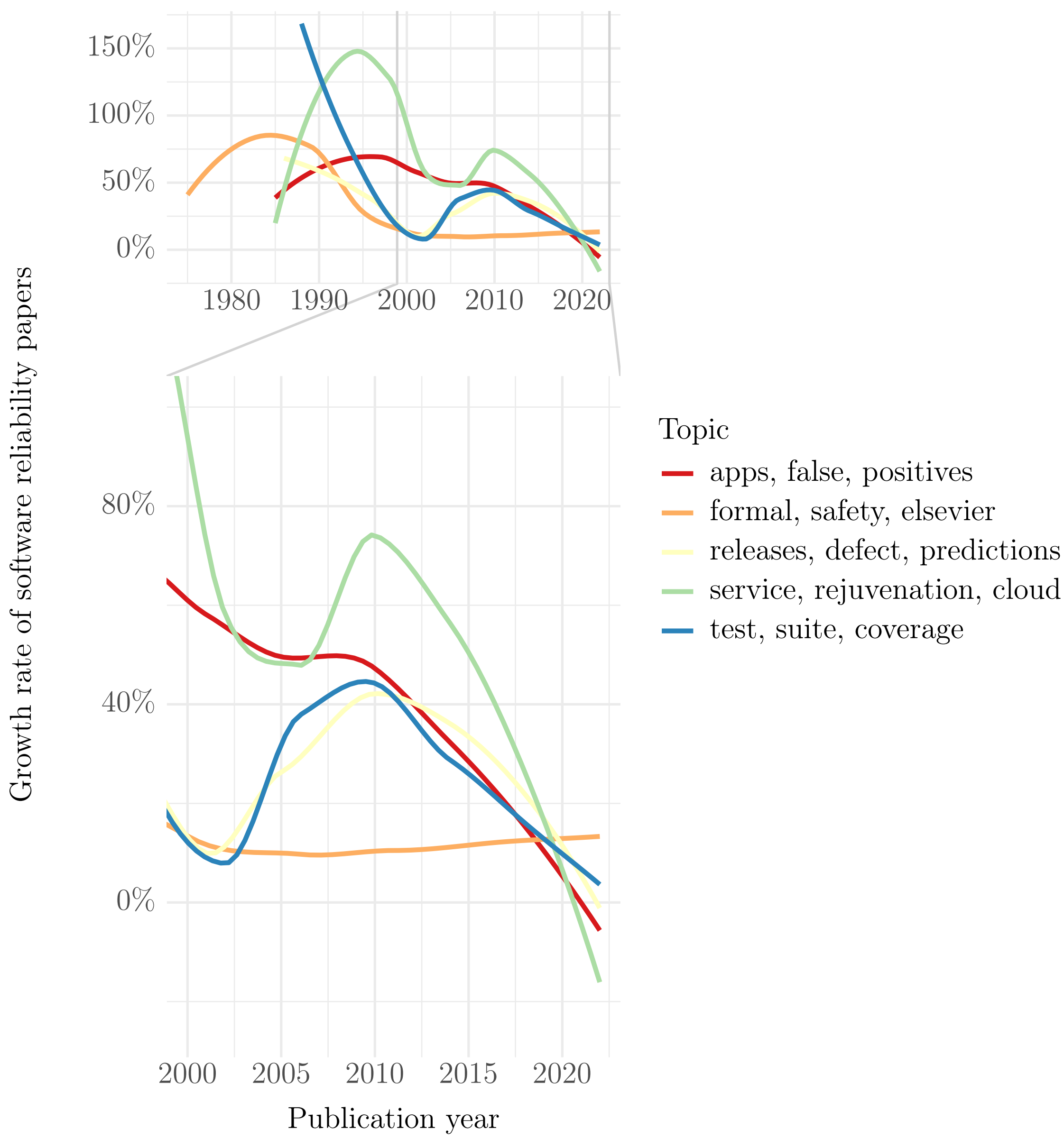

Figure B-3: Subtopic growth in software papers over time. Labels are based on the top three terms in the subtopic.

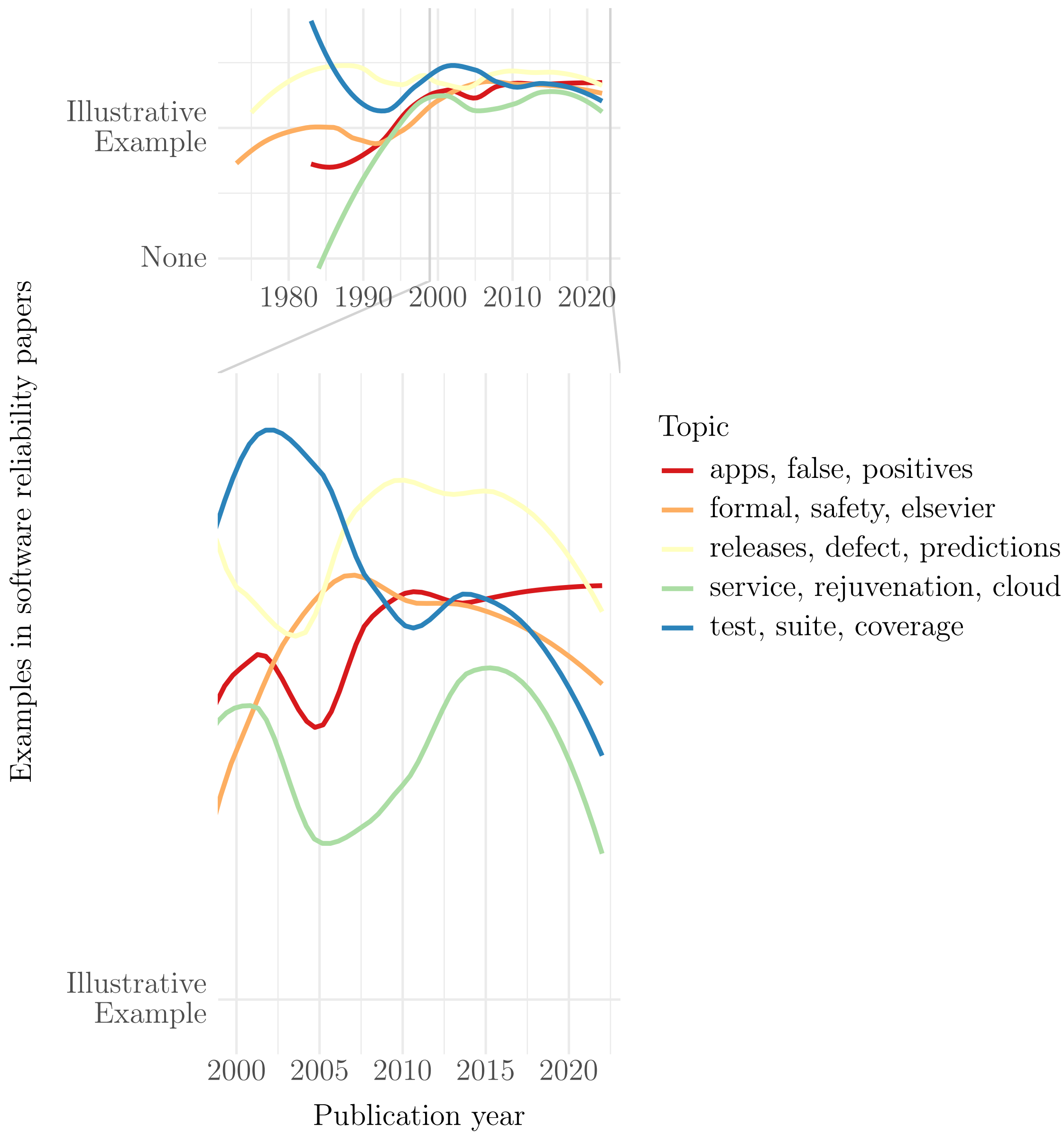

Figure B-4: Subtopic practicality in software papers over time. Labels are based on the top three terms in the subtopic.

B.2 Infrastructure domain

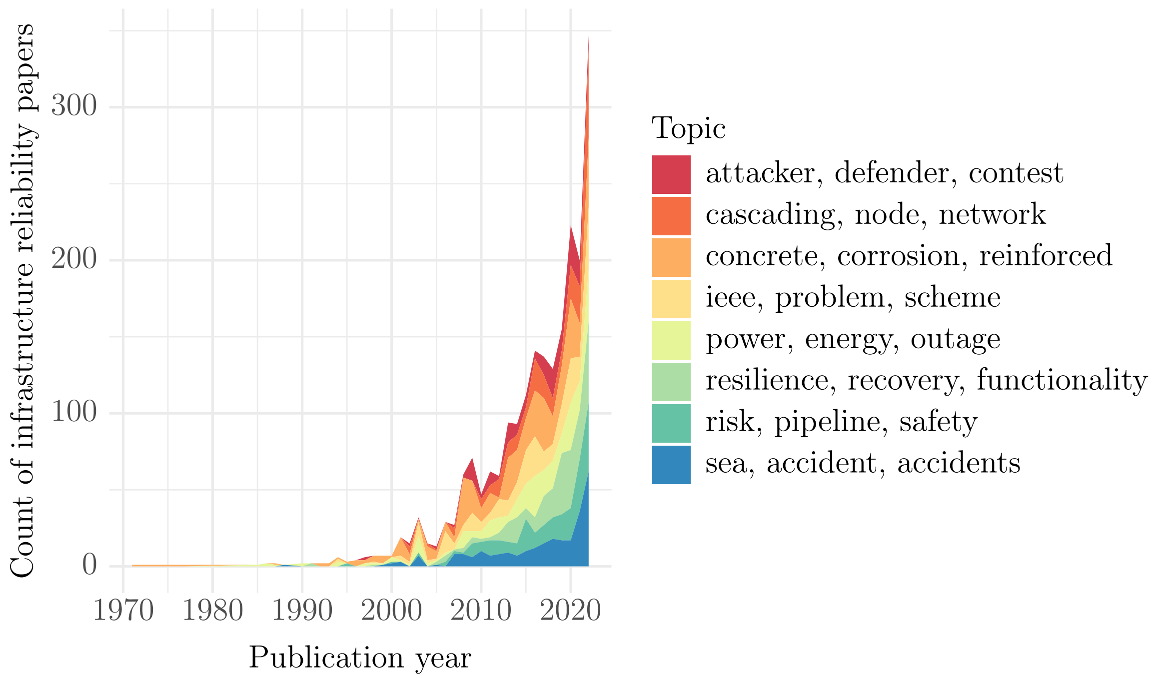

Figure B-5: Subtopic count in infrastructure papers over time. Topics are stacked according to count of publications in 2022. Labels are based on the top three terms in the subtopic.

