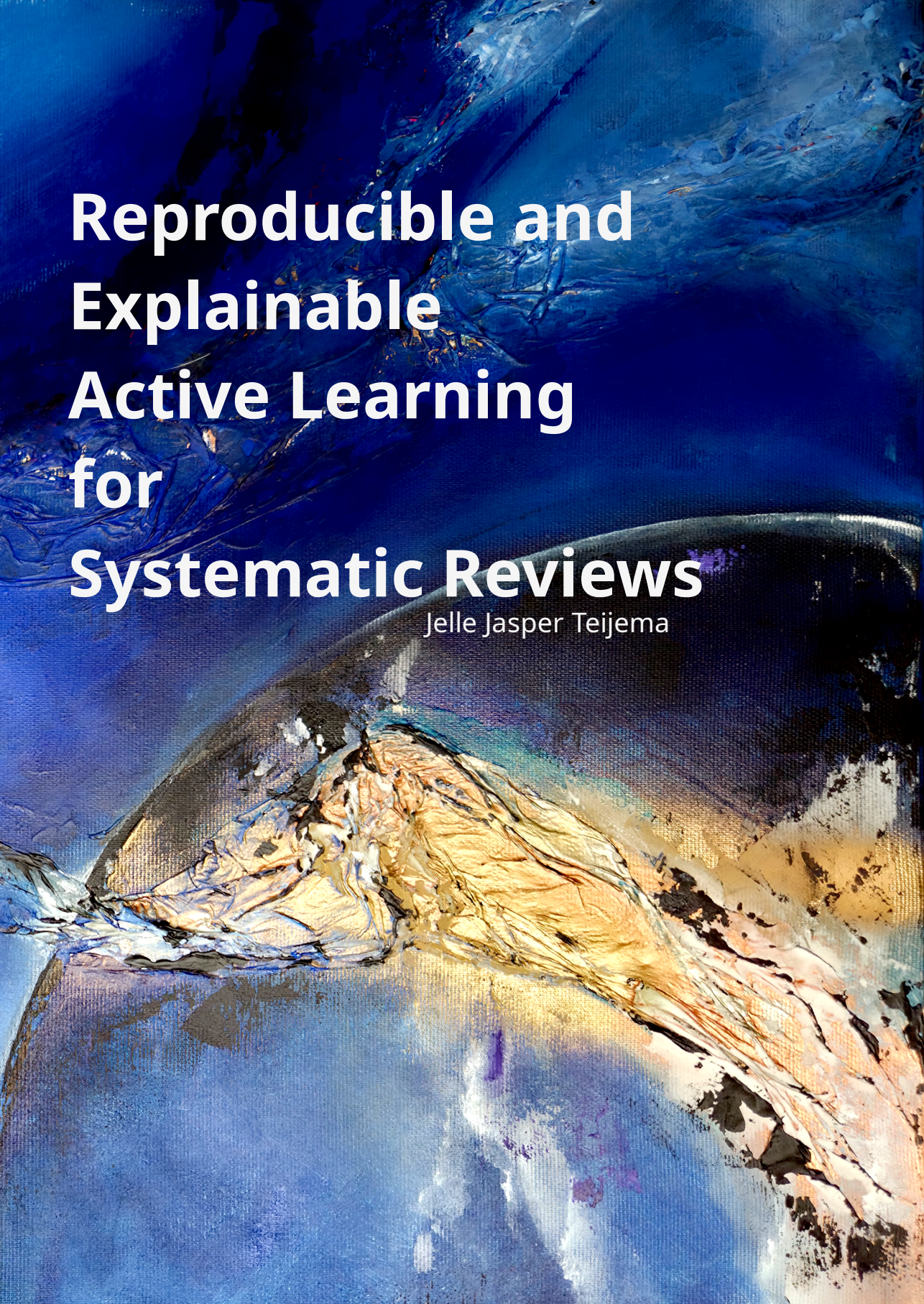


Reproducible and Explainable Active Learning for Systematic Reviews

Jelle Jasper Teijema

Reproducible and Explainable Active Learning for Systematic Reviews

An Applied Data Science Contribution Uniting Academic and Industrial Methodologies

Jelle Jasper Teijema

ISBN: 978-90-393-8059-8
Cover design: Jelle Jasper Teijema, Simone Mulder
Layout: Jelle Jasper Teijema
Print: Proefschriften.nl

Reproducible and Explainable Active Learning for Systematic Reviews

Reproduceerbaar en uitlegbaar gebruik van active learning in systematische reviews
(met een samenvatting in het Nederlands)

Proefschrift

ter verkrijging van de graad van doctor aan de
Universiteit Utrecht
op gezag van de
rector magnificus, prof. dr. ir. W. Hazeleger,
ingevolge het besluit van het College voor Promoties
in het openbaar te verdedigen op

donderdag 11 juni 2026 des middags te 12.15 uur

door

Jelle Jasper Teijema

geboren op 2 december 1996
te Amersfoort

Promotoren:

Prof. dr. A.G.J. (Rens) van de Schoot
Prof. dr. L. (Lars) Tummers

Copromotoren:

Dr. R.A. (Robert) Bagheri
Dr. M.J.S. (Matthieu) Brinkhuis

Beoordelingscommissie:

Prof. dr. A.P.J. (Antal) van den Bosch
Dr. ing. G.M. (Georg) Krempl
Dr. I. (Ioanna) Lykourentzou
Prof. dr. D.L. (Daniel) Oberski
Prof. dr. M.R. (Marco) Spruit

Fortune favors the prepared mind.

— From: *Louis Pasteur*

Contents

Chapter 1

Introduction

The rate of scientific publication is increasing rapidly (Bornmann & Mutz, 2015; Fortunato et al., 2018; Price, 1963). The number of papers published each year grows exponentially, with an estimated doubling time of about 17.3 years (Bornmann, Haunschild, & Mutz, 2021). Currently, in the systematic review-heavy biomedical field alone, more than 1 million papers are submitted to the PubMed database each year (Landhuis, 2016). This increase in publication rate supports faster dissemination of findings and accelerates the exchange of knowledge, but at the same time, leads to increased fragmentation of information.

Fortunato et al. (2018) show that, while the number of publications grows exponentially, the number of unique ideas rises only linearly. The amount of research produced per idea, therefore, increases exponentially. As a result, it becomes increasingly difficult for researchers to keep track of related work and to assemble a coherent overview of all available evidence in a field.

Systematic reviews assemble and summarize research scattered across studies. They allow researchers to build on consolidated evidence rather than having to navigate the literature. In this way, systematic reviews serve as convergence points that facilitate the reuse and dissemination of scientific knowledge. Systematic reviews are designed to use reliable, transparent, standardized procedures. However, the implementation of these standardized procedures necessitates a considerable investment of time, labor, and expertise. During the screening phase of the systematic review, thousands of scientific articles are checked for relevance. The labor cost of labeling literature is closely correlated to the number of studies available on a topic. As publication volume rises, the volume of material that must be assessed places a greater strain on the workflows used in evidence synthesis (Drozdz & Ladomery, 2024). Maintaining the current standards of systematic reviewing requires new approaches to reduce manual effort without compromising quality.

Advances in artificial intelligence (AI) and machine learning offer ways to reduce the labor required during a screening phase (Cohen, Hersh, Peterson, & Yen, 2006; Marshall & Wallace, 2019). Active learning is one branch of machine learning that has been shown to lower the number of records that must be manually screened (van de Schoot et al., 2021). By updating the model with user-provided labels, active learning can prioritize informative records and shorten the screening process, as relevant

records are found earlier. A benefit of this technique is the continued presence of the *human-in-the-loop*. The active learning cycle proceeds as follows:

- the model selects the most informative or highest-ranked record
- the user provides a label for the highest-ranked record
- the model re-trains on all labeled data
- a new ranking is produced

…and the cycle continues. Machine learning supports the user while leaving final decisions to them. In this way, active learning preserves human control over the decision process and serves as a support tool suited to academic contexts where accountability requirements are high. Although the technique has matured over the past decade, it remains an active research area, and its application to systematic reviewing continues to develop.

This dissertation is an Applied Data Science contribution that evaluates the performance of machine learning, through active learning, in accelerating the screening phase. The evaluation of performance can proceed without the user, as the screening cycle can be imitated with prelabeled datasets. Labels are incrementally provided to the active learning system to imitate user behavior during each active learning cycle. This approach allows for the simulation of performance in minutes rather than the lengthy periods associated with user-supplied label screening. Through this simulation-based setup, it becomes possible to examine techniques, models, or model hyperparameters to improve performance.

Still, simulation is only one element of an Applied Data Science study. Several other components must be in place before simulations can be meaningfully executed. This dissertation documents the complete execution of such a study, including data and environment exploration, questions of reproducibility, software development, and the implementation of simulation (support) software. The outcomes of the simulations then require evaluation, understanding, and explanation.

This dissertation aims to strengthen the application of active learning for accelerating the screening phase of systematic reviews. The goal is to make the screening process less labor-intensive for researchers conducting systematic reviews while remaining transparent and reproducible. All data, code, and workflows are openly available to support replication and further research[1].

[1]This data can be found as replication and simulation data in the List of Publications.

Chapter 2

Data Science in Academia and Industry

As written by Irizarry (2020), "The term data science was originally coined in academia, but the tech industry has mostly driven the proliferation of its use. [...] The increase in the popularity of the term coincides, not with these early attempts at defining a new term, but with the publication by *Harvard Business Review* of *'Data Scientist: The Sexiest Job of the 21st Century'*." Early references described data science as an extension of statistics and computing (Cleveland, 2001; Naur, 1974). Its rapid rise to prominence came with the growth of large-scale data collection and processing in commercial contexts, where data becomes a strategic resource. The industrial workspace gave data science its current form.

In industry, data science projects are often primarily driven by financial incentives or operational efficiency. Success is frequently measured by commercial value. Work is organized around rapid iteration and measurable outcomes, and evaluation prioritizes system functioning in production over theoretical novelty. Intellectual property is protected through technological advantage, often referred to as a technical moat, by leveraging proprietary data, private infrastructure, and closed development processes.

While these conditions limit transparency and the dissemination of knowledge, they accelerate technological progress (Ahmed, Das, Martin, & Banerjee, 2024). Industrial research environments have produced the majority of the defining breakthroughs in modern artificial intelligence, particularly in generative AI and large-scale machine learning[1]. From Word2Vec to BERT, and from GPT-3 to Gemini 3 Pro, nearly every cornerstone of modern natural language processing (NLP), machine learning (ML), and large language model (LLM) development came from industry research groups and primarily Google, OpenAI, and Microsoft (Gibney, 2024). Access to vast private (user)datasets, advanced hardware, and substantial funding enables advances that could not have been achieved within academia. Financial incentives have been essential for technological progress in this field.

Academic data science operates under a different incentive. Its primary output goal is usually the generation of generalized knowledge rather than immediate financial

[1] Most notably, the Transformer architecture was invented by Google Brain researchers in 2017, and explained in the scientific paper *Attention is All You Need* by Vaswani et al. (2017). Google holds a patent on the architecture but has generally permitted its wide use, which has been foundational for modern large language models.

gain. This leads to different strengths, such as a focus on theoretical understanding through methodological precision, leading to verifiable results. Academic research seeks to explain why models behave as they do, not only whether they perform well under commercial goals.

This focus on understanding supports research perspectives that may be de-prioritized under purely commercial incentives, such as studies on algorithmic bias, explainable AI, fairness across gender and culture, or the social and ethical consequences of automation (Hagendorff & Meding, 2021), as little financial gain is to be found in these topics. While industrial labs might invest in these areas to mitigate risk and ensure regulatory compliance, academia provides a venue for foundational research and exploration free from the conflicts of interest inherent in commercial product development.

2.1 Applied Data Science

Applied Data Science (ADS) focuses on using data science tools and techniques to solve real-world, domain-specific problems. In the formulation of Christiansen et al. (2022), ADS uses methods developed in Pure Data Science (Research Data Science) and adjusts them to the requirements of specific practical application contexts. It involves identifying where similar approaches recur across disciplines and determining how they can be adapted to domain constraints.

Leung, Pasi, and Wang (2023) note a similar divide between theoretical and practical data science. In contrast to Research Data Science, which centers on developing new analytical methods and theoretical foundations, ADS is driven by domain problems. Solving domain problems often requires research in its own right, even when the solution does not involve methodological novelty. The distinction between Research Data Science and Applied Data Science lies in their orientation: one advances new methods, the other advances the use and adaptation of existing methods to achieve domain goals. This focus is frequently formalized as Translational Data Science (TDS), which explicitly prioritizes the iterative process of bridging the gap between theoretical models and actionable, deployable solutions for end-users (Spruit, 2021).

2.1.1 Academic Contribution

The applied focus of ADS presents a challenge for both ADS educational programs and research projects within academic settings. Where lies the academic value of applied work? In response to this, Ruijer, Hassink, and Brinkhuis (n.d.) distinguishes between operational skills, which are the technical capabilities required to execute tasks, and reflective skills, which encompass critical thinking regarding ethical implications, legal boundaries, and methodological validity. While financial incentives mostly prioritize operational execution, the contribution of academic ADS is to enforce these reflective skills (Ruijer et al., n.d.). This ensures that technical proficiency is not used as a "black box," but is supported by an understanding of why a method works or fails. This dissertation follows this approach: it uses operational tools to generate empirical evidence while providing the reflection required for an academic contribution.

Reflective research on ADS as a topic in itself is limited in the academic literature (Dav-

enport & Malone, 2021). As ADS often takes place in other domains, much ADS work is published in domain-specific journals, which contributes to its low visibility as a distinct research area. Davenport and Malone (2021) note that central practical tasks such as deployment have long been under-emphasized in academic work, even though they are vital for successful data science projects.

There are signs that this is gradually changing: Scholarly attention to ADS practice is growing through reflective studies of workflows, reproducibility (Samuel, Löffler, & König-Ries, 2020), and MLOps (Kreuzberger, Kühl, & Hirschl, 2022). A rise in master's programs in ADS has been seen over the past decade (Table 2.1, Appendix A). Dedicated outlets for application-focused work are emerging: the *Journal of Applied Data Science* publishes reflective research on ADS and was accepted into Scopus in 2023 (Management of JADS, 2023). Major conferences now include explicit tracks for applied studies. The ADS track at KDD (KDD25 ADS chairs, 2025), for example, presents contributions based on unique real-world problem-solving and domain impact[2]. These developments show growing recognition that understanding how data science is carried out in practice is itself of scholarly value.

Institution	**Program**	**First graduates**
University of Gothenburg	MSc Applied Data Science	2019
University of Southern California	MS Applied Data Science	2020
San José State University	MS Applied Data Intelligence / Applied Data Science	2020
Utrecht University	MSc Applied Data Science	2021
University of Michigan	Master of Applied Data Science (MADS)	2021
Malmö University	MSc Computer Science: Applied Data Science	2023
HAN University of Applied Sciences	Master Applied Data Science	2025
University of Johannesburg	Master of Applied Data Science (CW)	2029

Table 2.1 Non-exhaustive list of examples of Applied Data Science master's programs and their first graduating cohorts.

2.1.2 Growing Influence of Industry

These developments are parallel with an increase in industry involvement in ADS research. Ahmed et al. (2024) report that papers with at least one industry co-author increased from 22% at leading AI conferences in 2000 to 38% in 2020, and the industry's share in large language model development rose from 11% in 2010 to 96% in 2021 (Owens, 2024). Additionally, the proportion of North American data science

[2]KDD is one of the leading international conferences on data mining and data science research.

PhDs entering industry after graduation increased from 21% in 2004 to nearly 70% by 2020 (Jurowetzki, Hain, Mateos-Garcia, & Stathoulopoulos, 2021).

This gradual shift can cause friction within data science communities, as shown by the controversy over ACL[3] 2025's decision to feature only industry keynote speakers. Critics argued that this choice reflected the growing dominance of corporate interests in a venue traditionally centered on academic research.

This friction extends beyond research venues into the educational programs preparing the next generation of data scientists. Ruijer et al. (n.d.) illustrates how the pressure for immediate professional applicability (and thus, operational skill) can erode the reflective components of a curriculum. 'In their analysis of a data science trainee program, they observed that participant feedback "overwhelmingly" prioritized operational skills training, with a "unanimous call" for technical execution and no explicit requests for reflective skills.' 'Although earlier designs of the program emphasized reflection, the co-creation process with practitioners gradually shifted the focus almost entirely toward operational skills.' This dynamic shows the increasing friction academia finds when maintaining its unique value: enforcing critical reflection in a field where the workforce is increasingly oriented toward operational efficiency, as demanded by industry.

2.2 Themes

The following section introduces four themes that can be found throughout this dissertation and represent a reflection on ADS as required in an academic setting. Each theme reflects on a strength or a weakness that emerges when academic and industrial approaches are compared. In some cases, the theme captures an element that academia excels at, but industry often overlooks. In other cases, it's the opposite.

2.2.1 Human-Centered Design

Human-centered design (HCD) provides a way to ground technical development in the contexts of the people who will use its results. ISO 9241-210 gives HCD as an approach that aims to make systems usable and useful by focusing on users' requirements and applying human-factors knowledge throughout development (ISO, 2010). Aragon, Guha, Kogan, Muller, and Neff (2022) proposes that data science gains value only when it serves people's decision-making, interpretation, and practice. Baumer (2017) argues that models and analyses are incomplete without attention to how people act upon results.

In industrial settings, HCD is often driven by strict market necessity. The commercial landscape is littered with products that failed precisely because they neglected user requirements or operational context. Consequently, successful industrial projects are compelled to prioritize utility and real-world viability. By contrast, Research Data Science is frequently driven by methodological novelty. This has value, but when used as the basis for ADS, it can lead to work that prioritizes technical innovation over immediate practical applicability. ADS, when carried out in academia, must balance

[3]ACL is the flagship conference of the Association for Computational Linguistics, a leading scientific society. 2025.aclweb.org/program/keynotes/

theoretical contribution with practical relevance to make up for the absence of the kind of natural HCD that exists in industrial environments.

This dissertation is an ADS contribution conducted in an academic context. While it engages with academic research goals, it adopts principles drawn from HCD in industry. In doing so, it connects academic data science with the practicality expected in ADS.

2.2.2 Software Usability

ISO 9241-11:2018 defines usability as *"the extent to which a product can be used by specified users to achieve specified goals with effectiveness, efficiency, and satisfaction in a specified context of use"* (ISO, 2018). In the context of software, this means a tool is considered highly usable if it meets the following criteria for its intended users:

- Effectiveness: Users can accomplish their intended tasks with the software.
- Efficiency: Tasks can be completed with minimal effort.
- Satisfaction: The experience of using the software is positive.

The development and distribution of domain-specific software forms a substantial part of ADS. In industry, software usability is often a defining characteristic of market-leading products. Financial incentives create a competitive environment where tools with poor user experience struggle to retain a user base, while those that prioritize usability thrive. Consequently, successful industrial software sets a high standard for ease of use. By contrast, the incentives for usability in academic software are lower. Academic software is frequently developed as a one-off proof-of-concept to accompany a publication, with far less emphasis on long-term usability or maintenance (Gregor et al., 2025). As a result, research code might lack polish (e.g., minimal documentation or clunky installation), limiting its reuse and causing many projects to fade away once the initial results are reported. Guidelines on effective coding practices are available, but they often presume a background in computer science (McIntosh, Kamei, Adams, & Hassan, 2016; Sadowski, Söderberg, Church, Sipko, & Bacchelli, 2018; Streiber et al., 2025).

In ADS, usable software translates to clear documentation, straightforward setup, and usable interfaces or APIs. Hunter-Zinck, De Siqueira, Vásquez, Barnes, and Martinez (2021) state that *"creating functional, usable, and maintainable software is increasingly essential to open-source scientific research"*. Using research software should be easy, making it simple for others to understand the methodology and reproduce results.

When these qualities are achieved, the software serves as a medium of research communication. Well-designed and well-documented tools are more likely to be reused by other researchers, leading to greater visibility and impact for the work. Studies show sharing usable code and data openly enhances impact, since the methodology is more likely to be adopted by others (Maitner et al., 2024; Streiber et al., 2025).

2.2.3 Reproducibility and Evidence

Data science emerged from mathematics and statistics, where proof meant reasoning that could be written down and checked. Early machine learning models were small enough that researchers could analyze their behavior through statistical learning theory (Hastie, Tibshirani, Friedman, et al., 2009). As models evolved and the complexity of algorithms and datasets grew (with higher dimensionality, nonconvex optimization, massive, heterogeneous datasets), this style of proof became harder to attain. The conditions needed for theory to work rarely exist in real-world machine learning.

Several authors note this shift toward a more computational and empirical style of science. Progress in machine learning is increasingly driven by large-scale computation, while theory often follows after empirical success (Andrews, 2025; Montévil, 2021). Relying on empirical success without theoretical grounding has been likened to "alchemy." Rahimi and Recht (Rahimi & Recht, 2017) argue that while alchemical trial-and-error can yield impressive results, it fails to provide the rigorous, verifiable knowledge necessary to safely deploy systems in critical domains like healthcare. Under these conditions, formal proof gives way to evidence derived from experiments: a model is accepted when it is shown to work.

This shift makes empirical performance the dominant proof standard. Benchmark results, comparisons to baselines, and ablation studies[4] replace theoretical claims. Yet methodologists warn that this style of research is fragile. Many studies present themselves as confirmatory even though their conclusions depend heavily on design choices (Herrmann et al., 2024). Proof in applied machine learning often only means that a system behaves well on the data presented.

These problems are amplified at the scale of current frontier systems. Current estimates of the cost of training models comparable to recent milestones reach tens of millions of dollars (June 2024), and are increasing at a rate of 2.4× per year (Figure 2.1 (Cottier, Rahman, Fattorini, Maslej, & Owen, 2024). These costs make validation increasingly inaccessible to academic researchers, and the financial case for verification of research diminishes rapidly. The result is that proof increasingly depends on claims from a small number of well-funded groups, even when partial code release is available (Semmelrock et al., 2025).

But computational cost is not the only reproducibility issue. Closed-source machine learning is infiltrating more than just machine learning, and its impact now extends beyond the field. Academic work increasingly uses closed-source models such as ChatGPT, Gemini, or Claude (Carammia, Iacus, & Porro, 2024; Laskar et al., 2023; Yan et al., 2023). These models are offered as remote services whose training data, parameters, and often even basic architectural details are proprietary trade secrets and thus often partially or fully publicly unavailable (Pei, 2025). Furthermore, these services are subject to silent updates, meaning a model may behave differently from one day to the next without version control. This purposeful opacity and volatility are in direct conflict with the fundamental academic principles. When empirical

[4]An ablation study aims to determine the contribution of a component to an AI system by removing the component, and then analyzing the resultant performance of the system (Sheikholeslami, 2019).

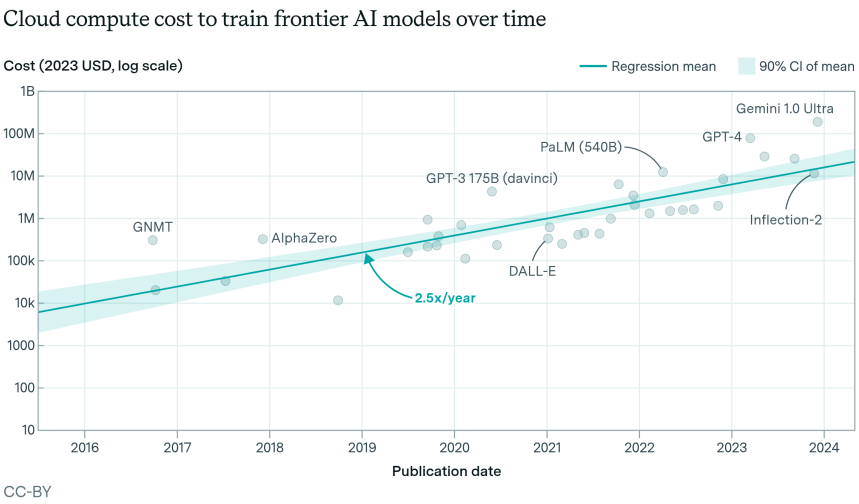


Figure 2.1 Estimated cloud compute costs for the final training run of frontier models.
Source: Based on *"Cloud compute cost to train frontier AI models over time"* by Ben Cottier (Epoch AI). Licensed under CC-BY 4.0. Adapted (Epoch AI logo removed). `https://epoch.ai/blog/how-much-does-it-cost-to-train-frontier-ai-models`

studies use outputs from such systems, other researchers cannot inspect the model or archive the exact version that was used. If the provider updates or withdraws the service, replication becomes impossible. As closed AI services diffuse into many parts of science, their opacity magnifies the reproducibility problems (Gibney, 2022; Kapoor & Narayanan, 2023).

Taken together, this trajectory shows a dispersal of proof. In the early stages, evidence lived inside the publication, as formal reasoning that could be checked line by line. As empirical performance became the standard, proof migrated outside the paper and into data, code, and computational results. With the rise of current frontier systems, it has shifted again, now concentrated in environments that only a handful of well-funded teams can access or rerun.

This raises a practical question for academic ADS. Our work is expected to produce functioning systems, yet it must also yield insight appropriate to academic standards. Proof cannot rely on theoretical guarantees anymore, nor can it rest entirely on empirical demonstrations that are unlikely to be reproduced, as the cost of verification is prohibitive.

2.2.4 FAIR Data

The FAIR Guiding Principles describe a framework for sharing and organizing research data. FAIR is an acronym for Findable, Accessible, Interoperable, and Reusable, and was originally developed as guidance for scientific data management (Mons et al.,

2017; Wilkinson et al., 2016). Although the acronym refers to data, the guidelines have since been extended to other research artifacts, such as software, workflows, and trained machine learning models (Lamprecht et al., 2020). For ADS, FAIR provides a guideline for thinking about the afterlife of datasets and tools.

The FAIR guidelines aim to increase the usefulness and reach of research output beyond its original project (Mons et al., 2017; Wilkinson et al., 2016), for both humans and machines. In ADS, the potential impact of FAIR is particularly high: ADS operates on digital artifacts. Reuse is not metaphorical but literal, in the form of datasets, models, and workflows. When these resources are described with persistent identifiers and rich, standardized metadata formats, they can be incorporated more easily into other projects (Jacobsen et al., 2020).

For data to be FAIR, it first needs to be shared. Regrettably, the practice of sharing in data science often falls short of the ideal. Studies show that many papers do not provide the artifacts needed to reuse or extend the work. Gundersen, Gil, and Aha (2018) report that "only about a third of the papers share the test data set [...] and only 8% of the papers share the source code of the AI method". Findings presented in chapter 3 show that this is similar for the domain of this dissertation.

Tenopir et al. (2020) surveyed thousands of scientists and found very positive attitudes toward sharing; over 85% were willing to share data and use others' data. However, the actual sharing rates are much lower. As they note, "attitudes towards data sharing [...] are mostly positive, [yet] practice does not always support data storage, sharing, and future reuse" (Tenopir et al., 2020). Multiple studies have identified common reasons for this shortfall, and a frequent theme is lack of time or resources: in one large survey, 60% of respondents cited lack of time and 40% lack of budget as a barrier to sharing data. In the same survey, 65% cited lack of time and 51% cited publishing pressure as a barrier to properly document code (Gelsleichter et al., 2025). Other reported obstacles include uncertainty about the right repositories or licenses to use, and worries that data "aren't clean," or that errors might be exposed. Gomes et al. (2022) analyze biological sciences researchers' attitudes and, like Tenopir et al. (2020), note a widespread recognition of openness. They note knowledge barriers and career concerns as limiting factors to sharing in practice.

These studies show that FAIR practices are widely supported in principle, and yet their implementation is comparatively sparse. Increasing the adoption of FAIR guidelines and expanding the availability of FAIR data is a difficult task, as the barriers are numerous and varied. Still, this dissertation aims to follow FAIR practices as far as possible. Code, data, and results are openly accessible with appropriate metadata.

2.3 Main Contributions

The themes introduced in section 2.2 emphasize that machine learning tools must align with users' cognitive processes, decision-making needs, and practical workflows. In the context of systematic reviews, these principles motivate not only the need for accurate automation but also the need for transparency, interpretability, and support for human judgment. Building on these themes, this section introduces the central Research Questions (RQs) guiding this dissertation:

- **RQ1:** How can active learning models be evaluated reproducibly across diverse systematic review datasets?
- **RQ2:** How can software tools and workflows be designed to support transparent and scalable experimentation?
- **RQ3:** How do different model configurations behave in various phases of the screening process?
- **RQ4:** How can we interpret active learning models in ways that align with human-centered design principles?

These RQs connect to five core contributions of the thesis, each corresponding to a major stage in the ADS workflow. Each contribution develops methods or insights that address aspects of human-centered design, whether by improving the clarity of model behavior, enhancing usability, or strengthening the collaboration between human reviewers and machine-learning systems. An overview of the RQs, core contributions, and their corresponding chapters can be found in Figure 2.2.

- **Part I** provides the research context and includes exploratory studies. These chapters investigate the current state of simulation-based evaluation, addressing **RQ1**.
- **Part II** introduces datasets and tools that support reproducible and scalable research. These contributions establish the transparent workflows and data standards required to answer **RQ2**.
- **Part III** focuses on modeling and evaluation, presenting empirical evidence from large-scale simulations. This study analyzes model behavior across different screening phases, responding to **RQ3**.
- **Part IV** addresses deployment, including practical guidance for users and real-world implementation and validation. This part presents the main applied contribution of this work.
- **Part V** presents methods for understanding and interpreting the results. It addresses the interpretability challenges raised in **RQ4**. Finally, it reflects on the implications of this work.

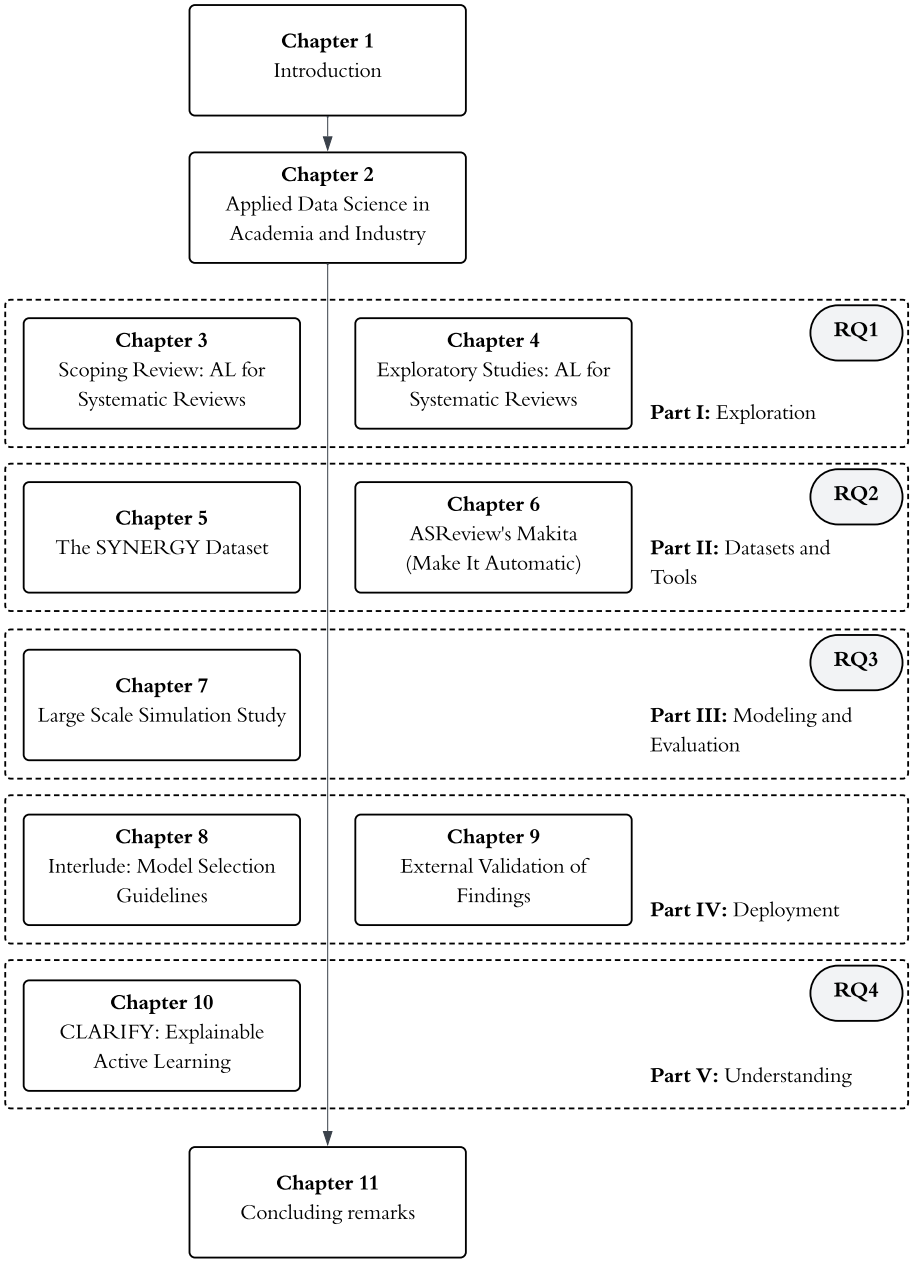


Figure 2.2 Overview

2.3.1 Part I - Research Context and Data Understanding

We present a scoping review of simulation-based active learning in systematic reviews, identifying methodological trends and open challenges in the field. Then, we present four exploratory studies that test the behavior and performance of active learning models under different conditions.

- Chapter 3 - We conduct the first scoping review on the use of simulations to evaluate active learning for accelerating the screening phase of systematic reviews. The review places focus on simulation design, dataset usage, and the methodological gaps that remain in the field. We find that active learning is endorsed across the reviewed literature. In addition, we find a need for the standardization of evaluation metrics for better alignment with the practical needs of reviewers, a need for better open data practices, and a need for a more diverse selection of used machine learning models.
- Chapter 4 - We conduct a set of four exploratory studies, investigating the current performance and usage of active learning under different model conditions. These studies examine the stability of active learning, the effects of deep learning classifiers, the performance of different model configurations, and the potential of model-switching strategies. The results show performance differences between lightweight and heavyweight models across different stages of the screening process, requiring further research to map this new complex behavior.

2.3.2 Part II - Data Preparation

This part has two practical research outputs: the SYNERGY dataset, which provides a curated benchmark for systematic review screening, and Makita, an open-source workflow generator for large-scale simulations.

- Chapter 5 - We introduce a new gold standard dataset for evaluation, the `SYNERGY` dataset, consisting of 26 high-quality systematic review datasets. These originate from manually conducted systematic reviews and are curated to ensure completeness, accuracy, and consistency. The dataset enables the evaluation of a wide range of technological applications for systematic review screening and is released in accordance with the FAIR data principles (Findability, Accessibility, Interoperability, and Reusability).
- Chapter 6 - We release ASReview's Makita (MAKe IT Automatic), a workflow generator for simulation studies. This open-source tool supports the execution of large-scale experiments in a user-friendly way. Makita enhances usability, transparency, and reproducibility in systematic review research. The software is actively maintained and provides clear documentation.

2.3.3 Part III - Modeling and Evaluation

We present a large-scale simulation study comparing model configurations in more than 29 thousand simulations. The results form the empirical foundation for the methodological recommendations presented later in the dissertation.

- Chapter 7 - We conduct a simulation study evaluating a wide range of models in over 29 thousand simulation runs. All experiments are performed using the SYNERGY dataset in combination with ASReview's Makita framework. To the best of our knowledge, this represents the largest simulation of its kind at the time of writing.

  The study is structured in two parts. The first part addresses the reliability of simulation studies, testing both inter-dataset and intra-dataset variability. The second part evaluates the performance of 92 model configurations, introducing the Normalized Recall Regret metric as a loss measure for comparison across models.

  Performance is analyzed at multiple levels: overall effectiveness, early-stage screening efficiency, and the ability to identify the final relevant record. These analyses directly respond to the open questions raised in chapter 4 regarding model behavior across the different phases of systematic reviewing. The study concludes with actionable recommendations for both researchers designing future evaluation studies and practitioners deploying active learning in systematic reviews.

2.3.4 Part IV - Deployment

To aid with the deployment of results found in part III, we offer users explanations and guidance on choosing the appropriate model. This takes the form of a blog and is included in this work as an interlude to the academic chapters. Then we apply the findings to a real-world context in collaboration with an industry partner. It evaluates current model parameters and supports the integration of active learning into systematic review workflows.

- Chapter 8 - We contribute to accessible guidelines for model selection in ASReview. The Model Selection Guide provides researchers with an overview of available feature extractors and classifiers, including their strengths and limitations, suitable use cases, and inner workings. The guide supports users in making informed decisions when setting up their systematic review, increasing the accountability of their work.
- Chapter 9 - We conduct a validation study of the current machine learning hyperparameters used in ASReview 2.0. Using datasets provided by the European Alliance of Associations for Rheumatology (EULAR), the study explores performance metrics associated with achieving a recall of 95% in at least 95% of cases. It examines combined metrics that indicate when this sensitivity target is reached. This report serves as the main evidence base for a large-scale initiative to modernize systematic review practices in a major European health organization.

2.3.5 Part V - Understanding

We introduce CLARIFY, a method for interpreting machine learning models in ASReview by identifying human-understandable concepts in their decision process. Finally, we conclude all studies and reflect on the broader role of Applied Data Science within academic research.

- Chapter 10 - We propose CLARIFY, a method for interpreting any machine learning model in ASReview by deriving concepts from calculated text embeddings. Following the positive evaluation of deep learning models in the active learning workflow, this post-hoc explainable AI (XAI) approach provides insight into the subcomponents of the decision process, addressing the reduced transparency that accompanies these models.
- Chapter 11 - The concluding remarks of the dissertation. It reflects on the position of Applied Data Science in an academic setting and revisits the themes introduced at the start of the thesis. The chapter discusses how the work approaches human-centered design, software usability, reproducibility, and FAIR data, and closes with an outlook on the future of ADS as an academic field.

Table 2.2 Overview of research contributions, methods, and publication status per chapter.

Chapter	Title	Methods	Pub. Status (April 29, 2026)
3	Simulation-based Active Learning for Systematic Reviews: A Scoping Review of Literature	Scoping Review of Literature	Published in *Journal of Information Science* (2025)
4	Active learning-based Systematic reviewing using switching classification models: the case of the onset, maintenance, and relapse of depressive disorders	Simulation Study, Case Study, Software Contribution	Published in *Frontiers in Research Metrics and Analytics* (2023)
5	SYNERGY-Open machine learning dataset on study selection in systematic reviews	Data Contribution	Data published on *DataverseNL* (2025)
6	Makita - A workflow generator for large-scale and reproducible simulation studies mimicking text labeling	Software Contribution	Published in *Software Impacts* (2024)
7	Large-Scale Simulation Study of Active Learning Models for Systematic Reviews	Simulation Study	Published in *International Journal of Data Science and Analytics* (2025)
8	Navigating the Maze of Models in ASReview	Research Communication	Preprint on *Zenodo* (2025)
9	External validation of machine learning hyperparameters for systematic review screening prioritization	Simulation Study	Technical Report (Completed) / Industry Guideline (In Preparation)
10	CLARIFY: Concept Level Active-Learning Ranker Interpreter for Systematic Reviews	Software Contribution	Conference Paper: BNAIC/BeNeLearn 2025

Part I

Research Context and Data Understanding

Chapter 3

Simulation-based Active Learning for Systematic Reviews: A Scoping Review of Literature

This chapter represents a scoping review accepted for publication in the *Journal of Information Science*. The text is a reproduction of the accepted work and was adapted solely for formatting and cross-referencing within this dissertation. No changes were made to the scientific content.

Teijema, J. J., Ribeiro, G., Seuren, S., Anadria, D., Bagheri, A., & van de Schoot, R. (2025). Simulation-based active learning for systematic reviews: A scoping review of literature. *Journal of Information Science*

Background: Active learning is a proposed method for accelerating the screening phase of systematic reviews. While extensively studied, evidence remains scattered across a fragmented body of literature.

Objective: This scoping review investigates whether active learning is recommended for systematic review screening and identifies areas needing further research.

Design: We screened 1887 records published since 2006 using ASReview, an active learning tool, and included 60 relevant studies. We also analyzed 238 of the 336 collected datasets for study design, dataset usage, and implementation.

Results: All 60 studies recommended active learning as a means to improve screening efficiency. Despite some methodological heterogeneity, consistent endorsement was found across the literature.

Conclusions: Active learning shows strong potential to support systematic review screening. Standardizing evaluation metrics, encouraging open data practices, and diversifying model configurations are key priorities for advancing this field.

3.1 Introduction

This paper investigates the use of active learning in the context of systematic reviews. While systematic reviews are the gold standard for evidence synthesis, they can be characterized by considerable time and labor demands. Active learning is a machine learning approach that has been widely tested in this context, yet results are fragmented across many studies. To bring clarity, we conduct a scoping review of simulation studies on active learning for systematic review screening, with the aim of mapping existing evidence, identifying gaps, and guiding future research.

The systematic review process involves manually searching databases for relevant literature, screening each record for relevance, and analyzing the resulting collection of research. Despite these demands, the systematic review is a widely used and trusted form of research because of its methodological approach, complete data collection, and critical appraisal of available evidence. The need for workload reduction (Marshall & Wallace, 2019) has motivated a new field of meta-systematic review research (Tsafnat et al., 2014), where research is aimed at increasing the efficiency with which systematic reviews can be performed to free up valuable time and resources. The traditional systematic review consists of many steps (Higgins et al., 2019), and overviews of the entire systematic review software environment exist, e.g. Jimenez et al. (2022); Van Dinter, Tekinerdogan, and Catal (2021).

Among all stages of a systematic review, the screening phase is arguably the most time-consuming and labor-intensive. Active learning (Settles, 2009), as we show in this paper, is especially appropriate in design and thus often employed to alleviate the burdensome process in this step. Active learning is already incorporated into numerous software tools (Adam, Wallace, & Trikalinos, 2021; Cowie, Rahmatullah, Hardy, Holub, & Kallmes, 2022; Jimenez et al., 2022; Khalil, Ameen, & Zarnegar, 2022; Mauricio & Gonzalez, n.d.; Pellegrini & Marsili, 2021; Robledo, Aguirre, Hughes, & Eggers, 2021; Scott et al., 2021; Tsou, Treadwell, Erinoff, & Schoelles, 2020; van de Schoot et al., 2021; Wagner, Lukyanenko, & Paré, 2022; L. L. Wang & Lo, 2021)[1], and studies have been done assessing the performance of active learning using simulations. However, study designs, datasets used, and active learning models compared vary widely across studies. The purpose of the current paper is to systematically review the existing literature on active learning applications in systematic reviewing.

Workload reduction in the screening phase is much sought after, but proves to be a complex task to accomplish. The relevance of a paper is often complex and nuanced, as the research question being addressed in the systematic review may be broad or multifaceted, and it can be difficult to determine which studies are directly relevant to the question. Because of this complexity, a human reviewer is needed with a good understanding of the nuances of the research question and the criteria for relevance. Text classification machine learning algorithms are found to be an effective tool for assisting these human reviewers. By being trained on a dataset of human-labeled data and distilling the subtleties present in the data, these algorithms can replicate human nuance. As a result, the utilization of machine learning algorithms is frequently attempted in research aimed at enhancing the efficiency of systematic reviews (Miwa,

[1] Identification of these tools is available in an OSF repository Stoel, Mourits, and van de Schoot (2023)

Thomas, O'Mara-Eves, & Ananiadou, 2014).

Thomas, McNaught, and Ananiadou (2011) identifies sub-processes within the systematic review process and provides commentary on which aspects of the process can be supported by various types of machine learning. The authors note that automatic classification poses a significant challenge for machine learning, but suggest that active learning (Settles, 2009) can be used to address this challenge. Thomas et al. (2011, 2017) propose active learning as an effective method of application of machine learning on text-based systematic reviewing.

Active learning is a machine learning approach in which the model incrementally trains itself during the labeling process. Instead of relying on a large pretrained model or a static training set, it begins with minimal labeled data and continuously updates its predictions as new labels are obtained. The model actively selects the requests with the highest likelihood of being relevant and requests human input. These new labels are immediately used to retrain the model, allowing it to improve in parallel with the screening task.

This setup enables the model to assist the reviewer early in the process, even before a large training set exists. With each new label, the model becomes more accurate in identifying which documents are likely to be relevant, and it prioritizes these for human screening. In this cycle, both the algorithm and the human reviewer improve over time: the model becomes better at selecting useful records, and the reviewer screens more efficiently. The ability to learn and support screening simultaneously makes active learning particularly well-suited for systematic reviews.

It is the nature of systematic reviews that makes the traditional training of machine learning algorithms for text classification of literature challenging. Systematic reviews address novel and evolving questions, and as a result, the labeled data required to train an algorithm to recognize relevance rarely exists. This absence of pre-collected data is, of course, in part, what motivates the conduct of systematic reviews in the first place. Active learning addresses this constraint by initiating learning without requiring a large labeled dataset, making it a suitable method for optimizing the screening process in systematic reviews.

Despite the advantages of incorporating active learning in the application of machine learning algorithms for the support of literature screening and the numerous simulation studies that have been conducted on this topic, an overview of the usage and performance of active learning is currently lacking. As a result, the research in this field is scattered, and there is a lack of consistency in research. From this field, one could not easily answer whether or not active learning is the answer to the challenges systematic reviewing brings.

The field of simulation studies for active learning-based systematic reviewing shows a lack of consistency and uniformity in study design and methodology. This includes variations in study design, dataset usage, the number of datasets utilized, the specific active learning models employed, the performance metrics applied, and more. As a result, it is difficult to draw broad conclusions or make comparisons between studies, hindering both the advancement of the field and the application of active learning in systematic review practice.

In light of the current state of the field, the main objective is to provide an overview of the performance of active learning for use during the screening phase of systematic review acceleration. We collect and present data extracted from simulation studies that deal with the performance and application of active learning in the acceleration of systematic reviews.

From the studies identified in our systematic search, the aim is to identify potential areas for future research and provide recommendations for future studies on active learning in systematic reviews. We extract information on the currently standard study design, dataset utilization and statistics, and machine learning applications in this field. Our study can serve as a reference point for anyone interested in simulating active learning performance and optimizing their systematic reviews or active learning-aided software tools.

In following the outlined goals, this study aims to achieve several objectives. First, it analyzes the study designs of the included simulation studies to assess the scale and methodology employed while collecting data on the availability of machine learning source code. Additionally, it gathers and presents information on the labeled datasets used to test performance, evaluating their accessibility and reproducibility. Finally, the study examines the *use of active learning models, reporting on the evaluations of different applications across the selected studies.* By integrating these elements, the research ultimately addresses the question of whether active learning should be recommended for accelerating systematic reviews.

3.2 Methodology

This review was reported in accordance with PRISMA. The PRISMA flowchart found in Figure 3.1 reports the steps taken during this scoping review, from data gathering to the final selection of included records, and is represented. The information extraction in this study is divided into three different categories: Study design, Datasets, and Models. Therefore, the methodology section of this paper is divided into sections following those categories.

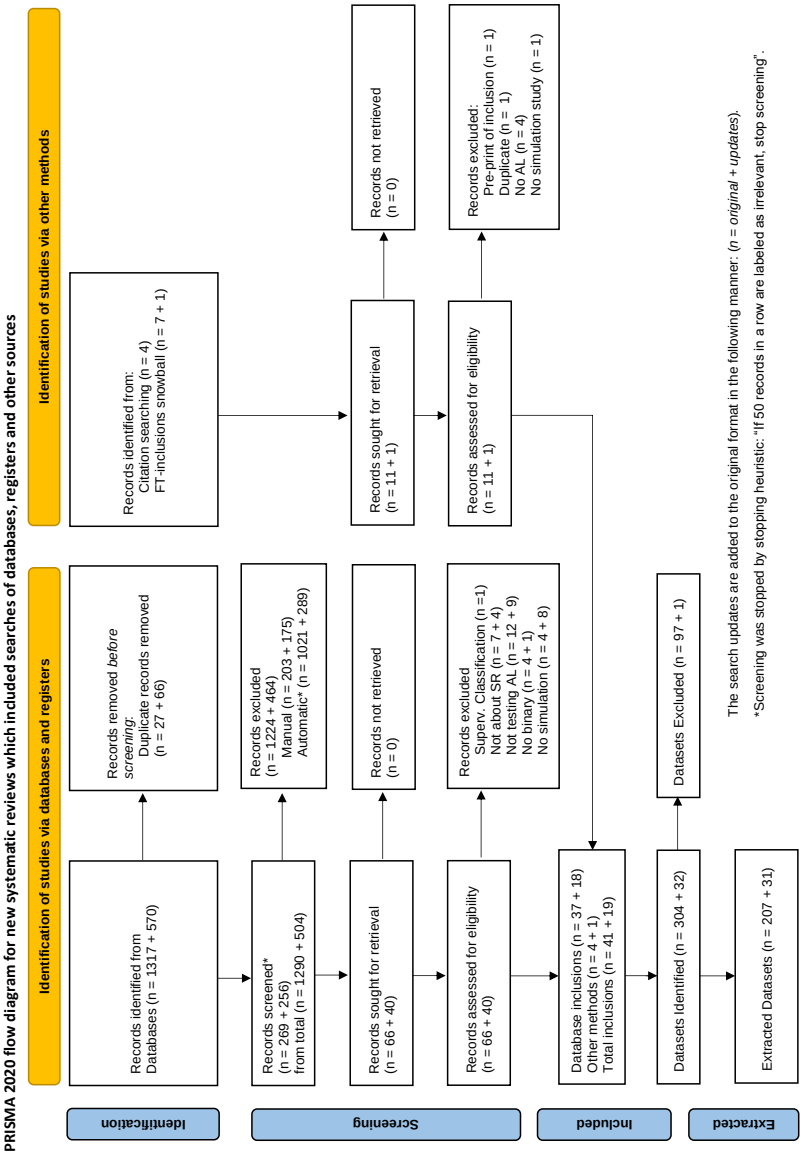


Figure 3.1 PRISMA flowchart representing the workflow for the current study, along with the amount of literature found and discarded in each step.

We searched the following databases for literature containing any variant of the term "systematic review" in combination with the term "active learning", published after 2005: Web of Science, Scopus, and EMBASE. The timeframe of "after 2005" was selected as the starting point for the analysis, as per a review by O'Mara-Eves, Thomas, McNaught, Miwa, and Ananiadou (2015), which identified the first documented application of techniques used for title and abstract screening in literature as occurring in 2006. The results show that this cutoff was enough; the earliest true simulation study found was done in 2010.

The database search used in our systematic review was performed first in September 2021 and repeated in March 2023 and August 2024. The exact search terms, along with other information, are available on the Open Science Framework as J. J. Teijema, van den Brand, Bagheri, and van de Schoot (2023).

The screening of abstracts for potential relevance was done by two screening researchers using active learning, through the use of the open-source software package ASReview, on version 0.19 for the primary screening, versions 1.1.1 and 1.6.2 for the literature updates. The application of active learning has been shown to result in a significant decrease in total screening time, as demonstrated in Ferdinands (2021); Muthu (2022); van de Schoot et al. (2021). This is achieved by prioritizing relevant records at the beginning of the screening process, allowing for partial screening with full results, thus reducing the overall number of records requiring screening.

The algorithm employed for both the original search and the first literature update is an implementation of *TF-IDF*, *Naïve Bayes*, and a *dynamic double resampling* algorithm. The second update changed models based on the simulation results in J. J. Teijema, de Bruin, Bagheri, and van de Schoot (2025), and opted for a retrieval optimized transformer, *mxbai-embed-large-v1*, together with a *Random Forest* classifier.

To initiate the active learning process, seven records known to be relevant and five randomly selected irrelevant records were identified and utilized as prior knowledge for the initialization of the active learning machine learning algorithm. For each literature update, the model is initiated using all relevant records from the previous searches, combined with a random irrelevant record. Specifically, the first update uses records from the initial search, while the second update incorporates records from both the first and second searches.

A stopping heuristic of 50 consecutive irrelevant records was set as the criterion for terminating the screening process and proceeding to the next phase of the systematic review. The consecutive count of irrelevant records was shared between screening researchers, and inclusions and uncertain exclusions were evaluated by both screening researchers.

To be considered relevant, a record must describe a simulation study that tests the performance of machine learning for systematic review screening, with at least one of the simulations utilizing active learning to apply the machine learning algorithm. Additionally, the dataset used in the simulation must be of a scientific nature, excluding materials such as legal documents or emails.

The first search yielded a total of 1290 articles, and the search updates yielded 504. The results from all three databases were combined and deduplicated based on DOI, title, and/or abstract in preparation for the screening phase. Following the stopping heuristic, screening was halted at 269 records out of the total 1290 (20%) for the initial screening phase and 91 of 223 (41%), 128 of 285 (45%) for the second screening phase. The performance of the first screening is depicted in Figure 3.2 and the search updates in Figure 3.3.

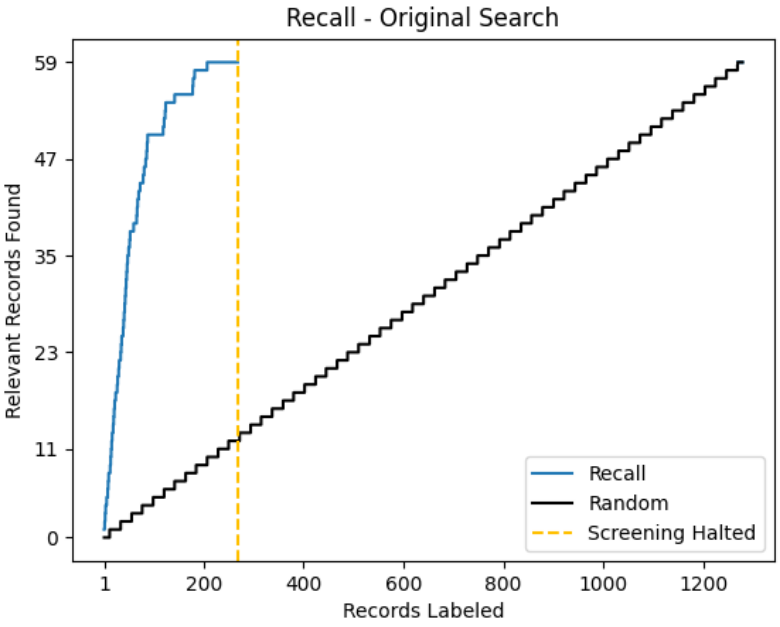
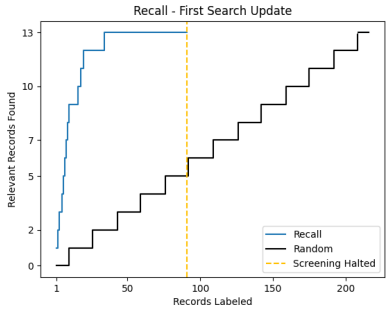
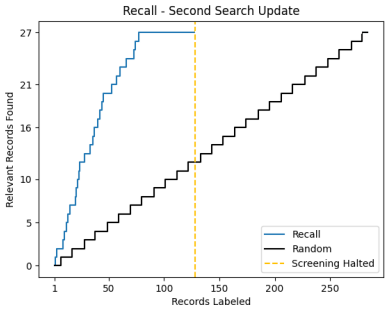


Figure 3.2 In this graph for the screening phase of the current scoping review, the blue line represents the number of relevant papers identified at a certain number of papers screened, a metric known as the recall. The dotted yellow line shows the moment the screening was halted, and the black line represents the hypothetical recall for random screening. The x-axis indicates the total dataset of records.

Combining the searches, of the total 525 records screened, 106 were deemed to be relevant based on their abstracts. These 106 papers were further assessed for eligibility based on their full text. This resulted in a final selection of 55 records deemed to be relevant to the study in accordance with the established inclusion criteria. Additional rounds of screening were conducted through citation searching of the reference lists of the final inclusions, the snowballing, and citation searching rounds. This led to the identification of an additional five records, bringing the total number of relevant records from 55 to 60, the final amount of relevant records.



(a) Recall graph for the first literature update phase.



(b) Recall graph for the second literature update phase.

Figure 3.3 Recall graphs for the first and second literature update phases of this scoping review.

3.2.1 Study design analysis

We search for simulation studies testing the performance of active learning in experiments or simulation studies to accelerate the screening phase in systematic reviews. From these papers, the analysis focuses on the study design of said papers. The papers were carefully reviewed, and the relevant data were organized into three separate tables. These tables can be found as supplementary materials in the persistent storage location[2]: Table 1 (paper and model information), Table 2 (dataset information), and Table 3 (linking datasets to their corresponding papers). From the identified papers, the following information was extracted:

- Record title
- Authors
- Publication year
- Number of datasets used for simulations
- Was the dataset originally prelabeled
- Which metrics are used to quantify the results
- Is the used dataset reported
  - If so, is the dataset accessible
  - If so, where is the dataset stored?
- Is the used code reported
  - If so, is the code accessible
  - If so, where is the code stored?

In assessing the availability of code and datasets for this study, the criterion employed was the feasibility of access through reasonable effort. Instances in which a link was provided but found to be non-functional were documented as unavailability. Similarly,

[2] `https://osf.io/t9hgm/files/osfstorage`, under results.

cases where code or datasets were described without providing an accessible link were also considered to be unavailable. Furthermore, if instructions for accessing code or datasets were provided but these instructions proved to be inoperable within a reasonable effort, such cases were also recorded as unavailable. Finally, if a link was directed to a program that was neither open source nor accessible due to a paywall, the program was deemed unavailable for this study.

The accessibility of each dataset was confirmed through manual verification (i.e., can the dataset be accessed through reasonable effort?).

3.2.2 Dataset analysis

The collection of datasets is not all-inclusive, as certain criteria must be met for a dataset to be considered. The dataset must pertain to a substantive topic (i.e., created for a systematic review, not for use in a simulation study) and be derived from scientific literature, thus excluding datasets such as those containing news articles. In addition, the dataset must either be prelabeled or be labeled during the study before the simulation starts to accurately evaluate the performance of active learning algorithms. This ensures that the datasets are representative of scientific fields and of relevance to systematic review studies.

From the literature, 336 datasets were extracted. From this, 238 datasets were selected to be valid for inclusion based on the previously mentioned inclusion criteria. The following variables were extracted:

- Dataset publication year
- Original author
- Collection author
- Originally prelabeled
- Data type (title, abstract, full-text)
- Original data purpose
- Field & topic
- Number of records in the dataset
- Number of inclusions
- Original dataset storage location

3.2.3 Model analysis

With model analysis, this study aims to reveal the model intricacies and provide a clearer understanding of how each model contributes to the performance of active learning, how often, and in what way the models are applied in the field.

Detailed data about the models utilized in each study were extracted. Specifically, we focused on:

- The type of machine learning model used

- Any customization applied to the model, if available, is referred to as the custom model name
- Are hyperparameter optimization techniques used
- The size of the batch used in active learning cycles

In total, information on 15 distinct models was gathered. Each of these models has integrated active learning techniques in some form, showing a diverse array of approaches used in this field. This analysis of different models will shed light on the design nuances that might influence the successful application of active learning in systematic reviews.

3.3 Results

The results section of this scoping review provides an analysis of the studies identified through the literature search process. To present the findings cohesively, the results are formatted in the same manner as the method section: subsection 3.3.1 Study Designs, subsection 3.3.2 Datasets, and subsection 3.3.3 Active Learning Models and Evaluation. Each category focuses on specific variables relevant to that topic and provides a detailed examination of the key findings and their significance in the context of active learning literature.

The 60 papers labeled as relevant can be described as studies that use active learning in their simulations to test its performance. These are studies that focus specifically on the performance of active learning to improve systematic reviews, often in the form of a case study or report on a systematic review that employed active learning as part of their machine learning implementation. Appendix Table B.1 presents the 60 simulation studies that were selected for inclusion in this scoping review.

3.3.1 Study designs of simulation papers

The first section of the results is about the study design. The information in this section provides an understanding of the scale of the studies included in this review.

The distribution of papers by year of publication can be observed in Figure 3.4, and provides insight into the rising popularity of the field.

Figure 3.5 shows the distribution of the number of datasets used for simulation studies. Most studies use only a single dataset, and the median number of datasets per study is 3.5. Moreover, it was found that most studies train their model on a title-abstract combination.

A remarkable observation can be found in the considerable quantity and diversity of metrics employed in the field. In Table 3.1 and Figure 3.6, only metrics with three or more instances are displayed; however, a total of 61 distinct evaluation metrics were identified to evaluate simulation performance. This substantial variation poses a significant challenge to the cross-comparison of models.

Figure 3.7 presents the distribution of the primary studies incorporated in this review, focusing on their compliance with open science principles. The figure delineates the

proportion of studies that provided access to their code, datasets, both, or neither. It reveals that while most studies report on their dataset and code, less than a quarter of the analyzed studies had both resources available. Factors contributing to the lack of shared code and datasets vary, including government restrictions exemplified in Howard et al. (2020), unintentional omissions of code and data in relevant publications, and persistence issues such as broken links or relocated resources over time. Regardless of the reasons for these missing resources, their absence can impede study reproducibility, verifiability, and overall utility.

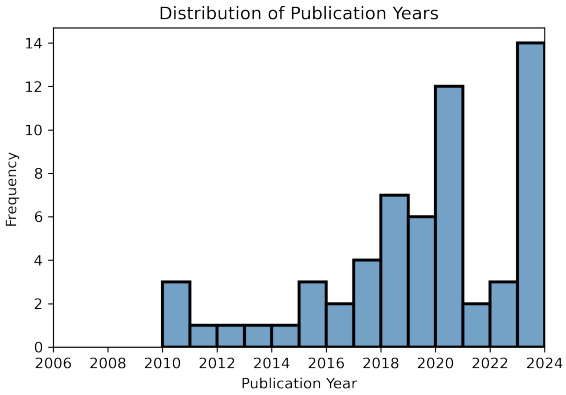


Figure 3.4 Histogram visualizing the number of papers per year. While the search was started in 2006, no papers running simulation studies were found between 2006 and 2010.

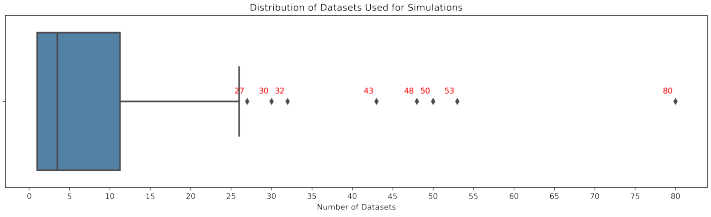


Figure 3.5 Boxplot with the number of datasets used in the included papers.

3.3.2 Datasets and usage

The second set of variables analyzed in this review considers the datasets reported in the studies, with particular attention given to the manner in which they were utilized. This information provides insight into how the datasets were initially intended to be used, the specific fields and topics they belong to, and the characteristics of the text and labeling status of the datasets.

Appendix Table B.2 lists open dataset collections with more than eight datasets for systematic review screening phase automation research, as identified in our searches.

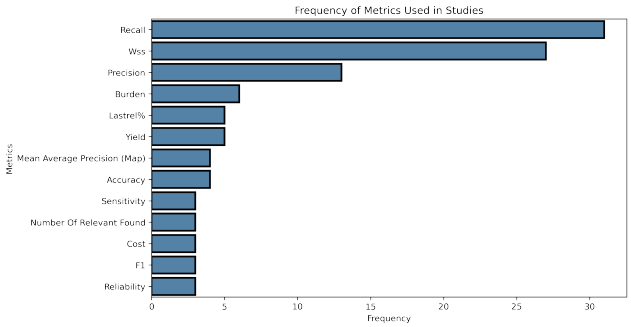


Figure 3.6 Barplot showing the frequency of metrics used in studies.

Metric	Description
Recall/Yield @ X	Number of relevant documents found at X screened, i.e. $TP/(TP+FN)$
WSS (often WSS@X%)	Work saved over sampling at $X\%$ relevant documents found (i.e., work decrease when compared to screening randomly)
Precision	$TP/(TP+FP)$
Burden	Percentage of studies that are manually labelled
Lastrel%	Percentage of candidate documents that need to be screened to get all the relevant documents
Mean Average Precision (map)	Average of precision at each recall position
Reliability	$lossr + losse$, with $lossr = (1-r)^2$, where r is the recall at the threshold, and $losse = (n/(R+100)*100/N)^2$, where n is the number of returned documents by the system up to the threshold, N is the size of the collection, and R the number of relevant documents.
Number of relevant found	Measure of recall at a set threshold
F1	$(2 * recall * precision)/(recall + precision)$
Cost	Extra percentage of documents screened to obtain a given level of estimated recall, using theoretical and actual $WSS@X\%$ values

Table 3.1 Descriptions of the most used metrics in the identified studies.

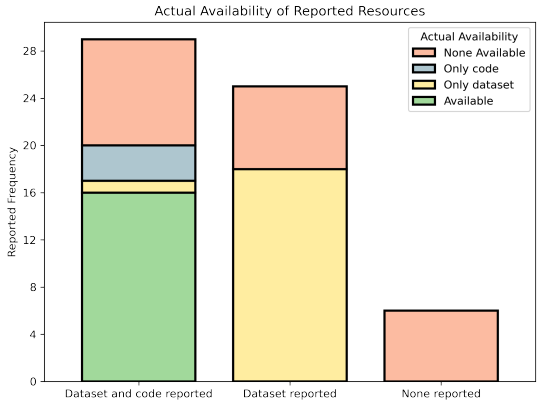


Figure 3.7 Stacked bar chart illustrating the actual availability of reported resources in relation to the frequency of their occurrence. The categories on the x-axis represent the types of resources reported, while the y-axis displays the reported frequency. The differently colored segments of each bar denote the actual availability, highlighting discrepancies between what was reported and the reality of access.

To facilitate organization and identification, a dataset_id and a collection_id are assigned to each record and dataset in the tables.

Figure 3.8 shows the distribution of research fields found in the collected datasets. For some fields, the usage of the systematic review format is more commonplace than in others, leading to these fields being overrepresented in the simulation papers.

Figure 3.9 illustrates the relationship between the number of records and the number of inclusions, excluding outliers for enhanced clarity. This visualization provides insight into the association between these two variables within the context of the analyzed datasets. A subsequent statistical analysis was conducted on the data, revealing no significant correlation between the number of records and the number of inclusions ($p = 0.17$, and thus not significant). This result suggests that, for the collected data, the two variables are independent and do not directly influence one another.

3.3.3 Active Learning Models and Evaluation

The third group of variables in this review presents the active learning models used in the studies and the methods used to evaluate and compare these models. This information provides insight into the active learning models utilized in the studies. Additionally, this group also includes the conclusions of the studies on the effectiveness of active learning.

The analysis revealed that approximately one quarter of the simulation studies incorpo-

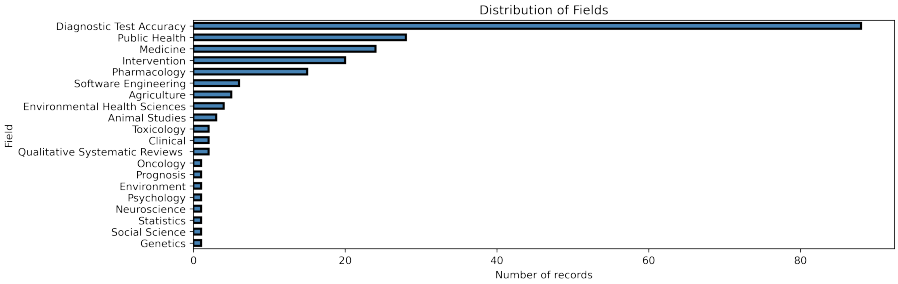


Figure 3.8 Horizontal bar chart depicting the distribution of fields across the datasets under review. The y-axis represents various fields, while the x-axis indicates the number of datasets in each field.

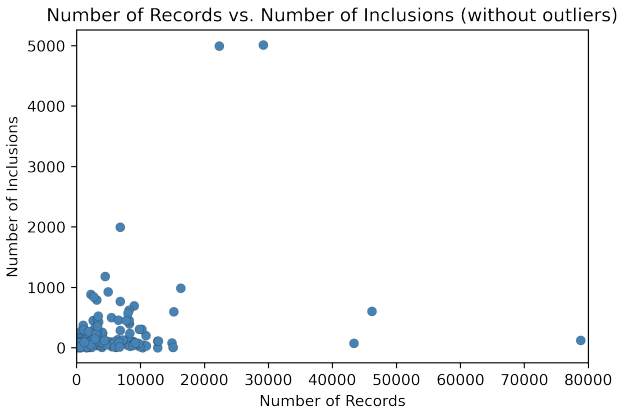


Figure 3.9 Scatterplot depicting the relationship between the number of records and the number of inclusions, with outliers removed for clarity. The x-axis represents the number of records, while the y-axis displays the number of inclusions. The plot highlights the association between these two variables within the context of the analyzed datasets.

rated some form of hyperparameter optimization for their machine-learning algorithms. The other studies either employed models that did not require optimization or relied on standard settings for their algorithms.

The code for the visualizations used in this paper is available as J. Teijema (2023). The datasets are available as J. J. Teijema, van den Brand, et al. (2023).

Finally, each paper was assessed on its evaluation of active learning. It was found that, despite identifying certain limitations, all 60 provided positive endorsements for using active learning to improve the efficiency of the screening phase in systematic reviews.

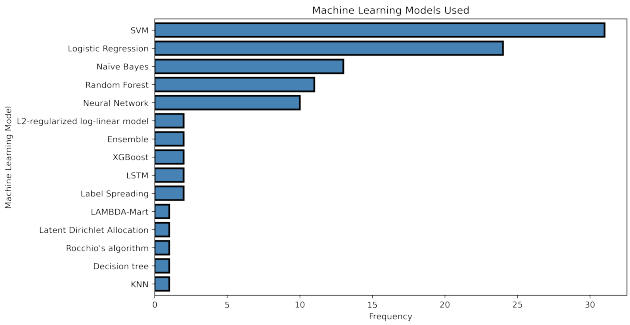


Figure 3.10 Horizontal bar chart showcasing the frequency of machine learning model choices employed in the studies. The y-axis represents various machine learning models, while the x-axis indicates their frequency of usage. This visualization highlights the popularity of different models within the context of the analyzed studies.

3.4 Discussion

Systematic reviews are top-quality research but require a lot of time and work, and the screening phase remains notably laborious and time-consuming. It is in this step that active learning, which has already found application in numerous software tools, can greatly contribute by reducing human effort. However, there's no solid overview of how well active learning performs in this role. We aimed to review existing literature on active learning in systematic reviews to map out the field, present its various approaches, methodologies, and findings, and ultimately determine if active learning is indeed the recommended solution for systematic review acceleration. We believe this work will serve as an important reference point, providing an understanding of the current state of the field and highlighting areas for future research and practical applications.

The review conducted in this study has thoroughly analyzed the current state of active learning-based systematic review screening, taking into account the variations in study design, dataset usage, active learning models employed, and performance metrics applied. Despite the diversity of examined studies, a shared theme emerged: our analysis consistently revealed the belief that active learning is recommended as a solution to the challenges posed by the implementation of machine learning for

systematic reviews. This finding not only supports the broader application of active learning in the field of systematic review practice but also serves as a strong foundation for both researchers in meta-systematic review research and practitioners considering the implementation of active learning in their own systematic review efforts.

The results section of this review offers an analysis of the identified studies, focusing on three key aspects: study design, dataset usage, and active learning models. The study design analysis provided insights into the scale and methodology of the included studies, as well as the distribution of papers by year of publication, number of datasets used, and the considerable diversity of metrics employed. The dataset analysis offered information on how datasets were utilized, their intended purposes, the specific fields they belong to, and the characteristics of the text and labeling status. The active learning models analysis presented the models used in the studies, the methods of evaluation and comparison, and the conclusions on the effectiveness of active learning.

One limitation encountered during this review is the diversity of metrics used in screened literature. This diversity makes cross-comparison of performance and models difficult. In order to effectively evaluate the performance of active learning in systematic reviews, a robust and consistent measurement tool is required. While previous studies have contributed to the development of sophisticated metrics, the absence of a universal standard and the variety of performance measures employed hinder direct comparisons of active learning methods' effectiveness.

O'Mara-Eves et al. (2015) provides an overview of the performance measures definitions used in studies of text mining for systematic reviews. This study presents clear and easily understandable documentation of the performance measures employed in various studies in this field. Their work serves as a valuable resource for researchers seeking to compare and evaluate the effectiveness of different text-mining methods for systematic reviews. However, we found that since then, the diversity of metrics has only increased, with a total of 61 different metrics identified. This further complicates the cross-comparison of active learning methods, posing a challenge for researchers aiming to draw broad conclusions about the field.

Another limitation identified in this review is the lack of dataset availability. Many datasets used in the studies are not open data or open science, which restricts the reproducibility of the research. Reproducibility is a vital aspect of the scientific method, as it allows researchers to validate the findings of a study and build upon the existing body of knowledge. Ensuring the reproducibility of research is essential for the credibility of scientific results. To address this issue, we encourage researchers to open up their datasets, adhering to open science principles, and facilitating the replication of their work by others.

Olorisade, Brereton, and Andras (2017) report that around 80% of the studies they assessed lacked sufficient information regarding dataset usage. Whilst our observations come to around 40% of assessed papers missing dataset information, this is still a significant portion. This lack of open data undermines the reproducibility of research in data science. To counter this, Olorisade et al. provide a framework for ensuring the reproducibility of research in data science, which can help researchers produce reliable and trustworthy results that can be validated and reused by others. By adopting this framework and sharing datasets, researchers can contribute to the advancement

of the scientific method and bolster the credibility of their findings in the active learning-based systematic review screening field.

Another limitation found by this review is the lack of cross-analysis between models. The majority of the reviewed papers employ either Support Vector Machines (SVMs) or Logistic Regression (LR), and most papers only compare against manual work, not against other models. Furthermore, the models are typically tested against unique datasets, making it challenging to compare their performance across different datasets. While the reviewed studies provide useful insights into the application of active learning models for systematic reviews, there is still a need for more extensive and comparative analyses across various models and datasets. Such research could help in identifying the most effective active learning models for systematic reviews and provide more standardized performance evaluation methods.

There are several areas of active learning-based systematic review screening that warrant further exploration. One critical area for future research is the development and standardization of metrics to evaluate active learning methods. With the proliferation of different metrics used in the field, there is a pressing need to identify the most appropriate metrics to use for evaluating active learning models.

Additionally, advocating for the use of open data practices could be beneficial in improving the availability of datasets and promoting collaborative research efforts.

There is a need to explore a wider variety of models to improve the understanding of active learning techniques. While Support Vector Machines (SVMs) and logistic regression models are currently popular choices, exploring a more extensive range of models may lead to improved performance and a better understanding of the strengths and weaknesses of different active learning techniques.

Future work in active learning-based systematic review screening should focus on standardizing metrics, promoting open data practices, and exploring a wider variety of models to improve the efficacy and transparency of research in this field.

3.4.0.1 Recommendations

As practical results of our analysis of 60 simulation studies, we find consistent evidence that active learning improves efficiency in systematic review screening compared with random or manual-only approaches. Across diverse designs and datasets, this conclusion appears repeatedly. From this body of work, three priorities can be formulated as initial steps toward a more standardized model: the inclusion of pre-existing evaluation criteria when introducing new metrics, increasing the availability of open datasets, and broadening the range of models compared. Taken together, these priorities provide a start for both researchers and practitioners, and can be seen as the beginning of a more unified framework for applying active learning in systematic reviewing.

3.4.1 Declarations

3.4.1.1 Data Availability

The datasets generated and analyzed during the current study are available in the Open Science Framework repository at the following URL: `https://osf.io/t9hgm`. The source code for 'Simulation-Based Active Learning for Systematic Reviews' is publicly accessible via our GitHub repository at the following URL: `https://doi.org/10.5281/zenodo.13361795`. These resources provide supplementary data, methods, and materials related to the study.

The OSF repository includes as much information as possible in the shared datasets within our interpretation of copyright restrictions, but this remains a limiting factor. Copyright restrictions constrain the extent to which full record sets can be made openly available, even though recent work has shown that reproducing systematic review datasets is often highly challenging without access to the original records (Neeleman, Leenaars, Oud, Weijdema, & van de Schoot, 2024).

Chapter 4

Active learning-based Systematic reviewing using switching classification models: the case of the onset, maintenance, and relapse of depressive disorders

This chapter is based on a study published in *Frontiers in Research Metrics and Analytics*. Revisions were made to the text to improve reading clarity while the results and core scientific content remain unchanged. Formatting and cross-referencing were adapted for internal consistency.

Teijema, J. J., Hofstee, L., Brouwer, M., de Bruin, J., Ferdinands, G., de Boer, J., Vizan, P., van den Brand, S., Bockting, C., van de Schoot, R., & Bagheri, A. (2023). Active learning-based systematic reviewing using switching classification models: The case of the onset, maintenance, and relapse of depressive disorders. *Frontiers in Research Metrics and Analytics*

Introduction: This study evaluates the performance of active learning-aided systematic reviews using a deep learning-based model compared to traditional machine learning approaches, and examines the potential benefits of model-switching strategies.

Methods: The study consists of four parts: (1) analysis of the performance and stability of active learning-aided systematic review; (2) implementation of a convolutional neural network classifier; (3) comparison of classifier and feature extractor performance; and (4) investigation of the impact of model-switching strategies on review performance.

Results: Lightweight models perform well in the early stages of simulations, whereas more complex models show increased performance in later stages. Model-switching strategies generally improve performance compared to using a single default classification model, in the right contexts.

Discussion: The findings support the use of model-switching strategies in active learning-based systematic review workflows. It is recommended to begin reviews with a lightweight model, such as Naïve Bayes or

logistic regression, and switch to a heavier classification model based on a heuristic rule when appropriate.

4.1 Introduction

Systematic reviews and meta-analyses aim to synthesize evidence within a specific scientific field by providing structured overviews of relevant literature (Moher et al., 2015). Conducting such reviews requires screening hundreds to tens of thousands of records, often to identify only a small number of relevant studies. This results in highly imbalanced datasets, where relevant records are typically sparse (commonly <5%) (chapter 5). An active learning-based systematic review pipeline can help address this challenge (Settles, 2012). In this approach, machine learning models prioritize records most likely to be relevant while continuously improving their ability to identify such records. Prior work has shown that active learning outperforms random screening across a range of feature extraction techniques and classifiers (Chapter 3), each with distinct strengths and limitations (Naseem, Razzak, Khan, & Prasad, 2021).