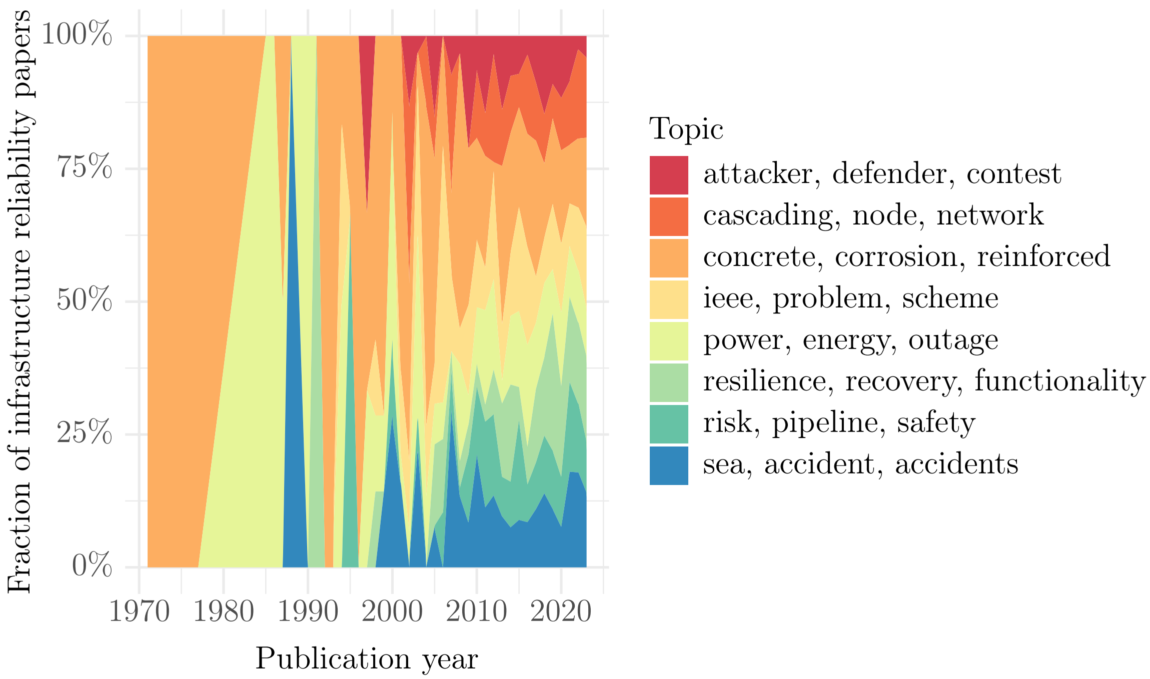

Figure B-6: Subtopic proportion in infrastructure papers over time. Topics are stacked according to count of publications in 2022. Labels are based on the top three terms in the subtopic.

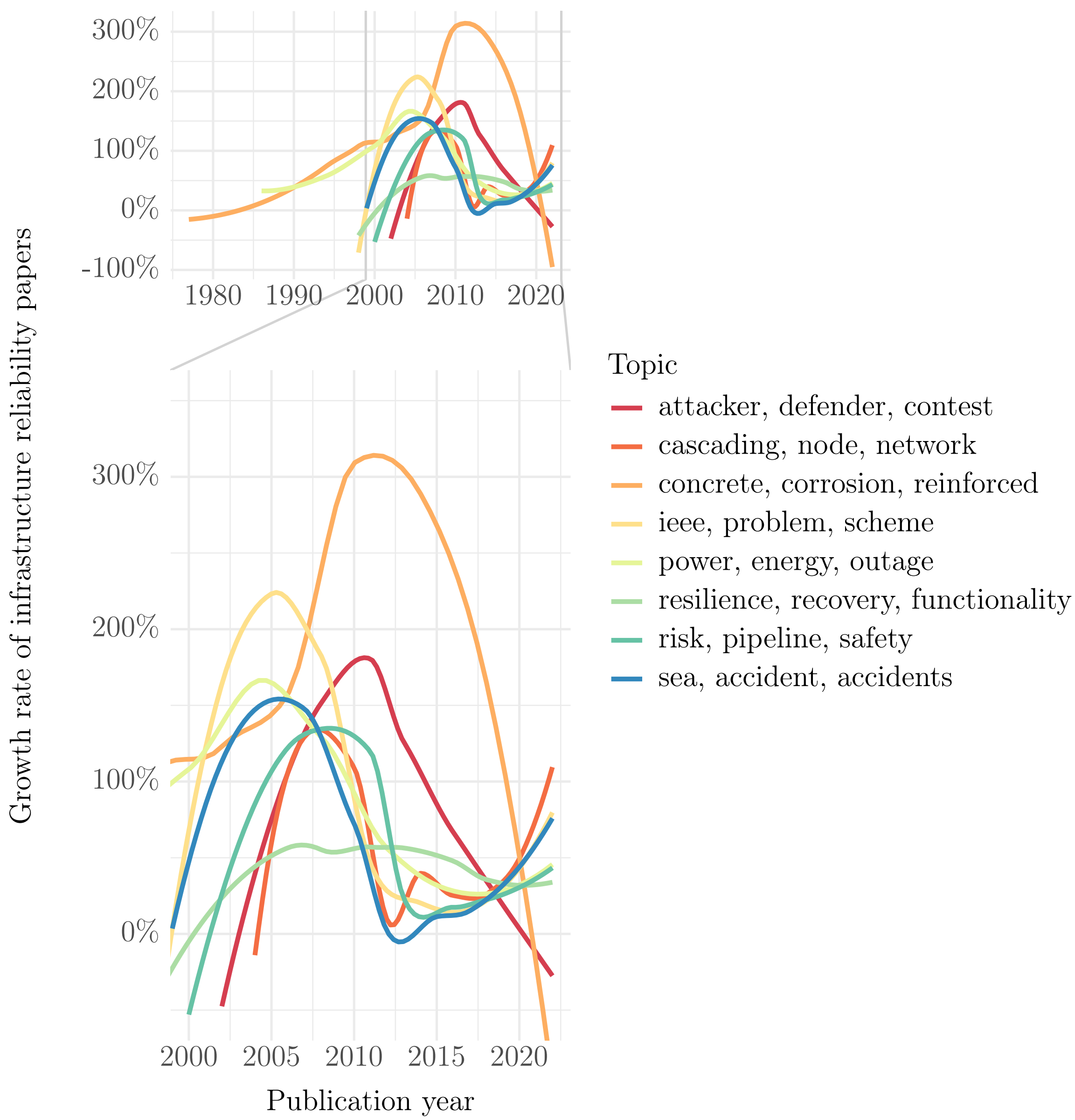

Figure B-7: Subtopic growth in infrastructure papers over time. Labels are based on the top three terms in the subtopic.

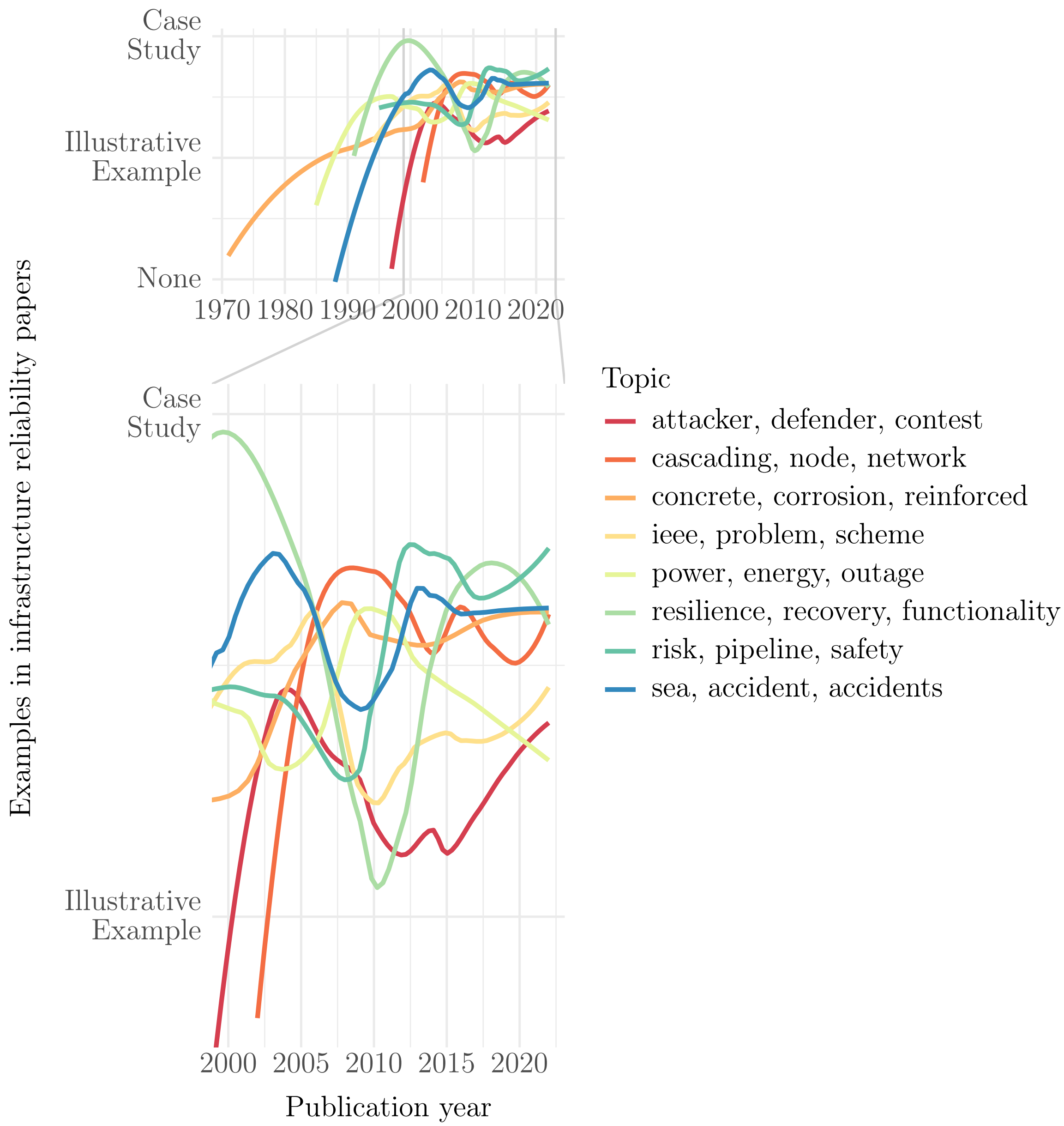

Figure B-8: Subtopic practicality in infrastructure papers over time. Labels are based on the top three terms in the subtopic.