Human-computer interaction plays a central role in the systematic review process, particularly in computer-aided systems where collaboration between human reviewers and machine learning models is essential for improving efficiency and accuracy. In linguistics, such systems enhance the identification and correction of functional styles in text (Savchenko & Lazebnik, 2022). In science education, they support the measurement of complex scientific reasoning (O. L. Liu, Lee, Hofstetter, & Linn, 2008). In healthcare, they assist in bridging research, policy, and practice (Best et al., 2009). In software development, they improve effectiveness and reduce defects (Tiwana, 2004). In physics-informed symbolic regression, they facilitate the discovery of meaningful symbolic expressions (Keren, Liberzon, & Lazebnik, 2023). Collectively, these applications emphasize the importance of user-friendly design, accurate and timely feedback, and an understanding of the cognitive processes underlying human decision-making when interacting with machine learning models. Building on these insights, the present work investigates an active learning-based pipeline that combines human expertise with machine learning capabilities to enhance the systematic review process.

Human-computer interaction is a crucial component of the systematic review process, particularly in computer-aided systems where collaboration between human reviewers and machine learning models is essential for efficiency and accuracy. Applications of computer-aided systems span a wide range of domains: in linguistics, they improve functional style identification and correction in texts (Savchenko & Lazebnik, 2022); in science education, they support the assessment of complex reasoning (O. L. Liu et al., 2008); in healthcare, they inform the research-to-policy and practice cycle (Best et al., 2009); in software development, they enhance effectiveness and reduce defects (Tiwana, 2004); and in physics-informed symbolic regression, they enable the discovery of meaningful symbolic expressions (Keren et al., 2023).

These examples highlight the importance of designing user-friendly interfaces, delivering accurate and timely feedback, and accounting for the cognitive processes underlying human decision-making when interacting with machine learning models. Building on these insights, the present study investigates an active learning-based

pipeline that integrates human expertise with machine learning to enhance the systematic review process.

The active learning pipeline designed to support systematic reviews consists of several steps that transform natural language text into structured representations suitable for predicting the relevance of records. Computational efficiency is of importance in this context. During the screening process, while a reviewer assesses the next record in the queue, the underlying model is retrained using the updated set of labeled data. Ideally, the model completes its retraining before the reviewer finishes reading the current record, ensuring that the subsequent abstract presented reflects the most recent relevance estimates. As such, limited computational time is a critical requirement in active learning for systematic reviewing.

A wide range of algorithms can be employed for text classification, from traditional approaches such as logistic regression and Naïve Bayes to more advanced methods including support vector machines and decision trees. However, the relationships among relevant records in systematic reviews are complex and not easily captured by surface-level lexical similarities. Conceptual overlap may exist even when vocabulary differs, while ambiguity, varying disciplinary perspectives, and temporal changes in terminology, known as concept drift (Y. Chen, Mani, & Xu, 2012; Gama, Žliobaitė, Bifet, Pechenizkiy, & Bouchachia, 2014), further complicate the task. These characteristics make it challenging for conventional algorithms to discern relevant studies, as they must extract meaning that lies beyond simple word matching.

Deep learning architectures, such as convolutional and recurrent neural networks, are more effective in uncovering these complex semantic relationships compared to classical machine learning algorithms. As shown by Rolnick and Tegmark (2017), deep neural networks can approximate sparse multivariate functions with exponentially greater efficiency than shallow networks performing the same task. The term deep learning refers to the multiple layers within such networks, where each successive layer captures increasingly abstract representations of the data. While shallow models are limited to one or two layers, deep networks may include many, constrained primarily by available computational resources. It is within these deeper layers that the network uncovers the latent structures essential for accurately modeling relevance in systematic review screening (Goodfellow, Bengio, Courville, & Bengio, 2016).

A convolutional neural network (CNN) approach is proposed to implement a deep neural network. CNNs are widely applied in text classification tasks (Collobert & Weston, 2008; Hughes, Li, Kotoulas, & Suzumura, 2017), yet, to our knowledge, have not previously been used to support systematic reviews. The convolutional layers that form the core of CNNs provide a specialized and computationally efficient alternative to standard dense layers. In dense, or fully connected, layers, each neuron connects to every neuron in the preceding layer, resulting in substantial computational cost. Convolutional layers, in contrast, connect only to a limited number of neighboring neurons and share the same weights across these connections. The reduced number of connections makes convolutional layers less computationally demanding while allowing them to extract locally related features from the input data. This property makes convolutional layers well-suited for text-based neural networks and, consequently, for applications within the systematic review process.

CNN models generally require substantially larger training datasets to achieve optimal performance (Montavon, Orr, & Müller, 2012). As demonstrated by Alwosheel, Van Cranenburgh, and Chorus (2018), neural network performance in classification tasks scales positively with sample size. For instance, Giga5, a widely used corpus for training deep learning models, contains nearly ten million documents (Parker, Graff, Kong, Chen, & Maeda, 2011). Evidence from Johnson, Rodeberg, and Wightman (2016) further shows that shallow neural networks can outperform deep neural networks when the amount of training data is limited. Deep models only begin to surpass shallower ones when trained on datasets containing millions of records, but perform comparatively poorly with smaller datasets of around 120,000 documents.

In contrast, systematic reviews typically include only a few thousand records (de Boer, Hofstee, Hindriks, & van de Schoot, 2021). Moreover, active learning for systematic reviewing often begins with only a handful of labeled records in the initial training iteration (van de Schoot et al., 2021). This makes it unlikely that a CNN would perform well during the early stages of the process. It is therefore more effective to start with a shallow classifier that performs robustly under data-scarce conditions, and to transition to a CNN model only once a sufficient number of labeled examples has been collected. In doing so, the model benefits from the strengths of both approaches: the stability of shallow classifiers at the start and the representational power of CNNs when more training data becomes available.

Another reason why switching to a CNN model can be beneficial is that the first set of relevant records in a systematic review is often relatively easy to identify, whereas the final records typically require considerably more effort from the active learning model (van de Schoot et al., 2021). The last-to-find records may differ semantically from those identified in the early phase of screening. When datasets contain multiple semantic clusters, the distribution of relevant records can also form clusters. Once a classifier has identified many records from one cluster, it may become overfitted to that cluster and consequently less effective at detecting records from others. The model can only begin identifying additional records within a cluster once at least one record from that cluster has been retrieved. Such clustered distributions can therefore create challenging situations during the classification process.

For the current study, we first demonstrate the advantage of using active learning over manual screening on a large labeled dataset containing more than 46 thousand records. Performance was evaluated using the Work Saved over Sampling (WSS) metric (Cohen et al., 2006), which measures efficiency relative to random reading. In addition, the Average Time to Discovery (ATD) (Ferdinands et al., 2023) of relevant records was computed to capture the occurrence of last-to-find papers. We then replicated the results of the original meta-analysis (M. E. Brouwer et al., 2019) to examine the potential impact of excluding these last-to-find papers from the evidence base.

In a second study, an optimized convolutional neural network was developed. The third study compared performance, measured in terms of WSS and computational time, across various combinations of classifiers (Naïve Bayes, Support Vector Machine, Logistic Regression, Random Forest, and a neural network classifier) and feature extraction techniques (TF-IDF, Doc2Vec, and SBERT), benchmarking them against

the newly developed 17-layer CNN model. In the fourth study, we investigated whether switching from a classical algorithm to a neural network improves performance relative to the best-performing method identified in the third study.

All simulations were conducted using the simulation mode of the open-source software ASReview. To ensure reproducibility, all scripts, code, and output are publicly available through persistent storage repositories (J. Teijema, 2021a, 2021b; J. Teijema, Van de Schoot, & Bagheri, 2022).

4.1.1 Data

The dataset used in this study comes from a systematic review-based meta-analysis focusing on the evidence for leading psychological and biological theories on the onset, maintenance, and relapse of depressive disorders M. E. Brouwer et al. (2019); Fu et al. (2021); Kennis et al. (2020). For this project, 18 researchers screened approximately 150,000 records for relevance, which took them 3 years. Within a sub-project of this project, the researchers screened over 46 thousand records for a question on psychological theories of depressive relapse. They identified only 63 eligible papers for the final meta-analysis (0.13% inclusion rate). In this project, only longitudinal and prospective studies were included to establish a hypothesized causality between the theories and depressive disorders for five leading psychological theories of relapse and recurrence of major depressive disorder: cognitive, diathesis-stress, behavioral, psychodynamic, and personality-based.

In the study by M. E. Brouwer et al. (2019), information about the dataset construction can be found, providing insights into the methodology and approach used by the researchers. For an understanding of the search strategy and selection process of relevant records, readers can refer to Appendix B of M. E. Brouwer et al. (2019), which contains the search keys used during the literature review. This information is important for replicating the process or conducting further research on the topic.

To establish a direct link and robust effects, any factor derived from one of the five theories needs to be assessed before the relapse or recurrence of major depressive disorder. The status of the disorder was required to be at least at two-time points prospectively through a clinical interview or expert opinion. The goal was to investigate the leading psychological theories, and thus all factors derived from that leading theory were pooled and analyzed. The primary outcome was the effect of the theory-derived factor on the risk of relapse or recurrence of major depressive disorder. The effect sizes Hazard Ratios (HR) and Odds Ratios (OR) for all factors were calculated using reported statistics from each study with the software program Comprehensive Meta-Analysis (Borenstein et al., 2021). The effect sizes were pooled using random-effects models, and the results were published. All pooled odds ratios and hazard ratios are available on the Open Science Framework as M. Brouwer and van de Schoot (2023).

4.1.2 Pre- and postprocessing

The data used in this study consist of the titles and abstracts of all the records identified in the search and their respective labeling decision (i.e., relevant/irrelevant). Before this data could be used for simulations, it needed to be pre-processed. The

researchers used several spreadsheets to manage the enormous number of records and labeling decisions. However, the simulations required a single file with three columns (title, abstract, labeling decision) and a low percentage of missing data. Therefore, all original files were merged, and missing abstracts were added. A description of the entire preprocessing procedure is found on the Open Science Framework as M. Brouwer et al. (2023).

For this study, the pre-processing procedure was continued on the dataset with the addition of stricter deduplication rules to increase the cleanliness of the dataset. On top of that, missing DOIs were obtained, and noisy labels were corrected in two rounds of quality checks. The deduplication scripts are available on Zenodo as van den Brand et al. (2021).

The exact number of records in the post-processed dataset is 46,376, of which 63 were included in the final meta-analysis. This ratio results in a relevance rate of less than 0.14%. The average abstract contains 218 words.

4.2 Study 1 - Active learning-aided systematic reviewing

The purpose of the first study is to increase confidence in active learning for systematic reviews. It investigates the work saved by using active learning, expressed in the WSS metric (Work Saved over Sampling). This metric is calculated from the ratio of effort saved compared to screening records randomly. The study also investigates the stability of the active learning aided systematic review by measuring the impact of skipping the last-to-find records of the original meta-analyses' calculations.

When using the active learning pipeline, not all records are screened. This method saves time but introduces a chance that relevant records are not suggested for screening, although it is unknown if this impact is equal to or smaller than the impact of screening fatigue losses. If the effect of missing the last-to-find records is low, this will lower the perceived risk of using this method. Study 1 aims to address this risk by answering the following research questions:

RQ1.1 How much time would the active learning application have saved during the systematic review that resulted in the Brouwer et al. dataset?

RQ1.2 What effect does the selected prior knowledge have on the average time to discover the relevant records?

RQ1.3 What is the impact of failing to discover the last-to-find records in the systematic review from the Brouwer et al. dataset?

4.2.1 Method

Using a pre-labeled dataset, such as the one used in this study, the labeling via the active-learning pipeline can be simulated, replicating the choices made by the reviewer, and training the model as it would during authentic use. Using these simulations, different models can be compared on how many records would have been found before the user stops reviewing. To answer RQ1.1 and RQ1.2, a simulation was run for each

relevant record, and differences between simulation records were examined.

For RQ1.3, the median last-to-find records were removed from the meta-analysis, and the Hazard Ratios (HR) and Odds Ratios (OR) were recalculated.

4.2.2 Setup

In study 1, we utilized the dataset to assess the efficiency of our active learning-based approach.

The simulation study was conducted with the default settings of ASReview v0.18. The default settings are classification by naïve Bayes combined with term frequency-inverse document frequency (TF-IDF) feature extraction approach for the active learning model. The number of runs was set equal to the number of inclusions in the dataset ($n = 63$). Every run started with training data consisting of only one relevant and ten randomly chosen irrelevant records (held constant across runs).

Randomly screening records and screening records using the active learning pipeline are compared using the WSS metric. This metric is defined as the percentage of papers a researcher does not have to screen. WSS@95% is measured at a recall level of 95%, meaning that it reflects the amount of work saved by using active learning at the cost of failing to identify 5% of relevant publications. Note that humans typically fail to find about 10% due to screening fatigue (Z. Wang, Nayfeh, Tetzlaff, O'Blenis, & Murad, 2020).

For the 63 included records, the Average Time to Discovery (ATD) was computed by taking the average of the time to discovery of all relevant records (Ferdinands et al., 2020). The time to discovery for a given relevant publication was computed as the number of records needed to screen to detect this record. All code to reproduce the simulation results and the output of the simulations can be found at Ferdinands, Teijema, De Bruin, Brouwer, and Van de Schoot (2022).

Finally, the original meta-analysis was redone, excluding the 5 and 10% last-to-find records (i.e., with the highest ATD). The results of the original meta-analysis and the new results are available on the Open Science Framework as M. Brouwer and van de Schoot (2023).

4.2.3 Results

Our findings build upon the Brouwer et al. dataset by demonstrating that active learning can significantly reduce screening time and efficiently identify relevant records in a systematic review. This suggests that our approach could potentially enhance the methodology used in Brouwer et al. (2019) study by increasing the speed and accuracy of the review process.

Figure 4.1 shows the simulation results of study 1, comparing the active learning-based approach to random reviewing, when testing on the Brouwer et al. dataset. It appeared that with active learning, on average, 92% (SD = 0.18; Min/max = 91.65/92.25) of the screening time (WSS) could have been saved compared to reading records at random. After screening only 5% of the total number of records, already

95% (SD = 0.35; Min/max = 95.16/96.77) of the relevant records were found. Based on these results, active learning shows significant time-saving potential compared to random reading.

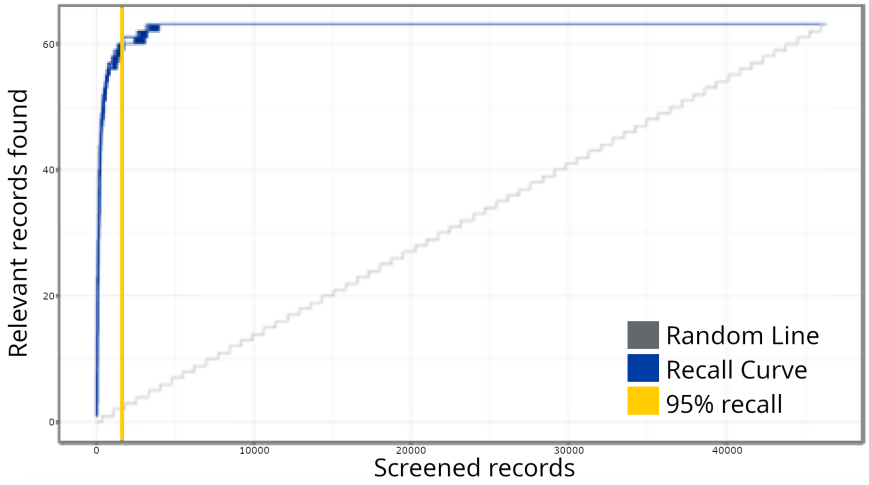


Figure 4.1 Simulation results of study 1. The absolute amount of relevant publications found is displayed on the y-axis, and the absolute amount of screened publications is on the x-axis. The solid blue lines are a combination of all the Recall curves, representing the relevant records found as a function of the screened publications for each of the 63 simulations.

Results show that excluding the 5 or 10% of last-to-find records from the analysis has no impact on analysis results. The conclusions drawn from these papers would have been similar when excluding the last. Even when excluding the last 10% of found records, the results overall remained alike for the analyses on time to relapse (Hazard Ratio) with an insignificant difference in pooled effects. For the odds ratios, the primary analyses (pooled effect sizes for the five leading theories) remained similar and differed only numerically for some subgroup analyses. When analyzing the effect of depressive symptoms on the predictive value of behavioral theories on the odds of depressive relapse, the effects changed from 'just'-significant to 'just' not-significant (Odds Ratio), which was due to one missing study. All other results were similar to the initial results. According to the original authors, neither the original paper's conclusion nor the clinical advice would have changed had the last-to-found records not been included in the review, indicating that these records are not of special relevance to the dataset.

Our results address the research questions as follows: For RQ1.1, we found that active learning saved an average of 92% of screening time compared to reading records at random during the systematic review that resulted in the Brouwer et al. dataset. In terms of RQ1.2, we observed that the prior knowledge had no impact on the average time to discover relevant records in the systematic review. Lastly, addressing RQ1.3, our analysis revealed that failing to discover the last-to-find records in the systematic review from the Brouwer et al. dataset did not impact the analysis results or clinical

advice, indicating these records were not of special relevance to the dataset. This suggests that stopping the review process earlier does not carry any particular risk associated with missing critical information, as the last records were not found to be more significant or influential compared to the others.

4.3 Study 2 - Development of deep neural networks

In ASReview, the implemented neural network is a feed- forward two-layer-based model. The goal of the second study is to propose an optimized deep neural network as a classification model. For this study, the chosen implementation of deep learning was a convolutional neural network consisting of 17 hidden layers. CNNs have been proven to be very effective in text classification problems (Hughes et al., 2017). No such neural network has been used for active learning in systematic reviewing before, to the best of our knowledge. However, this type of neural network is often used in hierarchical classification problems such as ordering records on relevance (Jaderberg, Simonyan, Vedaldi, & Zisserman, 2016). The convolutional layers found in a CNN have fewer connections than the fully connected layers often found in neural networks. The fewer connections and weights make convolutional layers cheaper in terms of memory and computing power needed. Their structure is designed not to be fully connected, opting to find local patterns first and combine them later. Reduced computational power is an essential feature, as every iteration in the classification re-trains the neural network. On the other hand, a fully connected neural network with a similar amount of layers as the implemented network would not be a feasible solution when considering the computational time in relation to the active learning pipeline.

In the second study, the only objective was to develop the CNN network. This leads to the following research question:

RQ2.1 Can a convolutional neural network be effective in text classification for active learning in systematic review?

4.3.1 Setup

The model implemented in this study has a comparable structure but with different layer sizes. Since this simulation study classifies collections of sentences, Doc2Vec was used as the feature extraction method instead of Word2Vec. As shown in Figure 4.2, the implemented model is made up of a combination of separable layers following:

- SeparableConv1D: this is a one-dimensional convolutional layer, mostly used for text, that can be used to detect features in a vector. This type of layer will detect patterns and connections within the records. The ReLu activation accompanying this layer has been beneficial for training deep neural networks(Glorot, Bordes, & Bengio, 2011). This layer has a size setting and a filter size setting (represented as K5 and K3). The size setting shows the number of filters (in this case, 256), and the filter size represents the sliding window in the convolution layer, 5 by 5, and 3 by 3, respectively.
- Dropout: this type of layer is used as a partial prevention for overfitting by setting a part of the nodes to 0 during each training step. Without Dropout,

a node can correct behavior for another node during training. This corrective behavior can lead to overfitting because these fused nodes do not generalize to unseen data. Dropout prevents this from happening and thus reduces overfitting (Srivastava, Hinton, Krizhevsky, Sutskever, & Salakhutdinov, 2014). Figure 2 shows what percentage of the nodes are dropped in each Dropout layer.

- MaxPooling1D: this layer reduces the network dimension size and generalizes patterns found by having kernels in the following layers by looking at relatively more data while keeping the same size.
- Dense: two Dense (or fully connected) layers are set up at the end of the CNN-based architecture, finalizing the network. These layers connect all patterns, which does not happen in the local-only convolutional layers. The number shown in Figure 4.2 represents the number of neurons.

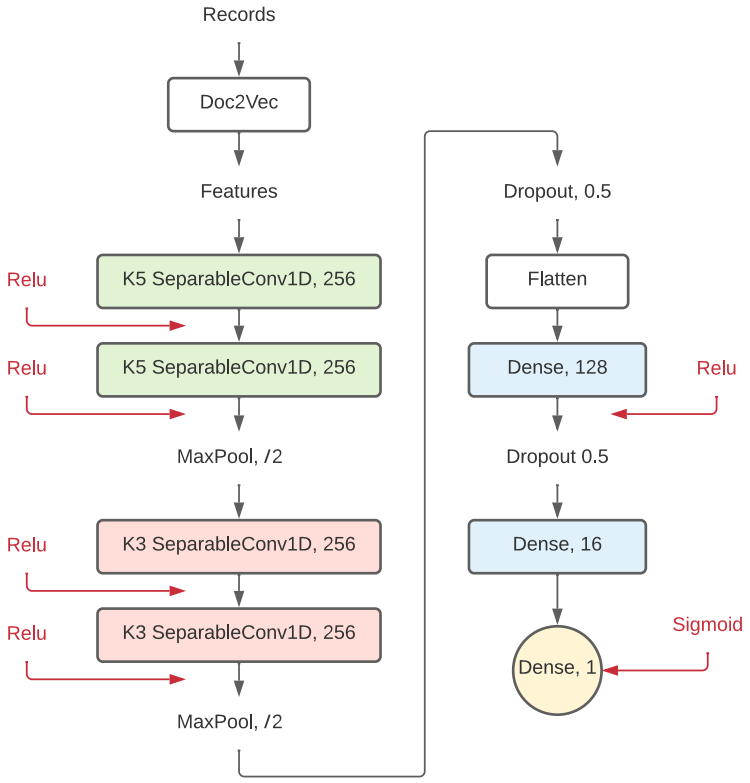


Figure 4.2 Proposed convolutional neural network model. Each element represents a different layer of the neural network. Numbers behind the layer title are settings for that layer.

As the size of the training data increases with each labeled record, so does the optimal amount of training epochs for the neural network. As a result, there is no universally optimal number of training epochs. A heuristic stopping rule was implemented to

compensate for a fluctuating training data size. This rule is based on the network loss delta to avoid having under or overfit networks.

For a neural network to work best, it needs to be optimized. The settings steering the behavior of this convolutional neural network were empirically optimized using the GridSearchCV function found in the Scikit-learn library. This grid search function cross-validates every setting five times[1] and records network accuracy as a performance metric for each run. The following settings were available for optimization: batch size, early stopping patience, early stopping delta, dropout rates, optimization method, kernel size, and filter size. The settings with the highest network accuracy were implemented in the final model.

To adjust for the possible sparsity of a dataset, a convolutional neural network usually adjusts its weights based on class imbalance. The implemented CNN in this study was modified not to calculate a class weight, as the ASReview software has an integrated balancer, making rebalancing the class weights redundant.

The implemented convolutional neural network is built from combinations of these dense neural network layers, separable convolutional layers, activation layers, pooling layers, and dropout layers. The resulting 17 hidden layers deep architecture shown in Figure 4.2 is published on GitHub and Zenodo as a plugin for ASReview.

As this network can handle a wider input size (as a result of being more computationally efficient), a companion feature extractor was created based on the current doc2vec implementation. Doc2vec can be a powerful feature extractor but fails to capture out-of-vocabulary words (Naseem et al., 2021). The standard doc2vec implementation has a vocabulary size of 40. The new feature extractor will be a wider doc2vec implementation with different vocabulary size. The vocabulary size for the new wider doc2vec feature extractor was set to 120 after 5-fold cross-validation in WSS@100% performance using 80, 120, and 250 as potential vocabulary sizes. This resulted vocabulary size should not be taken as universal vocabulary size but rather as near optimum for this dataset.

4.3.2 Results

The performance of the CNN is evaluated in the subsequent two studies by using it as a stand-alone classifier and a switch- to model for switching performance. It will be compared using the WSS@95% and WSS@100% metrics. In response to RQ2.1, our findings demonstrate that a convolutional neural network can be implemented effectively in text classification for active learning in systematic reviews.

4.4 Study 3 - Performance and Computation Time

The third study compares the classifier performance in terms of work saved over sampling for different combinations of classifiers with feature extraction techniques

[1] 5-fold cross-validation is the default setting in scikit-learn following version 0.22, which is widely adopted because it strikes a good balance between bias and variance. This helps to mitigate overfitting and ensures that the model generalizes well to new data.

and compares these combinations with the newly developed 17-layer CNN model. In this study, we aim to answer the following two questions:

RQ3.1 Which combination of feature extraction technique and classification method gives the best performance in terms of WSS for the Brouwer et al. dataset?

RQ3.2 How do the available models compare in terms of computational time and performance?

4.4.1 Method

All possible combinations of feature extraction techniques and classification methods are used in different simulations using the Brouwer dataset. Those simulations are then analyzed for performance and computational statistics. Computational time is presented for the feature extractor and the average iteration time. The order in which records are found in the simulations is registered, and a correlation between this order is calculated for each model.

4.4.2 Setup

This study combined all classifiers (naive Bayes, logistic regression, random forest, support vector machine, and a 2-layer neural network) with feature extraction techniques (TF-IDF, Doc2Vec, and SBERT) available in ASReview v0.18, plus the CNN model developed in Study 2. Only viable combinations were tested as it is impossible to test naive Bayes in combination with doc2vec and SBERT because the multinomial naive Bayes classifier cannot handle matrices containing negative values, which these feature extraction strategies generate in their representations. Moreover, the combination of a neural network and TF-IDF is not feasible because the feature matrices produced by TF-IDF are too wide to realistically employ in the implemented neural network due to limitations in working memory. The remaining combinations were used for simulations.

The results from study 1 show that the performance for simulations with different prior records is very similar, with a low standard deviation in performance. Based on these results, only 1 set of priors for the subsequent simulations was picked through a simulation seed. Furthermore, as study 1 found the last-to-find records of no particular relevance, and since human screening misses 10% of records on average, classifiers are compared at a WSS of 95%, judging performance more similar to real-world application.

The simulations were terminated when all relevant publications were found to save computational time. Running the simulations further would not influence the results, and termination reduces the computational time required to finish the simulation. Each simulation was initiated with 20 records of prior knowledge; ten included records and ten excluded records. The selected prior knowledge was the same for each simulation.

4.4.3 Results

While some combinations perform better than others, all simulations outperform random reading significantly. The simulation with the highest WSS@95% used Logistic Regression as a classifier, combined with SBERT as a feature extractor. This model combination found 95% of all records after screening 587 records, only 1.3% of all records. For comparison, on average with random reading, only one relevant record is expected to be found for every 750 screened papers. The recall of models can be seen in Figure 4.3, and the WSS@95 is provided in the first column in Table 4.1. To zoom in on the neural network models, we isolated the recall of these three models in Figure 4.4. As can be seen, the deeper network starts to outperform the lighter networks only at the very end of the simulation, finding the last records significantly faster than the other models. The best performance is nn-2-layer + SBERT, finding the 48th record significantly faster than the other models. Figure 4.5 shows the correlation matrix of cohesion between the order in which records were found (the rank order) for different classifiers and feature extractors. Note how the correlation is lowest between feature extractors but high for classifiers. Therefore, the order in which records are found is different for each model and is mainly caused by the different feature extractors.

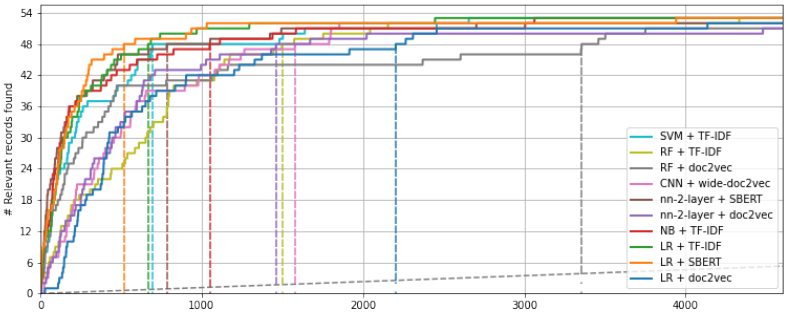


Figure 4.3 Recall curves for the simulation runs. The x-axis shows the number of screened records, truncated at 4,000 records (less than 10% of the full dataset is shown). The y-axis represents the number of relevant records identified. Of the 63 relevant records in the dataset, 10 were provided as prior knowledge; thus, the maximum on the y-axis is 53. The grey dotted line indicates random screening, while colored dotted lines mark the WSS@95% for each simulation.

Table 4.1 shows the computational time for each model. The feature extractor vs. the iteration time difference in computational time can be found. Especially, sBERT significantly increases computational time, followed by doc2vec. Most shallow classifiers are done by training a new model in a split second, and, as expected, the CNN takes much longer.

Our results address RQ3.1 and RQ3.2 as follows: The best- performing combination of feature extraction technique and classification method in terms of WSS for the Brouwer et al. dataset was Logistic Regression with SBERT as a feature extractor,

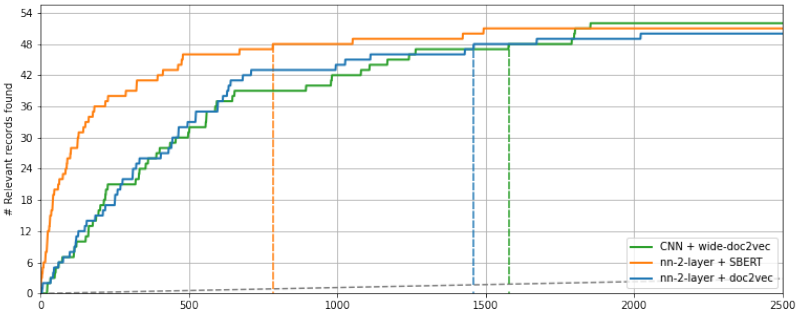


Figure 4.4 Neural network comparison at WSS@90%. The convolutional neural network with the wider doc2vec implementation and the two-layer neural network with both SBERT and doc2vec as feature extractors. When compared in finding the last record, only the convolutional neural network finds these records before the cutoff.

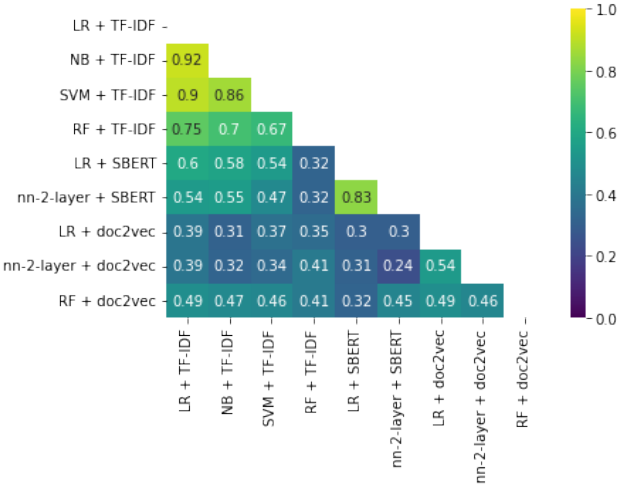


Figure 4.5 Rank order cohesion correlation matrix. This figure shows how similar or different the order can be between models.

Classifier + FE	WSS@95%	FE time	Median iteration time
LR + SBERT	94,21%	6:27:23.23	0:00:00.19
LR + TF-IDF	94,14%	0:00:23.35	0:00:00.05
nn-2-layer + SBERT	93,01%	6:58:30.89	0:00:02.79
NB + TF-IDF	92,81%	0:00:13.62	0:00:00.03
SVM + TF-IDF	92,69%	0:00:15.57	0:00:08.95
CNN + wide-doc2vec	92,34%	0:32:25.44	0:00:59.17
RF + TF-IDF	91,82%	0:00:15.56	0:00:02.45
LR + doc2vec	90,93%	0:18:01.80	0:00:00.02
RF + doc2vec	88,14%	0:15:42.61	0:00:00.57
nn-2-layer + doc2vec	86,57%	0:18:03.91	0:00:01.75

Table 4.1 Performance metrics for each simulation run, sorted by WSS@95%. Time in (hh:mm:ss).

achieving a WSS@95% of 94.21% screening time saved. In terms of computational time and performance, we found that SBERT significantly increased computational time, followed by Doc2Vec. Most shallow classifiers trained new models quickly, while the CNN took much longer. Notably, the deeper neural network started to outperform the lighter networks only at the very end of the simulation, finding the last records significantly faster than the other models, with the best performance being the 2-layer neural network with SBERT. The order in which records were found varied for each model, mainly caused by the different feature extractors.

4.5 Study 4 – Model Switching

The fourth study investigates the performance of the models when switching from one model to another, aiming to create a form of artificial paradigm shift. As different models struggle with different records, switching models might increase the performance of the pipeline. A re-representation of the information should have a significant transformative value for the machine learning algorithm. Here, we aim to answer the following research question.

RQ4.1: Can the performance of the active learning pipeline improve by switching models during the live review process?

4.5.1 Method

This study uses simulations of different machine learning models to investigate whether switching models during the active learning process can improve performance. An ASReview extension was developed to switch between models after a manually set number of records have been screened. The study compares the number of relevant records found after a certain percentage of screened records in the switched simulations to the values of the results of simulation study 3.

4.5.2 Setup

In the fourth study, model simulations from the third study were terminated after a stopping heuristic was reached (e.g., 50 irrelevant records are labeled consecutively) and continued with a different model to investigate if this increases performance. For the simulations, naive Bayes and TF-IDF were selected because it is the default in the software, and Logistic regression with SBERT was chosen as it was the best performing model from study 3. In aid of this switching process, an ASReview extension was developed to switch between models after a manually set number of records have been screened.

To quantify the performance of models after switching, the number of relevant records found after 1% (464 records), 1.5% (696), 2% (928), and 2.8% (1,391) of screened records in the switched simulations are compared to the values of the results of simulation study 3. The metric used for this is Relevant Records Found. The RRF@X% value represents the number of records found after X% of records are screened. The RRF values for switched simulations take this into account and thus represent X% of screened records, including those screened before switching.

4.5.3 Results

Figure 4.6 shows the performance of switching models from the original model. NB + TF-IDF and LR + SBERT serve as benchmark values since, in those simulations, the model was not switched from the starting model.

The stopping rule was triggered at 326 records for the naïve Bayes simulation, at which point 40 of the 53 relevant records had been identified (Figure 4.6a). For logistic regression, it was triggered at 367 records, with 45 relevant records found (Figure 4.6c). As shown in Figure 4.6b, switching from naïve Bayes + TF-IDF to another model almost always results in a performance increase, particularly when a different feature extractor is used. For LR + SBERT, the improvement is limited, since continuing with LR + SBERT already yields near-optimal performance.

Figure 4.7 presents the number of relevant records identified after screening X% of the dataset when switching to the CNN model. As expected, starting with a shallow model and subsequently switching to the CNN outperforms using the CNN alone throughout the screening process.

Figures 4.8a and 4.8b show the performance gains obtained by switching to a different classifier. Switching generally outperforms the default naïve Bayes classifier for nearly all models, including some that performed worse than naïve Bayes in Study 3. No average improvement is observed relative to the optimal classifier, logistic regression. However, certain models surpass logistic regression at specific stages, whereas logistic regression had been superior in all cases previously. In practice, the optimal model is unknown, and Naïve Bayes is the more likely starting point in real review settings.

For RQ4.1, the results of the fourth study show that the performance of active learning can be improved by switching models during the live review process. In particular, switching from naïve Bayes + TF-IDF to another model almost always led to a performance gain, especially when combined with a different feature extractor.

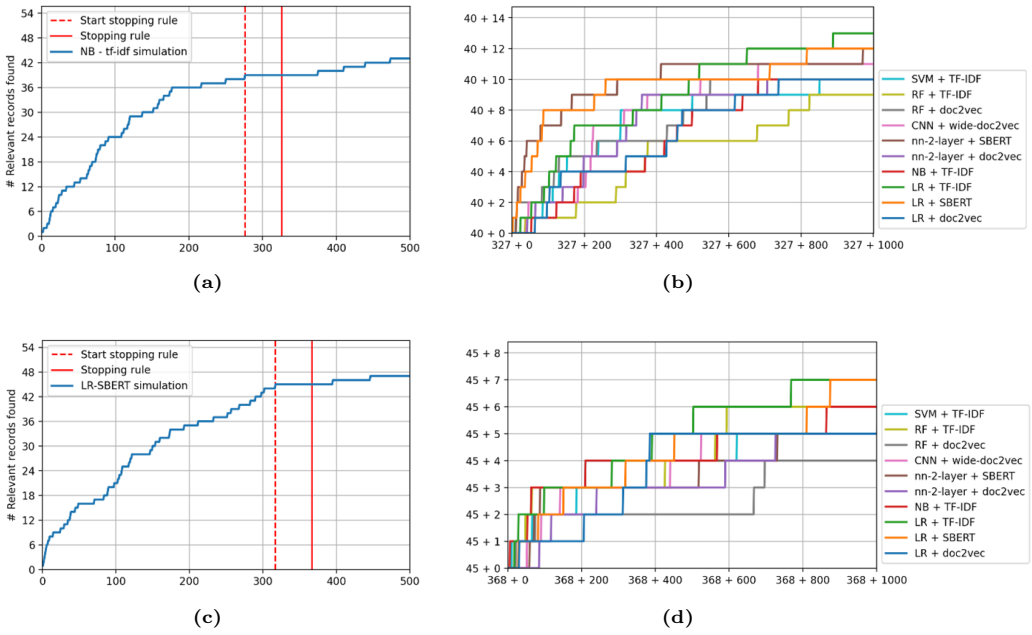


Figure 4.6 Study 4 simulation results. Panels (A) and (C) show recall curves for simulations starting with naïve Bayes (A) and logistic regression (C), before and after switching. Panels (B) and (D) present the recall curves of the other models after switching.