B.3 Nuclear domain

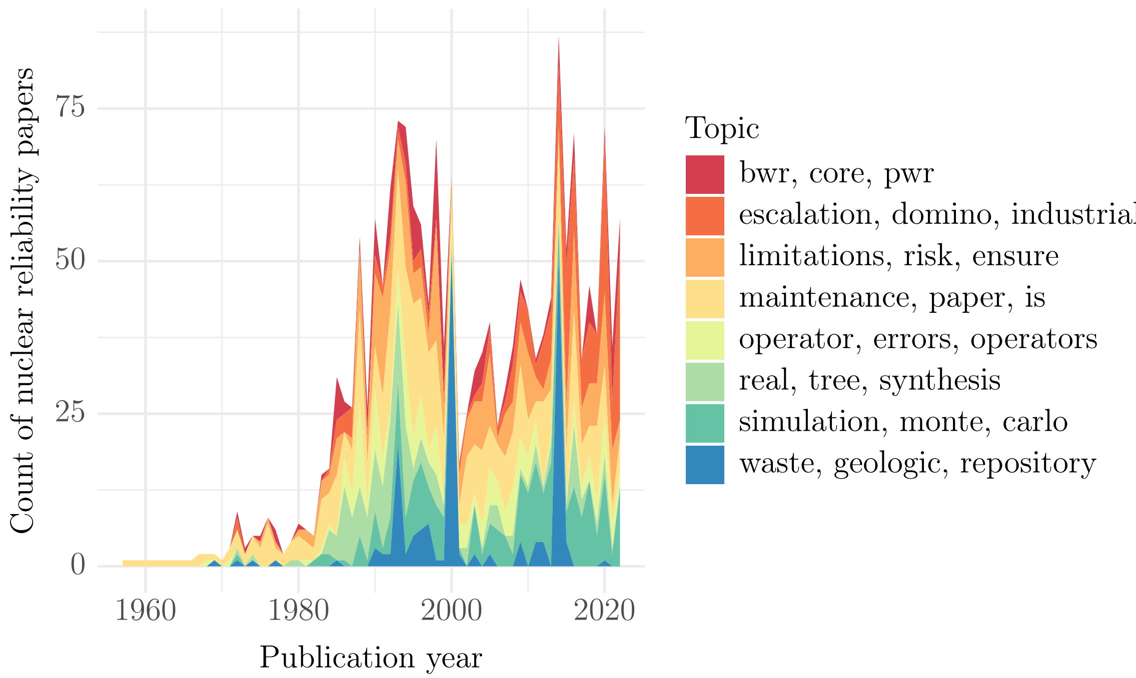

Figure B-9: Subtopic count in nuclear papers over time. Topics are stacked according to count of publications in 2022. Labels are based on the top three terms in the subtopic.

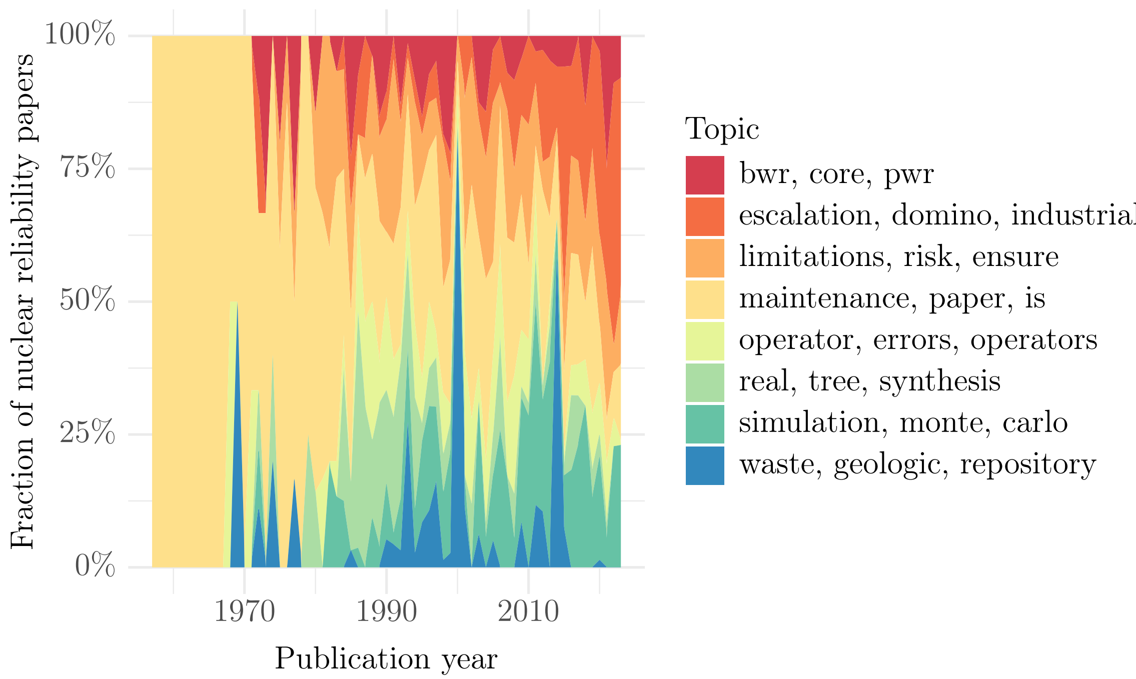

Figure B-10: Subtopic proportion in nuclear papers over time. Topics are stacked according to count of publications in 2022. Labels are based on the top three terms in the subtopic.

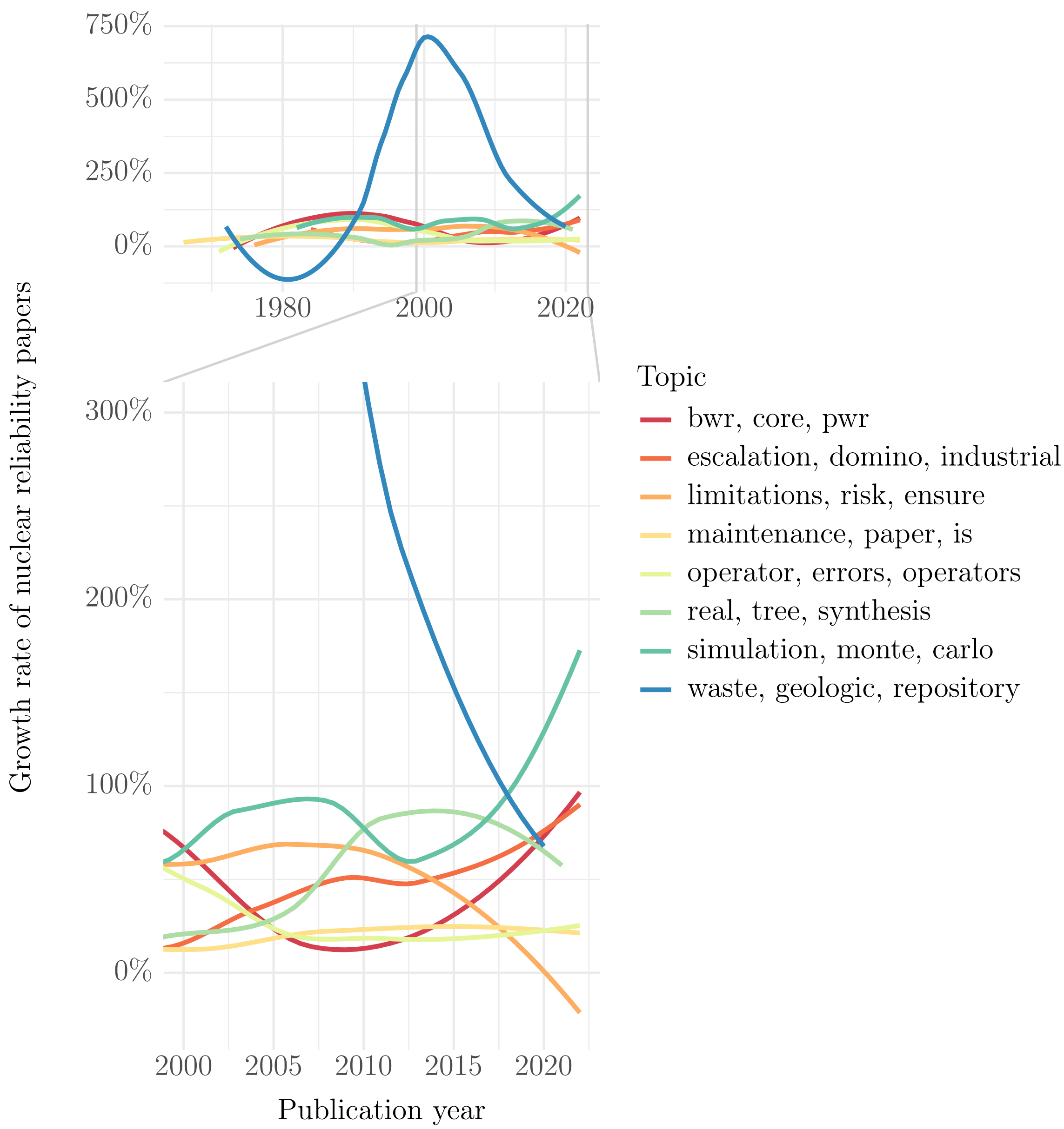

Figure B-11: Subtopic growth in nuclear papers over time. Labels are based on the top three terms in the subtopic.

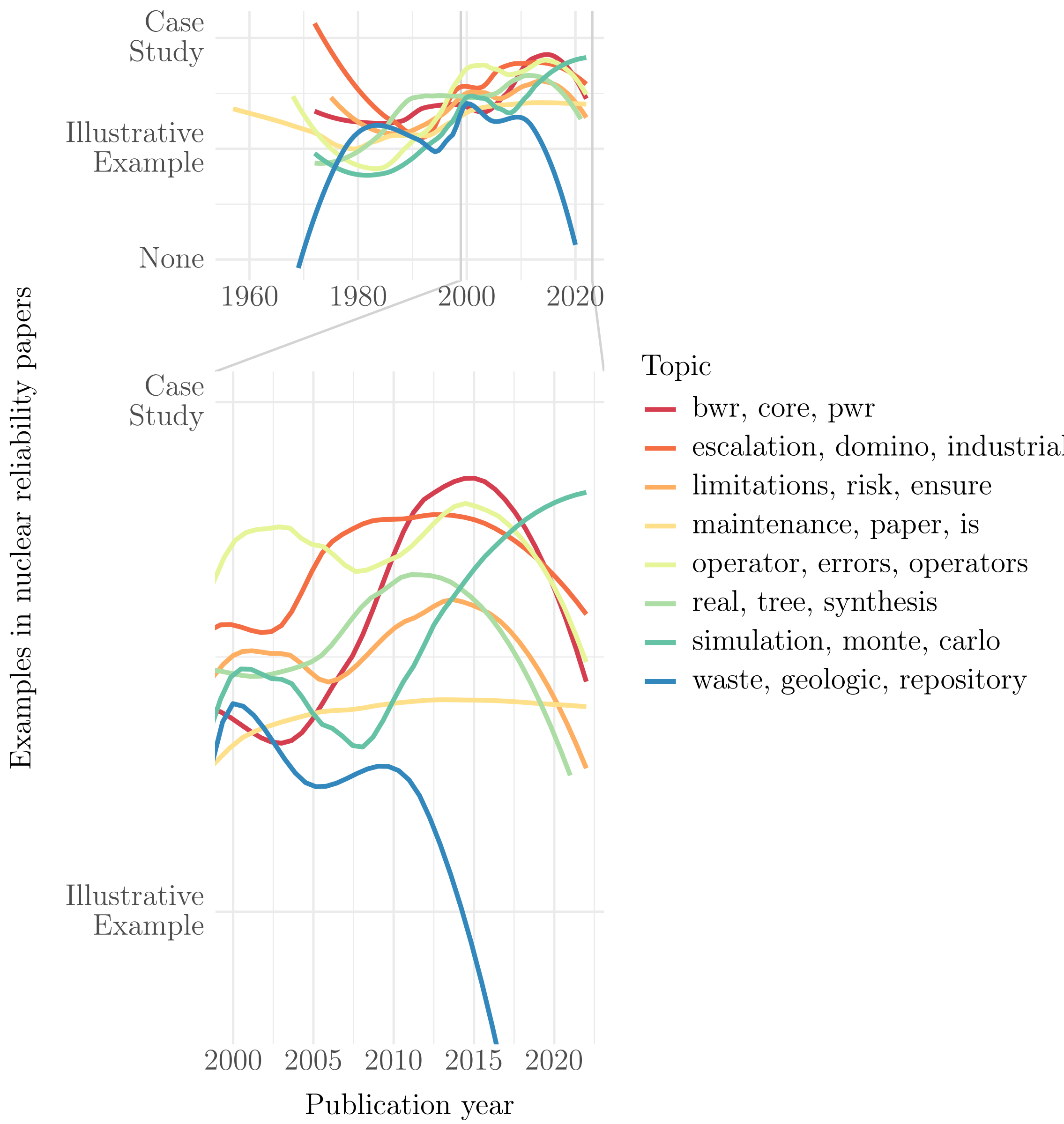

Figure B-12: Subtopic practicality in nuclear papers over time. Labels are based on the top three terms in the subtopic.

Appendix C

Analysis scripts

These scripts rely on several libraries for functionality. Python libraries are listed together here for convenience. Package versions were current as of May 7, 2023 with Python 3.10. Visualizations were generated with R 4.2.3 and those libraries are described in Section C.6. Up-to-date and executable source code is available at https://github.com/ckb2/rel-text-mine.

```
import requests
import pandas as pd
from urllib.parse import urlencode
import math
import concurrent.futures
import itertools
import time
import os
import progressbar
import numpy as np
from top2vec import Top2Vec
import umap.umap_ as umap
import hvplot.pandas
import openai
from thefuzz import fuzz
```

C.1 Data import and wrangling

```
## Get a list of search results

API_KEY = os.environ['ELSEVIER_API_KEY']
```

```python
BASE_URL = 'https://api.elsevier.com/content/ev/results?'
QUERIES = [
    r'(((("reliability engineering") WN ALL)) AND ({english} WN LA))'
]

year_range = range(2023, 1907, -1)
results = []

def get_results_by_year(query_year, base_url=BASE_URL, api_key=API_KEY):
    """
    Get results for a given query and year (as a tuple).
    Returns a list of results.
    """
    params = {
        'apiKey': api_key,
        'pageSize': 100,
        'query': query_year[0],
        'offset': 0,
        'startYear': query_year[1],
        'endYear': query_year[1]
    }

    url = base_url + urlencode(params)

    # Permit retry 3 times after a 1 second delay
    try_count = 0
    while try_count < 3:
        try:
            r = requests.get(url)
            n_results = r.json()['PAGE']['RESULTS-COUNT']
            print('.', end='')

            # EV API has a limit of 5000 results per query
            if n_results > 5000:
                raise('Too many results: ' + str(n_results))

            if n_results == 0:
                print('0', end='')
                break

            first_offset = params['pageSize']
            last_offset = (
                math.floor(n_results / params['pageSize'])
                * params['pageSize'] + first_offset
                )

            # Loop through each page
            for doc in r.json()['PAGE']['PAGE-RESULTS']['PAGE-ENTRY']:
                results.append(doc['EI-DOCUMENT']['DOC']['DOC-ID'])
            break
        except:
            print('e', end='')
            time.sleep(1)
            try_count = try_count + 1
```

```

    if n_results > 0:
        for offset in range(first_offset, last_offset, first_offset):
            params['offset'] = offset
            url = base_url + urlencode(params)

            # Permit retry 3 times after a 1 second delay
            try_count = 0
            while try_count < 3:
                try:
                    r = requests.get(url)
                    for doc in r.json()['PAGE']['PAGE-RESULTS']['PAGE-ENTRY']:
                        results.append(doc['EI-DOCUMENT']['DOC']['DOC-ID'])
                    print('.', end='')
                    break
                except:
                    print('e', end='')
                    time.sleep(1)
                    try_count = try_count + 1


# Only run if the file doesn't already exist
if not os.path.exists('data/search_results.csv'):
    with concurrent.futures.ThreadPoolExecutor() as executor:
        executor.map(
            get_results_by_year,
            itertools.product(QUERIES, year_range)
            )

    search_df = pd.DataFrame({'doc_id': results})
    search_df.to_csv('data/search_results.csv', index=False)

search_df = pd.read_csv('data/search_results.csv')

## Get the actual document metadata

BASE_URL = 'https://api.elsevier.com/content/ev/records?'

records_dict = {
    'doc_id': [],
    'doi': [],
    'title': [],
    'abstract': [],
    'doc_type': [],
    'year': [],
    'publisher': [],
    'source_title': [],
    'authors': [],
    'author_affiliations': [],
    'country_of_origin': []
}

res_len = search_df.shape[0]
docids_chunked = [search_df['doc_id'][i:i+50] for i in range(0,res_len,50)]
```

```python
result_count = 0

def get_records_by_chunk(docids, base_url=BASE_URL, api_key=API_KEY):
    """
    Get records for a given chunk of docids. Appends to the global records_df.
    """
    global bar
    global result_count

    params = {
        'docId': ','.join(docids),
        'apiKey': api_key
    }
    url = base_url + urlencode(params)

    # Permit retry 3 times after a 1 second delay
    try_count = 0
    while try_count < 3:
        try:
            r = requests.get(url)
            results = r.json()['PAGE']['PAGE-RESULTS']['PAGE-ENTRY']

            for document in results:
                with suppress(KeyError):
                    # Account for missing data (e.g. missing JSON keys)
                    doi = None
                    doi = document['EI-DOCUMENT']['DOCUMENTPROPERTIES']['DO']
                    doi = pd.NA if not doi

                    title = None
                    title = document['EI-DOCUMENT']['DOCUMENTPROPERTIES']['TI']
                    title = pd.NA if not title

                    abstract = None
                    abstract = document['EI-DOCUMENT']['DOCUMENTPROPERTIES']['AB']
                    abstract = pd.NA if not abstract

                    doc_type = None
                    doc_type = document['EI-DOCUMENT']['DOCUMENTPROPERTIES']['DT']
                    doc_type = pd.NA if not doc_type

                    # Publication year has several options
                    year = None
                    year = document['EI-DOCUMENT']['DOCUMENTPROPERTIES']['PY']
                    if not year:
                        year = document['EI-DOCUMENT']['DOCUMENTPROPERTIES']['YR']
                    if not year:
                        year = document['EI-DOCUMENT']['DOCUMENTPROPERTIES']['SD']
                    if not year:
```

```python
                year = document['EI-DOCUMENT']['DOCUMENTPROPERTIES']['PD_YR']
            if not year:
                year = document['EI-DOCUMENT']['DOCUMENTPROPERTIES']['CPR']
            year = pd.NA if not year

            publisher = None
            publisher = document['EI-DOCUMENT']['DOCUMENTPROPERTIES']['PF']
            if not publisher:
                publisher = document['EI-DOCUMENT']['DOCUMENTPROPERTIES']['PN']
            publisher = pd.NA if not publisher

            source = None
            source = document['EI-DOCUMENT']['DOCUMENTPROPERTIES']['MT']
            if not source:
                source = document['EI-DOCUMENT']['DOCUMENTPROPERTIES']['CF']
            if not source:
                source = document['EI-DOCUMENT']['DOCUMENTPROPERTIES']['RIL']
            source = pd.NA if not source_title

            authors = None
            authors = document['EI-DOCUMENT']['AUS']
            authors = pd.NA if not authors

            author_affiliations = None
            author_affiliations = document['EI-DOCUMENT']['AFS']
            author_affiliations = pd.NA if not author_affiliations

            country = None
            country = document['EI-DOCUMENT']['DOCUMENTPROPERTIES']['CO']
            if not country_of_origin:
                country = document['EI-DOCUMENT']['DOCUMENTPROPERTIES']['ML']
            if not country_of_origin:
                country = document['EI-DOCUMENT']['DOCUMENTPROPERTIES']['PL']
            if not country_of_origin:
                country = document['EI-DOCUMENT']['DOCUMENTPROPERTIES']['PLA']
            country = pd.NA if not country_of_origin

        (records_dict['doi_id']
            .append(document['EI-DOCUMENT']['DOC']['DOC-ID']))
        records_dict['doi'].append(doi)
        records_dict['title'].append(title)
        records_dict['abstract'].append(abstract)
        records_dict['doc_type'].append(doc_type)
        records_dict['year'].append(year)
        records_dict['publisher'].append(publisher)
```

```
                records_dict['source_title'].append(source)
                records_dict['authors'].append(authors)
                records_dict['author_affiliations'].append(author_affiliations)
                records_dict['country_of_origin'].append(country)

            bar.update(result_count + 1)
            result_count = result_count + 1
            break
        except Exception as e:
            time.sleep(1)
            try_count = try_count + 1

if not os.path.exists('data/records.csv'):
    widgets = [
        progressbar.Percentage(),
        progressbar.GranularBar(markers=' '),
        ' Chunk ', progressbar.widgets.Counter(), ' of ', str(len(docids_chunked)),
        ' | ', progressbar.ETA(),
    ]
    bar = progressbar.ProgressBar(widgets=widgets, max_value=len(docids_chunked)).start()

    with concurrent.futures.ThreadPoolExecutor() as executor:
        executor.map(get_records_by_chunk, docids_chunked)
    bar.finish()

    records_df = pd.DataFrame(records_dict)

    # Clean up the year column
    records_df['year'] = records_df['year'].str.extract(r'(\d{4})')

    # Filter out records with missing abstracts and drop dups
    records_df = records_df.query('@pd.notna(abstract)').drop_duplicates(subset='doc_id')
    records_df.to_csv('data/records.csv', index=False)

records_df = pd.read_csv('data/records.csv')
```