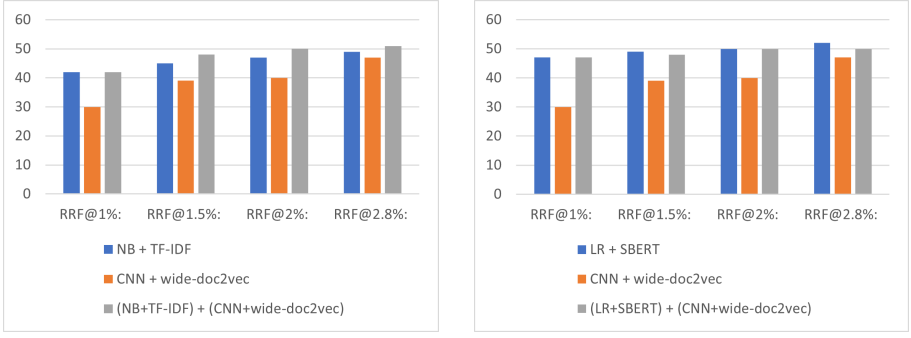
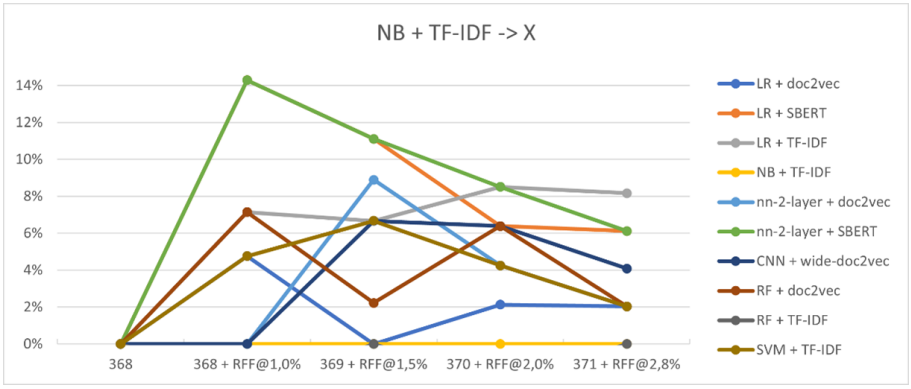
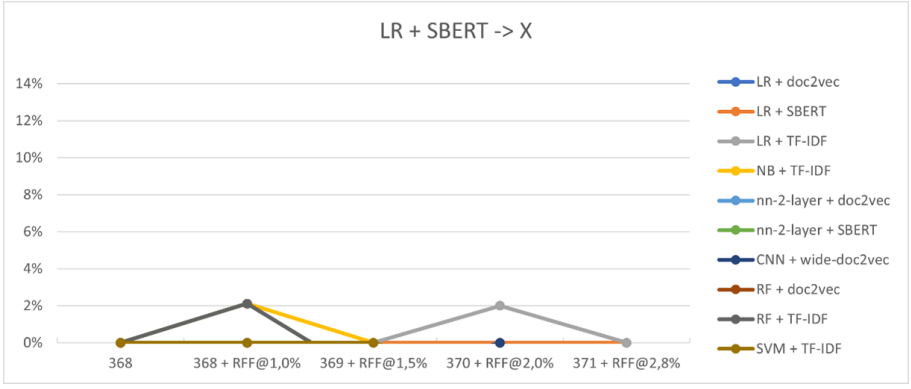


Figure 4.7 Study 4 switching results for the 17-layer CNN model combined with (A) naïve Bayes and (B) logistic regression. The y-axis indicates the cumulative number of relevant records found. RRF@X% denotes the number of relevant records retrieved after screening X% of the dataset.



(a)



(b)

Figure 4.8 Relative performance of classifiers after switching. Panel (A) shows performance when switching from Naïve Bayes to other models. Panel (B) shows performance when switching from logistic regression to other models.

Although no average improvement was observed relative to the optimal classifier, logistic regression, some models outperformed logistic regression at specific stages, whereas logistic regression had previously been superior in all cases. These findings indicate that model switching can enhance the performance of the active learning pipeline in live review settings.

4.6 Discussion

The primary objective of this study was to assess the performance of active learning–aided systematic reviews using a newly developed deep learning model and to examine whether switching between models improves performance. This work consisted of four connected studies

The first study examined the performance and stability of active learning in systematic reviews by analyzing the Average Time to Discovery (ATD) for each record. Subsequent re-analyses of the original meta-analyses were performed while excluding the 5% and 10% of records with the highest ATD values. The results indicate that excluding these last-to-find records did not substantially alter the overall conclusions of the meta-analyses. Across topics, both the hazard ratios for time-to-relapse analyses and the pooled effect sizes for the five leading theories remained largely consistent.

These findings have several implications. Active learning allows screening to be terminated before all records are assessed, which raises concerns about potentially missing relevant studies. Simulations suggest, however, that the probability of excluding meaningful records is very small. The analysis of last-to-find records confirms that they are not missed because of their scientific importance. In all simulated cases, these records were identified after screening less than ten percent of the total pool. Furthermore, human reviewers conducting random screening are known to overlook approximately ten percent of records, in part due to screening fatigue. Active learning mitigates this problem by reducing repetitive manual effort, thereby limiting human error. Collectively, these results support the use of active learning in systematic reviewing, demonstrating that it can substantially reduce screening time without compromising the integrity of the review outcomes.

The second study introduced a convolutional neural network (CNN) as a classifier within the active learning framework. The implementation, made openly available, was accompanied by a specialized Doc2Vec feature extractor to improve the model's representation of textual data. The third study evaluated this model's performance relative to traditional classifiers, measuring both efficiency and accuracy. The open-source release of these components ensures reproducibility and facilitates broader application within the systematic review community.

The third study provides an overview of model performance across different classifiers and feature extractors. While these results are specific to the dataset used, they demonstrate that although overall performance is strong, notable differences remain between model configurations. Selecting appropriate model combinations is therefore essential, as even small performance differences can translate into considerable time savings during screening. Substantial variation in computational time was also observed across classifiers and feature extraction methods. Detailed results are

available online (J. Teijema et al., 2022).

In terms of performance, the default model in ASReview v0.18 (Naïve Bayes with TF–IDF) ranked fourth in speed among all tested combinations. Logistic Regression with SBERT yielded the best performance for this dataset, identifying 95% of all relevant records after screening 587 documents. Logistic Regression with TF–IDF achieved nearly the same performance, while requiring significantly less computational time. These findings suggest that LR + TF–IDF may be the most efficient configuration for this dataset. Whether this observation holds across other datasets remains an open question and should be explored in future work. Only through further empirical validation can recommendations be made for adjusting the default model configuration in ASReview.

The study also compared several neural network–based classifiers. The results indicate that smaller networks identify the majority of relevant records more quickly, while the deeper convolutional neural network was the only model to identify all relevant records before termination. This suggests that deeper architectures may be better at detecting records that are conceptually distant from the main body of relevant studies. If last-to-find records are indeed distinct in content, the CNN's ability to capture more complex relationships could explain its superior performance. Determining whether these records differ systematically in content or terminology is an important question for future research.

Notably, the neural network performed well despite the limited size of the dataset. Deep learning models typically require millions of samples to train effectively, yet here the CNN achieved strong results with only a few thousand examples. This finding indicates that, when properly configured, deep networks may offer advantages even in data-scarce domains such as systematic reviewing.

Finally, the sequence in which classifiers identified relevant records was compared using rank-order correlation. The results show that the lowest correlations occur between feature extractors, rather than between classifiers. This suggests that the feature extractor primarily determines which information is captured from each record, thereby shaping the underlying network of learned patterns. The classifier's role is then to prioritize these patterns in order of likely relevance. Consequently, the structure of the learned representation depends more on the chosen feature extractor, while the overall screening efficiency, expressed as WSS, is influenced more strongly by the classifier. Together, these findings indicate that model performance is not dominated by either component alone. While WSS@95% depends mainly on classifier efficiency, the correlation of identified records is largely driven by the feature extractor.

The fourth study examined the effect of switching from one model to another once a predefined heuristic switching rule was met. Performance was measured for switches originating from both the default model and the best-performing model identified in Study 3. Each model was tested as a potential target for switching, including the newly developed convolutional neural network from Study 2.

The underlying expectation was that lighter models would perform best in the early stages of screening, while heavier models would improve performance in later stages. The results confirm this hypothesis. Models that initially performed less effectively

than lighter models were able to match or surpass their performance once the switch occurred.

In the simulations of Study 3, Naïve Bayes with TF–IDF ranked as the fourth-fastest model on average. After the switching point, however, almost every model outperformed the Naïve Bayes baseline, which subsequently aligned with the lowest-performing configurations from the earlier simulations. Even the top-performing model from Study 3, logistic regression, was outperformed by other models after the switch in some scenarios.

In practical terms, the optimal model configuration in a systematic review is rarely known in advance. Reviewers often begin with the default model rather than the optimal one. The results of this study therefore, demonstrate that, on average, switching models during the screening process is the preferable strategy when the optimal configuration is unknown.

4.6.1 Limitations

Based on the study presented in this paper, we identified the following major challenges in the domain of active learning-aided systematic reviews:

1. **Model selection and performance:** Choosing appropriate models for a dataset is critical, as even minor performance differences can save considerable effort. While different models were evaluated in this study, further investigation is needed to determine whether the results are specific to this dataset or more generally applicable.
2. **Re-training frequency:** Determining the optimal frequency for re-training a model remains a significant challenge. The performance differences between re-training after every newly found record, after every n records, or training only once are not yet clear.
3. **Training time trade-offs:** Balancing training time against performance is essential. In practice, if a model requires long training times, iterations may be skipped. This could allow a faster but less accurate model to outperform a slower, more accurate one under certain conditions.
4. **Optimal model switching point:** Identifying when in the active learning process to switch models is challenging. One candidate is the point at which the order of records stabilizes, but further investigation is required to confirm this.
5. **Content-related differences in record findability:** Understanding how differences in record content affect their findability and ranking may improve active learning-aided systematic review performance. This area requires further research.
6. **Generalizability of results:** The simulation results reported here are specific to the Brouwer dataset and cannot be directly generalized. A benchmark platform containing multiple datasets with diverse topics and characteristics is recommended to enable broader empirical comparison of model performance.

4.6.2 Future Work

For future datasets, such as the depression disorder dataset examined in this study, researchers may benefit from using active learning to extend their original search. Active learning reduces the screening effort, enabling more records to be reviewed and potentially more relevant studies to be identified. In addition, the findings suggest that experimenting with model-switching strategies could be worthwhile, as they may further improve performance in certain contexts. Future work could therefore explore model switching as a potential extension of the active learning-based systematic review workflow. Finally, active learning itself could be investigated in other areas of mental health research, such as anxiety disorders or substance use, to assess its broader applicability.

4.7 Conclusion

The main conclusion of this study is that models exhibit different strengths at different stages of the screening process. Some classifiers perform best in the early phases of the review, while others are more effective in later stages. This pattern is most pronounced for heavier models, such as the two-layer neural network and the convolutional neural network, which transition from weaker to stronger performance as more data becomes available.

These findings highlight the practical value of model-switching strategies. On average, switching models yielded higher performance than relying on a single default classifier. Future implementations of active learning–based systematic reviews may further benefit from ensemble or hybrid approaches that combine these strengths dynamically. Until such approaches are fully developed, a pragmatic strategy is to begin reviews with a lightweight model, such as Naïve Bayes or logistic regression, and to apply heuristic-based switching to more complex classifiers once sufficient data have been labeled.

4.7.1 Data Availability Statement

The datasets presented in this study are available in an online repository: https://-doi.org/10.5281/zenodo.6799805.

Part II

Data Preparation

Chapter 5

SYNERGY-Open machine learning dataset on study selection in systematic reviews

This chapter synthesizes methodological sections regarding the SYNERGY dataset from Chapters 7 and 10. To avoid redundancy, overlapping descriptions were omitted from those chapters and consolidated here. The text presented below was written solely for this dissertation and uses material from the source papers together with original content. Although I am not the primary author of the SYNERGY dataset publication itself, I did contribute to the data collection process. This contribution was supported by a fellowship from the Hofvijverkring in The Hague.

As of May 26, 2026, the SYNERGY dataset has accumulated over 13 million downloads on DataverseNL.

> This chapter describes the dataset referenced in the following citation:
> De Bruin, J., Ma, Y., Ferdinands, G., Teijema, J. J., & van de Schoot, R. (2023). SYNERGY - Open machine learning dataset on study selection in systematic reviews (Version V1) [Dataset]. DataverseNL. https://-doi.org/10.34894/HE6NAQ

The SYNERGY dataset is the most diverse and high-quality data collection currently available for the study of automated screening in systematic reviews. It consists of 26 datasets, each corresponding to a completed systematic review. These datasets were constructed by retrieving scientific records from recognized bibliographic databases and annotating them with binary inclusion labels based on predefined eligibility criteria. Each row in a dataset represents a publication (e.g., journal article, preprint, or report), including its title, abstract, and an inclusion/exclusion label indicating whether it was selected during the manual screening process. Additionally, each dataset in SYNERGY is annotated with inclusion and exclusion criteria. Full dataset details are provided in Appendix Table C.1.

SYNERGY spans a wide range of disciplines, including medicine, psychology, biology, and computer science. Although approximately half of the reviews fall under the broad "medicine, NOS" category, the collection includes diverse topical areas: 3 datasets from computational sciences, 7 from psychology (some of which intersect with medicine), and 3 from biology (also with medical overlap).

The datasets vary substantially in size and prevalence of relevant records. Dataset sizes range from 238 to 48,375 records. Based on size, they can be grouped as follows:

- Very small: $< 1{,}000$ records (7 datasets)
- Small: $1{,}000-3{,}000$ records (7 datasets)
- Medium: $4{,}000-10{,}000$ records (10 datasets)
- Very large: $> 30{,}000$ records (2 datasets)

The proportion of relevant records also varies:

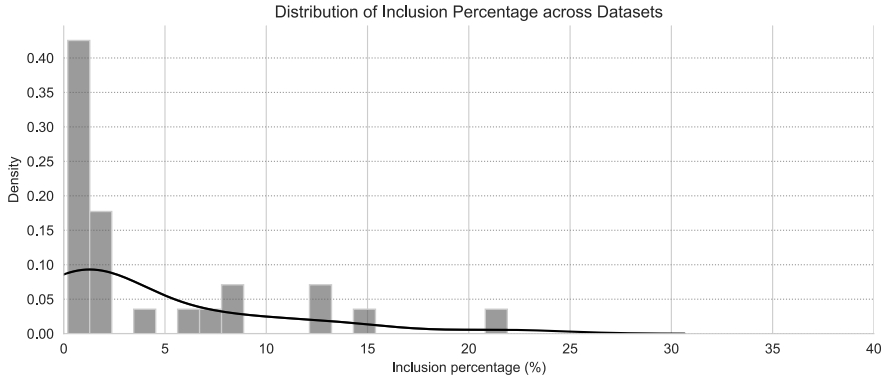
- Very rare: $< 0.5\%$ (4 datasets)
- Rare: $0.5-0.99\%$ (6 datasets)
- Average: $1.0-2.2\%$ (7 datasets)
- Frequent: $4-15\%$ (8 datasets)
- Abundant: $> 20\%$ (1 dataset)

Figure 5.1a shows the distribution of inclusion ratios across all 26 SYNERGY datasets. The inclusion ratio for each dataset is computed as:

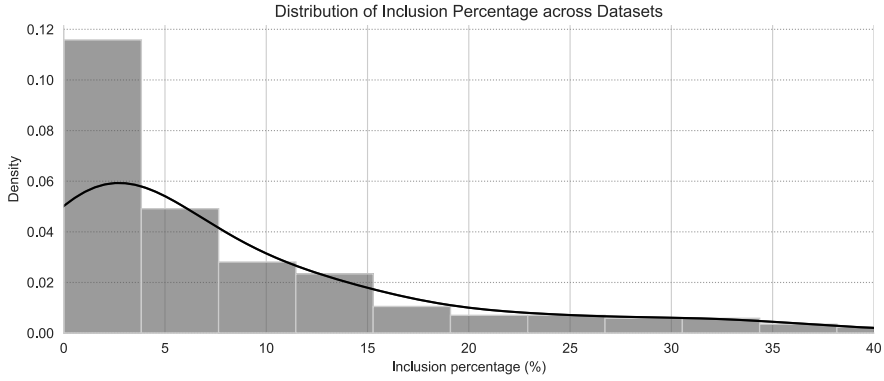
$$\text{Inclusion ratio} = \frac{\text{Included records}}{\text{Total records}}$$

The figure combines a binned histogram and a kernel density estimate (KDE) line to provide both count-based and smoothed views of the distribution.

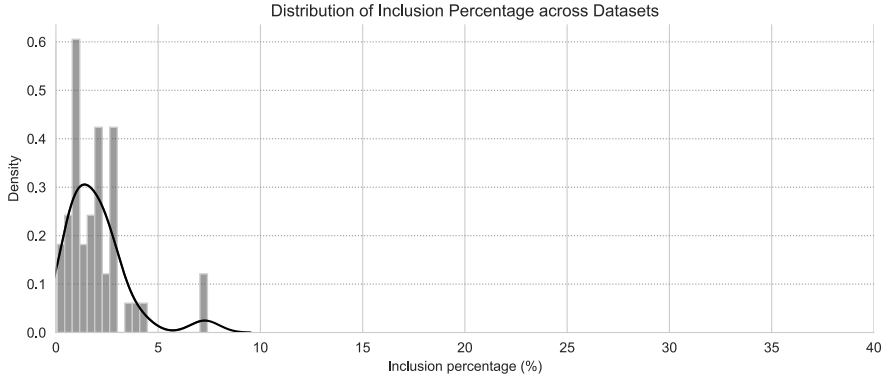
Most SYNERGY reviews fall in the lower range of inclusion ratios, and the distribution extends upward with several higher-prevalence cases. This produces a moderate spread (mean 4.3 percent, median 1.5 percent, sd 5.6). When viewed alongside the broader scoping review collection from chapter 3 and the EULAR dataset from chapter 9,



(a) SYNERGY dataset (26 reviews)



(b) Extracted reviews from chapter 3 (224 reviews)



(c) Extracted reviews from chapter 9 (45 reviews)

Figure 5.1 Inclusion ratio across independent review populations. Each plot shows a histogram with KDE overlay, showing the distribution of included records relative to total screened records in each dataset.

SYNERGY sits between the two. The scoping review collection shows much higher and more variable inclusion ratios (mean 8.8 percent, sd 11.3), while the EULAR reviews cluster tightly at the low end (mean 2.0 percent, sd 1.5, median 1.8 percent).

These differences show how inclusion ratios can differ between fields. Protocol strictness, database selection, and the maturity of a research area also likely play a role. Against this background, the SYNERGY dataset provides a useful middle profile, making it well-suited as a representative benchmark for modelling and method evaluation. Still, expanding the SYNERGY dataset to include a broader range of topics and prevalence levels would strengthen its role as a benchmark.

The SYNERGY dataset plays a central role in this dissertation. It provides the foundation for the large-scale simulations presented in later chapters. Beyond its use here, the dataset is used frequently within the Utrecht University AI-aided Knowledge Discovery Lab, where it supports multiple ongoing research projects. As of May 26, 2026, it has accumulated over 13 million downloads on DataverseNL. Its open availability contributes to its wide reach and supports research in automated screening and evidence synthesis.

Chapter 6

Makita—A workflow generator for large-scale and reproducible simulation studies mimicking text labeling

This chapter is a reproduction of an article published in *Software Impacts*. Adjustments were limited to formatting and cross-referencing. Additionally, Figure 6.2 was updated with additional information. No changes were made to the scientific content. The current latest version, v1.1, was developed in close collaboration with the lead maintainer of ASReview Lab. Version 1.1 is developed for ASReview 2.0 and includes new features, a restructured codebase, and several bug fixes. The update improves usability, reliability, and support for larger and more complex simulation workflows.

Teijema, J. J., van de Schoot, R., Ferdinands, G., Lombaers, P., & de Bruin, J. (2024). Makita—A workflow generator for large-scale and reproducible simulation studies mimicking text labeling. *Software Impacts, 21*, 100663. https://doi.org/10.1016/j.simpa.2024.100663

This paper introduces ASReview Makita, a tool designed to enhance the efficiency and reproducibility of simulation studies in systematic reviews. Makita streamlines the setup of large-scale simulation studies by automating workflow generation, repository preparation, and script execution. It uses Jinja and Python templates to create a structured, reproducible environment that aids both novice and expert researchers. Makita's flexibility allows for customization to specific research needs, ensuring a repeatable research process. This tool represents an advancement in the field of systematic review automation, offering a practical solution to the challenges of managing complex simulation studies.

de Bruin, J., & Teijema, J. (2025). *ASReview Makita: A workflow generator for simulation studies using the command line interface of ASReview LAB (v1.1)* [Software]. Zenodo. https://doi.org/10.5281/zenodo.15720003

Nr	Code metadata description	Value
C1	Current code version	V0.9.0
C2	Permanent link to code/repository used for this code version	`https://github.com/asreview/asreview-makita`
C3	Permanent link to reproducible capsule	`https://codeocean.com/capsule/5417339/tree/v1`
C4	Legal code license	MIT License
C5	Code versioning system used	git
C6	Software code languages, tools, and services used	Python, Jinja
C7	Compilation requirements, operating environment,s and dependencies	Prog. Language :: Python :: 3.7 Prog. Language :: Python :: 3.8 Prog. Language :: Python :: 3.9 Prog. Language :: Python :: 3.10 Prog. Language :: Python :: 3.11 ASReview jinja2 cfgtemplater
C8	If available, link to developer documentation/manual	`https://github.com/asreview/asreview-makita/blob/main/README.md`
C9	Support email for questions	`asreview@uu.nl`

Table 6.1 Code metadata for ASReview Makita.

6.1 Summary

The field of accelerating the screening phase of systematic reviews with advanced machine learning methods is rapidly evolving (chapter 3). A simulation study involves mimicking the screening process for a systematic review of a human in interaction with an Active learning model. The simulation reenacts the screening process as if a researcher were using a machine learning model to prioritize the order of papers being screened. The performance of one or multiple models can then be measured by performance metrics, such as the Work Saved over Sampling, recall at a given point in the screening process, or the average time to discover a relevant record. However, setting up a simulation study can be a time-consuming and error-prone process, especially since reproducibility is of key importance.

This chapter presents ASReview's Makita (**MAK**e **IT A**utomatic) (J. J. Teijema, Van de Schoot, Ferdinands, Lombaers, & De Bruin, 2023). Makita is the precursor to a reproducible simulation study. It streamlines the simulation study design process for systematic reviews using ASReview (van de Schoot et al., 2021), providing a generative framework to simplify creating and running large-scale simulations. Using Makita templates, different study workflows can be generated to fit the study's needs. If a study requires a unique template, a custom template can be used. Its implementation through the command-line interface aims to make reproducible and repeatable research easy and efficient, to assist both novice and expert researchers.

6.2 Statement of need

Although tools such as ASReview LAB (ASReview LAB developers, 2023) offer various ways to simulate the screening process in systematic reviews via their user interface, there is a need for automation in setting up the research environment for large-scale simulation research. Setting up the structure of a simulation study manually is prone to mistakes and a tedious task, especially when the scale of the simulation increases. ASReview Makita fills this gap by automating the workflow setup, preparing GitHub repositories, documentation, pre-/post-processing code, and generating execution scripts.

Simplifying reproducibility and maintaining an organized folder structure are key elements in scientific research. They ensure that experiments can be reliably repeated and built upon by other researchers. A well-organized directory makes it easier to understand the workflow and locate files, and contributes to the transparency and credibility of the study (Lombaers, de Bruin, & van de Schoot, 2024).

6.3 Technical Functionality

Using a combination of Jinja-based templates and Python templates, ASReview Makita automatically generates a hierarchical folder structure, a README.md (including descriptions, instructions, file tree, and data statements), any additional code used for pre-and post-processing, and a batch or shell execution script. Makita offers code for, among others, extracting dataset statistics (developers, 2025a), extracting simulation performance metrics such as Time to Discovery (Ferdinands et al., 2023), merging those metrics into easy-to-read tables, generating word clouds (developers, 2022), and plotting the results (developers, 2025c). Makita assures that all steps of the simulation study are stored and thus reproducible and transparent.

The Jinja-based templates handle study structure, while accompanying Python templates add extended functionality. A range of standard templates is available, specifically tailored for ASReview simulations. Overall, the architecture provides a modular and flexible framework, allowing users to easily adapt the tool to their specific research needs.

What Makita does:

- Setting up a workflow for running a large-scale simulation study
- Preparing a GitHub repository, including a README file
- Automating the many lines of code needed
- Creating an execution script for running the simulation study with just one line of code
- Making research fully reproducible
- Allowing for custom templates to accommodate specific research questions

What Makita does not do:

- Executing jobs or tasks itself
- Writing the study

While Makita was originally developed for use with ASReview's simulation CLI, Makita's design allows it to be integrated with any other CLI tool via a customized template, broadening its applicability across different large-scale research environments. Makita can be used locally, on a server, or in combination with Docker and Kubernetes.

Very-large-scale simulation studies have been successfully run using Makita, with over 29,000 simulations in a single study, using 25 different datasets and 92 different simulation models (chapter 7. The study implemented Makita within a Kubernetes cluster, generating custom templates on the fly for each of the cluster nodes' specific needs (J. J. Teijema, 2023a).

6.4 Software Scope

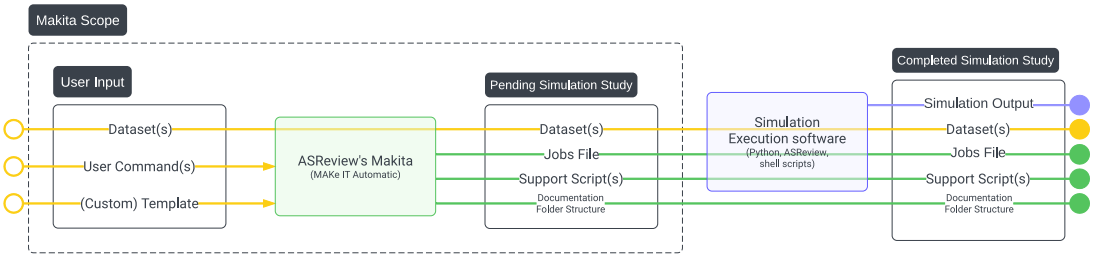


Figure 6.1 Diagram illustrating the general workflow of Makita. The "Makita Scope" section defines the specific functionalities covered in this paper.

The software discussed in this paper focuses on the section 'Makita Scope' of Figure 6.1. For starting the creation of a simulation study, three elements are necessary: (1) one or more datasets that provide the input data, (2) user-defined commands that specify operational settings, and (3) a choice between using a default template or a custom template provided by the user. Together, these are represented as 'User Input' in the figure.

Next is the 'Pending Simulation Study' section, which sets up but does not start the simulations. It organizes datasets, the jobs file, support scripts, documentation, and folder structure, preparing everything needed for the simulation study.

The 'Completed Simulation Study' section deals with the results after the simulations have run via execution of the jobs file. It includes output (such as, but not limited to: simulation files, plots, metric files, and updated documentation), input datasets, used code scripts for processing results, and documentation. Ideally, additional details should be added to the documentation to explain the goal of the study, results, and methods used, for future reference and reproducibility.

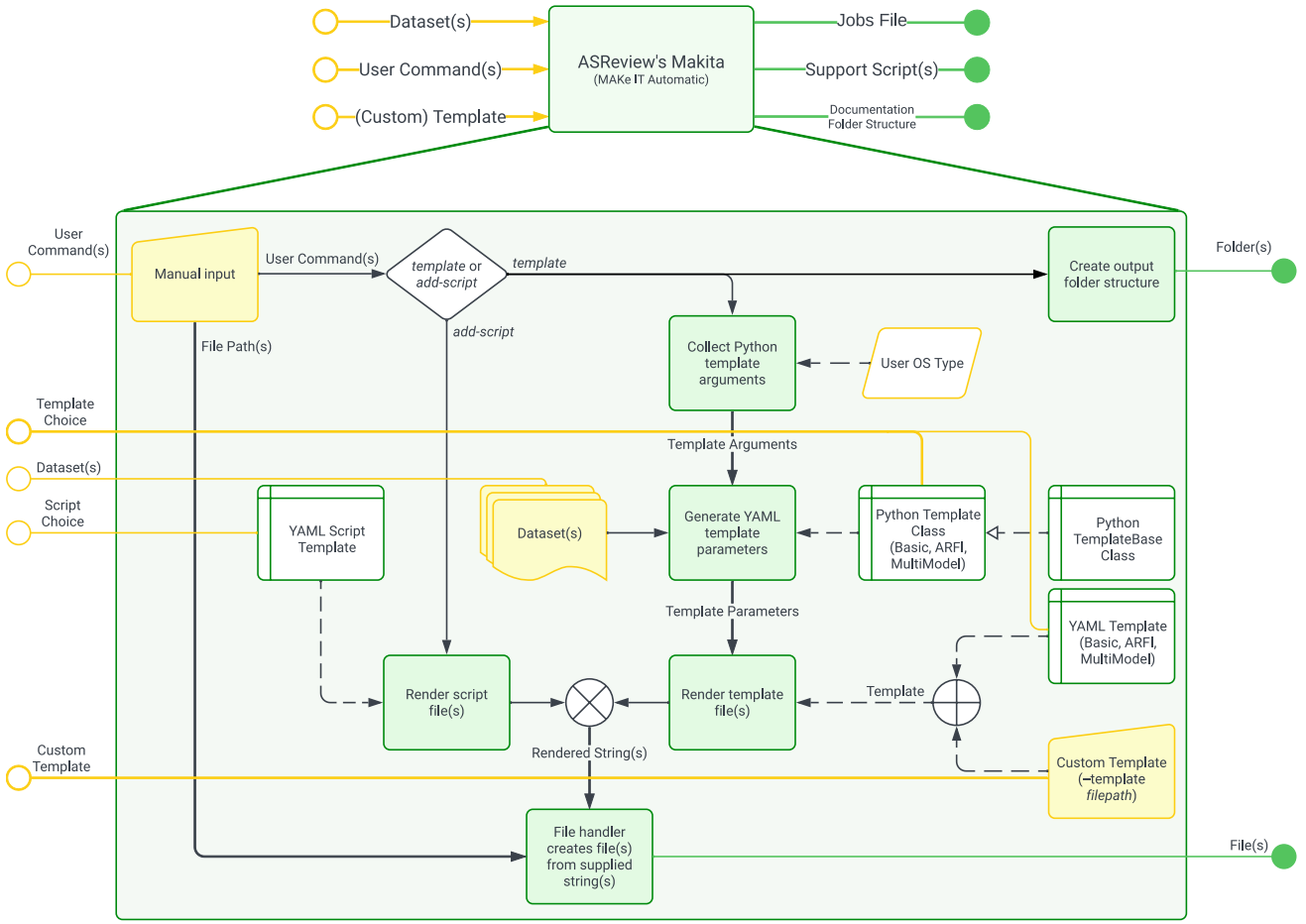


Figure 6.2 Diagram illustrating the sequence of operations of Makita. Key processes include the selection of input datasets, the choice of templates, the collection of template arguments, and the rendering of files from templates. Makita supports custom study design through user-defined templates and adapts the jobs file to the user's operating system. The final outputs are organized into a structured folder system as specified by the generated jobs file.

6.5 Software Architecture

Makita starts with the `MakitaEntryPoint`, which handles the execution of user commands via the `argparse` library. The entry point defines several commands, with the primary ones being template setup and script addition.

For template configuration, the user specifies various operational parameters, such as the *template name*, *job file type*, *dataset*, *and output locations*, *initialization seeds*, *model configurations*, and more, which allows for precise control over the simulation settings. The command setup supports dynamic handling of template parameters, allowing both default and user-provided templates. The template system uses Python's entry point mechanism to load templates, ensuring modularity and extensibility.

After the rendering of both scripts and documentation, files are centrally handled by the `FileHandler` class, which executes file operations such as adding, overwriting, and generating files from templates using the Jinja2 templating engine. The architecture is designed to be adaptive to different operating systems, providing tailored job file generation (e.g., *.bat* for Windows and *.sh* for Unix-like systems) across platforms.

6.6 Usage

Upon creating a 'data' folder with the desired datasets, the study structure is generated by running the Makita command for the 'basic' template. In this case, a 'n_runs' argument is added to the command, indicating that 100 simulations with different seeds are needed for the study. Executing the generated jobs file starts the simulation tests, producing output logs and metrics within the created folder structure. This allows for easy access to performance metrics and study results. Below are the file tree results for running the basic template and executing the jobs file. File trees are generated with scientific ordering, following Scitree (De Bruin, 2023).

```
> asreview makita template basic -n_runs 100
```

Folder structure after Makita generation, before the execution of the 'jobs.bat' file:

```
Makita_basic/
├── README.md
├── jobs.bat
├── data/
│   └── generic_labels.csv
└── scripts/
    ├── get_plot.py
    ├── merge_descriptives.py
    ├── merge_metrics.py
    └── merge_tds.py
```

After the execution of the 'jobs' file, the following additional files will populate the

folder structure.

```
└── output/
    ├── figures/
    │   ├── plot_recall_sim_generic_labels.png
    │   ├── wordcloud_generic_labels.png
    │   ├── wordcloud_irrelevant_generic_labels.png
    │   └── wordcloud_relevant_generic_labels.png
    ├── simulation/
    │   └── generic_labels/
    │       ├── descriptives/
    │       │   └── data_stats_generic_labels.json
    │       ├── metrics/
    │       │   ├── metrics_sim_generic_labels_0.json
    │       │   ├── ...
    │       │   └── metrics_sim_generic_labels_99.json
    │       └── state_files/
    │           ├── sim_generic_labels_0.asreview
    │           ├── ...
    │           └── sim_generic_labels_99.asreview
    │
    └── tables/
        ├── metrics_sim_all.csv
        ├── metrics_sim_all.xlsx
        ├── data_descriptives_all.csv
        ├── data_descriptives_all.xlsx
        ├── metrics/
        │   ├── metrics_sim_generic_labels.csv
        │   └── metrics_sim_generic_labels.xlsx
        └── time_to_discovery/
            ├── tds_sim_generic_labels.csv
            └── tds_sim_generic_labels.xlsx
```

6.7 Impact Overview

Enhanced Accessibility and Efficiency in Research ASReview Makita significantly reduces the time and complexity involved in setting up simulation studies. Its ability to swiftly create reproducible workflows allows researchers, especially those new to the field, to initiate and evaluate their simulation studies in mere minutes. This accelerated process not only saves time but also encourages a broader exploration of research questions, expanding the horizons of systematic review research.

Reproducibility and Reliability The software's emphasis on reproducibility is an important element in improving the quality of research. By ensuring that each step of the simulation study is meticulously recorded and can be replicated, Makita enhances the reliability and credibility of research findings. On top of recording every step in Makita's process, it writes basic documentation for later reference.

This documentation is crucial in a field where the accuracy and consistency of data processing directly influence the outcomes and interpretations of systematic reviews.

Ongoing Research and Contributions The impact of Makita is evidenced by its usage in multiple research projects, including many unpublished exploratory studies, and in studies such as Campos et al. (2024); Neeleman et al. (2024); Oude Wolcherink, Pouwels, van Dijk, Doggen, and Koffijberg (2023); Romanov et al. (2024). Makita is used in both government and commercial organizations, its open-source nature allowing for easy adaptation in any setting. These organizations include, but are not limited to, the *PBL Netherlands Environmental Assessment Agency*, the Dutch *National Institute for Public Health and the Environment*, and private institutions. Its usage in these projects highlights its utility and relevance in modern research settings.

Part III

Modeling and Evaluation

Chapter 7

Large-Scale Simulation Study of Active Learning Models for Systematic Reviews

This chapter presents a large-scale simulation study published in the *International Journal of Data Science and Analytics*. Minor revisions were made for clarity and formatting. The section describing the SYNERGY dataset was excluded here to prevent duplication with the material presented in chapter 5.

Teijema, J. J., de Bruin, J., Bagheri, A., & van de Schoot, R. (2025). Large-scale simulation study of active learning models for systematic reviews. *International Journal of Data Science and Analytics*, 1-22.

Purpose: Despite progress in active learning, evaluation remains limited by constraints in simulation size, infrastructure, and dataset availability. This study advocates for large-scale simulations as the gold standard for evaluating active learning models in systematic review screening.

Methods: Two large-scale simulations, totaling over 29 thousand runs, assessed active learning solutions. The first study evaluated 13 combinations of classification models and feature extraction techniques using high-quality datasets from the `SYNERGY` dataset. The second expanded this to 92 model combinations with additional classifiers and feature extractors.

Results: In every scenario tested, active learning outperformed random screening. The performance gained varied across datasets, models, and screening progression, ranging from considerable to near-flawless results.

Conclusion: The findings demonstrate that active learning consistently outperforms random screening in systematic review tasks, offering significant efficiency gains. While the extent of improvement varies depending on the dataset, model choice, and screening stage, the overall advantage is clear. Since model performance differs, active learning systems should remain adaptable to accommodate new classifiers and feature extraction techniques. The publicly available results underscore the importance of open benchmarking to ensure reproducibility and the development of robust, generalizable active learning strategies.

7.1 Introduction

Methodologies used to reduce the screening labor for systematic reviews are continually being introduced (Beller et al., 2018; Marshall & Wallace, 2019; O'Connor et al., 2019; Olorisade et al., 2017; Olorisade, De Quincey, Andras, & Brereton, n.d.; Thomas et al., 2011, 2017; Van Dinter et al., 2021). Especially the use of active learning for prioritization in systematic review screening (Cohen et al., 2006; Hashimoto, Kontonatsios, Miwa, & Ananiadou, 2016; Settles, 2009; B. C. Wallace, Trikalinos, Lau, Brodley, & Schmid, 2010) has seen significant progress and innovation. This application of active learning has been integrated into several screening software tools (Adam et al., 2021; Cowie et al., 2022; Howard et al., 2020; Jimenez et al., 2022; Khalil et al., 2022; Mauricio & Gonzalez, n.d.; Pellegrini & Marsili, 2021; Przybyła et al., 2018; Robledo et al., 2021; Scott et al., 2021; van de Schoot et al., 2021; Wagner et al., 2022; B. Wallace, 2012; L. L. Wang & Lo, 2021), employing a variety of machine learning models to improve prioritization efficiency. However, while model development has been rapid, evaluation remains inconsistent. Additionally, many tools lack the flexibility to incorporate different machine learning models, limiting their applicability in research settings.

A simulation study emulates user labeling decisions using pre-labeled data, which enables the recreation of a systematic review's precise performance. By adjusting parameters such as prior knowledge, feature extractors, and, classifiers and then re-running the simulations with the same pre-labeled data, performance evaluations of machine learning models can be conducted (where in the current paper a model refers to a configuration comprising a feature extraction and classifier). A single simulation can provide insights, but the true value emerges with replicability and scalability.

Empirical support for active learning in screening prioritization largely relies on simulations. While these simulations are generally implemented adequately, many studies would benefit from larger, broader, and more reproducible simulations to strengthen their conclusions and practical relevance. Improving simulation quality helps maximize time savings, as the choice of model can translate into hours or even days of work saved. However, performance simulation studies in this field face several limitations, including minimal use of data, a lack of diversity in studied domains, limited model comparisons, and the use of non-standardized metrics, as shown in the scoping review of chapter 3. Addressing these challenges would improve the reliability and generalizability of active learning in systematic review screening.

First, the median number of datasets used in simulation studies evaluating the performance of multiple active learning models was under four datasets. The limited median number of datasets used in these studies may constrain the generalizability of the findings, as it is likely that the performance for a single or couple of datasets is not interpretable as general performance.

Some studies have incorporated multiple datasets (Carvallo, Parra, Lobel, & Soto, 2020; Molinari & Kanoulas, 2022; C. R. Norman, Leeflang, Porcher, & Neveol, 2019; Zou & Kanoulas, 2020). However, studies that use more than four datasets still predominantly focus on medical reviews, as shown in Figure 7.1, based on data from chapter 3. Even the high-quality Cohen et al. (2006) dataset, often considered the

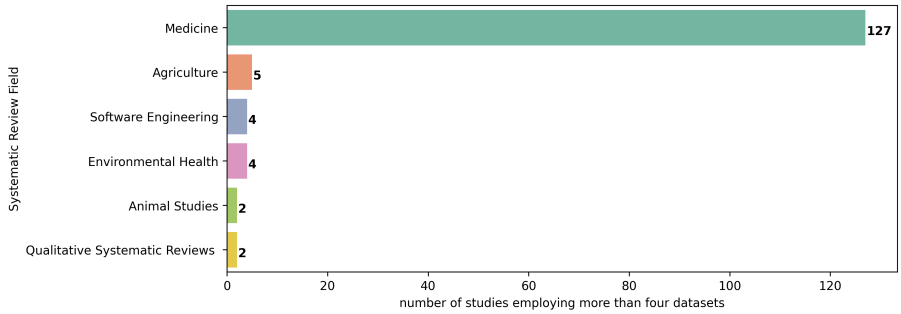


Figure 7.1 Distribution of fields in simulation studies employing more than four datasets, as found in J. J. Teijema, van den Brand, et al. (2023).

gold standard in the field, is limited to drug class reviews. Expanding the range of datasets beyond medical topics would further enhance the generalizability of active learning models across different domains and disciplines.