C.2 Topic modeling

```
## Create top2vec model

corpus = records_df['abstract'].tolist()
document_ids = records_df['doc_id'].tolist()

if not os.path.exists('data/top2vec_model.mdl'):
    model = Top2Vec(corpus, speed='learn', document_ids=document_ids, workers=10)
    _ = model.hierarchical_topic_reduction(num_topics=11) # 11 topics is the sweet spot
```

```python
    model.save('data/top2vec_model.mdl')
model = Top2Vec.load('data/top2vec_model.mdl')
_ = model.hierarchical_topic_reduction(num_topics=11) # reduction isn't saved

## Apply labels

# Make labels
model_words = model.topic_words_reduced[model.doc_top_reduced,0:3]
labels = np.array([', '.join(x) for x in model_words])
topic_ids = model.doc_top_reduced

reduced_df = pd.DataFrame({
    'label': labels,
    'topic_id': topic_ids,
    'doc_id': model.document_ids
})
reduced_df.label = reduced_df.label.astype('category')

# Join reduced_df with records_df to get full dataset
records_df = records_df.join(reduced_df.set_index('doc_id'), on='doc_id')

## Visualize model

# Create UMAP model for visualization. This will take ~1 min to run.
vectors = model.document_vectors
mapping = umap.UMAP(
    n_neighbors=100,
    min_dist = 0.0,
    n_components=2,
    metric='cosine',
    verbose=True,
    n_epochs=1000
    )
reduced_fit_transform = mapping.fit_transform(vectors)
transform_df = pd.DataFrame(reduced_fit_transform, columns=['x', 'y'])
transform_df['doc_id'] = model.document_ids
transform_df.to_csv('data/transform_df.csv', index=False)

# Plot the reduced dimensionality data
records_df.join(transform_df.set_index('doc_id'), on='doc_id').hvplot(
    'x',
    'y',
    by='label',
    kind='scatter',
    width=1500,
    height=1000,
    size=2,
    alpha=0.2,
    legend=False,
    yaxis=False,
    xaxis=False,
    hover_cols=['doc_id', 'title']
).opts(bgcolor='#111111')

```

```python
## Generate topic list

# Get most relevant documents for each topic
rep_docs = pd.DataFrame()
for topic_num in model.get_topics(reduced=False)[2]:
    result = model.search_documents_by_topic(topic_num=topic_num, num_docs=1)
    row = {
        'topic_num': topic_num,
        'topic_words': ', '.join(model.topic_words[topic_num,0:3]),
        'doc_count': model.topic_sizes[topic_num],
        'title': records_df.query(f'doc_id == "{result[2][0]}"')['title'].values[0]
    }
    rep_docs = pd.concat([rep_docs, pd.DataFrame(row, index=[0])])

rep_docs.to_csv('data/topic_list.csv', index=False)

## Filter out domain topics

# Filled chart with non-relevant categories filtered out
pd.options.mode.chained_assignment = None
filtered_labels = [
    'bug, developers, software',
    'infrastructure, transportation, disruptions',
    'nuclear, plants, reactor'
    ]
records_df_filtered = records_df.query('label not in @filtered_labels')
records_df_filtered['label'] = records_df_filtered['label'].cat.remove_unused_categories()

## Friendly topic names and timing

# Establish names for each label
label_names = {
    'engineering, organizations, development': 'Management',
    'weibull, estimation, estimators': 'Statistics',
    'cut, minimal, binary': 'Modeling',
    'human, experts, hra': 'Risk Assessment',
    'preventive, replacement, maintenance': 'Maintenance',
    'charts, chart, shewhart': 'Quality Control',
    'rul, prediction, prognostic': 'Prognostics',
    'electron, silicon, oxide': 'Physics of Failure',
}

# Establish timing for each name
name_timing = {
    'Management': 1, # Applies from start of project
    'Statistics': 4, # Can only be used once testing starts
    'Modeling': 3, # Can be used effectively at PDR+
    'Risk Assessment': 2, # Can be used in concept +
    'Maintenance': 6, # Applies once product is in field
    'Quality Control': 5, # Applies after product is in production
    'Prognostics': 6, # Applies after product is in field
    'Physics of Failure:': 3 # Can be used in PDR+
```

```python
}

# Apply those names and timings to the dataframe
records_df_filtered['label_name'] = records_df_filtered['label'].map(label_names)
records_df_filtered['timing'] = records_df_filtered['label_name'].map(name_timing)
records_df_filtered.to_csv('data/records_df_filtered.csv', index=False)
```

C.3 Classification

```python
# Randomly sample 1000 documents from the full dataset
if not os.path.exists('data/sample.csv'):
    sample = records_df.sample(1000)[['doc_id', 'abstract']]
    sample.to_csv('data/sample.csv', index=False)

# Load sample from csv file with annotations
sample = pd.read_csv('data/sample.csv')
sample['real_world'] = sample['real_world'].astype('int')

openai.api_key = os.environ['OPENAI_API_KEY']

def classify_abstract(abstract):


    delay = 1 / (3500 * 60) * 10 # (s) 3500 requests per minute times 10 workers
    time.sleep(delay)
    system_prompt = (
        'Provided academic abstracts as prompts, classify them as one of the following: '
        '0: no explicit mention of an example, '
        '1: mentions an illustrative example or demonstration, or '
        '2: explicitly mentions a case study. '
        'If an abstract mentions a case study and an example, classify it as 2. '
        'Respond only with the classification ID number (0, 1, or 2).'
    )

    prompt = abstract

    while True:
        try:
            response = openai.ChatCompletion.create(
                model="gpt-3.5-turbo",
                max_tokens=1,
                temperature=1,
                messages=[
                        {"role": "system", "content": system_prompt},
                        {"role": "user", "content": prompt}
                    ]
                )
            break
        except:
```

```
            time.sleep(60) # wait 1 minute and try again
            pass
    try:
        classification = response['choices'][0]['message']['content']
    except:
        classification = '-1'

    return classification

# Map the abstracts to the classification. This will cost money.
if not os.path.exists('data/sample_classified.csv'):
    sample['classification'] = sample['abstract'].apply(classify_abstract)
    sample.to_csv('data/sample_classified.csv', index=False)

sample = pd.read_csv('data/sample_classified.csv')

# Check our accuracy
tf = 0 == sample['real_world'] - sample['classification']
tf.apply(int).sum()/len(tf)

if not os.path.exists('data/records_df_examples.csv'):
    # This will cost LOTS of money.
    from pandarallel import pandarallel
    pandarallel.initialize(
        progress_bar=True,
        verbose=0,
        nb_workers=10
        )

    records_df['examples'] = records_df['abstract'].parallel_apply(classify_abstract)
    records_df.to_csv('data/records_df_examples.csv', index=False)

records_df = pd.read_csv('data/records_df_examples.csv')
records_df.label = records_df.label.astype('category')
```

C.4 Domain topics

```
model = Top2Vec.load('data/top2vec_model.mdl')
_ = model.hierarchical_topic_reduction(num_topics=11)
def get_sub_topics(topic_num, filename, target_num_topics=5, model=model, records_df=records_df):

    def get_topic_documents(model, records_df, topic_num, reduced=True):
        global corus
        n_docs = model.get_topic_sizes(reduced=reduced)[0][topic_num]

        topic_docs = model.search_documents_by_topic(
            topic_num,
            n_docs,
```

```
            return_documents=False,
            reduced=reduced
            )
        return records_df[records_df['doc_id'].isin(list(topic_docs[1]))]

    recs = get_topic_documents(model, records_df, topic_num)
    corpus = recs['abstract'].tolist()
    document_ids = recs['doc_id'].tolist()
    sub_model = Top2Vec(corpus, speed='learn', document_ids=document_ids, workers=10, verbose=False)
    if sub_model.get_num_topics() > target_num_topics:
        sub_model.hierarchical_topic_reduction(num_topics=target_num_topics)

        reduced_df = pd.DataFrame({
            'sub_label': np.array([', '.join(x) for x in sub_model.topic_words_reduced[sub_model.doc_top_reduced,0:5]]),
            'sub_topic_id': sub_model.doc_top_reduced,
            'doc_id': sub_model.document_ids
        })
    else:
        reduced_df = pd.DataFrame({
            'sub_label': np.array([', '.join(x) for x in sub_model.topic_words[sub_model.doc_top,0:5]]),
            'sub_topic_id': sub_model.doc_top,
            'doc_id': sub_model.document_ids
        })

    reduced_df.sub_label = reduced_df.sub_label.astype('category')
    output_df = reduced_df.merge(records_df.set_index('doc_id'), on='doc_id')
    output_df.to_csv(filename, index=False)
    return (output_df.groupby(['sub_label'])
        .count()['doc_id']
        .sort_values(ascending=False)[0:target_num_topics]
        )
```

C.5 Document pool validation

```
def get_reference_coverage(filename, title=1, records_df=records_df):
    with open('data/refs_zio.txt', 'r') as f:
        references = f.read().splitlines()

    titles = []
    for item in references:
        titles.append(item.split('.')[title].strip())

    matches = []

    def get_matches(titles, records_df=records_df):

        global matches # Shared across threads

        for title in titles:
```

```
            for record in records_df.title:
                if fuzz.ratio(title, record) > 90:
                    matches.append(title)
                    break

    titles_chunked = [titles[i:i+10] for i in range(0,len(titles),10)]
    with concurrent.futures.ThreadPoolExecutor() as executor:
        executor.map(get_matches, titles_chunked)

    print(len(matches))
    print(len(matches)/len(references))
```

C.6 Visualizations

```
library(tidyverse)
library(lubridate)
library(scales)
library(svglite)
library(ggforce)

# Load the full dataset from reduced_df_joined_filtered.csv
filtered_df <- read_csv("data/records_df_filtered.csv")

# Arrange topics manually (we choose ascending order of count in 2022)
topics <- c(
    "Risk Assessment",
    "Modeling",
    "Physics of Failure",
    "Management",
    "Quality Control",
    "Maintenance",
    "Statistics",
    "Prognostics"
    )

## Plot cluster map

transform_df <- read_csv("data/transform_df.csv")
filtered_df %>%
    left_join(transform_df, by = "doc_id") %>%
    ggplot(aes(x = x, y = y, color = factor(label_name, levels=topics))) +
    geom_point(size = 0.2, alpha = 0.1) +
    theme_void() +
    scale_color_brewer(palette="Spectral") +
    labs(
        color = "Topic",
        # title = "Reliability paper clustering"
    ) +
    guides(colour = guide_legend(override.aes = list(alpha = 1, size=4))) +
    theme(
        plot.title = element_text(hjust = 0.5, margin = margin(t = 0, r = 0, b =
        ↪ 10, l = 0)),
```

```r
        panel.background = element_rect(fill = 'black', color = 'black')     )

ggsave("figures/res_paper_clustering.svg", width=6, height=4, dpi=300)

## Timing score

filtered_df %>%
    group_by(year) %>%
    summarise(timing_score = mean(timing, na.rm=TRUE)) %>%
    ggplot(aes(x=year, y=timing_score)) +
    geom_smooth(span = 0.4, color = "black") +
    # geom_line() +
    theme_minimal() +
    labs(
        # title = "Mean timing of reliability activities over time",
        x = "Publication year",
        y = "Phase addressed in reliability papers",
    ) +
    theme(
        axis.title.y = element_text(margin = margin(t = 0, r = 30, b = 0, l = 0)),
        axis.title.x = element_text(margin = margin(t = 10, r = 0, b = 0, l = 0)),
        axis.text.y = element_text(lineheight = 1.1, size = 10)

        ) +
    scale_y_continuous(breaks=c(1, 2, 3, 4, 5, 6), labels=c("Planning", "Concept", "System\nDesign", "Detailed\nDesign", "Testing", "Production"))


ggsave("figures/res_timing_score.svg", width=6, height=4, dpi=300)

## Trend plotting functions for top-level and domains

plot_fill <- function(data_filename, plot_filename, title) {

    # For cases with comma separated terms, take only the first three
    input_df <- read_csv(data_filename) %>%
        mutate(sub_label = strsplit(sub_label, ", ") %>%
        sapply(function(x) paste(x[1:3], collapse=", ")))

    # Account for missing years
    all_values <- expand.grid(
        sub_label = unique(input_df$sub_label),
        year = unique(input_df$year)
    )

    summary_df <- input_df %>%
        group_by(sub_label, year) %>%
        summarise(count = n()) %>%
        ungroup()

```

```
    summary_df <- full_join(all_values, summary_df, by = c("sub_label", "year")) %>%
        replace_na(list(count = 0))

    p <- summary_df %>%
        ggplot(aes(x=year, y=count, fill = factor(sub_label))) +
        geom_area(position="fill") +
        theme_minimal() +
        labs(
            # title = paste0("Fraction of topics in the ", title, " subtopic over time"),
            x = "Publication year",
            y = paste0("Fraction of ", title, " reliability papers"),
            fill = "Topic",
        ) +
        theme(
            axis.title.y = element_text(margin = margin(t = 0, r = 10, b = 0, l = 0)),
            axis.title.x = element_text(margin = margin(t = 10, r = 0, b = 0, l = 0))
            ) +
        scale_fill_brewer(palette="Spectral") +
        scale_y_continuous(labels=scales::percent)
    print(p)
    ggsave(plot_filename, width=6, height=4, dpi=300)
}

plot_count <- function(data_filename, plot_filename, title) {

    # For cases with comma separated terms, take only the first three
    input_df <- read_csv(data_filename) %>%
        mutate(sub_label = strsplit(sub_label, ", ") %>%
        sapply(function(x) paste(x[1:3], collapse=", ")))

    # Account for missing years
    all_values <- expand.grid(
        sub_label = unique(input_df$sub_label),
        year = unique(input_df$year)
    )

    summary_df <- input_df %>%
        group_by(sub_label, year) %>%
        summarise(count = n()) %>%
        ungroup()

    summary_df <- full_join(all_values, summary_df, by = c("sub_label", "year")) %>%
        replace_na(list(count = 0))

    p <- summary_df %>%
        filter(year < 2023) %>%
        ggplot(aes(x=year, y=count, fill = factor(sub_label))) +
        geom_area(stat="identity") +
        theme_minimal() +
```

```r
        labs(
            # title = paste0("Count of topics in the ", title, " subtopic over time"),
            x = "Publication year",
            y = paste0("Count of ", title, " reliability papers"),
            fill = "Topic",
        ) +
        theme(
            axis.title.y = element_text(margin = margin(t = 0, r = 10, b = 0, l = 0)),
            axis.title.x = element_text(margin = margin(t = 10, r = 0, b = 0, l = 0))
            ) +
        scale_fill_brewer(palette="Spectral")
    print(p)
    ggsave(plot_filename, width=6, height=4, dpi=300)
}

plot_growth <- function(data_filename, plot_filename, title, ylim) {
    p <- read_csv(data_filename) %>%
        mutate(sub_label = strsplit(sub_label, ", ") %>%
        sapply(function(x) paste(x[1:3], collapse=", "))) %>%
        group_by(year, sub_label) %>%
        summarise(count = n()) %>%
        filter(year < 2023) %>%
        group_by(sub_label) %>%
        mutate(growth_rate = (count - lag(count)) / lag(count)) %>%
        ggplot(aes(x=year, y=growth_rate, fill = factor(sub_label), color = factor(sub_label))) +
        geom_smooth(span = 0.8, se=FALSE) +
        theme_bw() +
        labs(
            # title = paste0("Growth rate of topics in the ", title, " subtopic over time"),
            y = paste0("Growth rate of ", title, " reliability papers"),
            x = "Publication year",
            color = "Topic",
        ) +
        theme(
            axis.title.y = element_text(margin = margin(t = 0, r = 10, b = 0, l = 0)),
            axis.title.x = element_text(margin = margin(t = 10, r = 0, b = 0, l = 0)),
            panel.border = element_blank(),
            axis.ticks.x = element_blank(),
            axis.ticks.y = element_blank(),
            zoom.x = element_rect(fill = NA, color = "lightgray"),
            zoom.y = element_rect(fill = NA, color = NA)
            ) +
        scale_color_brewer(palette="Spectral") +
        scale_fill_brewer(palette="Spectral") +
        guides(fill = "none") +
        facet_zoom(xlim=c(2000, 2022), ylim=ylim, zoom.size=3, horizontal=FALSE) +
```

```r
        scale_y_continuous(labels=scales::percent)
    print(p)
    ggsave(plot_filename, width=6, height=4, dpi=300)
}

plot_examples <- function(data_filename, plot_filename, title, ylim) {
    p <- read_csv(data_filename) %>%
        mutate(sub_label = strsplit(sub_label, ", ") %>%
        sapply(function(x) paste(x[1:3], collapse=", "))) %>%
        filter(year < 2023, examples %in% c(0, 1, 2)) %>%
        group_by(year, sub_label) %>%
        summarise(example_score = mean(as.numeric(examples), na.rm=TRUE)) %>%
        ggplot(aes(x=year, y=example_score, fill = factor(sub_label), color = factor(sub_label))) +
        geom_smooth(span = 0.6, se=FALSE) +
        theme_bw() +
        labs(
            # title = paste0("Mean example content of topics in the ", title, " subtopic over time"),
            x = "Publication year",
            y = paste0("Examples in ", title, " reliability papers"),
            color = "Topic",
        ) +
        theme(
            axis.title.y = element_text(margin = margin(t = 0, r = 10, b = 0, l = 0)),
            axis.title.x = element_text(margin = margin(t = 10, r = 0, b = 0, l = 0)),
            panel.border = element_blank(),
            axis.ticks.x = element_blank(),
            axis.ticks.y = element_blank(),
            zoom.x = element_rect(fill = NA, color = "lightgray"),
            zoom.y = element_rect(fill = NA, color = NA),
            axis.text.y = element_text(lineheight = 1.1, size = 10)
            ) +
        scale_color_brewer(palette="Spectral") +
        scale_fill_brewer(palette="Spectral") +
        scale_y_continuous(breaks=c(0, 1, 2), labels=c("None", "Illustrative\nExample", "Case\nStudy")) +
        guides(fill = "none") +
        facet_zoom(xlim=c(2000, 2022), ylim=ylim, zoom.size=3, horizontal=FALSE)
    print(p)
    ggsave(plot_filename, width=6, height=4, dpi=300)
}
```

Appendix D

References

Adachi, Takao and Bruce R. Ellingwood (Jan. 2008). "Serviceability of earthquake-damaged water systems: Effects of electrical power availability and power backup systems on system vulnerability". en. In: *Reliability Engineering & System Safety* 93.1, pp. 78–88. ISSN: 0951-8320. DOI: 10.1016/j.ress.2006.10.014. URL: https://www.sciencedirect.com/science/article/pii/S0951832006002493 (visited on 04/30/2023).

Ahmed, Qadeer, Syed Asif Raza, and Dahham M. Al-Anazi (2021). "Reliability-based fault analysis models with industrial applications: A systematic literature review". en. In: *Quality and Reliability Engineering International* 37.4. _eprint: https://onlinelibrary.wiley.com/doi/pdf/10.1002/qre.2797, pp. 1307–1333. ISSN: 1099-1638. DOI: 10.1002/qre.2797. URL: https://onlinelibrary.wiley.com/doi/abs/10.1002/qre.2797 (visited on 04/08/2023).

Alkaff, Abdullah (May 2021). "Discrete time dynamic reliability modeling for systems with multistate components". en. In: *Reliability Engineering & System Safety* 209, p. 107462. ISSN: 0951-8320. DOI: 10.1016/j.ress.2021.107462. URL: https://www.sciencedirect.com/science/article/pii/S0951832021000302 (visited on 04/30/2023).

An, Dawn, Nam H. Kim, and Joo-Ho Choi (Jan. 2015). "Practical options for selecting data-driven or physics-based prognostics algorithms with reviews". en. In: *Reliability Engineering & System Safety* 133, pp. 223–236. ISSN: 0951-8320. DOI: 10.1016/j.ress.2014.09.014. URL: https://www.sciencedirect.com/science/article/pii/S0951832014002245 (visited on 04/15/2023).