Third, it was found that most simulation studies that compared active learning models typically involved no more than three distinct models. From the evaluated studies, we identified 13 combinations of well-performing classifiers and feature extractors that are frequently utilized. Examples of most often used classifiers include Logistic Regression, Random Forest, and Support Vector Machines, paired with feature extraction techniques like TF-IDF and word embeddings. Additionally, a review of the active learning software tools[1] reveals a predominant use of Support Vector Machine in software.

Fourth, although many diverse metrics exist (O'Mara-Eves et al., 2015) for the evaluation of performance, most major studies focus on a single one. The most commonly used is the *Work Saved over Sampling @ 95%* (WSS@95%) metric (Cohen et al., 2006). While useful for evaluating performance at a fixed recall level, WSS@95% provides no insight into how a model performs in scenarios such as quickly retrieving relevant records after a cold start or identifying the last-to-find records (Byrne, Hofstee, Teijema, De Bruin, & van de Schoot, 2024; Ferdinands et al., 2023; Harmsen et al., 2024). Evaluating scenarios like retrieval efficiency in the first 100 papers or the ability to find the final relevant paper would help tailor models to specific screening challenges.

These limitations set the stage for our study on the diversity of data and models in simulation studies.

The `SYNERGY` dataset (chapter 5), used in our simulations, is the most diverse collection of systematic review datasets currently available. It spans multiple disciplines, including medicine, psychology, computational sciences, and biology, making it suitable for testing active learning methods across fields. In addition to covering various research fields, the dataset includes a wide range of dataset sizes and relevance densities,

[1] github.com/Rensvandeschoot/software-overview-machine-learning-for-screening-text#overview-of-available-models

allowing models to be tested under different screening conditions, from rare relevant records to high-prevalence scenarios.

Moreover, new models are continuously being developed, such as deep learning architectures and ensemble methods, which show promise in various applications (Bojanowski, Grave, Joulin, & Mikolov, 2016; T. Chen & Guestrin, 2016; Feng, Yang, Cer, Arivazhagan, & Wang, 2020; S. Lee, Shakir, Koenig, & Lipp, 2024). However, these newer models have yet to be widely adopted in simulation studies, which indicates a gap between model development and their application in active learning simulations.

The gap between the frequent use of certain models in active learning software tools and their evaluation with simulation studies can largely be attributed to the complexities involved in setting up and running simulations, especially on a large scale. Establishing a robust simulation infrastructure is a significant undertaking. While some software allows for limited simulation capabilities (Hamel et al., 2020), most of the code used in the 48 simulation studies reviewed was custom-made.

To facilitate large-scale simulations, software such as ASReview (van de Schoot et al., 2021) is essential, enabling seamless integration of various models. Additionally, workflow generators like ASReview's Makita (chapter 6) play a crucial role in setting up repeatable and reproducible simulations. These tools make it possible to leverage larger datasets, such as `SYNERGY`, for more extensive evaluations. However, the scale of such datasets results in a substantial number of simulation runs, necessitating adequate infrastructure to ensure efficient execution (Romanov et al., 2024). This study demonstrates the full simulation pipeline, providing a framework for future research in active learning for systematic review screening.

In the current study, we conduct two large-scale simulations to evaluate the performance of the active learning-based pipeline across a broad range of systematic review datasets. The **first simulation study** focuses on the 13 combinations of classifiers and feature extraction techniques identified in the systematic review, exploring both inter- and intra-dataset variability. These simulations use only two records to start the active learning cycle. One relevant and one irrelevant document, which together, are known as the *Prior Knowledge*.

In the **second simulation study**, we expand our analysis to 92 feature extractor (Table C.2) and classifier (Table 7.1) combinations, selecting models that have performed well in natural language processing but are rarely used in active learning for systematic reviews. This includes both pre-trained and newly trained models, evaluated for their potential to improve systematic review performance and precision. The amount of prior knowledge is increased based on findings from the first study and insights from previous research (Byrne et al., 2024).

Our study has two key objectives:

1. To analyze variability in simulation studies, both across datasets and within individual datasets, as well as differences between models.
2. To evaluate model performance across different phases of the simulation: early screening, final screening, and overall effectiveness.

7.1.1 Background AI-aided screening

Screening prioritization is explained in Cohen et al. (2006); van de Schoot et al. (2021), and in-depth in Box 1 of Lombaers et al. (2024). These offer a detailed exploration of active learning, current simulation research, and challenges in assessing active learning models in systematic reviews. This background provides the context for understanding the methods and applications of active learning.

Systematic reviews are a method for synthesizing evidence to answer specific research questions (Cumpston et al., 2019). This process typically involves several phases: formulating a research question, designing a search strategy, screening records for relevance, and synthesizing the findings. Among these, the screening phase, where researchers evaluate large sets of titles and abstracts, is especially time-intensive and is the focus of this work. Screening prioritization is part of the PRISMA checklist, as mentioned as Priority screening in box 3 of Page et al. (2021).

Active learning is a form of machine learning that does not require a fully labeled dataset. Instead, its performance is refined in iterative cycles through interactions with human reviewers. At each iteration, the model requests labels from the human reviewer, learns from that new information, and improves its predictions. The enhanced model then selects records more accurately, enabling the human reviewer to label only the most informative items. This positive feedback loop is especially suitable for systematic review screening, which generally begins with little labeled data and generates these labels as part of the review process.

Although an active learning cycle can start with no data, prior knowledge accelerates initial training and avoids a *cold start*. Without prior knowledge, the model must rely on random screening. Even a small number of labeled documents improves early performance. In practice, reviewers often already know of some relevant documents, even if those items do not perfectly match the research question.

In a systematic review pipeline, the model improves with each labeling cycle, reaching sufficient accuracy to identify relevant documents earlier than random screening. This accelerated discovery leads to an increasingly sparse distribution of relevant documents, while many irrelevant documents remain. Once no new relevant items appear within a defined interval, screening can stop, reducing the manual workload.

Throughout this work, the expression 'the machine learning model', or simply, 'model', refers to the combined process of feature extraction and classification.

7.2 Methodology

7.2.1 Data

All simulations in this chapter use the `SYNERGY` dataset, described in detail in Chapter 5 and Appendix Table C.1. For the first study, the `Walker_2018` dataset (48,375 records) is used with a reduced number of simulation runs due to its computational demands. In the second study, this dataset is downsampled to 4,837 records using stratified sampling while preserving the original class distribution (before: label 0 - 0.984248, label 1 - 0.015752; after: label 0 - 0.984288, label 1 - 0.015712). This reduction allows

for efficient experimentation while maintaining representative label ratios.

7.2.2 Overview Simulation Design

We perform simulations on 26 pre-labeled datasets derived from existing systematic reviews. In the first simulation study, we run all permutations of the relevant record, classifier, feature extractor, and SYNERGY dataset. In the second simulation study, we increase the classifier and feature extractor pool to 92 combinations but remove the relevant records from the permutations.

$$N_{\text{sim, S1}} = 13_{\text{models}} \cdot n_{\text{relevant}}$$

$$N_{\text{sim, S2}} = 92_{\text{models}}$$

$$N_{\text{sim, total}} = \sum_{i=1}^{n_{\text{datasets}}} (N_{\text{sim, S1},i} + N_{\text{sim, S2},i})$$

The first simulation study is used to gauge the reliability and stability of the results in preparation for the second study, where we replace iterating over all relevant records with running a single simulation per combination per dataset. To combat the extra instability in this new format, we increase the amount of prior knowledge, as the increased prior knowledge will reduce the amount of performance fluctuation in the first cycles of the simulation. Based on the findings of Byrne et al. (2024), we set the prior knowledge to a level that ensures stability, using a number of records that will minimize early-cycle variability while maintaining a realistic screening scenario.

Our study aims to achieve a high degree of reproducibility by adopting an open-source approach and making all data and code openly accessible (Lombaers et al., 2024). The simulations are run on the open-source cloud platform Exoscale[2]. We developed a custom Docker image (J. J. Teijema, 2023a), which is available to the public. An in-depth explanation of the Docker image's functionality can be found on its GitHub[3]. The processing tasks within the Docker image are managed using a Kubernetes cluster[4].

7.2.3 Models

The classifiers in this study are trained during runtime, both for users using the software and for simulation studies. At each iteration of the active learning process, a new classifier is trained on the current set of labeled data, enabling adaptation to the dataset as labeling progresses. This approach is particularly suited to systematic reviews. In this context, which often involves frontier research, researchers are typically the first to construct the dataset as part of the review process. Therefore, no pre-existing labeled data is available for training classifiers beforehand. The strength

[2] exoscale.com

[3] Software available at github.com/jteijema/asreview-simulation-project

[4] kubernetes.io

of the active learning approach lies in its ability to dynamically adapt to the data as new labels become available during the screening process.

The methodology for feature extractor training varies. Simpler feature extractors, such as TF-IDF and Doc2Vec, can generate embeddings without pre-training the weights. In contrast, transformer-based extractors, like MiniLM and Sentence-BERT, are pre-trained and used without fine-tuning on a specific dataset, leveraging their general-purpose embeddings instead.

Hyperparameter optimization for the machine learning models was not performed in this study. Instead, optimized parameters were adopted directly from the original ASReview software package where available. For models not included in the ASReview software package, developer-recommended settings were used.

The feature extractors utilized in the first simulation study are TF-IDF, Doc2Vec, MiniLM, and Sentence-BERT (all-mpnet-base-v2). The classifiers selected are Logistic Regression, Naive Bayes, Random Forest, and Support Vector Machine. The reasoning behind the selection of these models for the first simulations is based on frequently used models in other simulation studies (chapter 3).

Note that Naive Bayes cannot use feature matrices containing negative vectors. As such, when feature extractors Doc2Vec, MiniLM, or all-mpnet-base-v2 are used, which produce negative vectors, Naive Bayes cannot be employed as a classifier. Therefore, the only viable pairing involving Naive Bayes is with TF-IDF, which limits the total number of models evaluated to 13.

Models: TF-IDF + Logistic Regression, TF-IDF + Naive Bayes, TF-IDF + Random Forest, TF-IDF + Support Vector Machine, Doc2Vec + Logistic Regression, Doc2Vec + Random Forest, Doc2Vec + Support Vector Machine, MiniLM + Logistic Regression, MiniLM + Random Forest, MiniLM + Support Vector Machine, Sentence-BERT + Logistic Regression, Sentence-BERT + Random Forest, and Sentence-BERT + Support Vector Machine.

For the second study, the scope is expanded to include additional models. MiniLM is replaced with larger models that share a similar architecture. In total, the second study evaluates 13 feature extractors and 8 classifiers. However, certain limitations arise due to the nature of the embeddings and classifiers:

- Neural networks cannot process sparse embeddings because the input layer would need to be excessively wide (e.g., matching the vocabulary size, which exceeds 40,000 dimensions).
- Naive Bayes cannot handle negative embeddings.

Taking these constraints into account, the total number of combinations evaluated is calculated as follows:

- 6 classifiers compatible with all embeddings $\times$ 13 feature extractors = 78 combinations

- 3 Naive Bayes classifiers compatible with positive embeddings only = 3 combinations
- 11 neural network simulations compatible with specific embeddings = 11 combinations

In total, this results in 92 unique combinations evaluated in the second study. A complete list of evaluated feature extractors is found in Appendix Table C.2 and classifiers in Table 7.1.

Name	**Description**	**Speed (Current Study)**	**Study**
Naive Bayes	Based on Bayes' theorem with independence assumptions between features.	Fast ($0.02s$)	1 & 2
Logistic Regression	Predicts probabilities for binary outcomes based on linear combinations of features.	Fast ($0.19s$)	1 & 2
k-nearest neighbors	Classifies based on the majority label among the k-nearest samples.	Fast ($0.62s$)	2
Random Forest	Ensemble of decision trees, improving prediction accuracy through averaging.	Medium ($1.04s$)	1 & 2
AdaBoost	Boosts the performance of decision trees through a focus on incorrectly classified instances.	Medium ($2.45s$)	2
Neural Network	2 layered, fully-connected. Learns complex patterns using two layers of interconnected nodes.	Slow ($5.07s$)	2
Support Vector Machine	Finds the hyperplane that best separates different classes in the feature space. Speed depends on dataset size.	Slow ($6.86s$)	1 & 2
XGBoost	Scalable optimized gradient boosting model.	Slow ($16.34s$)	2

Table 7.1 Classifiers used in the second phase of the simulations, along with classifier description and approximate speed.

Table C.2 and Table 7.1 categorize the processing speed of classifiers and feature extractors during their active learning cycles. Table C.2 categorizes text embedding speeds as 'Fast' (seconds), 'Medium' (minutes), or 'Slow' (hours) and details the features each method extracts.

Basic methods focus on word occurrence and frequency, while word co-occurrence

and context capture relationships between words. Semantic meaning emerges with sufficient context, reflecting deeper word significance. Syntax analyzes sentence structure, and attention dynamically weights word relevance, enabling focus on meaningful text parts. Language-agnosticism is noted for methods applicable across multiple languages. Together, these features provide tools for interpreting text across various dimensions.

Table 7.1 similarly assigns 'Fast' to processing times of less than one second per cycle, 'Medium' to less than three seconds, and 'Slow' to more than three seconds. These values were calculated from the results of this study. This analysis is based on Table 4.1 from chapter 4, which presents similar statistics. Comparisons of identical classifiers and feature extractors between the two studies yield consistent results, reinforcing the reliability of these findings.

7.2.4 Prior Knowledge

In our first simulation study, we conduct one simulation for each relevant record, together with 10 fixed irrelevant records, using the ARFI (All Relevant - Fixed Irrelevant) template from ASReview-Makita (chapter 6) to minimize inter-dataset variation. This standardized method ensures consistency across datasets and increases reproducibility for large-scale simulation studies. The total number of simulations conducted in study one is detailed in Table C.1.

In the second study, we use a prior knowledge set consisting of five relevant records and ten irrelevant ones, selected at random but kept constant across models. Increasing prior knowledge helps reduce variability caused by differences in how informative individual prior records are.

Using the MultiModel template from Makita, we establish a consistent simulation template that encompasses every permutation of classifiers, feature extractors, and datasets, while ensuring that the prior knowledge remains unchanged for each dataset.

7.2.5 Evaluation

The study yields two sets of results. The direct performance results of the simulations and the meta-analyses focus on the variability present within these findings.

7.2.5.1 Simulation Study 1 (Reliability)

Simulation study one first assesses inter-dataset variability, then intra-dataset variability, via the performance results between datasets, models, and prior knowledge settings. To evaluate this variability, we use the Loss across datasets and models. Simulation study one also introduces APD Heatmaps.

Recall curves Recall curves are a common method for visualizing simulation results, depicting the fraction of relevant records found versus the fraction of screened records. By stacking recall curves from multiple simulations, we can better assess active learning performance. In this study, $13 \times 25 = 325$ stacked recall curves from the first study are available on the persistent results website.

This paper highlights a selection of these stacked recall curves to illustrate examples of good, average, and poor performance, as well as the influence of prior knowledge. The figures include the *perfect performance* curve, representing the optimal scenario where all relevant records are identified before encountering any irrelevant ones. For datasets with a higher proportion of relevant records, this curve is naturally less steep.

Normalized Recall Regret The Normalized Recall Regret metric quantifies the overall performance of an active learning model by measuring how the recall curve is distributed between the optimal and the worst possible screening performance. Regret is commonly used to measure the difference between the actual performance and an ideal benchmark. Our contribution normalizes this value, resulting in a value between 0 and 1.

Unlike point-based metrics like WSS or Recall, the Normalized Recall Regret provides a holistic assessment by evaluating the area under the recall curve (AUC), and can therefore be treated as a loss function. It is computed as the difference between the optimal AUC and the actual AUC, divided by the difference between the optimal AUC and the worst AUC.

- Optimal AUC: This is the area under a *perfect recall curve*, where relevant records are identified as early as possible. Mathematically, it is computed as

$$Nx \times Ny - \frac{Ny \times (Ny - 1)}{2}$$

  where Nx is the total number of records, and Ny is the number of relevant records.

- Worst AUC: This represents the area under a worst-case recall curve, where all relevant records appear at the end of the screening process. This is calculated as

$$\frac{Ny \times (Ny + 1)}{2}$$

- Actual AUC: This is the area under the recall curve produced by the model during the screening process. It can be obtained by summing up the cumulative recall values for the labeled records.

$$\text{Normalized Recall Regret} = \frac{Ny \times \left(Nx - \frac{Ny-1}{2}\right) - \sum \text{Cumulative Recall}}{Ny \times (Nx - Ny)} \tag{1}$$

For simplicity and ease of interpretation, we refer to Normalized Recall Regret as *Loss* throughout the paper, as it quantifies the loss of recall. A loss value of 0 represents perfect performance, while a loss of 1 corresponds to the worst possible performance. A loss of 0.5 indicates that the model's performance is midway between these outer values. However, it does not necessarily imply random screening. While lower loss values generally indicate better performance, the interpretation of a specific loss score depends on how the recall is distributed throughout the screening process.

Variability We visualize the performance per dataset using the inter-dataset boxplot. Each box in this boxplot is a combined performance that includes all classifiers, feature extractors, and prior knowledge settings for a single dataset. This will provide a representation of the performance per dataset, giving insight into inter-dataset performance variation.

To visualize intra-dataset variability, the inter-dataset boxplot is split into 25 separate boxplots, one for each dataset (except Walker_2018, which has too few simulations for a fair comparison). Unlike the inter-dataset boxplot, where model performances were combined, these dataset-specific boxplots separate the 13 models into individual boxes. Within each boxplot, the performance range is determined solely by the selection of prior knowledge, allowing for an assessment of its impact on performance and a direct comparison between models.

Average Pair Distance Heatmap Recall curves do not provide insights into the specific discovery order of individual records. This limits their utility for cluster identification. Some stacked recall curves exhibit a distinct split shape due to unique discovery time groups, as illustrated in Figure 7.2. Here, the only variable leading to these different curves is the specific record used as prior knowledge. We hypothesize that these distinctive paths could be attributed to clusters of records that are highly interrelated, yet show little correlation to the rest of the dataset. In cases where the prior knowledge exists in one of these clusters, the recall graph is likely to trace the unique, separate curve.

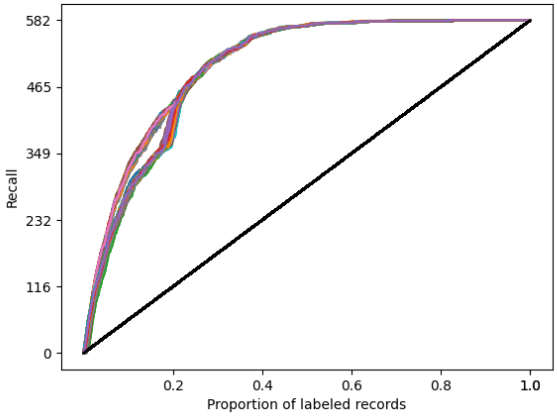


Figure 7.2 Hypothetical example recall curve with a clear indication for clustering records.

Using Time to Discovery, we generate visualizations in the form of heat maps. To create the heatmaps, we organize the data from the simulations into a three-dimensional array, indexed by record ID, Time to Discovery TD, and simulation number n. We measure the distance between discovery for each pair of records across all simulations, with the distance being the number of records between the discovery of a first and second record, defined as

$$d_{i,j} = |TD_i - TD_j| \tag{2}$$

where TD_i and TD_j are the Time to Discovery of the first and second records, respectively, and $d_{i,j}$ represents the absolute difference between TD_i and TD_j.

We compute the log-transformed average across simulations to create the Average Pair Distance (APD) array using

$$APD_{i,j} = \ln\left(\frac{1}{N}\sum_{n=1}^{N} d_{i,j,n}\right) \tag{3}$$

In this array, both axes correspond to record IDs i and j for all IDs, and each cell value denotes the average distance in the discovery sequence between two specific records. $APD_{i,j}$ denotes the log-transformed average pair distance between records i and j, N is the total number of simulations, and $d_{i,j,n}$ is the distance between records i and j for simulation n.

The data is then visualized as a color-coded heatmap. The APD heatmap can be sorted by the average discovery sequence to present a clear view of potential 'hotspots' - clusters of records that appear closely related in their discovery sequence, but relatively independent from the rest of the dataset.

7.2.5.2 Simulation Study 2 (Performance)

The second study evaluates the performance of 92 models across three key stages of the screening process. The analysis begins with ranking models based on Loss. Next, overall performance is assessed by examining effectiveness at different points along the WSS curve. Starting performance is measured to identify models that excel at retrieving relevant records early in the process (first 100 records), which is particularly important for mitigating the cold start problem in active learning. Finally, Last-to-Find performance is analyzed to determine how well models identify the most challenging records in the final stages.

Overall Performance Overall performance is evaluated by plotting Work Saved over Sampling values in increments from $WSS@10\%$ to $WSS@100\%$, capturing each model's trajectory throughout the screening process. While previous analyses focused on Loss, this evaluation measures WSS at predefined thresholds for a more complete assessment.

A heatmap of the top 20 performing models is presented, followed by a graph showing WSS across all models. If even more granular data is desired, we present an interactive persistent website [5]. A sample of this interactive website is given.

Starting Performance To assess which models perform best in the early stages of screening, the number of relevant records identified within the first 100 screened records (including prior knowledge) is measured. This benchmark is based on expert analysis indicating that approximately 100 records can be screened within an hour. This approach helps us pinpoint the models that are most suitable for active learning

[5] Live: `https://jteijema.github.io/synergy-simulations-website/models.html`, Persistent: `http://doi.org/10.5281/zenodo.13169790`

sessions where time is limited. Effectively, we count the number of relevant records with a Time to Discovery (TD) (Ferdinands et al., 2023) below 100 for each simulation.

A key challenge in active learning is the cold start problem, where models initially lack sufficient training data to make accurate relevance predictions. The starting performance metric helps assess how well different models overcome this limitation by effectively utilizing prior knowledge and quickly retrieving relevant records.

The theoretical lower bound for this metric is 5 relevant records, as each simulation begins with 5 relevant records provided as prior knowledge. The theoretical upper bound is given by:

$$\frac{1}{26}\sum_{d=1}^{26}(\min(r_d,(100-10))) = 50.84$$

where d represents the datasets and r_d represents the total number of relevant records for each dataset. Since 10 irrelevant records are included as prior knowledge, they are subtracted from the 100-record maximum. This calculation results in a theoretical upper limit of 51 records found.

Starting performance is particularly relevant for end-users under time constraints, such as those conducting rapid reviews or exploratory screening. While prior knowledge can help mitigate the cold start problem in active learning, its availability varies. When prior knowledge is limited, models with strong starting performance become important, as they are better able to retrieve relevant studies early in the process.

In the same time-sensitive scenarios, feature extractor computational speed also becomes a factor. Faster embeddings enable a quicker start, allowing for more screening within a limited time frame. Models that combine high starting performance with low computational cost are therefore the most suitable for time-constrained tasks.

Last-to-Find Performance The last-to-find section looks at the WSS@100% metric, which evaluates the work saved when *all* relevant records are found. This will identify models that are most effective at finding the final relevant records in the screening process. A bar chart ranks models based on WSS@100% performance values, with additional analysis on embedding calculation speed to illustrate the trade-offs between model complexity and computational cost.

For end-users of active learning in systematic review support, last-to-find performance is particularly relevant. This metric is critical because active learning aims to optimize the search process, stopping the screening when the stopping rule is triggered. Minimizing the number of missed records at this point is essential to avoid missing relevant records. The number of missed records depends, in effect, on the stopping rule and the model's performance in identifying the final relevant records (Bron, van der Heijden, Feelders, & Siebes, 2024). If a model excels at finding easily identifiable papers but struggles with more difficult ones, it may lead to a gap in discovery, leading to an overestimation of dataset sparsity. This can occur if the model is overly focused on easily classified records and lacks robustness against noise, potentially causing relevant studies to remain undetected.

Beyond ranking models by their ability to find difficult records, the analysis also

shows the trade-off between model performance and computational requirements. More complex models may improve retrieval but also require greater computational resources, which affects practical implementation.

7.2.6 Infrastructure

The simulations are performed using the simulation functionality of ASReview (simulation 1: v1.2, simulation 2: v1.5) (ASReview LAB developers, 2023), and facilitated through a Kubernetes cluster powered by 4 CPU-optimized processing nodes, amassing a total of 128 processing cores. The simulations are executed in a cloud infrastructure, and the results are stored in a persistent S3 bucket.

Our procedure is primarily a juggling act between managing the element sizes on the cluster and controlling simulation overhead. On one hand, packing all simulations into a single pod[6] is inadvisable due to the inherent strength of Kubernetes being its ability to distribute tasks across pods. Conversely, segmenting all simulations into individual pods leads to unmanageable processing overhead by necessitating a distinct simulation environment for each job. Although workload queue systems presented a viable option, this introduces significant complexity and is, based on literature, opted against (Romanov et al., 2024). We establish a system wherein a single pod is set up for a single template run. In the first simulation study, the template provided to the pod is the ARFI template, and for the second study, the MultiModel template.

Step-by-step We provide a detailed step-by-step guide of the simulation execution process:

- Job files[7] are created for each combination of variables under investigation.
- These job files are automatically dispatched to the cluster for processing.
- Each job is allocated a minimum of four CPU cores. If the cluster has sufficient memory resources available, the job is provided with the necessary processing power. If not, it is kept waiting in a jobs queue.
- The Docker image generates a Makita workflow specific to the dataset and models, as designated in the job.
- Following this, the cluster proceeds to run all simulations detailed in the Makita workflow using the ASReview simulation back-end and subsequently extracts the simulation metrics.
- These metrics are sent to an S3 storage bucket, providing a repository from which further analysis can be performed.

[6] A 'pod' in Kubernetes refers to a single instance of a running process or application. It is the smallest and simplest unit of deployment in Kubernetes, encapsulating one or more containers and associated resources. Pods enable the grouping and management of containers within a Kubernetes cluster.

[7] A 'job' in Kubernetes refers to a resource that manages the execution of a specific task or job within a cluster. It represents a one-time task that runs to completion, rather than continuously running like other Kubernetes resources. A job ensures that a pod completes the assigned task before considering the job as finished

7.2.7 Availability of Results and Replicability

The visualization results of this simulation study are made available as a GitHub Page (J. J. Teijema, 2024). The webpage features recall curves for each simulation conducted during the study, covering all datasets. It also covers the per-model performance for each classifier and feature extractor used in this study. This totals 325 stacked recall graphs for simulation study one, representing the 27001 recall curves collectively, and 21 stacked model performance graphs for study two. These visuals allow any researcher to dissect and analyze our study results in depth. The persistent repository includes the necessary instructions to run the website locally using the Python built-in web server functionality, allowing further users to easily re-host the website and its functionality in the event the website is no longer available.

To promote research persistence and replicability, all raw results from this study are made available on DataverseNL as J. J. Teijema (2023b). By providing these resources, we aim to encourage further exploration and utilization of our findings in the active learning for systematic reviews community.

7.3 Results

7.3.1 Simulation Study 1

Figure 7.3 reflects the performance results using the Loss for every individual dataset. The difference in mean performance is high between datasets, ranging from just marginally better than random sampling of records to near-flawless results. The range of performance within a single dataset changes from one dataset to another and indicates how differently various machine learning models perform. For some datasets, the large range means that some models perform much better or worse than others, while a small range suggests that all models perform similarly.

Following this, we examine the performance of each model and dataset in the first simulation study. The intra-dataset variability presented in Figure 7.4 showcases performance for all simulations in study 1. Here, a large box and whiskers indicate that the selected prior knowledge significantly impacts simulation performance, while a small range suggests a limited impact.

While the top-performing datasets (e.g., *Jall_2012, Leenaars_2019, others*) show very similar results across most models, there is a noticeable difference in performance between models for other datasets (e.g., *Appenzeller-Herzog_2019, Bos_2018, others*). This variability is more significant for some datasets than others. In most cases, the performance range for a single dataset and model is relatively narrow in the intra-dataset evaluation.

Figure 7.5 presents four examples from the 325 stacked recall curves generated in simulation study 1, along with two recall curves from study 2. The recall curves are compared to the ideal performance represented by the perfect line. The first, *Van de Schoot 2018 - Naive Bayes with TF-IDF*, demonstrates good performance. *Jeyaraman 2020 - Logistic Regression with TF-IDF* performs moderately, with a steep initial section but declining performance in later stages, suggesting a different model might be needed to optimize performance. *Moran 2021 - Logistic Regression with TF-IDF*

is an example of a dataset that is challenging to classify, likely due to factors beyond our experimental setup. Finally, *Oud 2018 - Random Forest with all-mpnet-base-v2* shows a recall curve where the performance is more influenced by the selection of prior knowledge than on average. The two recall curves from the second simulation study include every simulation run on a dataset in this study, combining the performance across all models.

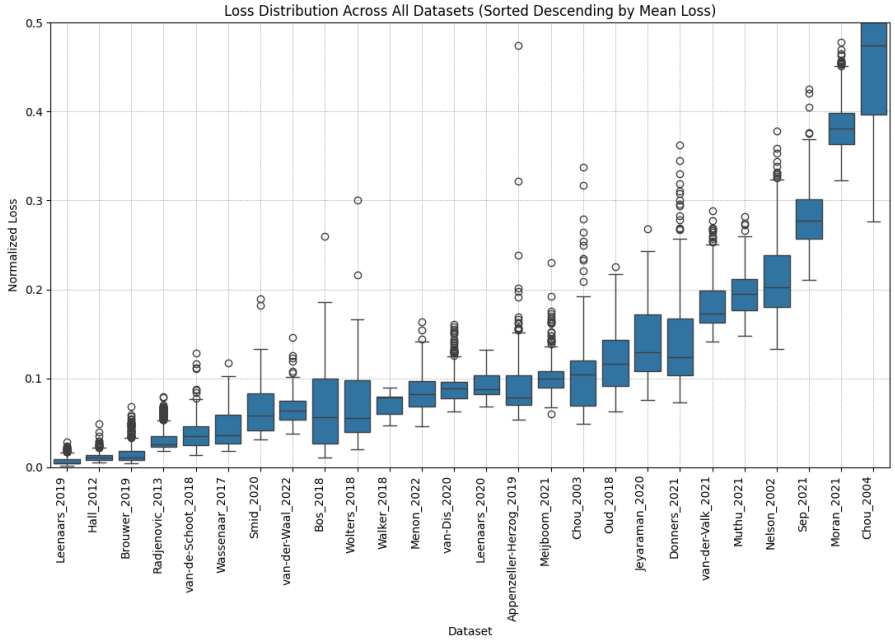
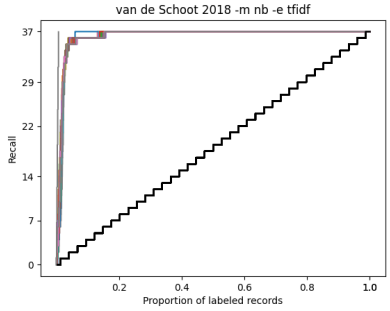


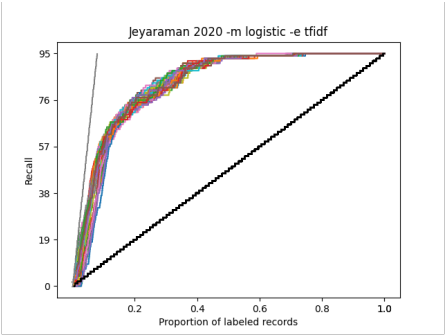
Figure 7.3 Illustration of the variability in Loss across (inter) datasets. Datasets are in ascending order, based on their mean Loss values, from best to worst. The plot aims to highlight the dispersion in model performance when applied to different datasets

Average Pair Distance Heatmap We observe patterns in the recall curve of the "Jeyarama_2020_-m_logistic_-e_all-mpnet-base-v2" dataset that suggest the presence of potential clusters. We select five records from the recall curve that lie closely together but are distinct from the other curves, identifying them as a potential cluster. The recall curve (1), cluster subset (2), and corresponding APD heatmap with the main cluster of records (3) and subcluster (4) are shown in Figure 7.6. When examining the documents from the subset, we find a significant overlap between the subcluster of identified records from the recall curve and the observed subcluster in the APD heatmap. These consistent observations across different datasets provide evidence for the existence and influence of clustering in record discovery.

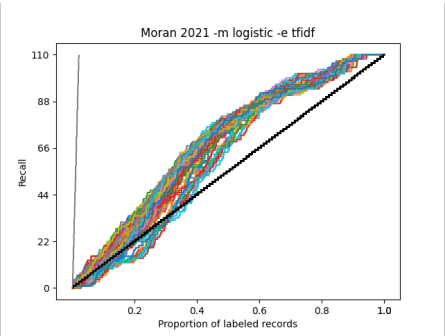
Figure 7.4 The intra-dataset variability for each model and dataset combination, represented by Loss. Ordered based on the order of Figure 7.3 and Mean Model Loss.



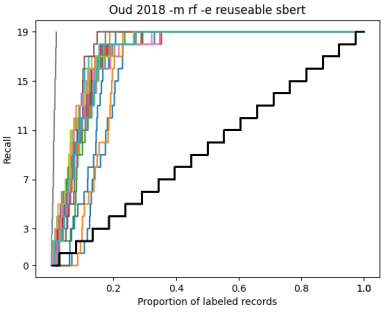
(a) Van de Schoot 2018 - Naive Bayes with TF-IDF



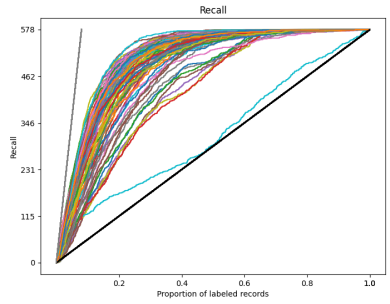
(b) Jeyaraman 2020 - Logistic Regression with TF-IDF



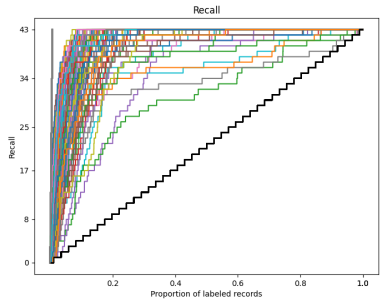
(c) Moran 2021 - Logistic Regression with TF-IDF



(d) Oud 2018 - Random Forest with all-mpnet-base-v2



(e) Leenaars 2020 - Every Model



(f) Radjenovic 2013 - Every Model

Figure 7.5 A selection of recall curves from over 29 thousand available. In panels a through d, each line represents a single simulation using a different record as prior knowledge. In panels e and f, each line represents a single simulation with a unique feature extractor-classifier combination.

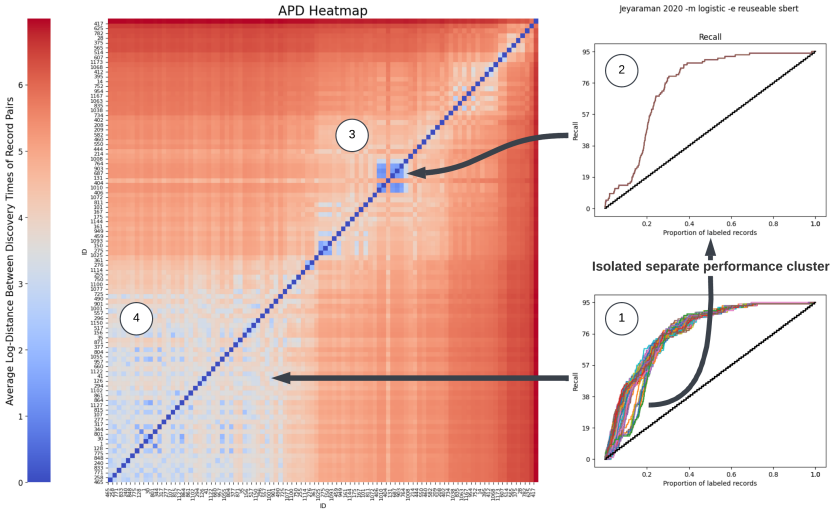


Figure 7.6 Combined representation of the Average Pair Distance (APD) heatmap (3, 4) and recall curves for the dataset Jeyaraman 2020 with the Logistic Regression model and all-mpnet-base-v2 feature extraction. The heatmap uses color coding to indicate pair distances, where the pair distance is defined as the log difference in discovery time between two records. Red signifies larger distances (greater differences in discovery time), while blue represents smaller distances. Adjacent to the heatmap, the lower right corner displays all recall curves for the dataset (1), while the upper right corner shows a subset of recall curves corresponding to a specific cluster (2).

7.3.2 Simulation Study 2

Ranking Model Performance Figure 7.7 presents the mean loss for each Classifier - feature extractor combination, with the standard error represented by the black error bars. Lower loss values indicate better model performance. From this plot, it becomes clear that some models perform better than others, but no decisive best model can be selected.

While some models perform better on average, the best-performing model varies significantly across datasets. As shown in Table 7.2, out of the 26 available datasets, 14 different classifier-feature extractor combinations achieved the lowest loss at least once. This suggests that no single model consistently outperforms others across all datasets. The most frequent top performer, "mxbai-embed-large-v1 transformer with Random Forest," was the best model in only 7 of 26 cases, while several other combinations appeared just once or twice.

Model Combination	**Top Performer Count**
Random Forest with mxbai-embed-large-v1	7
Random Forest with all-mpnet-base-v2 (hierarchical mean)	3
Naive Bayes with TF-IDF	2
Naive Bayes with scaled Doc2Vec	2
XGBoost with Onehot	2
Neural Network (2-layer) with all-mpnet-base-v2 (hierarchical mean)	2
Neural Network (2-layer) with mxbai-embed-large-v1	1
XGBoost with mxbai-embed-large-v1	1
XGBoost with TF-IDF	1
Naive Bayes with OneHot	1
Random Forest with scaled Doc2Vec	1
Random Forest with Onehot	1
Logistic Regression with all-mpnet-base-v2 (hierarchical mean)	1
Logistic Regression with all-mpnet-base-v2	1

Table 7.2 Frequency with which each model achieved the best performance across datasets.