Angelov, Dimo (Aug. 2020). *Top2Vec: Distributed Representations of Topics*. DOI: 10.48550/arXiv.2008.09470. URL: http://arxiv.org/abs/2008.09470 (visited on 04/08/2023).

ASQ (2023). *What is Reliability? Quality & Reliability Defined | ASQ*. URL: https://asq.org/quality-resources/reliability (visited on 04/23/2023).

Aven, Terje (June 2017). "Improving the foundation and practice of reliability engineering". en. In: *Proceedings of the Institution of Mechanical Engineers, Part O: Journal of Risk and Reliability* 231.3. Publisher: SAGE Publications, pp. 295–305. ISSN: 1748-006X. DOI: 10.1177/1748006X17699478. URL: https://doi.org/10.1177/1748006X17699478 (visited on 04/22/2023).

Balkus, Salvador and Donghui Yan (May 2022). *Improving Short Text Classification With Augmented Data Using GPT-3*. arXiv:2205.10981 [cs]. DOI: 10.48550/arXiv. 2205.10981. URL: http://arxiv.org/abs/2205.10981 (visited on 04/08/2023).
Bhattacharya, Ritwik and Biswabrata Pradhan (2018). "Bayesian design of life testing plans under hybrid censoring scheme". en. In: *Quality and Reliability Engineering International* 34.1. _eprint: https://onlinelibrary.wiley.com/doi/pdf/10.1002/qre.2240, pp. 93–106. ISSN: 1099-1638. DOI: 10.1002/qre.2240. URL: https://onlinelibrary.wiley.com/doi/abs/10.1002/qre.2240 (visited on 04/30/2023).
Blanks, Henry S. (1998). "The challenge of quantitative reliability". en. In: *Quality and Reliability Engineering International* 14.3, pp. 167–176. ISSN: 1099-1638. DOI: 10.1002/(SICI)1099-1638(199805/06)14:3<167::AID-QRE154>3.0.CO;2-L. URL: https://onlinelibrary.wiley.com/doi/abs/10.1002/%28SICI%291099-1638%28199805/06%2914%3A3%3C167%3A%3AAID-QRE154%3E3.0.CO%3B2-L (visited on 04/22/2023).
Blei, David M., Andrew Y. Ng, and Michael I. Jordan (Mar. 2003). "Latent dirichlet allocation". In: *The Journal of Machine Learning Research* 3.null, pp. 993–1022. ISSN: 1532-4435.
Bradley, Dr Edgar (Dec. 2022). *Reliability Engineering: A Life Cycle Approach*. 2nd ed. Boca Raton: CRC Press. ISBN: 978-1-00-332648-9. DOI: 10.1201/9781003326489.
Breneman, James E., Chittaranjan Sahay, and Elmer E. Lewis (Apr. 2022). *Introduction to Reliability Engineering*. English. 3rd edition. Wiley.
Brown, Tom B. et al. (July 2020). *Language Models are Few-Shot Learners*. DOI: 10.48550/arXiv.2005.14165. URL: http://arxiv.org/abs/2005.14165 (visited on 04/08/2023).
Carnerud, Daniel (2017). "Exploring research on quality and reliability management through text mining methodology". English. In: *The International Journal of Quality & Reliability Management* 34.7. Num Pages: 40 Place: Bradford, United Kingdom Publisher: Emerald Group Publishing Limited, pp. 975–1014. ISSN: 0265671X. DOI: 10.1108/IJQRM-03-2015-0033. URL: https://www.proquest.com/docview/1927782784/abstract/BD2FDA981D8E4453PQ/1 (visited on 04/16/2023).
Carot, V. and J. Sanz (May 2000). "Criticality and sensitivity analysis of the components of a system". en. In: *Reliability Engineering & System Safety* 68.2, pp. 147–152. ISSN: 0951-8320. DOI: 10.1016/S0951-8320(00)00011-9. URL: https://www.sciencedirect.com/science/article/pii/S0951832000000119 (visited on 04/30/2023).
Chowdhery, Aakanksha et al. (Oct. 2022). *PaLM: Scaling Language Modeling with Pathways*. arXiv:2204.02311 [cs]. DOI: 10.48550/arXiv.2204.02311. URL: http://arxiv.org/abs/2204.02311 (visited on 04/08/2023).
Contreras, Diana et al. (Aug. 2022). "Accuracy of a pre-trained sentiment analysis (SA) classification model on tweets related to emergency response and early recovery assessment: the case of 2019 Albanian earthquake". en. In: *Natural Hazards* 113.1, pp. 403–421. ISSN: 1573-0840. DOI: 10.1007/s11069-022-05307-w. URL: https://doi.org/10.1007/s11069-022-05307-w (visited on 04/16/2023).
Devlin, Jacob et al. (May 2019). *BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding*. arXiv:1810.04805 [cs]. DOI: 10.48550/arXiv. 1810.04805. URL: http://arxiv.org/abs/1810.04805 (visited on 04/08/2023).

Economou, M. (Jan. 2004). “The merits and limitations of reliability predictions”. In: *Annual Symposium Reliability and Maintainability, 2004 - RAMS*, pp. 352–357. DOI: 10.1109/RAMS.2004.1285474.

Emura, Takeshi and Hirofumi Michimae (2022). “Left-truncated and right-censored field failure data: Review of parametric analysis for reliability”. en. In: *Quality and Reliability Engineering International* 38.7, pp. 3919–3934. ISSN: 1099-1638. DOI: 10.1002/qre.3161. URL: https://onlinelibrary.wiley.com/doi/abs/10.1002/qre.3161 (visited on 04/16/2023).

Farsi, Mohammad Ali and Enrico Zio (June 2019). “Industry 4.0: Some Challenges and Opportunities for Reliability Engineering”. In: *International Journal of Reliability, Risk and Safety: Theory and Application* 2.1. Publisher: Aerospace Research Institute, Ministry of Science, Research and Technology, IRAN, pp. 23–34. ISSN: 2676-3346. DOI: 10.30699/ijrrs.2.1.4. URL: http://www.ijrrs.com/article_91148.html (visited on 04/22/2023).

Fazlollahtabar, Hamed and Seyed Taghi Akhavan Niaki (Nov. 2017). *Reliability Models of Complex Systems for Robots and Automation*. en. CRC Press. ISBN: 978-0-203-70417-2. DOI: 10.1201/b22491. URL: https://www.taylorfrancis.com/books/mono/10.1201/b22491/reliability-models-complex-systems-robots-automation-hamed-fazlollahtabar-seyed-taghi-akhavan-niaki (visited on 01/22/2023).

Forcina, Antonio et al. (2020). “Reliability allocation methods: A systematic literature review”. en. In: *Quality and Reliability Engineering International* 36.6, pp. 2085–2107. ISSN: 1099-1638. DOI: 10.1002/qre.2675. URL: https://onlinelibrary.wiley.com/doi/abs/10.1002/qre.2675 (visited on 01/18/2023).

Freeman, Laura J., Rebecca M. Medlin, and Thomas H. Johnson (Jan. 2019). “Challenges and new methods for designing reliability experiments”. In: *Quality Engineering* 31.1. Publisher: Taylor & Francis Ltd, pp. 108–121. ISSN: 08982112. DOI: 10.1080/08982112.2018.1546394. URL: https://search.ebscohost.com/login.aspx?direct=true&db=bth&AN=135863014&site=ehost-live&scope=site&authtype=shib&custid=s8978330 (visited on 04/22/2023).

Goel, Parveen Singh (1998). “Integration of quality, reliability, and durability engineering concepts in product development”. English. ISBN: 9780591795912. Ph.D. United States – Michigan: Wayne State University. URL: https://www.proquest.com/docview/304454179/abstract/73D9B094DE3A4052PQ/1 (visited on 05/07/2023).

Grootendorst, Maarten (Mar. 2022). *BERTopic: Neural topic modeling with a class-based TF-IDF procedure*. arXiv:2203.05794 [cs]. DOI: 10.48550/arXiv.2203.05794. URL: http://arxiv.org/abs/2203.05794 (visited on 02/09/2023).

Higgins, Julian P. T. et al. (2019). *Cochrane Handbook for Systematic Reviews of Interventions*. eng. 2nd ed. Wiley Cochrane Series. Newark: John Wiley & Sons, Ltd. ISBN: 978-1-119-53665-9. DOI: 10.1002/9781119536604.

Hussain, A. Z. M. O. and D. N. P. Murthy (Dec. 2003). “Warranty and optimal reliability improvement through product development”. en. In: *Mathematical and Computer Modelling*. Stochastic models in engineering, technology, and management 38.11, pp. 1211–1217. ISSN: 0895-7177. DOI: 10.1016/S0895-7177(03)90122-1. URL: https://www.sciencedirect.com/science/article/pii/S0895717703901221 (visited on 04/29/2023).

Kelly, Dana L. (Jan. 1992). “Probabilistic analysis of flow control as an alternative to level control for BWR ATWS”. en. In: *Reliability Engineering & System Safety* 36.2, pp. 87–94. ISSN: 0951-8320. DOI: 10.1016/0951-8320(92)90089-4. URL: https://www.sciencedirect.com/science/article/pii/0951832092900894 (visited on 04/30/2023).

Larsen, Peder and Markus von Ins (Mar. 2010). “The rate of growth in scientific publication and the decline in coverage provided by Science Citation Index”. en_US. In: *Scientometrics* 84.3. Publisher: Akadémiai Kiadó, co-published with Springer Science+Business Media B.V., Formerly Kluwer Academic Publishers B.V. Section: Scientometrics, pp. 575–603. ISSN: 0138-9130, 1588-2861. DOI: 10.1007/s11192-010-0202-z. URL: https://akjournals.com/view/journals/11192/84/3/article-p575.xml (visited on 05/01/2023).

Le, Quoc V. and Tomas Mikolov (May 2014). *Distributed Representations of Sentences and Documents*. en. URL: https://arxiv.org/abs/1405.4053v2 (visited on 04/09/2023).

Leech, C. (1985). “An introduction to quality assurance in the information processing industry”. In: *Software World*. Softw. World (UK) 16.1. Place: UK, pp. 7–8. ISSN: 0038-0652.

Li, Xiang, Wei Zhang, and Qian Ding (Feb. 2019). “Deep learning-based remaining useful life estimation of bearings using multi-scale feature extraction”. en. In: *Reliability Engineering & System Safety* 182, pp. 208–218. ISSN: 0951-8320. DOI: 10.1016/j.ress.2018.11.011. URL: https://www.sciencedirect.com/science/article/pii/S0951832018308299 (visited on 04/30/2023).