Overall Performance Figure 7.8 illustrates the performance of all active learning models using Work Saved over Sampling. The plot shows that the WSS values generally increase as the simulation progresses, demonstrating the effectiveness of active learning in reducing the number of records that need to be manually screened. However, the WSS values consistently decrease towards the end as the models search for the last-to-find relevant records. The results shown in this figure represent the average performance across all datasets included in the simulation. As this figure obscures specific model performances on individual datasets, this plot is not highly

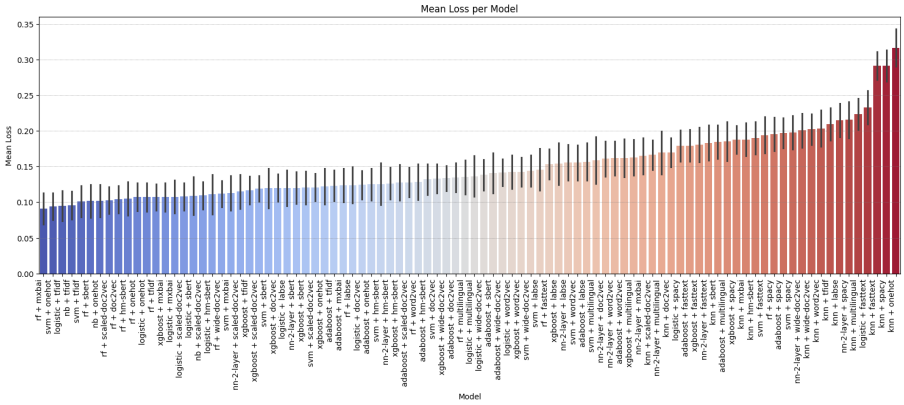


Figure 7.7 Mean Loss per Model with Standard Error .

informative for detailed analysis. For a more granular view, separate figures that display the performance of individual models are available on the interactive persistent website. Two of such figures, namely the model performance images for mxbai and logistic regression[8], are included in the image.

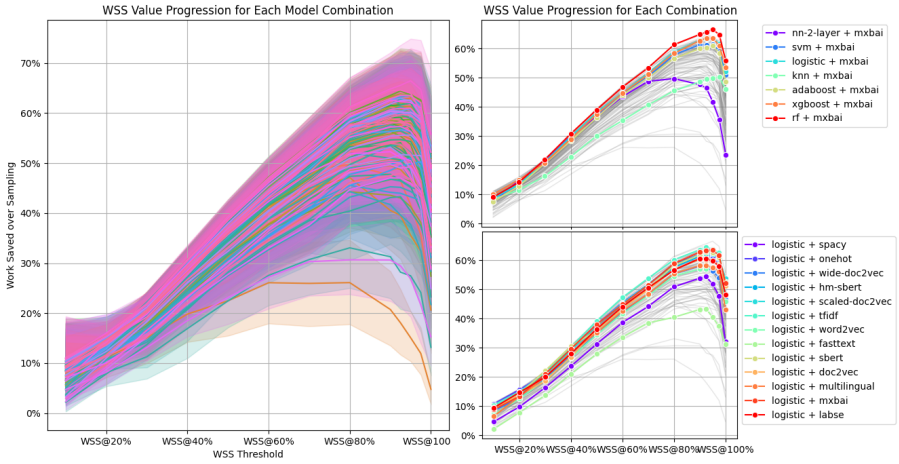


Figure 7.8 Plot showing the performance of all active learning models, in Work Saved over Sampling. Also showing the derivative plots only using mxbai or logistic regression.

Starting Performance Figure 7.9 shows the average number of relevant records found after screening 100 records in the simulation. The hue indicates the type of feature extractor used, in terms of the calculation speed of the embedding computation.

[8] https://jteijema.github.io/synergy-simulations-website/models.html#mxbai, https://jteijema.github.io/synergy-simulations-website/models.html#logistic

Along with the graph, Table 7.3 shows how often each model performs best in terms of finding the most relevant records within 100 screened. As the amount of relevant records found is less granular than the WSS score previously used, multiple models can tie for the top performance. Out of 92 models evaluated, 44 models reached the top-performing spot at least once. The table highlights models that achieved this distinction more than once.

Model Combination	**100 Record-Top Performer Count**
Random Forest with all-mpnet-base-v2 (hierarchical mean)	6
Random Forest with mxbai-embed-large-v1	5
Naive Bayes with Onehot	4
Support Vector Machine with Onehot	3
Naive Bayes with scaled Doc2Vec	2
Naive Bayes with TF-IDF	2
Random Forest with Doc2Vec	2
Random Forest with all-mpnet-base-v2 (head-only)	2
Logistic Regression with Onehot	2

Table 7.3 Frequency with which each model achieved the highest performance across the first 100 records. Only models that outperformed others on more than one occasion are included.

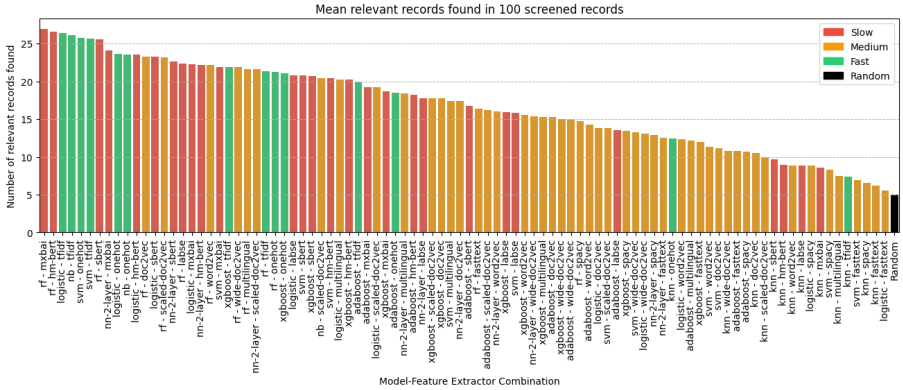


Figure 7.9 The average number of relevant records found per model in the first 100 screened records. The lower limit is 5 records (prior knowledge). The color indicates the feature extractor embedding speed.

Last-to-Find Performance In Figure 7.10, the performance of each model is shown using work saved over sampling after all relevant records have been found (WSS@100%). The (flat) *random sampling* bar represents the performance without using active learning, therefore its work saved is 0. The speed of the used feature extractor is again given by the hue of the bar.

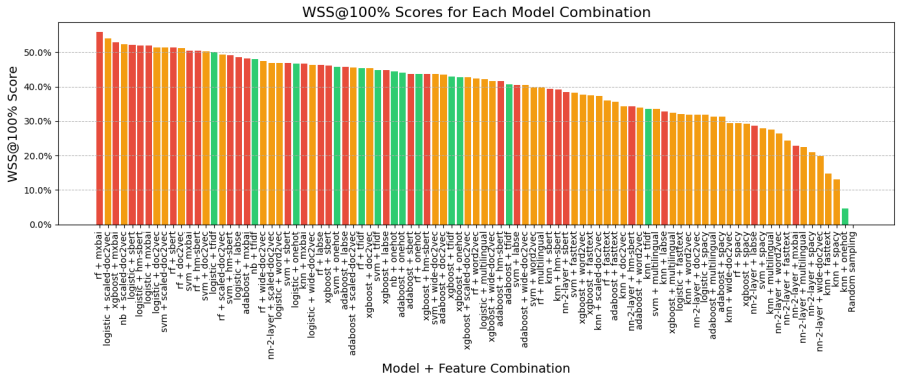


Figure 7.10 The mean WSS@100% values for each model, measured after every record has been found, ordered by work saved over the sampling. The color indicates the computation embedding speed of the feature extractor.

7.4 Discussion

This study aimed to analyze variability in simulation studies, both across datasets and within individual datasets, as well as to evaluate model performance across different screening phases. The simulation results confirm that active learning-based systematic review screening consistently outperforms random sample screening across all tested scenarios. Our work builds on the groundwork of many others, such as van de Schoot et al. (2021); Van Dinter et al. (2021), by conducting large-scale simulations that systematically assess 13 commonly used classifier-feature extraction combinations, along with 92 additional models. By incorporating a broader set of datasets and models than prior studies, our results provide a more comprehensive assessment of active learning in systematic review screening.

The performance differences between datasets are informative (Ferdinands, 2021; van de Schoot et al., 2021). Some datasets consistently yield low Loss scores (Hall, Beecham, Bowes, Gray, & Counsell, 2012; Leenaars et al., 2019), suggesting easier classification, while others produce higher Loss scores with low variation between models, indicating a more challenging task (Muthu & Ramakrishnan, 2021; Sep, Vellinga, Sarabdjitsingh, & Joëls, 2021). The most revealing datasets show significant differences in performance between models (Donners et al., 2021; Oud, Arntz, Hermens, Verhoef, & Kendall, 2018). While these datasets may provide insights into what model-specific advantages lead to better performance, those same datasets can also lead to bias when used exclusively for the comparison between models. Single or low-count dataset experiments risk overly optimistic or pessimistic outcomes due to dataset-specific biases; for robust conclusions, models should be validated across diverse datasets rather than limited samples.

Some datasets, such as Chou_2004 or Moran_2021, show negligible performance improvements compared to random screening, reflecting realistic situations where active learning may offer little benefit. However, the current study shows that even the weakest models generally detect enough of a pattern to slightly outperform

random sampling. If no classifier, regardless of its complexity, manages to identify such a pattern in a given dataset, it strongly suggests that the dataset itself lacks any exploitable signal.

The influence of specific dataset characteristics, such as size or topic, remains unclear. No significant correlation was found between identifiable dataset features and model preference. With a larger number of datasets, stronger statistical power might reveal such correlations if they exist, but 26 is insufficient to draw conclusions. This result also suggests that commonly used dataset descriptors may not capture the factors influencing model effectiveness.

The study investigated whether an optimal model consistently outperforms others across systematic review datasets. Results indicate that some models significantly outperform others, including:

- Random Forest with mxbai-embed-large-v1
- Random Forest with all-mpnet-base-v2 (hierarchical mean)
- Naïve Bayes with TF-IDF
- Naïve Bayes with scaled Doc2Vec
- XGBoost with One-hot encoding
- Neural Network (2-layer) with all-mpnet-base-v2 (hierarchical mean)

No model achieved superior performance across all datasets. The model with the lowest average loss is still outperformed in 19 out of 26 datasets. These findings confirm that no classifier-feature combination is universally optimal. Instead, the optimal model varies depending on the specific dataset, highlighting the need for a flexible, adaptable approach when creating active learning software.

For general performance, both simple and complex models rank among the top performers, indicating that either type can excel depending on the dataset. For starting performance, the best models identified on average 27 out of a theoretical maximum of 51 relevant records, within the first 100 records screened. Compared to general performance, simpler models tend to rank higher in starting performance. More complex models, which rely on a larger number of parameters and features, typically require more data to generate effective screening orders (chapter 4). Since starting performance is evaluated with limited data, simpler models tend to perform relatively better. However, some complex models also perform well, particularly those using pre-trained feature extractors. Models like *hm-bert* and *mxbai* are pre-trained, while others, like *Word2Vec* and *Doc2Vec*, are trained during simulation. Because the pre-trained models have already been exposed to large amounts of data, the general rule that complex models require more data remains valid; these models have simply already encountered more data.

For last-to-find records (Byrne et al., 2024; Ferdinands et al., 2023; Harmsen et al., 2024), advanced models significantly outperform simpler models due to their ability to detect contextual and semantic relationships. Additionally, the "last few difficult documents" phenomenon is consistently observed across datasets. While the trends

in starting and last-to-find performance align with expectations, it is valuable that our findings confirm them empirically.

Despite the widespread use of Support Vector Machines (SVM) in active learning (Ambert, Cohen, Burns, Boudreau, & Sonmez, 2013; Carey, Harte, & McCullagh, 2021; Przybyła et al., 2018; Yu, Carver, Rothermel, & Menzies, 2022), our results suggest that SVM can underperform compared to several other classifiers. This raises the question of whether its common use is justified or whether alternative classifiers should be considered more frequently. However, since SVM has shown strong results in prior studies, further research should investigate whether better hyperparameter tuning or architectural adjustments could improve its performance.

The Normalized Recall Regret metric provides a broader assessment of active learning performance than point-based metrics like Work Saved over Sampling ($WSS@X\%$ (Cohen et al., 2006)). While $WSS@X\%$ is widely used, there is no unified metric for evaluating systematic review screening models (O'Mara-Eves et al., 2015). Unlike $WSS@X\%$, which evaluates performance at a fixed recall level, our metric captures the overall effectiveness of a model across the entire screening process. This makes it useful for both general model evaluation and optimization. By treating regret as a Loss function, model performance can be compared more directly across datasets, supporting better model selection and pipeline optimization.

Recall graphs illustrate active learning performance but do not capture clustering dynamics. The current study introduces a new visualization method that reveals how records are discovered over time, providing deeper insight into dataset structure. Certain recall curves display distinct shapes. Visualizing the average discovery sequences in APD heatmaps highlights underlying clusters. The observed recall curve shapes correlated with these identified clusters. This indicates both the existence of clusters and the validity of the assumption that these can be identified by the shape of the recall curve.

The current study has limitations. While many model settings were explored, not all possible configurations were covered. Other variables, such as different samplers and balancers, were left unexplored. This study considers a model to be a combination of a feature extractor and a classifier, rather than each component separately. The interplay between the feature extractor and classifier within a model can influence overall performance for better or worse, and some combinations were not compatible. This analysis focuses exclusively on the results for each complete model combination to avoid the complexities and data incompleteness of dissecting the contributions of feature extractors and classifiers independently.

Another important aspect not covered is hyperparameter optimization. Many classifiers have tunable parameters that can significantly impact performance, and future research should explore whether certain models could achieve even better results with fine-tuned parameters.

Future studies should focus on identifying and analyzing more dataset characteristics, as this might lead to a better understanding of the relationship with model performance. Advanced feature extraction techniques that capture more complex lexical categories, alongside topic expert-driven dataset analysis, could help uncover

underlying patterns. This might improve performance predictions and lead to more informed model selection strategies.

There is a need to enhance the performance of underperforming datasets, as they offer the most room for improvement. An open question is whether or not these datasets are performing as well as possible, or if yet undiscovered patterns exist that could further improve classification outcomes. Regardless of whether improvement is possible, stabilizing their performance across multiple simulations and reducing variability is crucial to ensure consistent and reliable results. Researchers should also consider contributing new screening data to the `SYNERGY` dataset to make it even more relevant and broad-based.

A promising direction is to treat the time-to-discovery of records as time-to-event data. This would open up survival analysis, a well-established branch of statistics, for use in our framework. Applying preexisting tools like the Kaplan-Meier estimator and accounting for censored data (where some records remain undiscovered), would allow for deeper statistical analyses to compare discovery rates across models and datasets, uncovering factors that influence efficiency.

In the evolution of systematic review automation, Large Language Models (LLMs) present a promising candidate for enhancing classification tasks. Given their capabilities in natural language understanding, LLMs have the potential to (semi-)automate the classification process. A hybrid approach, combining active learning strategies with LLM-driven classification, could offer a balanced solution. In the current active learning pipeline, some dataset segments go unscreened when the dataset appears sufficiently sparse, and the stopping rule is reached (Bron et al., 2024). LLMs could address this gap by automatically screening these overlooked portions, while human experts focus on a subset of ambiguous or high-importance cases. This would facilitate a more efficient and reliable review process, enabling researchers to better manage large volumes of data. The integration of LLMs into the classification pipeline could therefore contribute significantly to the stability and accuracy of classification, warranting its exploration in future studies.

7.4.1 Recommendations

Our recommendations for end-users assume the use of an "average" dataset. Evaluating the unique characteristics of the datasets so they may lead to more tailored model recommendations falls beyond the scope of this study. We provide general guidance to support researchers in applying our approach to systematic reviews across various domains.

When screening in a **limited time** or for a limited number of records (in our experiments, 100 records or one hour of screening time), we recommend using a combination of either Naive Bayes or Logistic Regression with TF-IDF. This recommendation is based on the results shown in Table C.2, Table 7.3, and Figure 7.9. Although not the best performer, these models rank a very close third and fourth out of 92 models. The reason for this recommendation over the number one and two models is that these models are computationally lightweight. When computational time is a limiting factor, the time saved by using a faster model allows for screening more documents, leading to more data, which leads to a larger performance boost than

using a slower transformer to embed the dataset, given the similar performance levels. Another point to consider is the explainability factor. Even when computational time is not a concern, these less complex models offer a significantly more interpretable process compared to transformer models.

For the **last-to-find records**, screening often involves reviewing a larger number of documents. In such cases, more complex models tend to be more time-efficient. Notably, *Random Forest* with the *mxbai-embed-large-v1* embedding consistently performs significantly in identifying these difficult records, making it the recommended option. This recommendation also applies to switching models. If a model change is planned during the review process, this model is the preferred choice.

Finally, we recommend that platforms facilitating active learning remain open to the implementation of new machine learning algorithms as open-source projects. This study demonstrates that new approaches can improve performance. Given the rapid pace of these developments, the open extensibility of software supporting active learning is the most obvious and sustainable option.

7.4.1.1 User considerations

Users should consider the following when selecting their approach in performing a systematic review supported by active learning:

1. **Available time**: If time is limited and the focus is on screening efficiently, lightweight models such as Naive Bayes or Logistic Regression with TF-IDF are ideal. These models save computational time, enabling more documents to be screened in less time, without a significant drop in performance.
2. **Scope of the search**: If time is not a constraint and the goal is to either increase the scope of the search (e.g., retrieve more data from the database) or ensure high recall (e.g., increase the emphasis on finding all potentially relevant records), then complex models such as Random Forest with *mxbai-embed-large-v1* are recommended. This is especially suitable for users willing to invest more time in achieving exhaustive results.
3. **Optimal workflow**: To balance efficiency and thoroughness, users should follow the SAFE procedure (Boetje & van de Schoot, 2024). This work provides a workflow that ensures an evidence-based strategy for determining when to stop screening and switch between models if needed. The selected models for this procedure are those recommended in the previous section.

7.4.2 Conclusion

Empirical evidence is the foundation of any scientific discipline, especially in data science and machine learning. In a rapidly progressing field like active learning for systematic reviews, which is fundamentally empirical, it is crucial to base the adoption of new methodologies on robust, large-scale evidence.

This large-scale simulation study evaluated active learning strategies for systematic reviews, testing whether an optimal model consistently outperforms others across

multiple datasets. No such model was found. Instead, different models performed best at different stages of the review process and across different datasets.

These results highlight the importance of large-scale empirical evidence in systematic review simulations and set a higher standard for future research in this field.

7.4.3 Statements and Declarations

7.4.3.1 Acknowledgements

We would like to acknowledge the invaluable contributions of the software engineers at VSHN AG for their technical assistance. Their expertise assisted us in the establishment and maintenance of our computational infrastructure. Their commitment to maintaining high-quality infrastructure was instrumental in the successful execution of this large-scale simulation study.

We thank the EU Open Clouds for Research Environments Project (OCRE) project, our funding body, which provided the necessary resources and support that enabled us to carry out this study.

Chapter 8

Interlude: Model Selection Guideline

This chapter departs from the standard academic format to present a blog post aimed at ASReview users and lay readers of this dissertation. Included as a demonstration of research communication, the text translates technical complexity into practical model selection guidelines. Consequently, this work appears as an interlude rather than a formal peer-reviewed article. The material was further adapted into an educational video. A still from the video alongside a QR code for direct access is provided in Figure 8.1.

> Teijema, J. J. (2024, February 13). Navigating the maze of models in ASReview. ASReview.ai. Retrieved October 7, 2025, from *https://asreview.nl/blog/asreview-model-selection-guide/*
>
> Teijema, J. J. (2025). *Navigating the maze of models: ASReview Model Selection Guideline.* Zenodo. https://doi.org/10.5281/zenodo.17899247
>
> [ASReview TV]. (2024, June 5). Navigating the Maze of Models in ASReview [Video]. Youtube.com. https://www.youtube.com/watch?v=pQ1t60_RF0c

Figure 8.1 Still from the video *"Navigating the Maze of Models in ASReview"*, uploaded by ASReview TV on YouTube. Includes a QR code containing the URL of the video.

Navigating the Maze of Models in ASReview

The following work introduces key concepts such as feature extraction and classification, explains the implementation of models including TF-IDF, Doc2Vec, and SBERT, and outlines how these models interact with classifiers within an active learning pipeline. The post is designed to help researchers select suitable model combinations based on their dataset and review objectives. There's also a discussion on computational considerations and performance implications.

8.1 Introduction

Starting a systematic review can feel like navigating through a maze, with countless articles and endless decisions. Enter ASReview – your trusty AI tool. ASReview isn't just a one-trick pony; it's more like a Swiss Army knife. Equipped with a variety of models, designed to tackle every different type of dataset.

In this work, we're diving into these models. Every model has their special powers and quirks. From the classic predictability of traditional classifiers to the complex power of neural networks, we'll explore how each model can turn the task of literature screening into your own optimized research plan. Whether you're a seasoned researcher or just dipping your toes into the ocean of systematic reviews, join us through the powerful and diverse models available in ASReview.

8.2 What Are Models?

ASReview is a tool heavily dependent on machine learning algorithms. When you begin a new systematic review in ASReview, you have the opportunity to select which models you will use. ASReview, with its strong academic roots, emphasizes the values of openness and fairness. And being scientists, we don't limit you to a model we think is best for you; instead, we give you the freedom to choose. After all, you are the Oracle. But with this freedom comes a question: which combination of models will you select for your review?

ASReview uses Active Learning at its core. Active learning is a cycle in which you and an AI take turns doing what they do best to finish the screening phase of a systematic review as efficiently as possible. What you do better than anyone or anything is the labeling of literature for your study. What AI does best is efficiently and quickly going through tons and tons of data to provide you with the critical piece of data to

be labeled. Together, you make an incredible team! Want to know more about how active learning works? This work explains it in great detail.

So what exactly is AI? AI is a broad term, but we will let you in on a little secret — AI is just statistics on a large scale!

Don't tell anyone, though! By keeping this in mind, AI goes from being magic to something that can be understood. And that's exactly what we will do in this work. Explain the small elements that together make up the powerful AI pipeline.

The ASReview active learning pipeline is made up of five elements:

- A Feature Extractor algorithm
- A Classification algorithm
- A Query Strategy
- A Balancing strategy
- A stopping rule [1]

These elements are *algorithms*. Give an algorithm some input, and in return, it will give you some output based on the rules of the algorithm. By stringing these algorithms together, we create an AI pipeline that will help you in your systematic review screening. In this work, we will touch on feature extractors and classifiers.

8.3 Feature Extractors

Machines do not understand text in the way humans do. Machines are adept at processing numbers, but not so much at processing words or sentences. This limitation holds even for advanced models like the newest GPT models, which, despite their seemingly humanlike interaction, convert text into numerical representations before analyzing what it says. And since ASReview runs on a machine, we want to represent the text as numbers. Many rows of numbers. One row of numbers for each abstract, and we do this using *feature extractors*.

Feature extractors are a crucial component in the field of machine learning. At its core, a feature extractor is a tool that transforms text data into a numerical format, a process often referred to as *vectorization*. It takes *features* from the text and *extracts* them. The result is a *vector*, a row of numbers in a table.

So, what are features? Features are types of information. It can be the frequency of a word in a text, but also the location of a word in a sentence (position). And it can even be more complex; a feature can be the semantic meaning of a word, or the words a word references (attention). Any piece of information we can derive from an abstract using algorithms can be coded as a feature.

In the context of ASReview, feature extractors enable machine learning models to process and analyze vast amounts of text. From the features extracted from text,

[1] github.com/asreview/asreview/discussions/557

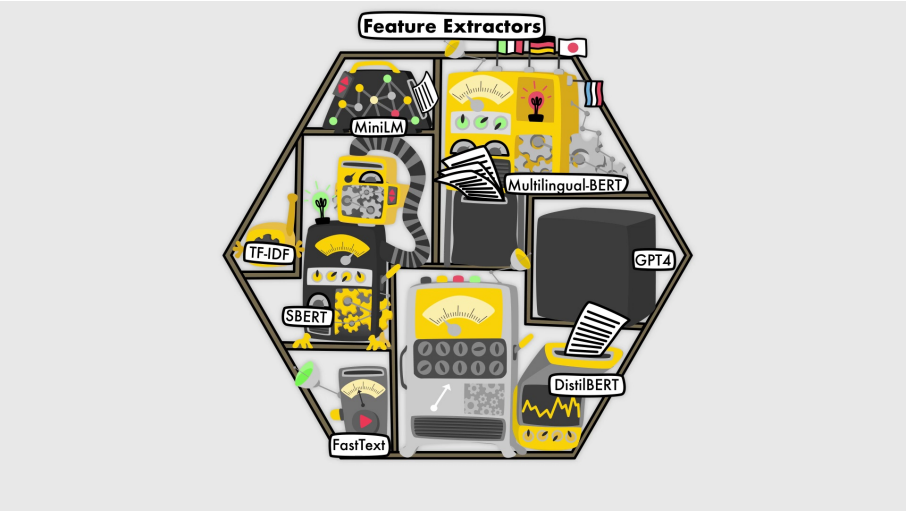


Figure 8.2 Different types of feature extractors.

Abstract_1: *You* *like* *cookies*.

becomes

	You	*like*	dogs	*cookies*
Abstract_1	1	1	0	1

Figure 8.3 Representation of an embedding

models can identify patterns, trends, and correlations, which might be indicative of the relevance or irrelevance of an article to a particular systematic review.

ASReview has different feature extractors available for you to select. Below is an (often simplified) overview of options, together with PROS and CONS for each model.

8.3.1 TF-IDF

TF-IDF (Sparck Jones, 1972), short for Term Frequency-Inverse Document Frequency, is a feature extraction technique used in text mining and information retrieval. It is a feature extractor available in ASReview. It operates on two key concepts:

- **Term Frequency (TF):** This measures how frequently a term occurs in your abstract.
- **Inverse Document Frequency (IDF):** This measures how often the term appears across all of your abstracts. IDF is calculated by dividing the total number of documents by the number of documents containing the term.

The TF-IDF value is obtained by multiplying these two figures. This results in a numeric value for a word, signifying its importance in an abstract relative to your dataset. By doing this for every word in every abstract, we effectively transform text into numbers, with each column representing a specific word and its corresponding TF-IDF score and each row representing an abstract, allowing for subsequent machine learning analysis.

PROS:

- **Lightweight**: TF-IDF has near-instant computation.
- **Compatibility**: TF-IDF produces no negative values, and will work with Naïve Bayes *(Naïve Bayes? You'll read about NB in the classifiers section!)*, while other feature extractors won't.
- **Simple**: Good in datasets with consistent terms for classification.
- **Interpretability**: TF-IDF is easily interpretable when evaluating the feature matrix.

CONS:

- **Loss of word order and context**: TF-IDF does not incorporate the order of words and their context into its representation. This can lead to a loss of meaning, especially in documents where phrasing and context are important. Without this information, do you know the difference between "not really" and "not really"? How about the difference between "bark" and "bark"?
- **Ignores Semantics**: TF-IDF doesn't put different words with the same meaning together. It is thus up to the classifier to reconnect similar words. *Identical* and *indistinguishable* are very different words for TF-IDF. Yet they are identical/indistinguishable for us.

What do the vectors look like for TF-IDF? Here, each column is a different word. The final table will have one column for each word found across all abstracts. Each value is calculated as (Term Frequency / Document Frequency).

	Word 1 (TF-IDF Score) **I**	Word 2 (TF-IDF Score) **like**	Word 3 (TF-IDF Score) **cookies**
Abstract 1	0.15	0.07	0.20
Abstract 2	0.00	0.12	0.18
Abstract 3	0.22	0.05	0.00

Figure 8.4 TF-IDF Vector Representation

8.3.2 Doc2Vec

Compared to TF-IDF, the Doc2Vec (Le & Mikolov, 2014) model is a lot more complex. Doc2Vec is a simple form of *neural network*. This means that instead of using an

algorithm made by a human to perform its logic, it has learned its logic by training on data first, and uses this learned logic to perform its task. Doc2Vec learns this logic in three steps:

1. During the initial setup, each document in your dataset is randomly assigned a vector. These vectors are not meaningful yet.
2. Then, the model is trained. Training works as follows: the model reads a few words from an abstract and tries to predict the next word. It does this by using the vectors of the surrounding words and the vector assigned to this specific document. The prediction is a probability distribution over all words in the vocabulary – essentially, a guess at which word comes next.
3. If the prediction is wrong, the model adjusts the vectors slightly, using a method called *backpropagation* [2]. Over many iterations, the predictions get better and better, as the vectors start capturing the essence of the words and documents – their meaning, usage, and context.
4. After sufficient training, the vectors stabilize. They don't change much anymore because the model has become good at prediction. These trained vectors are the output of your Doc2Vec model, which you will use for your classification.

> ***Neural networks.*** *The term neural network is indeed derived from the neurons one finds in our brains. However, they work, learn, and function quite differently. A neural network is modeled after a brain, like an airplane is modeled after a bird. The metaphor only works up to a point. The machine learning implementation of the brain has diverged significantly from how the brain works.*

The great thing about learning how a word is used in language is that we learn how similar words are used. And on top of that, it also learns in *what context* a word is used. Doc2Vec learns the meaning of words in a text by observing how they appear in relation to other words, essentially learning the associations and patterns from their usage. By doing this repeatedly across many contexts and documents, the model builds a *multi-dimensional space* where words with similar contexts and meanings are positioned closer to each other. And what is a *multi-dimensional space*? It's just rows and columns of numbers! The rows represent the abstracts, and the columns represent *dimensions of meaning*.

> ***Not One Meaning Per Dimension:*** *In the multi-dimensional space of Doc2Vec, each dimension does not correspond to an interpretable meaning or concept. Instead, each dimension represents a feature learned from the text data. Sadly, these features are often abstract and not easily interpretable. Too bad!*

This *multi-dimensional space* makes Doc2Vec particularly powerful for tasks like document similarity, where you want to understand how closely related different documents are based on their content.

[2] https://towardsdatascience.com/understanding-backpropagation-algorithm

PROS:

- **Contextual and Semantic Understanding**: Doc2Vec is very good at understanding the context and semantics of words within documents. It captures the nuanced relationships between words based on their usage in different contexts, offering a richer representation of text data.
- **Preservation of Order**: Unlike simpler models like TF-IDF, the representation that Doc2Vec makes uses the order of words. This helps in understanding the narrative or the flow of ideas in a document, which can be crucial for text analysis where the specific ordering of words has a lot of impact (think of legal documentation, for example, or a medical patient history).
- **No outside bias**: Doc2Vec is trained only on your data, and therefore cannot include bias from outside of your dataset.

CONS:

- **Computational Intensity**: Doc2Vec is computationally demanding, particularly with large datasets. After all, it is learning a language from scratch! Keep this in mind when selecting Doc2Vec as a feature extractor, as this might pose challenges in terms of processing time and resource requirements.
- **Hyperparameter Sensitivity**: The performance of Doc2Vec can be sensitive to the choice of hyperparameters (the gears and knobs to optimize the learning capacity of the model). While ASReview's hyperparameters are verified extensively with countless simulations, it is challenging to determine the optimal settings for Doc2Vec that would suit every user's specific review scenario.
- **Low Explainability**: The dimensions in the vector representations produced by Doc2Vec are abstract and not easily interpretable. This might be a concern in scientific applications where the quality of the feature extractor is important.
- **Single Language:** Doc2Vec will not work for datasets with multiple languages.

What do the vectors look like for Doc2Vec? Here, each column is an abstract concept dimension as extracted from the documents. The vectors will have as many dimensions as set via the **vector_size** hyperparameter. For ASReview's Doc2Vec model, there are 40 of these dimensions.

	Concept 1 (Vector Value)	Concept … (Vector Value)	Concept 40 (Vector Value)
Abstract 1	1.24	-0.33	0.75
Abstract 2	0.87	0.45	-1.05
Abstract 3	-0.56	0.67	0.32

Figure 8.5 Doc2Vec Vector Representation

8.3.3 SBERT

SBERT (Reimers & Gurevych, 2019), short for Sentence-BERT, is a *pre-trained* BERT model (**B**idirectional **E**ncoder **R**epresentations from **T**ransformers), that's specialized in evaluating sentences. It uses a transformer architecture, and *yes*, this is the same architecture that powers the latest GPT models! SBERT is designed to create embeddings for entire paragraphs, making it very useful for semantic similarity analysis.

> ***Embeddings:*** *Embeddings are the same as vectors, which are simply rows of numbers. However, embedding is the conventional term when dealing with transformers. Confusing, we know...*

Just like Doc2Vec, SBERT builds a *multi-dimensional space* where dimensions represent meanings. Sadly, explaining exactly how transformers work is outside of the scope of this work. Out of the three feature extractor models we've seen, SBERT is the only one that inherently understands sentence context *before training*. It comes pre-packaged with the ability to read and speak before you've used it. Because it can be pre-trained, there are many specialized versions of SBERT trained for specific contexts. Specialized models exist that are trained on only scientific literature, or even models that are trained on many different languages at the same time, enabling them to process multilingual datasets. Examples such as SciBert (Beltagy, Lo, & Cohan, 2019), LaBSE (Feng, Yang, Cer, Arivazhagan, & Wang, 2022), or BERTje (de Vries et al., 2019) show the versatility of the BERT model

The pre-trained model that ASReview uses is called "all-mpnet-base-v2" [3]. This model was tested and was found to perform well in ASReview. It was trained on over 1 *billion* sentences, including but not limited to most of Reddit from 2015 to 2018, Yahoo Answers, Simple Wikipedia, the Semantic Scholar Open Research Corpus, and much more [4].

All-mpnet-base-v2 maps your abstracts to 768 different dimensions. The Doc2Vec model used in ASReview has only 40 dimensions. However, the dimensions for Doc2Vec are derived from your own dataset, while the dimensions for SBERT are predetermined during its training.

PROS:

- **Unchallenged Superior Theoretical Performance**: Transformers are widely regarded as having the best theoretical performance among all feature extractors. Its advanced understanding of semantics and context makes it extremely good at interpreting text.
- **Multilingual and Domain-Specific Models**: SBERT's versatility is further enhanced by its range of models that cater to different languages (den Boer et al., 2024) and specialized domains, offering tailored solutions for diverse research needs.

[3] https://huggingface.co/sentence-transformers/all-mpnet-base-v2
[4] https://huggingface.co/sentence-transformers/all-mpnet-base-v2#training-data

CONS:

- **Computational and Memory Intensity**: Preparing SBERT for use in your project can be time-consuming. Expect a long preprocessing time and a large RAM requirement. For processing times, check out J. J. Teijema, Hofstee, et al. (2023).
- **Complexity and Opacity**: Like Doc2Vec, SBERT's internal mechanisms are complex. This complexity often makes it challenging to decipher why the model makes specific decisions or interpretations.
- **Variable Performance Across Datasets**: Although SBERT excels in understanding contexts and semantics like no other model, in practice, we've seen that for ASReview, not all datasets benefit from this. Think, is your inclusion criterion only derivable from semantics and contexts? It's important to consider whether a model like TF-IDF might suffice.
- **Potential Bias**: Since SBERT is pre-trained on existing data, there's a risk that it may inherit biases present in its training data (Jentzsch & Turan, 2022). Consider this aspect for your study, as it might impact the objectivity and validity of your findings.

What does the embedding look like for SBERT? Where for Doc2Vec the dimensions are derived from the dataset you trained on, for SBERT, the columns are dimensions that were developed and set during the pre-training of this transformer model.

	Semantic Dimension 1 (Vector Value)	Semantic Dimension ... (Vector Value)	Semantic Dimension 768 (Vector Value)
Abstract 1	0.82	-0.47	0.29
Abstract 2	-0.35	0.91	-0.63
Abstract 3	0.10	0.55	0.75

Figure 8.6 SBERT Vector Representation

8.3.4 Visually representing the feature extractors

Using our newfound knowledge of feature extractors, we can create embeddings for sentences. For the following demonstration, we use TF-IDF, Doc2Vec, and SBERT to create embeddings for the following sentences:

```
["I love cookies", "I love cake", "I hate cookies", "I hate cake", "I like
desserts", "My cat's name is Meowzers"]
```

Using a technique called PCA, we can reduce these highly dimensional rows of data into two dimensions. Useful for explaining embeddings, because we can visualize two dimensions as a scatter plot! While simplified, you can interpret points on our scatter plot as relatedness. Points close together are similar, and points far away are dissimilar.

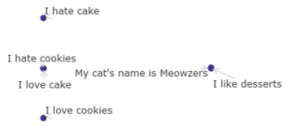
TF-IDF	Doc2Vec	SBERT
		
TF-IDF only takes into account the frequency of word appearances. We can see that the words *hate* and *love* are close together, just like *cake* and *cookies*. For TF-IDF, these are important words. They don't appear in a lot of documents (low IDF) but do appear in some documents (high TF). On the other side, we see that *I like desserts* is very far removed from the rest, even though it is semantically very related. TF-IDF sees that *I* appears in a lot of documents, so that's not important. And while *like* and *love* might seem related to you, for TF-IDF these are totally different words.	For Doc2Vec, we can see that every sentence aside from *My cat's name is Meowzers* is close together. Even though *I like desserts* and *I love cake* are made up of almost entirely different words, Doc2Vec has learned that the sentence structure and context are very similar for those words, and thus the sentences are related. What we also see is that it could not deduct that *I hate cookies* and *I love cookies* are very different in meaning. It did not have enough context, and thus found these sentences to be very similar, even though for us, they're not.	For SBERT, we see the sentences are strongly clustered on meaning. We see that *I love cake* and *I like desserts* are nearly at the same location, whereas in TF-IDF they were on opposite sides. SBERT has learned language from a lot of sources before its use today and knows inherently that *liking* and *loving* are similar in meaning. If you ask the model to guess whether the word *love* is more appropriate for *cake* than *hate*, it will have a strong opinion. This bias is near unavoidable to this type of model.

Figure 8.7 Captured Features per Feature Extractor

8.4 Classifiers

After processing text into a numerical format using feature extractors, the next step for ASReview is classification. Classifiers are algorithms that sort data into categories, based on data found in those categories (labeled data). The key to good classifiers is the ability to accurately find what features make each class unique. In the case of ASReview, there are two classes. Relevant, and Irrelevant.

Normally, in machine learning, you would start your classification after collecting *labeled data*. Labeled data is data where each data point has its own label. For ASReview, this would be the combination of *text* and its label, *relevance*. Using this data, the machine learning algorithm can predict decisions on *unlabeled data*.

However, doing a systematic review introduces a unique challenge. The research you are doing is unique! It is very likely that nobody before you had the idea to select data for your exact research question. Unlike typical machine learning scenarios, where you start with a substantial amount of labeled data, in systematic reviews, you likely start with limited labeled data or no labeled data at all. This is where the concept of active learning becomes important.

The challenge for models in ASReview lies in being efficient in both the early and later stages of active learning. In the early stages, when labeled data is limited, the model needs to be robust enough to make accurate predictions with very little information. This is important because the early predictions significantly boost the performance of the whole screening phase. If the model performs poorly at this stage, it might be that it takes a long time for the active learning cycle to kick into gear.

Then, for each piece of data that you label, the model adapts and refines its predictions. In the later stages of the active learning cycle, the challenge shifts to efficiently combing through the increasingly sparse dataset to identify any remaining relevant articles, which are often few, scattered, and hard to find. The classifier now needs to switch from making generalized predictions on very little data to making very accurate predictions on much, much data.

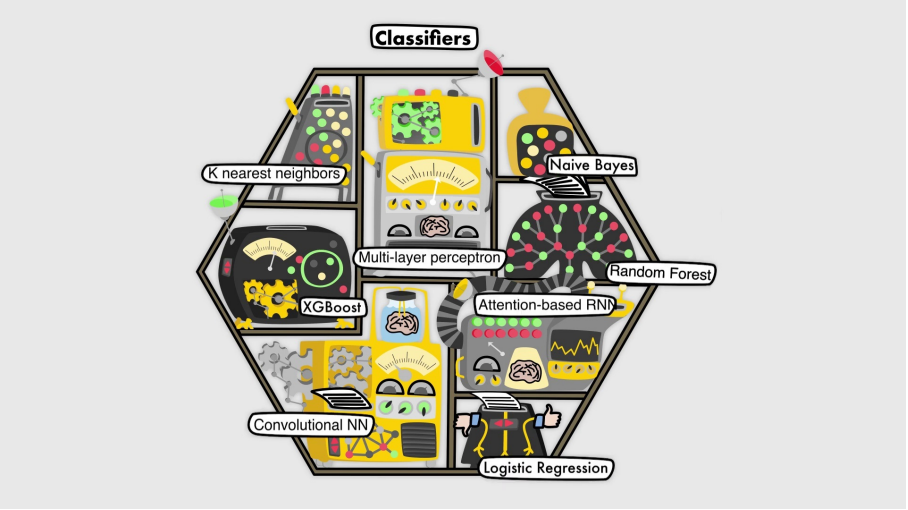


Figure 8.8 Different types of classifiers.