Lisnianski, Anatoly, Ilia Frenkel, and Lev Khvatskin (2021). “LZ-Transform and Inverse LZ-Transform of a Discrete-State Continuous-Time Markov Process”. en. In: *Modern Dynamic Reliability Analysis for Multi-state Systems: Stochastic Processes and the Lz-Transform*. Ed. by Anatoly Lisnianski, Ilia Frenkel, and Lev Khvatskin. Springer Series in Reliability Engineering. Cham: Springer International Publishing, pp. 53–84. ISBN: 978-3-030-52488-3. DOI: 10.1007/978-3-030-52488-3_3. URL: https://doi.org/10.1007/978-3-030-52488-3_3 (visited on 04/30/2023).

Liu, X. et al. (July 1999). “The properties of 2.7 eV cathodoluminescence from SiO2 film on Si substrate”. en. In: *Journal of Physics D: Applied Physics* 32.14, p. 1563. ISSN: 0022-3727. DOI: 10.1088/0022-3727/32/14/302. URL: https://dx.doi.org/10.1088/0022-3727/32/14/302 (visited on 04/30/2023).

Luo, Jinjian et al. (Oct. 2021). “A Runtime Monitoring Based Fuzzing Framework for Temporal Properties”. In: *2021 IEEE International Symposium on Software Reliability Engineering Workshops (ISSREW)*, pp. 300–301. DOI: 10.1109/ISSREW53611.2021.00089.

Malamud, Carl (Oct. 2021). *The General Index*. URL: http://archive.org/details/GeneralIndex (visited on 04/17/2023).

Mancuso, A. et al. (Aug. 2016). “Risk-based optimization of pipe inspections in large underground networks with imprecise information”. en. In: *Reliability Engineering & System Safety* 152, pp. 228–238. ISSN: 0951-8320. DOI: 10.1016/j.ress.2016.03.011. URL: https://www.sciencedirect.com/science/article/pii/S0951832016000776 (visited on 04/30/2023).

Maurya, Ankur and Divya Kumar (2020). “Reliability of safety-critical systems: A state-of-the-art review”. en. In: *Quality and Reliability Engineering International* 36.7. _eprint: https://onlinelibrary.wiley.com/doi/pdf/10.1002/qre.2715, pp. 2547–2568. ISSN: 1099-1638. DOI: 10.1002/qre.2715. URL: https://onlinelibrary.wiley.com/doi/abs/10.1002/qre.2715 (visited on 04/16/2023).

McInnes, Leland, John Healy, and James Melville (Feb. 2018). *UMAP: Uniform Manifold Approximation and Projection for Dimension Reduction*. en. URL: https://arxiv.org/abs/1802.03426v3 (visited on 04/09/2023).

Meeker, William Q. and Yili Hong (Jan. 2014). “Reliability Meets Big Data: Opportunities and Challenges”. In: *Quality Engineering* 26.1. Publisher: Taylor & Francis _eprint: https://doi.org/10.1080/08982112.2014.846119, pp. 102–116. ISSN: 0898-2112. DOI: 10.1080/08982112.2014.846119. URL: https://doi.org/10.1080/08982112.2014.846119 (visited on 04/22/2023).

Miller, George A. (Mar. 1956). “The magical number seven, plus or minus two: Some limits on our capacity for processing information”. English. In: *Psychological Review* 63.2. Num Pages: 81-97 Place: Washington, US Publisher: American Psychological Association (US), pp. 81–97. ISSN: 0033-295X. DOI: 10.1037/h0043158. URL: https://www.proquest.com/docview/614291982/abstract/80A3F6F3B034957PQ/1 (visited on 04/30/2023).

Nor, Ahmad Kamal M., Srinivasa R. Pedapati, and Masdi Muhammad (Sept. 2021). “Reliability engineering applications in electronic, software, nuclear and aerospace industries: A 20 year review (2000–2020)”. en. In: *Ain Shams Engineering Journal* 12.3, pp. 3009–3019. ISSN: 2090-4479. DOI: 10.1016/j.asej.2021.02.015. URL: https://www.sciencedirect.com/science/article/pii/S2090447921001179 (visited on 04/16/2023).

Pascual, Rodrigo et al. (Mar. 2017). “Optimal repairable spare-parts procurement policy under total business volume discount environment”. en. In: *Reliability Engineering & System Safety* 159, pp. 276–282. ISSN: 0951-8320. DOI: 10.1016/j.ress.2016.10.034. URL: https://www.sciencedirect.com/science/article/pii/S0951832016307311 (visited on 04/30/2023).

Plessis, Alfred Du et al. (Jan. 2014). “The thirty greatest reliability challenges”. In: *2014 Reliability and Maintainability Symposium*. ISSN: 0149-144X, pp. 1–6. DOI: 10.1109/RAMS.2014.6798478.

Quigley, John and Lesley Walls (Dec. 2003). “Cost–benefit modelling for reliability growth”. In: *Journal of the Operational Research Society* 54.12. Publisher: Taylor & Francis _eprint: https://doi.org/10.1057/palgrave.jors.2601633, pp. 1234–1241. ISSN: 0160-5682. DOI: 10.1057/palgrave.jors.2601633. URL: https://doi.org/10.1057/palgrave.jors.2601633 (visited on 04/29/2023).

Raheja, Dev and Louis J. Gullo (2012). *Design for Reliability*. 1st ed. John Wiley & Sons, Ltd. URL: https://onlinelibrary.wiley.com/doi/10.1002/9781118310052 (visited on 01/28/2023).

Rahman, Md Farhadur et al. (Feb. 2016). *HDBSCAN: Density based Clustering over Location Based Services*. en. URL: https://arxiv.org/abs/1602.03730v2 (visited on 04/09/2023).

Ralph L., Keeney and Detlof von Winterfeldt (Jan. 1988). "Probabilities are useful to quantify expert judgments". en. In: *Reliability Engineering & System Safety* 23.4, pp. 293–298. ISSN: 0951-8320. DOI: 10.1016/0951-8320(88)90041-5. URL: https://www.sciencedirect.com/science/article/pii/0951832088900415 (visited on 04/30/2023).

Rehurek, Radim and Petr Sojka (May 2010). "Software Framework for Topic Modelling with Large Corpora". In: (visited on 04/08/2023).

Righi, Angela Weber, Tarcisio Abreu Saurin, and Priscila Wachs (Sept. 2015). "A systematic literature review of resilience engineering: Research areas and a research agenda proposal". en. In: *Reliability Engineering & System Safety*. Special Issue on Resilience Engineering 141, pp. 142–152. ISSN: 0951-8320. DOI: 10.1016/j.ress.2015.03.007. URL: https://www.sciencedirect.com/science/article/pii/S0951832015000654 (visited on 01/18/2023).

Sheu, Shey-Huei, Yuh-Bin Lin, and Gwo-Liang Liao (Mar. 2006). "Optimum policies for a system with general imperfect maintenance". en. In: *Reliability Engineering & System Safety* 91.3, pp. 362–369. ISSN: 0951-8320. DOI: 10.1016/j.ress.2005.01.015. URL: https://www.sciencedirect.com/science/article/pii/S0951832005000517 (visited on 04/30/2023).

Sievert, Carson and Kenneth Shirley (2014). "LDAvis: A method for visualizing and interpreting topics". en. In: *Proceedings of the Workshop on Interactive Language Learning, Visualization, and Interfaces*. Baltimore, Maryland, USA: Association for Computational Linguistics, pp. 63–70. DOI: 10.3115/v1/W14-3110. URL: http://aclweb.org/anthology/W14-3110 (visited on 02/20/2023).

Sundaram, Girish and Daniel Berleant (Nov. 2022). *Automating Systematic Literature Reviews with Natural Language Processing and Text Mining: a Systematic Literature Review*. arXiv:2211.15397 [cs]. DOI: 10.48550/arXiv.2211.15397. URL: http://arxiv.org/abs/2211.15397 (visited on 01/18/2023).

Swartz, Aaron (2008). *Guerilla Open Access Manifesto*. URL: http://archive.org/details/GuerillaOpenAccessManifesto (visited on 04/17/2023).

Thornton, Anna C. (Feb. 2021). *Product Realization: Going from One to a Million*. English. 1st edition. Hoboken, NJ, USA: Wiley. ISBN: 978-1-119-64953-3.

Touvron, Hugo et al. (Feb. 2023). *LLaMA: Open and Efficient Foundation Language Models*. arXiv:2302.13971 [cs]. DOI: 10.48550/arXiv.2302.13971. URL: http://arxiv.org/abs/2302.13971 (visited on 04/08/2023).

Tranfield, David, David Denyer, and Palminder Smart (2003). "Towards a Methodology for Developing Evidence-Informed Management Knowledge by Means of Systematic Review". en. In: *British Journal of Management* 14.3, pp. 207–222. ISSN: 1467-8551. DOI: 10.1111/1467-8551.00375. URL: https://onlinelibrary.wiley.com/doi/abs/10.1111/1467-8551.00375 (visited on 01/18/2023).

Ulrich, Karl T. (2020). *Product design and development*. eng. Seventh edition. New York, NY: McGraw-Hill Education. ISBN: 978-1-260-04365-5.

Vaswani, Ashish et al. (Dec. 2017). *Attention Is All You Need*. arXiv:1706.03762 [cs]. DOI: 10.48550/arXiv.1706.03762. URL: http://arxiv.org/abs/1706.03762 (visited on 04/08/2023).

Yang, Su-Fen and Smiley W. Cheng (2011). “A new non-parametric CUSUM mean chart”. en. In: *Quality and Reliability Engineering International* 27.7. _eprint: https://onlinelibrary.wiley.com/doi/pdf/10.1002/qre.1171, pp. 867–875. ISSN: 1099-1638. DOI: 10.1002/qre.1171. URL: https://onlinelibrary.wiley.com/doi/abs/10.1002/qre.1171 (visited on 04/30/2023).

Zio, E. (Feb. 2009). “Reliability engineering: Old problems and new challenges”. en. In: *Reliability Engineering & System Safety* 94.2, pp. 125–141. ISSN: 0951-8320. DOI: 10.1016/j.ress.2008.06.002. URL: https://www.sciencedirect.com/science/article/pii/S0951832008001749 (visited on 01/15/2023).

Zio, Enrico (Dec. 2016). “Some Challenges and Opportunities in Reliability Engineering”. In: *IEEE Transactions on Reliability* 65.4. Conference Name: IEEE Transactions on Reliability, pp. 1769–1782. ISSN: 1558-1721. DOI: 10.1109/TR.2016.2591504.