Below, we discuss five types of classifiers used in ASReview, providing a basic understanding of how each works.

Naive Bayes Classifier (classifiers.NaiveBayesClassifier: nb) The Naive Bayes classifier is based on Bayes' Theorem, which deals with probability. This classifier assumes that the presence of a particular feature in a class is unrelated to the presence of any other feature. It's like determining the likelihood of enjoying a movie based on liking the genre, regardless of the movie's director or cast. Naive Bayes is particularly known for its simplicity and speed.

Random Forest Classifier (classifiers.RandomForestClassifier: rf) Imagine a forest where each tree gives a vote on whether an article is relevant or not. The Random Forest classifier builds multiple decision trees and merges their results to get more accurate and stable predictions. Each tree in the 'forest' considers a random subset of features and makes a decision. The majority vote among all trees determines the final classification. This method is effective because it reduces the risk of overfitting (fitting too closely to a particular set of data).

Support Vector Machine Classifier (classifiers.SVMClassifier: svm) Support Vector Machine (SVM) classifiers find the best boundary that separates different categories of data. Imagine plotting all your data on a graph, and the SVM

finds a line (or hyperplane in multiple dimensions) that best divides and categorizes your data points. It's particularly effective in high-dimensional spaces.

Logistic Regression Classifier (classifiers.LogisticClassifier: logistic) Despite its name, Logistic Regression is a classification method, not a regression method. It estimates the probability that a given input point belongs to a certain class. It is a method that borrows heavily from the field of statistics.

Neural Network Classifier (classifiers.NN2LayerClassifier: nn-2-layer) A *fully-connected 2-layer dense neural network*. In a fully connected neural network, each node in one layer is connected to every node in the next layer. The network is trained by letting it make predictions on the labeled data. If the prediction is wrong, it adjusts the values of the neurons. If the predictions are right, we leave them as is. That way, the neurons naturally get better at predictions. These networks are capable of learning complex patterns but require more data and computational power compared to other classifiers.

8.4.1 Processing times

We all have places to be, tasks to accomplish, and deadlines to meet. Even though ASReview will already save you incredible amounts of time (averages range from 67% to 92%) (van de Schoot et al., 2021), understanding and managing the processing times you will find in ASReview is worth thinking about.

Feature extractors are unavoidably computationally intensive. It involves analyzing and converting each piece of text into a numerical format that machine learning algorithms can understand. However, the silver lining is that this is a one-time process. Once the text has been converted into embeddings, these numerical representations can be used repeatedly without the need for recalculating them. This initial investment in processing time pays off in the long run as it improves the foundation for all active learning cycles.

> **Timing:** The following indications are based on J. J. Teijema, Hofstee, et al. (2023), one of our simulation studies, performed in ASReview, using a dataset containing 46,376 records.

From quickest to slowest embedding calculation time:

1. TF-IDF – times ranged from 13 to 23 seconds
2. Doc2Vec – times ranged from 15 to 17 minutes
3. SBERT – times ranged from 6 to 7 hours

In contrast to feature extractors, classifiers in ASReview are repeatedly calculated, once per active learning cycle. In each cycle of active learning, the system builds the classifier from the ground up, and then asks for new predictions. The processing time for classifiers can vary based on the complexity of the algorithm, the size of the dataset, and the computational resources available.

1. **nb**: The fastest model around, ideal for fast computations ±0.03 seconds per cycle
2. **logistic**: Low processing cost, scales well with increasing data size. ±0.10 seconds per cycle
3. **rf**: Medium resource-intensiveness, involves building multiple decision trees. ±1.5 seconds per cycle
4. **nn-2-layer**: Processing-intensive due to its multi-layered architecture. ±2.4 seconds per cycle
5. **svm**: Complexity increases with data size and feature count, leading to higher processing demands for larger datasets. ±9 seconds per cycle

Note how each of these classifiers takes under 10 seconds to process. If you, the labeler, take more than this processing time to assess and label an abstract, ASReview will wait for you to finish before restarting the active learning cycle, and the calculation time of the classifier will have no impact on ASReview. Only when you take less time, ASReview will skip an active learning cycle and continue with the previous iteration of the classifier. Even then, this will have little impact on the performance of ASReview.

8.5 Feature Extractors and Classifier Interactions

Considering what we've touched on in this blogpost, your final question might be: *So, what model is best for me?* Ultimately, it is impossible to provide you with a *best* model combination. Only you know the situation of your research question and environment. Therefore, you are the best judge of models!

Models can have complex interactions. Some classifiers will more efficiently utilize the embeddings from certain feature extractors than others. Some classifiers can only use embeddings with positive numbers (which is why Naïve Bayes only works with TF-IDF). Some classifiers will require huge amounts of memory when using certain feature extractors. Therefore, we provide you with certain predefined combinations in ASReview, selected and verified by us to work well, based on many, many simulations.

The combinations we selected for you in ASReview are:

- ELAS Ultra - Rapid and excellent-performing model for most use cases.
  - ELAS_u3 - TF-IDF + Naive Bayes
  - ELAS_u4 - TF-IDF (with bigrams) + SVM
- ELAS Multilingual - Designed for multilingual datasets.
  - ELAS_l2 - multilingual-e5-large (similar to SBERT) + SVM
- ELAS Heavy - Focuses on the semantic understanding of text.
  - ELAS_h3 - mxbai-embed-large-v1 (similar to SBERT) + SVM

Plugins: *If you're interested in more models for ASReview, why not install asreview Dory* [5] *(developers, 2025b), or even create your own?* [6]

[5] https://github.com/asreview/asreview-dory
[6] https://github.com/asreview/template-extension-new-model

ASReview offers a wide range of models for conducting systematic reviews, each with its unique strengths and limitations. The choice between feature extractors like TF-IDF, Doc2Vec, and SBERT, and classifiers such as Naïve Bayes, Random Forest, and Logistic Regression, depends on your dataset and research question. Ultimately, your knowledge of the subject matter and the specific demands of your systematic review will allow you to make the right choice.

Part IV

Deployment

Chapter 9

External validation of machine learning hyperparameters for systematic review screening prioritization

This chapter is based on a technical report for the European Alliance of Associations for Rheumatology. It constitutes a near-exact copy of the original text and was adapted solely for formatting and internal cross-referencing within this dissertation.

> Teijema, J. J., Westerbeek, E., Van der Kuil, T., Bischof, L., van de Schoot, R., External validation of machine learning hyperparameters for systematic review screening prioritization, *on behalf of the European Alliance of Associations for Rheumatology.*

This study evaluates model configurations for accelerating the screening phase of systematic reviews, focusing on the validation of hyperparameters in a newly optimized active learning framework. Forty-five curated datasets containing 202,177 scientific studies, of which 1.5% were included in systematic reviews, were used to assess model performance and sensitivity thresholds. Simulations were conducted using three model configurations, applying time-to-event analysis to track recall progress over screening effort.

Results show that the lightweight `elas_u4` model provides great performance while requiring significantly less computational power than `elas_h3`. In most cases, screening either one-fourth of the total records or observing a consecutive streak of 10% irrelevant records is already sufficient to identify 95% of records. However, the relationship between screening fraction and irrelevant streaks suggests that combined metrics could further reduce screening effort without compromising recall.

9.1 Introduction

ASReview 2.0 (de Bruin et al., 2025) optimized its hyperparameters using the Synergy dataset (chapter 5). Although this dataset is large enough to suggest generalizability, external validation remains necessary. To assess how well these optimized settings transfer to other contexts, this study performs an external validation using an independent collection of datasets. This work represents the first large-scale validation of the ASReview 2.0 hyperparameter optimization beyond the Synergy dataset and its accompanying publications.

Usage of ASReview can be simulated to evaluate performance of settings, models (J. J. Teijema, Ribeiro, et al., 2025), and recall sensitivity heuristics based on historical data (Boetje & van de Schoot, 2024; Bron et al., 2024), (chapter 3). In this report, simulations are conducted to validate the performance of ASReview when applied to new datasets.

The target of this study is to identify metrics (Ferdinands et al., 2023) and model settings under which the system achieves a recall of 95% in at least 95% of cases. This sensitivity target is intentionally conservative, reflecting standards commonly applied in medical contexts, such as those used by EULAR, where missing relevant studies carries substantial risk. In doing so, this paper serves as an external validation and replication of the model configurations and hyperparameter optimization performed in the ASReview 2.0 development work, now evaluated on an independent collection of datasets. The paper describes the steps taken to run the simulations, reports the resulting performance metrics, and assesses whether the optimized settings meet this predefined sensitivity threshold.

9.2 Methodology

9.2.1 Data

A total of 49 datasets were received from EULAR (Bischof, 2025). Each dataset corresponds to a previously conducted systematic review and includes titles, abstracts, and binary full-text inclusion labels.

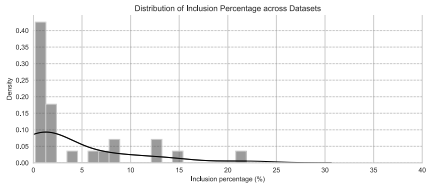
The datasets were cleaned and assessed for eligibility based on a set of criteria. Duplicates introduced *after* the screening processes, particularly those processed using Rayyan, which produces noisy outputs, were removed. Special attention was given to duplicates with conflicting labels; where two identical records had differing inclusion labels, these were reconciled manually.

Records with inclusion labels but missing abstracts were excluded. These entries were considered to introduce noise and do not reflect realistic screening conditions. Datasets were excluded if the number of included studies was insufficient for meaningful simulation (fewer than four inclusions).

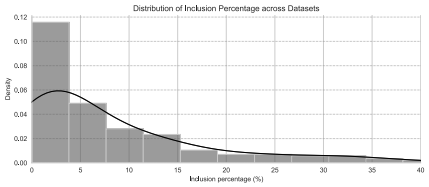
Four datasets were excluded from the simulations: two due to having too few included records, one lacking abstracts, and one that did not follow a standard systematic review structure. After these exclusions, 45 datasets remained eligible for analysis, also available on OSF.

The retained datasets contained an average of 4,493 records (SD = 2,851), with an average of 67 included studies (SD = 46). In total, the dataset contains 202,177 scientific studies, with 3,036 included records.

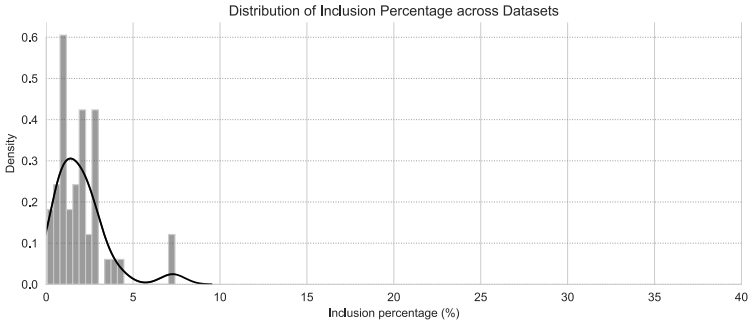
The distribution of inclusion ratios in the EULAR dataset is noticeably lower than in the datasets from chapter 3 and chapter 5. The other collections contain reviews with broader and generally higher inclusion ratios, whereas the EULAR reviews cluster toward the lower end of the range, as shown in Figure 9.1.



(a) SYNERGY dataset (26 reviews).



(b) Reviews from chapter 3 (224 reviews).



(c) Distribution of inclusion ratios in the EULAR dataset (45 reviews).

Figure 9.1 Comparison of inclusion ratios across datasets. The top panels show the distributions of inclusion ratios for the SYNERGY dataset and the scoping review collection from chapter 3. The bottom panel shows the EULAR dataset, where inclusion ratios are concentrated at the lower end of the range, indicating that substantially fewer records tend to be included relative to the number screened.

9.2.2 Data Splitting and Stratified Sampling

The European Alliance of Associations for Rheumatology (EULAR[1]) is preparing a guideline on the use of ASReview (van de Schoot et al., 2021), an active learning tool designed to accelerate the screening phase of systematic reviews. To support this effort, EULAR provides several datasets for analysis.

To establish the relationship between screening effort and the predefined sensitivity targets, the datasets were divided into training and validation subsets. Metrics and corresponding thresholds (e.g., the proportion of records that must be screened to

[1] https://www.eular.org/

reach a recall of 95%) were derived from the training data and subsequently validated on the held-out subsets to assess the generalizability of the threshold.

A 90-10 test-validation split using stratified sampling was used. Stratification was performed on the proportion of relevant records in the dataset. To create strata, quantile binning was applied, initially targeting five bins. Missing values in the stratification variable were imputed using the median.

Balance between the training and test subsets was assessed using the standardized mean difference (SMD), calculated with the pooled standard deviation. For $q = 5$ bins on the 'ratio' variable, the resulting split achieved excellent balance, with an SMD below 0.02 (Table 9.1).

Variable	Train Mean	Test Mean	Train SD	Test SD	SMD
Ratio of relevant records	0.01972	0.019533	0.016119	0.010856	0.012236

Table 9.1 Balance check results for stratified sampling on the `ratio` variable, using five quantile bins. An SMD below 0.05 indicates good covariate balance between the training and test subsets.

9.2.3 Simulation Procedure and Metrics

Simulations were executed using ASReview Makita (Chapter 6) version 1.1 (ASReview LAB developers, 2025), which allows for structured, reproducible runs across multiple datasets and model configurations. Rather than using all available models in ASReview, the selection was limited to three configurations: `h3`, `u3`, and `u4`, available in ASReview 2.0 and ASReview Dory (developers, 2025b) version 1.1.1. These were chosen based on results from Study 2 of chapter 7. These model combinations are identified as optimal for the many different datasets of Synergy (chapter 5). The `u3` configuration replicates the default model combination from ASReview v1, while `u4` represents the updated "Ultra" configuration with refined hyperparameters. The `h3` configuration belongs to the "Heavy" series, optimized for higher computational capacity and potentially improved stability on complex datasets.

For each dataset and model combination, three simulation runs were performed. Each run has different prior knowledge, to negate the variability introduced by the prior knowledge selection. The choice for three simulation runs is derived from chapter 7, on the variability of simulation results. Resulting metrics were averaged across these runs and merged with dataset descriptors for downstream analysis.

In addition to standard simulation metrics, time-to-event analysis was applied. Here, the event of interest is reaching 95% recall, and the time variable is the number of records screened. A benefit is that our use case has no censored events, as all simulations eventually reach the target recall, making the evaluation straightforward. Additionally, because there are no censored events, the Kaplan–Meier estimators can be interpreted as the empirical probability (ECDF) that a run has hit the target by fraction s.

The certainty curve (Kaplan–Meier curve) provides a visualization of the fraction of datasets in which the 95% recall threshold has been reached by a given screening percentage. Curves are plotted both overall and per model, with 95% confidence intervals shown as shaded bands.

We compute the minimum safe streak length, defined as the longest sequence of irrelevant records encountered before reaching 95% recall:

$$S = \max\left(r_1 - 1, \max_{1 \leqslant i < T}(r_{i+1} - r_i - 1)\right) + 1$$

In the formula, r_i denotes the screening rank (position) of the i-th relevant record, and T is the total number of relevant records needed to reach the 95% recall threshold. The expression $(r_{i+1} - r_i - 1)$ represents the number of consecutive irrelevant records between two relevant ones. The maximum of these values, plus one, defines S, the minimum safe streak length.

This provides the minimal streak length of irrelevant records observed at the point where the sensitivity target is reached. Simulation outcomes are visualized using Kaplan–Meier plots, box-plots, and density graphs to support guideline development.

To further explore how dataset composition influences model performance, datasets were binned by their inclusion ratio to create a split between high- and low-inclusion datasets. Although both key performance metrics (screening fraction and minimum safe streak) are normalized as percentages of the total dataset size, they are not normalized for inclusion ratio. Because datasets with higher inclusion ratios naturally require screening more records before reaching saturation, their performance curves tend to taper off more slowly compared to sparse datasets. As a result, performance characteristics can differ between these groups. To examine this effect, datasets were divided into high- and low-inclusion ratio bins (50 - 50 split), and Kaplan–Meier curves were compared between them. This exploration is not for direct application, as inclusion ratios are unknown in real-world screening, but it provides insight into how dataset composition influences performance and stopping rule behavior in historical data.

9.2.4 Reporting Outcomes

To assess when the predefined sensitivity target of 95% recall in 95% of cases is reached, three outcome metrics are extracted from the simulations. These metrics are formalized in this section and serve as the basis for evaluating model performance against the target.

First, the certainty curve is visualised using Kaplan–Meier plots. This curve shows the fraction of simulations that have reached 95% recall as a function of the fraction of records screened. A vertical reference line is used to mark the screening percentage at which 95% of simulations achieve the recall target. The plots also include 95% pointwise confidence intervals, calculated using Greenwood's formula, providing an estimate of uncertainty around the cumulative probability of reaching the recall threshold. These visualizations offer a direct probabilistic estimate of the screening

effort required to meet the sensitivity target.

Second, the minimum safe streak of irrelevant records is computed per simulation. This value represents the longest run of irrelevant records encountered before reaching 95% recall. Expressing this value as a percentage of the dataset size ensures comparability across datasets. This metric can inform stopping rules that terminate screening after a sufficiently long sequence of non-relevant records.

Third, model-specific performance is reported, allowing users to make informed choices about which configuration to use in practice. This includes comparisons of screening effort and performance variation between models.

These outcome metrics can be interpreted individually or in combination to examine when the sensitivity target is reached. The results presented here provide empirical input for future guideline development within EULAR.

9.3 Results

9.3.1 Certainty Curves

The Kaplan–Meier estimator curves show the cumulative probability of reaching the 95% recall threshold across increasing screening fractions. Figure 9.2 presents these curves per model, with vertical reference lines indicating the fraction at which 95% certainty is achieved. These results show 95% recall efficiency between model configurations.

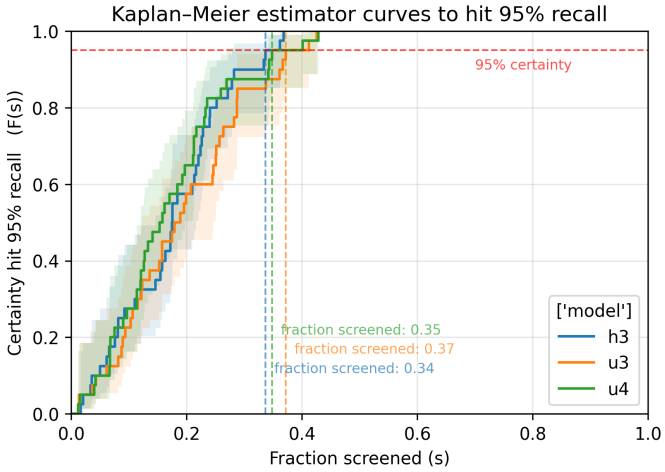


Figure 9.2 Kaplan–Meier estimator curves for reaching 95% recall, grouped by model. The shaded areas represent 95% pointwise confidence intervals computed using Greenwood's formula. Vertical dashed lines indicate the screening fraction at which 95% certainty is reached.

Performance curves are also shown after binning datasets into low and high inclusion

ratio groups (Figure 9.3). These curves are not useful for practical application, but provide insight into the effect of the inclusion ratio.

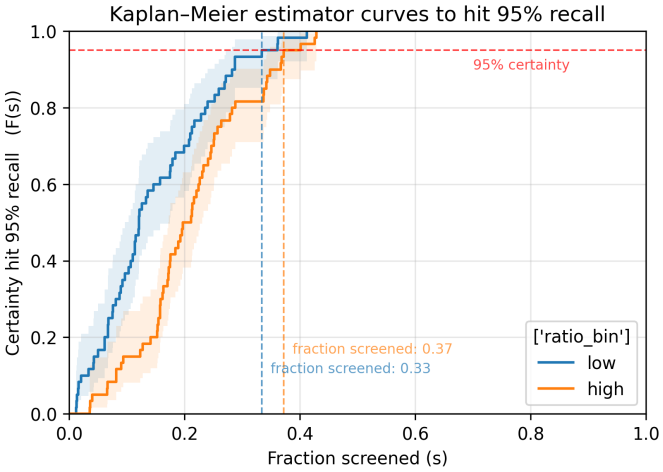


Figure 9.3 Kaplan–Meier estimator curves binned by inclusion ratio (low vs. high).

9.3.2 Minimum Safe Streak

The minimum safe streak of irrelevant records, expressed as a percentage of dataset length, is summarized per model in Figure 9.4. This shows the longest run of irrelevant records observed before hitting 95% recall. The 95th percentile is highlighted. An additional density view of the same metric is provided in Figure 9.5.

9.3.3 Model Comparison

Overall model performance is compared using loss, with loss being the Normalized Recall Regret from 7.2.5.1. Figure 9.6 shows the loss distribution across datasets. Figure 9.7 compares the screening fractions required to reach 90%, 95%, and 100% recall for each model. Filled dots indicate the median, empty dots indicate outliers.

9.3.4 Model performance

Among the tested model configurations, the `u4` model shows the strongest overall performance, achieving the lowest average loss across all datasets, as shown in Figure 9.6. However, the `h3` model displays slightly faster certainty in reaching 95% recall in the Kaplan–Meier curve, and has fewer outlier datasets. The difference between `u4` and `h3` lies within the margin of error and is therefore not statistically significant (Figure 9.2). In contrast, the `u3` model performs markedly worse, with performance metrics falling outside the confidence intervals of both `h3` and `u4`, although it still performs significantly better than random sampling.

Based on these results, `u4` is recommended as the default model configuration for future systematic reviews, as it offers great performance and requires significantly

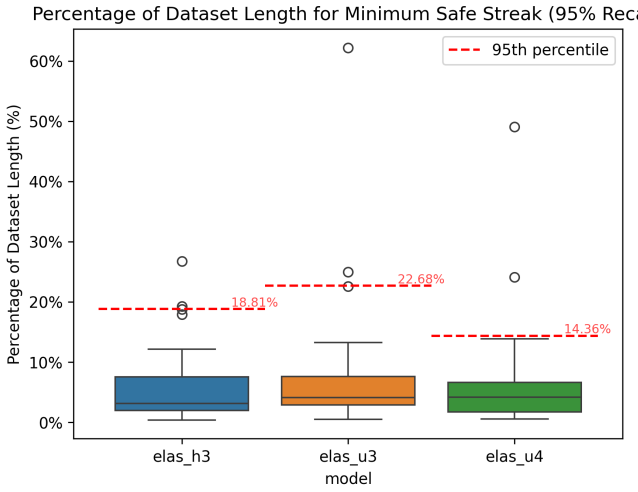


Figure 9.4 Minimum safe streak length (95% recall) as a percentage of dataset length, by model. Dashed lines indicate the 95th percentile threshold.

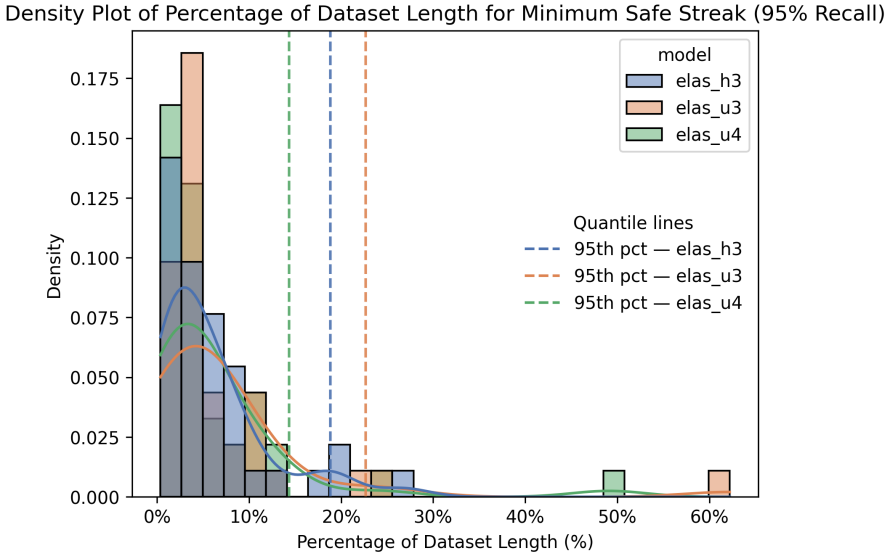


Figure 9.5 Distribution of minimum safe streak length as a percentage of dataset length.

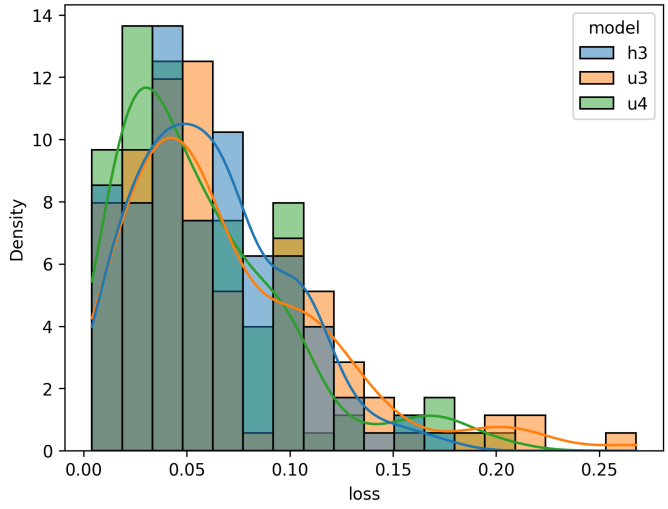


Figure 9.6 Distribution of loss across all datasets, stratified by model. Loss is defined as (1 − recall) at a fixed screening threshold.

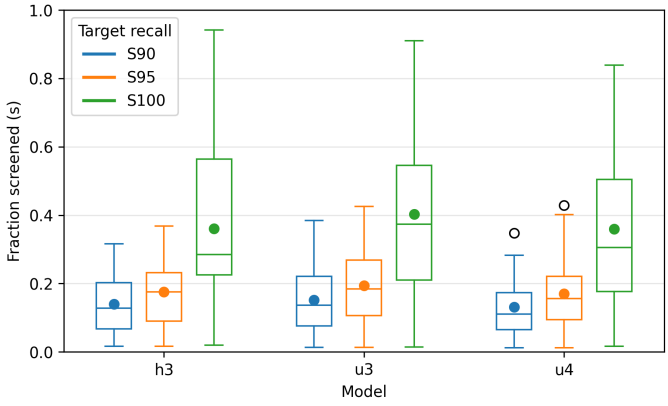


Figure 9.7 Boxplots showing the screening fraction required to hit 90%, 95%, and 100% recall, grouped by model. Dots indicate the median per configuration.

less computational power than h3. However, h3 does show improved performance on difficult datasets, having fewer extreme outliers. The further identification of minimal metrics corresponding to the 95% sensitivity target is based on the u4 and h3 models.

A sensitivity of 95% is an appropriate operational target, albeit highly conservative. Performance degrades notably when increasing to 100% recall, due to the diminishing returns associated with the final few relevant records, as seen in Figure 9.7. Meanwhile, the difference between 90% and 95% recall is relatively small. This validates a 95% recall in 95% of cases as a suitable target.

9.3.5 Observed Metrics Corresponding to Target Sensitivity

This analysis identifies the minimal metric values required to achieve a sensitivity of 95% recall in 95% of cases across the test datasets. The results provide empirical boundaries that could guide future stopping rule development. With forty datasets in the test set, this allows for a maximum of two datasets not reaching the 95% recall criterion. The reported values, therefore, represent the minimal observed conditions under which the 95% sensitivity level is maintained across the collection. The thresholds are the lowest metric combinations that satisfy this requirement.

Two key metrics are reported: the fraction of records screened to reach 95% recall and the minimum safe streak of consecutive irrelevant records, both expressed as percentages of the total dataset.

For the u4 model, the 95% recall threshold is reached after screening 34.8% of the dataset when using only a fraction-based condition, and after observing 13.9% consecutive irrelevant records when using only a streak-based condition. For the h3 model, 95% recall is reached after screening 33.7% of the dataset, and after observing 18.8% consecutive irrelevant records.

In practice, screening around 25% of the dataset or encountering a consecutive streak of 10% irrelevant records is already sufficient to reach the 95% recall target for 85% of cases. However, to ensure maximum conservativeness and to safeguard against rare edge cases, particularly in medical applications, more conservative figures are used.

As shown in Figure 9.3, datasets with a low inclusion ratio typically reach 95% recall earlier than those with high ratios. Low-inclusion datasets tend to align with fraction-based thresholds, while high-inclusion datasets often require longer screening and exhibit higher streak-based values. This relationship suggests that both metrics capture complementary aspects of model performance.

Following the solitary metrics, we give combined thresholds using both fraction and streak metrics. For the u4 model, 95% recall is achieved when screening 26.0% of the dataset and observing a consecutive irrelevant streak of 9.8%, or alternatively 23.6% screened with an 11.4% irrelevant streak. For the h3 model, the corresponding values are 28.0% screened and a 2.9% irrelevant streak.

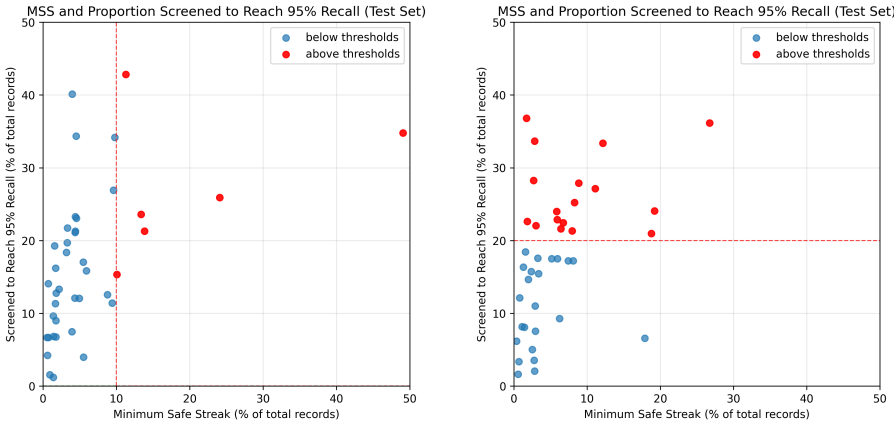
A screening phase should only terminate when both criteria are satisfied. The two conditions are independent and operate in parallel, meaning that the streak of irrelevant records may begin and end before, during, or after the absolute screening

threshold is reached.

The metric values are summarized in Table 9.2. Their relationship is visualized in Figure 9.8, which shows the minimum safe streak length against the proportion screened for both models, with red points indicating datasets outside the identified metric boundaries.

Model	Fraction-based-only (% screened)	Streak-only (% irrelevant)	Combined thresholds (%)
u4	34.8	13.9	26.0 screened + 9.8 streak
u4 (alt.)	–	–	23.6 screened + 11.4 streak
h3	33.7	18.8	28.0 screened + 2.9 streak

Table 9.2 Minimal observed metric values corresponding to 95% recall in 95% of cases across datasets. Both the fraction-based-only and streak-only conditions show the screening or streak length required when used independently. Combined thresholds represent empirically derived metric pairs achieving the same sensitivity. All values are expressed as percentages of the total dataset size.



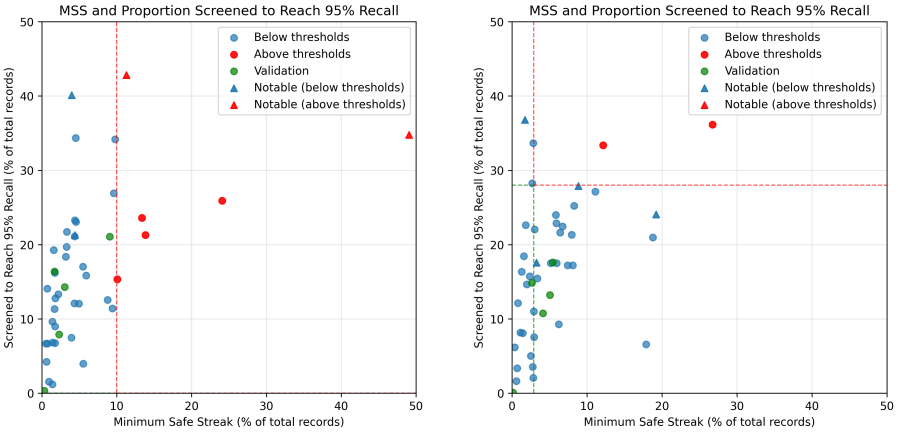
(a) u4 model **(b)** h3 model

Figure 9.8 Minimum safe streak length versus proportion screened to reach 95% recall for the u4 and h3 models on the test set. Red points indicate datasets outside the combined metric thresholds.

9.3.6 Validation

The reported metrics were validated using the held-out validation set to assess their performance on unseen data. The same evaluation criteria were applied: achieving 95% recall and satisfying both components of the rule.

As shown in Figure 9.9, all datasets in the validation set meet these criteria. Across



(a) u4 model **(b)** h3 model

Figure 9.9 Minimum safe streak length and the proportion of records screened to reach 95% recall. This figure includes the hold-out set, as well as 4 notable datasets referenced in the discussion.

both the test and validation sets, this corresponds to a sensitivity of 95.6% with respect to the 95% recall target.

9.4 Discussion

An observation from the analysis of minimum safe streak distributions (Figure 9.4) is that the `h3` model shows no outliers exceeding 30% of the dataset length, whereas both `u3` and `u4` do. These outliers correspond to four datasets from the same systematic review, which consistently underperform across all models (notable datasets in Figure 9.9). Three of them are the most extreme cases in the distribution for `u4` and substantially influence the sensitivity thresholds.

This finding should not be interpreted as a reflection on dataset quality or review methodology. Instead, it likely reflects particularly complex inclusion criteria that are difficult to model algorithmically, an established limitation of certain active learning systems (see section 10.2). The datasets themselves remain valid and valuable for analysis.

Interestingly, the more complex `h3` model performs more robustly on these difficult cases, suggesting potential advantages when facing challenging inclusion criteria. This reduced difficulty of outlier datasets was observed in the original hyperparameter training, de Bruin et al. (2025), too. As we have no practical method to identify such cases beforehand, this observation can only be used as an indication for future work: developing predictors of task complexity to inform adaptive model selection in systematic review automation.

9.4.1 Limitations

Overall, the validation confirms that the proposed two-part metric thresholds generalize well to this dataset. It is important to emphasize that this validation applies specifically to the EULAR dataset collection used in this study. It should not be assumed to extend automatically to other domains, research topics, or screening conditions. Differences in study design, inclusion complexity, or terminology distribution may influence model behavior and the effectiveness of the stopping criteria. The reported thresholds are based on empirical points derived from the observed data. The results reflect the minimal observed screening fractions and streak lengths at which the predefined sensitivity target was reached, rather than representing the full distribution of performance across datasets.

The effectiveness of a stopping rule based on historical data is conditional on the completeness of the dataset. Even if screening is optimally efficient, relevant studies may still be missing if the initial search strategy is too narrow (van de Schoot et al., 2025). That study shows that combining traditional database searches with full-text and AI-based methods can help uncover additional relevant papers that would otherwise be missed.

The metrics used in this work are related to the dataset size. While this approach works well across most datasets, it may become less practical for extremely large datasets. In such cases, the absolute number of records corresponding to the same percentage threshold may result in unnecessarily long screening sessions. Although these very large datasets are uncommon (see section 3.3), future research should examine at what scale the current rule begins to lose efficiency and whether additional adjustments are needed for projects involving unusually large datasets.

9.5 Conclusion

This study validated the hyperparameters optimized in ASReview 2.0 on an independent dataset collection provided by EULAR. The results show that the `u4` model, originally optimized on the Synergy dataset, also performs best on the EULAR datasets, despite differences in topic, scope, and screening context. This indicates that the optimized model generalizes well beyond its original training environment.

The `h3` model performs slightly less efficiently overall but shows greater robustness in complex or atypical datasets. This suggests that while `u4` is the preferred configuration for most screening scenarios, `h3` may offer advantages in cases with more challenging inclusion criteria.

In addition to validating model performance, this study identified the metric values corresponding to the predefined sensitivity target of 95% recall in 95% of datasets. This target is intentionally conservative, allowing at most one dataset out of twenty to fall short, reflecting the focus on completeness in medical evidence synthesis. The metrics were confirmed on the validation set and shown to hold for unseen data. Combining multiple metrics leads to efficient attainment of the target sensitivity and can guide future development of historical data-based stopping rules.

9.6 Open Materials and Reproducibility

All tools, simulations, and analysis materials used in this work are open source and publicly available, ensuring full transparency and reproducibility.

The analyses were performed using the ASReview ecosystem:

- ASReview LAB: `https://github.com/asreview/asreview`
- ASReview Makita (for automated simulation workflows): `https://github.com/asreview/asreview-makita`
- ASReview Dory (model repository for the `h3` model): `https://github.com/asreview/asreview-dory`

The datasets provided by EULAR, including metadata and descriptors, are available on the Open Science Framework (OSF):

> Bischof, L. (2025, September 29). ASReviewSLR - EULAR Working Group on AI-Driven Screening. Retrieved from osf.io/t8knj

All simulation materials are available on DataverseNL:

> Teijema, J. J. (2025). Replication Data for: Technical report on the usage of active learning for systematic review screening phase acceleration, on behalf of the European Alliance of Associations for Rheumatology. DataverseNL. Retrieved from https://doi.org/10.34894/YSXMVV doi: 10.34894/YSXMVV

This repository includes:

- All simulation project files generated with ASReview Makita.
- Metric tables, model performance summaries, and dataset descriptors.
- All figures from this report, as well as per-dataset and per-model performance figures, are included here.
- All scripts used for analysis, data processing, and figure generation.

Together, these materials provide complete reproducibility of all results presented in this work. This setup enables full reproducibility of the simulations and analysis.

Part V

Interpretation

Chapter 10

CLARIFY: Concept Level Active-Learning Ranker Interpreter for Systematic Reviews

This chapter presents a paper accepted to BNAIC 2025, the 37th Benelux Conference on Artificial Intelligence. The content was adapted for formatting purposes and internal cross-referencing. Descriptions of the dataset were omitted to avoid redundancy with chapter 5.

Teijema, J. J., Bagheri, A., & van de Schoot, R. (2025). CLARIFY: Concept-Level Active-learning Ranker Interpreter For sYstematic reviews. *Accepted for the 37th Benelux Conference on Artificial Intelligence and the 34th Belgian-Dutch Conference on Machine Learning (BNAIC/BeNeLearn 2025)*., Namur, Belgium.

Transformer-based ranking models have recently advanced active-learning tools for accelerating systematic reviews. However, the internal criteria they use to rank documents are opaque, limiting their utility in scientific decision-making. We introduce CLARIFY, a post-hoc explainability method for active-learning applications that (i) automatically derives high-level concepts from the model's embedding space, (ii) quantifies each concept's influence on ranking, and (iii) links the most influential concepts to sentences via occlusion and re-projection without retraining or manual intervention. Evaluated on three `SYNERGY` systematic review datasets, CLARIFY uncovers latent concepts and represents them in a human-understandable manner. Similarity metrics show discernible relations between these concepts and elements of the inclusion criteria. By making model reasoning transparent, CLARIFY supports accountable, evidence-based decision-making in systematic-review screening. Our open-source work can be found on GitHub/Zenodo.

10.1 Introduction

Machine learning, and more specifically Active learning (AL), is rapidly gaining ground in the field of systematic reviews as the key approach for semi-automating the screening phase (van de Schoot et al., 2021). As the volume of published literature continues to grow exponentially, the need for systems that assist with managing this information overload becomes increasingly urgent. Traditionally, these systems have been treated as black boxes, with most available software being closed-source[1]. The software provides a ranking or suggestions, but does not explain why a given reference was deemed relevant or not.

Researchers require insight into model decision-making processes to understand how predictions are made, build trust in the model, and make informed decisions based on its outputs. Transparency, accountability, and interpretability remain essential, especially in domains such as health, where inclusion decisions must be both defensible and reproducible. Opacity can hinder trust and adoption of AI assistance (Ribeiro, Singh, & Guestrin, 2016). Reviewers are understandably cautious about relying on a model's inclusions and exclusions without understanding the model's reasoning.

In the traditional screening phase, human reviewers manually assess each abstract for relevance and select studies for inclusion in the next stages of analysis. Machine learning tools like ASReview aim to reduce this burden by predicting which documents are most likely to meet the review's inclusion criteria, using an iterative process called active learning (de Bruin et al., 2025). When simple models are used, for example, TF-IDF (term frequency-inverse document frequency) and regression models, interpretability is relatively straightforward: one can trace decisions to specific words or combinations thereof (Sebastiani, 2002). This helps the reviewer understand and justify the tool's behavior, which is vital in contexts where methodological rigor is non-negotiable, and lack of transparency is one of the main barriers to implementation (Markus, Kors, & Rijnbeek, 2021).

However, recent developments in the field of active learning for systematic reviews show that more complex natural language processing (NLP) models, such as transformers (large language models, LLMs), can not be ignored for their performance (Chapter 7). These models better capture the nuances of human decision-making but do so in inherently opaque ways. Their logic operates in high-dimensional spaces, often diverging from the symbolic reasoning used by humans. This presents a fundamental challenge: if researchers are to remain accountable for the inclusion and exclusion decisions made with the aid of machine learning, they need a way to understand cx xthe basis for those decisions, even if that understanding comes after the fact.

Explainable AI techniques have been shown to complement active learning workflows effectively. A general framework of Explainable Active Learning (XAL) exists (Ghai, Liao, Zhang, Bellamy, & Mueller, 2020), in which local explanations are provided during the annotation process to improve annotator understanding and model trust. This framework demonstrates that explanations can enhance user engagement and decision confidence in iterative labeling tasks. However, the work also cautions that explanations may introduce cognitive biases, emphasizing the need for carefully de-

[1] github.com/Rensvandeschoot/software-overview-machine-learning-for-screening-text

signed, domain-sensitive interpretation methods. While these findings are not situated in the context of systematic reviews, they suggest that integrating explainability into active learning can be beneficial, particularly when model decisions carry scientific or clinical weight. To fill this gap, in the current study, we propose a new method for post-hoc, concept-based explanation of models used in systematic review screening software, namely CLARIFY. Our goal in CLARIFY is to create a tool for decomposing the internal representations of neural networks, thereby providing insight into the decision-making process of automated active learning-based screening tools such as ASReview. Rather than requiring retraining or architectural changes, CLARIFY operates after the model has completed its predictions, making it well-suited for integration into established review workflows. The contributions in this study are to:

1. Introduce CLARIFY, the first Explainable Active Learning pipeline for systematic review screening.
2. Provide a pipeline that integrates feature extractors and classification models with concept activation vectors, enabling human-readable concept scores without the need for retraining or architectural changes.
3. Reuse the active-learning state directly within the explanation pipeline
4. Conduct experiments on multiple review datasets to demonstrate the generalizability of the approach.

The rest of the paper is structured as follows. Section 10.2 explores related work on active learning for systematic reviews and explainable machine learning. Section 10.3 details the proposed CLARIFY pipeline and its integration within existing active learning frameworks such as ASReview. Section 10.4 reports on the results and Section 10.5 discusses limitations and future work, and Section 10.6 concludes.

10.2 Related Work

In the domain of systematic reviews, recent work evaluates the performance of various active learning strategies across multiple review datasets. The results of the study show that the difficulty of applying active learning is not confined to a particular research domain. Instead, the work suggests that a possible explanation for difficulty could be attributed to factors such as the complexity of inclusion criteria used to identify relevant publications (Ferdinands et al., 2023). Rathbone, Hoffmann, and Glasziou (2015), as cited in Gates, Johnson, and Hartling (2018), observes that the complexity of inclusion criteria can substantially affect the precision of automated screening tools. In their evaluation of Abstrackr, they note that imprecise population definitions (e.g., "young adults") and reviews structured around multiple key questions pose challenges for automated classification. Gates et al. extend this observation by showing that tasks with broad or heterogeneous criteria (e.g., descriptive analyses with no restriction by intervention or outcome) led to poor specificity and minimal workload savings. Ferdinands et al. (2023) suggest that variability in active learning performance may also stem from the complexity of the criteria themselves, even across otherwise comparable domains. These results underscore the importance of well-defined and narrowly scoped inclusion criteria in enabling effective automation, and potentially in making the classifier's logic more interpretable.

More generally, Vilone and Longo (2021) provides a comprehensive taxonomy of explainable AI methods and their application domains. Their review emphasizes the distinction between global and local explanations, model-agnostic versus model-specific approaches, and the varying interpretability needs across domains. Although the review does not address active learning, it offers a conceptual framework for situating post-hoc explanation techniques, such as the one proposed here, within the broader XAI landscape.

Jourdan et al. (2023) introduce COCKATIEL, a post-hoc, concept-based, model-agnostic explainer for neural text classifiers. It finds latent concepts in final-layer representations, ranks their importance, and maps them to text spans via occlusion, requiring only a non-negative embedding and no retraining.

Based on these developments, this study combines elements of XAL, ASReview's active learning approach, and COCKATIEL into a new framework, CLARIFY, which is a new method of explainable AI for systematic review screening optimization. To our knowledge, this is the first explainability method specifically designed for active learning in the context of systematic reviews.

While popular explainability methods such as LIME (Ribeiro et al., 2016), SHAP (Lundberg & Lee, 2017), or attention-based visualizations have been applied to NLP tasks, they are not well-suited for our setting. First, SHAP and LIME focus on local feature attributions, which are often unstable in high-dimensional embedding spaces and do not yield coherent or reusable semantic structures across documents. Attention-based methods, while attractive due to their direct integration in transformer architectures, have been shown to lack fidelity and can mislead users about model causality (Jain & Wallace, 2019). Moreover, these techniques typically provide token-level or word-level attributions, which do not align well with the sentence-level, concept-driven reasoning that systematic reviewers use. Our approach instead emphasizes latent concept discovery, enabling higher-level, reusable explanations that are better suited to capturing structured decision criteria such as inclusion rules.

10.3 Methods

10.3.1 CLARIFY Architecture

In an active learning cycle, scientific records are iteratively screened and re-prioritized, as implemented in frameworks such as ASReview. Each abstract is first transformed into an embedded representation via a feature extractor $h(x)$, yielding a matrix A that encodes the semantic features of the record collection. For this proof-of-concept, we employ the `mxbai-embed-large-v1` model as the embedding function. This model has shown good performance across a wide range of models in simulations (S. Lee et al., 2024).

Once embedded, the active learning system iteratively trains a classifier $c(x)$, updating the model each time new user labels are provided. At each step, the classifier ranks the remaining unlabeled documents by their estimated relevance, reordering the review queue accordingly. When the user classifies additional records, these are added to the

labeled set, and the classifier is retrained. This process continues until the screening task is completed.

To provide insight into this classifier's decision process, we apply the CLARIFY explanation method in three stages. First, we factorize the embedding matrix A using Non-negative Matrix Factorization (NMF), yielding two low-rank matrices: a concept alignment matrix U, and a concept base matrix W. The columns of W are interpreted as latent "concepts" learned from the data, while each row of U quantifies the degree to which a given document aligns with those concepts. D. D. Lee and Seung (1999) show that imposing non-negativity produces a **parts-based representation**: each column Wk of W captures a latent concept, and any document vector is reconstructed solely by non-negative mixes of those parts (D. Lee & Seung, 2000; D. D. Lee & Seung, 1999). Non-negativity forces negative alignments toward zero, resulting in sparse representations where each document aligns with only a subset of concepts. This sparsity enhances interpretability by allowing us to disregard concepts with (near) zero alignment, effectively identifying which parts are irrelevant for a given document.

In the second step, we compute the global importance of each discovered concept by perturbing concept activations U directly using Sobol sampling, and measuring the resulting variance in classifier output using Total Sobol indices. This uses the latest trained generation of the classifier $c(x)$ in the active learning cycle of ASReview. These indices quantify the variance in the model output attributable to perturbations in each concept's activation, capturing both direct and interaction effects.

The third step involves estimating local contributions: we assess which parts of an abstract contribute to a document's alignment with each concept. We mask individual sentences and create perturbations of the abstract. For each perturbation, we re-embed the perturbed abstract using the same feature extractor $h(x)$ and project it into the concept space using the fixed concept base W using the NMF transformation. This projection results in a new alignment matrix, revealing the new alignment for each perturbed abstract, minus the alignment for the masked sentence to the concept base W. A strong shift in alignment suggests that the masked sentence is strongly associated with the affected concept. While COCKATIEL performs occlusion at the word level, we found that sentence-level masking is more suitable for systematic reviews. In CLARIFY, we therefore apply this approach, since inclusion and exclusion criteria are often satisfied, or violated, within single, self-contained sentences.

For practical deployment, we propose applying CLARIFY in an on-the-fly fashion: explanations are generated for the document currently at the top of the active learning queue, for the next document the screener is expected to assess. While sentence-level occlusion explanations (step three) are recomputed for each document, both the NMF decomposition and the Sobol-based global concept importances can be reused across iterations. This makes CLARIFY efficient enough to be integrated into an interactive screening workflow.

Importantly, concept extraction is performed using only the embeddings of positively labeled documents. Since the top-ranked document is selected by the model as most likely to be relevant, it is most informative to explain its alignment with inclusion-related concepts. Attempting to extract concepts from documents predicted to be irrelevant would shift the focus toward exclusion justification, which is not aligned

with how active learning operates in ASReview.

10.3.2 Dataset Selection

This study uses the SYNERGY dataset (Chapter 5) for evaluating CLARIFY. From the full collection of 26 systematic reviews, a subset of datasets is selected based on prior classification performance, shown in Table 10.1.

Following Ferdinands et al. (2023), datasets with consistently poor model performance are excluded, as this may reflect inconsistently applied or vague inclusion criteria. Among the remaining high-performing datasets, selection is further refined based on the interpretability and clarity of inclusion and exclusion rules. Preference is given to datasets where these criteria are well defined, distinct, and plausibly reflected in the text.

Name	Relevant records	Total Records	Topic
Hall_2012 (Hall et al., 2012)	104	8793	Computer science
Jeyaraman_2020 (Jeyaraman, Muthu, & Ganie, 2021)	96	1175	Medicine
Menon_2022 (Menon, Struijs, & Whaley, 2022)	74	975	Medicine

Table 10.1 Datasets used for the evaluation of CLARIFY

10.3.3 Implementation

The CLARIFY explainability method was adapted to operate within the ASReview framework. While the original implementation is based on the PyTorch ecosystem, ASReview incorporates a range of models and utilities implemented in both scikit-learn and TensorFlow, necessitating cross-framework integration. To address this, a hybrid pipeline was developed that extracts final-layer embeddings from ASReview, formats them for compatibility with the decomposition and attribution modules, and returns sentence-level explanations for the top-ranked documents. All code, along with configuration files, results, and documentation, is published openly via GitHub and archived on Zenodo to ensure transparency and reproducibility.

We refactored CLARIFY into a self-contained ASReview plug-in, replacing the PyTorch code with scikit-learn-compatible components and a lightweight NMF-based concept module. The pipeline now (i) extracts transformer embeddings through MXBAI, (ii) normalizes them once via a shared min-max scaler, (iii) restricts concept factorization and Sobol attribution to the positively labeled subset, and (iv) returns sentence-level heat-maps through a fast occlusion routine built directly on ASReview's feature interface.

Figure 10.1 shows the schematic overview of the final CLARIFY architecture embed-

ded in the ASReview active-learning-based system, and the pseudo-code representation of this work can be found in Appendix D.1.

10.4 Results

10.4.1 Concept Importance and Sentence-Level Highlights

To evaluate whether the CLARIFY yields useful and interpretable outputs in the context of systematic review screening, we applied it to a select subset of high-performing datasets from the `SYNERGY` benchmark. The central question was whether the discovered concepts extracted from the model's internal representations can be useful to human reviewers, either by giving insight into the model's decision mechanism or by aligning with the inclusion criteria.

We present the results in two stages. First, we present the direct outcomes of the method, including concept importance rankings and highlighted abstracts with sentence-level alignment. These results are shown alongside each dataset's inclusion criteria to support alignment analysis. Second, we present the normalized cosine similarity between the identified concepts and the inclusion criteria to serve as a quantitative evaluation of the concept's usefulness in regard to the inclusion criteria.

Figure 10.2 shows global concept importance per dataset, computed as total Sobol indices on embeddings of positively labeled records. The red horizontal line marks the selection threshold, defined as the mean plus one standard deviation of the positive-importance distribution ($(\mu + \sigma)$). Concepts above this threshold are used in the later analyses. In these runs, Hall and Jeyaraman each yield two selected concepts; Menon yields three. The threshold is a pragmatic heuristic; other cut-offs (for example, a top-quantile rule or a fixed number of important concepts) are also reasonable, providing little difference.

We observe that importance drops sharply after roughly concept 10. We keep all 20 bars visible for transparency, since NMF was run with k=20. In our setup, scikit-learn's NMF initialization uses NNDSVDa, an SVD-based initializer that is energy ordered. On our positive-only data, this tends to concentrate mass in early components, yielding lower importance at higher indices (Boutsidis & Gallopoulos, 2008). Using `nndsvdar` or `random` spreads variance more evenly and can lift late-index importances, although the set retained after the $(\mu + \sigma)$ threshold is largely uninfluenced.

We set k=20 for NMF to avoid collapse into one dominant factor when k is too small. Overcompleting the basis lets the model express variation, after which Sobol ranking identifies the few concepts that affect the classifier; almost half have near-zero importance. Because NMF is initialization-sensitive, the amount and the exact indices above the threshold can change between runs, but the pattern is consistent: a small set of high-importance concepts and a long tail of negligible ones, consistent with COCKATIEL's outcomes.

Inclusion Criteria Hall 2012

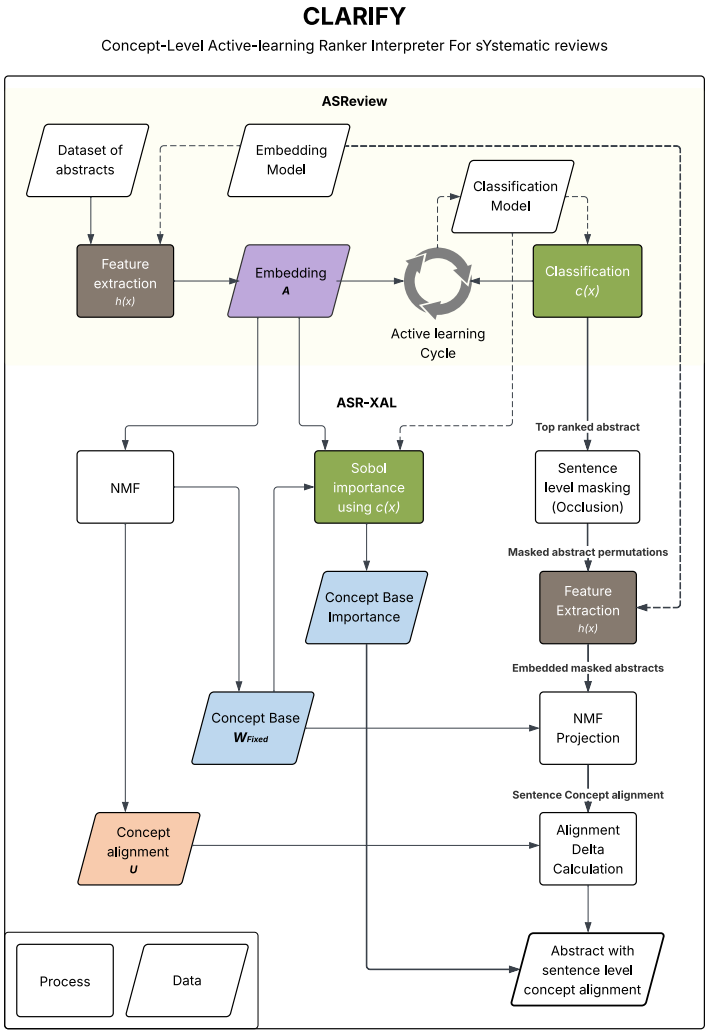
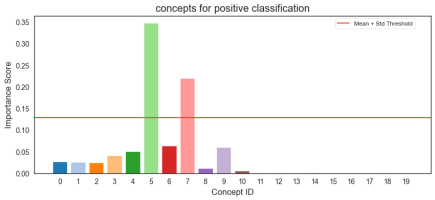
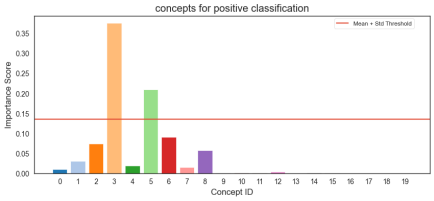


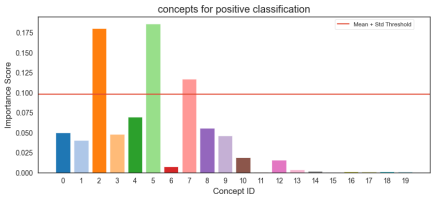
Figure 10.1 Overview of CLARIFY explainable active learning pipeline embedded in ASReview. The figure represents the important components of the pipeline using process and data blocks.



Hall



Jeyaraman



Menon

Figure 10.2 Global concept importance across datasets. Threshold $\mu + \sigma$ shown in red.

- An empirical study
- Focused on predicting faults in units of a software system
- Faults in code is the main output (dependent variable)

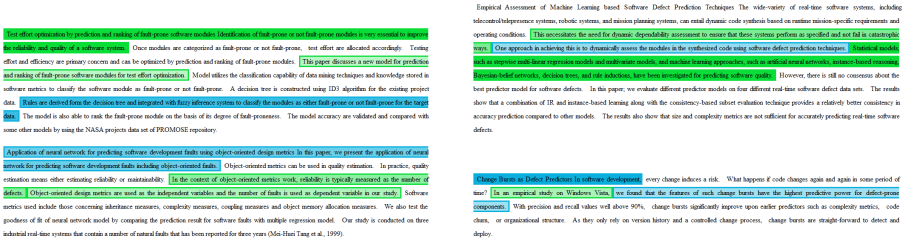
Inclusion Criteria Jeyaraman 2020

- Patients with knee osteoarthritis
- Intervention with MSC therapy
- Comparator: usual care
- Outcomes: VAS for Pain, WOMAC, Lysholm, WORMS, KOOS, and adverse events
- Study design: Randomized controlled trials

Inclusion Criteria Menon 2021

- Explicitly identified as a "systematic review" in the title
- Assessed the effect of a non-acute, non-communicable, environmental exposure on a health outcome
- Included studies in people or mammalian models

Figure 10.3, Figure 10.4, and Figure 10.5 present abstracts of the datasets that were included in the study and labeled as relevant. The color indicates concept alignment;



Test effort optimization by prediction and ranking of fault-prone software modules Identification of fault-prone or not fault-prone modules is very essential to improve the reliability and quality of a software system. Once modules are categorized as fault-prone or not fault-prone, test effort are allocated accordingly. Testing effort and efficiency are primary concern and can be optimized by prediction and ranking of fault-prone modules. This paper discusses a new model for prediction and ranking of fault-prone software modules for test effort optimization. Model utilizes the classification capability of data mining techniques and knowledge stored in software metrics to classify the software module as fault-prone or not fault-prone. A decision tree is constructed using ID3 algorithm for the existing project data. Rules are derived from the decision tree and integrated with fuzzy inference system to classify the modules as either fault-prone or not fault-prone for the target data. The model is also able to rank the fault-prone module on the basis of its degree of fault-proneness. The model accuracy are validated and compared with some other models by using the NASA projects data set of PROMOSE repository.

Application of neural network for predicting software development faults using object-oriented design metrics In this paper, we present the application of neural network for predicting software development faults including object-oriented faults. Object-oriented metrics can be used in quality estimation. In practice, quality estimation means either estimating reliability or maintainability. In the context of object-oriented metrics work, reliability is typically measured as the number of defects. Object-oriented design metrics are used as the independent variables and the number of faults is used as dependent variable in our study. Software metrics used include those concerning inheritance measures, complexity measures, coupling measures and object memory allocation measures. We also test the goodness of fit of neural network model by comparing the prediction result for software faults with multiple regression model. Our study is conducted on three industrial real-time systems that contain a number of natural faults that has been reported for three years (Mei-Huei Tang et al., 1999).

Empirical Assessment of Machine Learning based Software Defect Prediction Techniques The wide-variety of real-time software systems, including telecontrol/telepresence systems, robotic systems, and mission planning systems, can entail dynamic code synthesis based on runtime mission-specific requirements and operating conditions. This necessitates the need for dynamic dependability assessment to ensure that these systems perform as specified and not fail in catastrophic ways. One approach in achieving this is to dynamically assess the modules in the synthesized code using software defect prediction techniques. Statistical models such as stepwise multi-linear regression models and multivariate models, and machine learning approaches, such as artificial neural networks, instance-based reasoning, bayesian belief networks, decision trees, and rule inductions, have been investigated for predicting software quality. However, there is still no consensus about the best predictor model for software defects. In this paper, we evaluate different predictor models on four different real-time software defect data sets. The results show that a combination of IR and instance-based learning along with the consistency-based subset evaluation technique provides a relatively better consistency in accuracy prediction compared to other models. The results also show that size and complexity metrics are not sufficient for accurately predicting real-time software defects.

Change Bursts as Defect Predictors In software development, every change induces a risk. What happens if code changes again and again in some period of time? In an empirical study on Windows Vista, we found that the features of such change bursts have the highest predictive power for defect-prone components. With precision and recall values well above 90%, change bursts significantly improve upon earlier predictors such as complexity metrics, code churn, or organizational structure. As they only rely on version history and a controlled change process, change bursts are straight-forward to detect and deploy.

Figure 10.3 A selection of highlighted abstracts with all concepts present from the Hall Dataset

Injectable Cultured Bone Marrow–Derived Mesenchymal Stem Cells in Varus Knees With Cartilage Defects Undergoing High Tibial Osteotomy A Prospective, Randomized Controlled Clinical Trial With 2 Years' Follow-up To analyze the results of the use of intra-articular cultured autologous bone marrow–derived mesenchymal stem cell (MSC) injections in conjunction with microfracture and medial opening-wedge high tibial osteotomy (HTO).Fifty-six knees in 56 patients with unicompartmental osteoarthritic knees and genu varum were randomly allocated to the cell-recipient group (n = 28) or control group (n = 28). Patients who had a joint line congruity angle of more than 2°, malalignment of the knee from femoral causes, a fixed flexion deformity, or age older than 55 years were excluded. All patients underwent HTO and microfracture. The cell-recipient group received intra-articular injection of cultured MSCs with hyaluronic acid 3 weeks after surgery, whereas the control group only received hyaluronic acid. The primary outcome measure was the International Knee Documentation Committee (IKDC) score at intervals of 6 months, 1 year, and 2 years postoperatively. Secondary outcome measures were Tegner and Lysholm clinical scores and 1-year postoperative Magnetic Resonance Observation of Cartilage Repair Tissue (MOCART) scores.The median age of the patients was 51 years with a mean body mass index of 23.85. Both treatment arms achieved improvements in Tegner, Lysholm, and IKDC scores. After adjustment for age, baseline scores, and time of evaluation, the cell-recipient group showed significantly better scores. The effect of treatment showed an added improvement of 7.65 (95% confidence interval [CI], 3.04 to 12.26; P = .001) for IKDC scores, 7.61 (95% CI, 1.44 to 13.79; P = .016) for Lysholm scores, and 0.64 (95% CI, 0.10 to 1.19; P = .021) for Tegner scores. Magnetic resonance imaging scans performed 1 year after surgical intervention showed significantly better MOCART scores for the cell-recipient group. The age-adjusted mean difference in MOCART score was 19.6 (95% CI, 10.5 to 28.6; P < .001).Intra-articular injection of cultured MSCs is effective in improving both short-term clinical and MOCART outcomes in patients undergoing HTO and microfracture for varus knees with cartilage defects.Level II, randomized controlled trial.

Comparison of Bone Marrow Aspirate Concentrate and Allogeneic Human Umbilical Cord Blood Derived Mesenchymal Stem Cell Implantation on Chondral Defect of Knee: A assessment of Clinical and Magnetic Resonance Imaging Outcomes at 2-Year Follow-Up Biological repair of cartilage lesions remains a significant clinical challenge. A wide variety of methods involving mesenchymal stem cells (MSCs) have been introduced. Because of the limitation of the results, most of the treatment methods have not yet been approved by the Food and Drug Administration (FDA). However, bone marrow aspirate concentrate (BMAC) and human umbilical cord blood derived mesenchymal stem cells (hUCB-MSCs) implantation were approved by Korea FDA. The aim of this study was to evaluate clinical and magnetic resonance imaging (MRI) outcomes after two different types of MSCs implantation in knee osteoarthritis. Fifty-two patients (52 knees) who underwent cartilage repair surgery using the BMAC (25 knees) and hUCB-MSCs (27 knees) were retrospectively evaluated for 2 years after surgery. Clinical outcomes were evaluated according to the score of visual analogue scale (VAS), the International Knee Documentation Committee (IKDC) subjective, and the Knee Injury and Osteoarthritis Outcome Score (KOOS). Cartilage repair was assessed according to the modified Magnetic Resonance Observation of Cartilage Repair Tissue (M-MOCART) score and the International Cartilage Repair Society (ICRS) cartilage repair scoring system. At 2-year follow-up, clinical outcomes including VAS, IKDC, and KOOS significantly improved (P < 0.05) in both groups; however, there were no differences between two groups. There was no significant difference in M-MOCART [1-year (P = 0.261), 2-year (P = 0.353)] and ICRS repair score (P = 0.655) between two groups. Both groups showed satisfactory clinical and MRI outcomes. Implantation of MSCs from BMAC or hUCB-MSCs is safe and effective for repairing cartilage lesion.However, large cases and a well-controlled prospective design with long-term follow-up studies are needed.

Cell-Seeded Autologous Chondrocyte Implantation: A Simplified Implantation Technique That Maintains High Clinical Outcomes Background: The use of autologous chondrocyte implantation (ACI) remains limited, even though multiple studies have demonstrated success rates exceeding 75%. The procedure is perceived as invasive and technically challenging, presenting barriers to more widespread adoption. Purpose/Hypothesis: The objective of this study was to investigate whether outcomes and the failure rate of a simplified ACI technique (cs-ACI) were comparable with those of the more complicated traditional technique of a chondrocyte suspension injected under a collagen membrane (cACI). We hypothesized that the change in technique would not negatively affect outcomes. Study Design: Cohort study; Level of evidence, 3. Methods: Thirty-nine patients treated with the cs-ACI technique fulfilled the inclusion requirements. A group of 45 patients treated previously with standard cACI was used as a comparison. The functional outcomes were prospectively collected both preoperatively and postoperatively at the last follow-up. Failure was defined as any graft removal of more than 25% of the original defect size. Magnetic resonance imaging was performed postoperatively, and scans were assessed using a modified MOCART (magnetic resonance observation of cartilage repair tissue) scoring system. Results: Group demographics were not significantly different, except for the defect size and mean follow-up: 4.09 years in the cACI group and 2.46 years in the cs-ACI group. Significant improvements were seen in all outcome measures except the Tegner score from the preoperative baseline to the latest follow-up for both the cACI group (International Knee Documentation Committee [IKDC] score from 42.0 to 63.4; Knee injury and Osteoarthritis Outcome Score [KOOS]–Pain subscore, from 58.7 to 77.1; Lysholm score, from 57.2 to 69.7; and Tegner score, from 3.5 to 4.2) and the cs-ACI group (IKDC score, from 45.6 to 68.0; KOOS–Pain subscore, from 66.6 to 84.7; Lysholm score, from 53.7 to 75.4; and Tegner score, from 3.2 to 3.8). No significant difference was found between the groups at the latest follow-up. The failure rate at 2 years was not significantly different, while the total failure rate over the entire study period was significantly lower in the cs-ACI group than the cACI group (5% vs 24%, respectively). The overall MOCART score was not significantly different between the groups. Conclusion: The treatment of full-thickness articular cartilage defects with a simplified cell-seeded ACI technique demonstrated no significant differences in the failure rate and patient-reported outcomes when compared with a standard technique utilizing interrupted sutures and the injection of a cell suspension under a collagen membrane.

Subchondral bone or intra-articular injection of bone marrow concentrate mesenchymal stem cells in bilateral knee osteoarthritis: what better postpone knee arthroplasty at fifteen years? A randomized study Purpose There is an increasing number of reports on the treatment of knee osteoarthritis (OA) using mesenchymal stem cells (MSCs). However, it is not known what would better drive osteoarthritis stabilization to postpone total knee arthroplasty (TKA): targeting the synovial fluid by injection or targeting on the subchondral bone with MSCs implantation. Methods A prospective randomized controlled clinical trial was carried out between 2000 and 2005 in 120 knees of 60 patients with painful bilateral knee osteoarthritis with a similar osteoarthritis grade. During the same anaesthesia, a bone marrow concentrate of 40 mL containing an average 5727 MSCs/mL (range 2740 to 7540) was divided in two equal parts: after randomization, one part (20 mL) was delivered to the subchondral bone of femur and tibia of one knee (subchondral group) and the other part was injected in the joint for the contralateral knee (intra-articular group). MSCs were counted as CFU-F (colony fibroblastic unit forming). Clinical outcomes of the patient (Knee Society score) were obtained along with radiological imaging outcomes (including MRIs) at two year follow-up. Subsequent revision surgeries were identified until the most recent follow-up (average of 15 years, range 13 to 18 years). Results At two year follow-up, clinical and imaging (MRI) improvement was higher on the side that received cells in the subchondral bone. At the most recent follow-up (15 years), among the 60 knees treated with subchondral cell therapy, the yearly arthroplasty incidence was 1.3% per knee-year, for the 60 knees with intra-articular cell therapy, the yearly arthroplasty incidence was higher ($p = 0.01$) with an incidence of 4.6% per knee-year. For the side with subchondral cell therapy, 12 (20%) of 60 knees underwent TKA, while 42 (70%) of 60 knees underwent TKA on the side with intra-articular cell therapy. Among the 18 patients who had no subsequent surgery on both sides, all preferred the knee with subchondral cell therapy. Conclusions Implantation of MSCs in the subchondral bone of an osteoarthritic knee is more effective to postpone TKA than injection of the same intra-articular dose in the contralateral knee with the same grade of osteoarthritis.

Figure 10.4 A selection of highlighted abstracts with all concepts present from the Jeyaraman Dataset

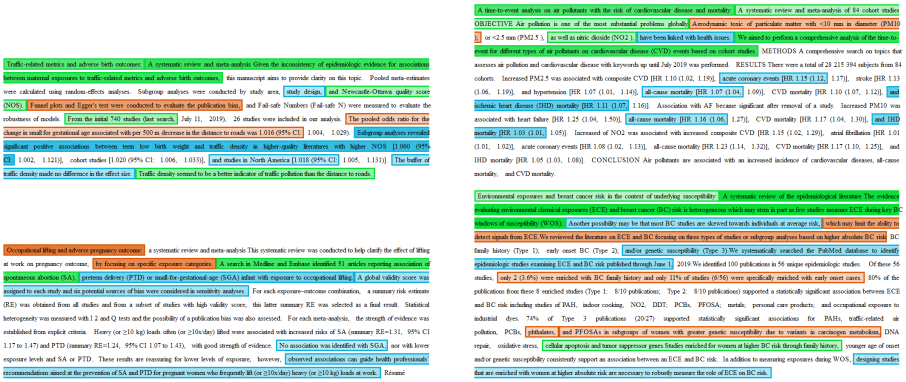


Figure 10.5 A selection of highlighted abstracts with all concepts present from the Menon Dataset

values below a set threshold are omitted as the alignment with a concept is deemed too weak to be relevant. The abstracts were selected based on concept occurrence. Not all abstracts contain all concepts; some abstracts have fewer or no above-threshold sentences. The computations for occlusion, embedding, NFM projection, and delta required for visualizing alignment per abstract take an average of 21 seconds per abstract. Timings were obtained on a 2021 4-core laptop-class CPU. The granularity of the abstracts is sentence-based.

10.4.2 Quantitative Evaluation of Concept Usefulness

As shown in the Inclusion Criteria for Hall 2012, this study applied three inclusion criteria. Using cosine similarity, we calculate the similarity between the embedding of each criterion and each learned concept vector. This allows us to assess whether certain concepts align more strongly with specific criteria. A high similarity score for a given criterion–concept pair suggests that the concept captures semantic information directly related to that criterion, while low scores indicate weak or no alignment.

To calculate similarity, the inclusion criteria are embedded using $h(x)$; the same feature extractor is applied during model training. These embeddings are then normalized and compared, via cosine similarity, to the learned concept base W from the NMF decomposition. Along with the inclusion criteria, the similarity scores are computed for unrelated baseline sentences, providing a reference level. Finally, the scores are normalized and visualized in a bar plot, with dashed horizontal lines indicating the mean baseline similarity for each concept. The baseline is the mean similarity to the embeddings of unrelated sentences.

Figure 10.6 shows the output of this process. We select the concepts identified as important in the CLARIFY process and compare them to the inclusion criteria for a dataset. For the Hall dataset, 2 important concepts were discovered (concept 5 and concept 7), and 3 inclusion criteria were used for the creation of the dataset (identified as criteria 1, 2, and 3). After calculations, all three inclusion criteria have similarity

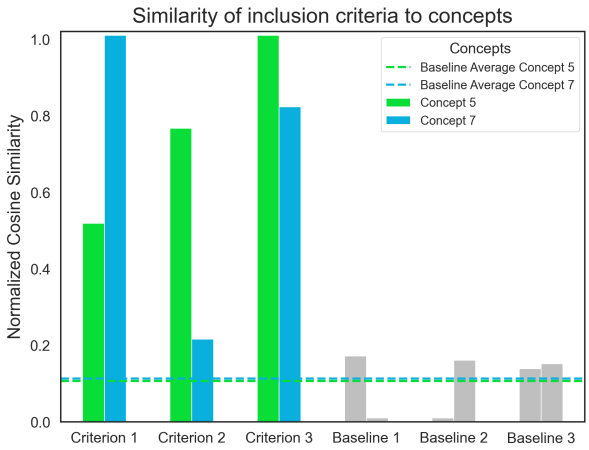


Figure 10.6 Normalized cosine similarity for the Hall_2012 dataset between concepts and inclusion criteria, with baseline similarity shown as dashed lines. The plot shows the different similarities between the inclusion criteria and concepts.

scores that clearly rise above the baseline for each concept. Inclusion criterion 1 shows a strong alignment with the first important concept, while criterion 2 is more strongly aligned with the second important concept. Criterion 3 exhibits comparable similarity to both concepts. This pattern indicates that the discovered concepts lie within the semantic space of the inclusion criteria. The model's concept structure is organized along dimensions that correspond to the review's decision rules, supporting the hypothesis that concept discovery near the classifier (in the final hidden layer) recovers decision-relevant signals.

10.5 Discussion

10.5.1 Role of concept positioning

Concept extraction is performed on the embeddings produced by the feature extractor $h(x)$, immediately prior to classification $c(x)$. This location within the pipeline prioritizes the extraction of latent representations that are closely aligned with the classifier's decision function over the interpretability of the concepts, as this is most useful for ASReview.

The hypothesis in this work is that these representations reflect the semantic signals relevant to the inclusion criteria, to the extent that such signals are captured by the model. The goal is not to reconstruct the input or surface linguistically grounded structures, but to identify internal signals that influence classification outcomes. Extracting concepts too early risks overfitting to shallow lexical patterns; extracting them too late risks reducing them to direct encodings of the predicted label.

This is not to say that directly encoding the classification prediction as a concept is

useless. By masking each sentence in turn, re-embedding the perturbed text, and comparing the change in predicted relevance to the full abstract, we obtain sentence-level alignment scores visualized in Figure 10.7. We interpret the final classification output as a single concept and quantify each sentence's effect on the predicted relevance. This provides local accountability for a specific abstract. However, it does not reveal the intermediate semantic factors the model relies on. It shows the impact sentences have on the classification probability, not which latent dimensions structure the decision.

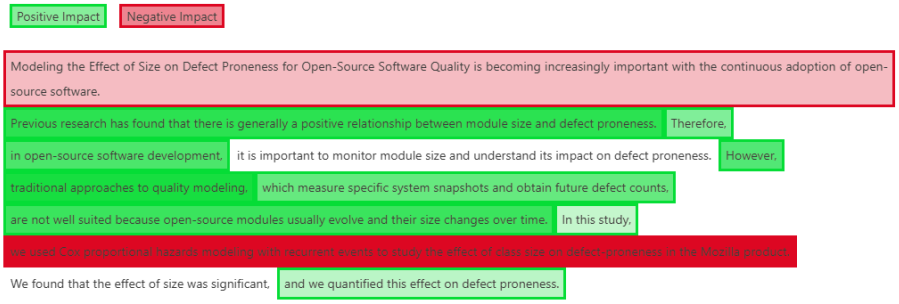


Figure 10.7 Final layer accountability highlighted at the sentence level for the abstract from the Hall 2012 dataset.

Reversing this perspective, CLARIFY does not provide direct accountability for the final classification in the way shown in Figure 10.7. Instead, it identifies latent concepts that represent intermediate decision signals within the model. These should not be interpreted as explanations for the final prediction. If the goal is to assess the contribution of individual sentences to the final classification, final-layer occlusion, as in this example, is appropriate. Conversely, occlusion applied to the final hidden layer is suited to revealing which semantic dimensions influence the decision.

In practice, we could use final-layer occlusion for sentence-level accountability and final hidden-layer occlusion to explain which semantic dimensions drive decisions.

10.5.2 Cognitive biases and interpretability limitations

The assumption that inclusion criteria are encoded in interpretable units is difficult to support. It is often used to explain the workings of CNNs for images: a complex task like digit recognition is decomposed into simple visual components such as loops, straight lines, and intersections, and recomposed layer by layer into digit identities. Early layers are frequently interpretable; they learn edges, curves, and simple shapes. But move deeper into the network, and the visualizations quickly degrade. Later layers do not resemble meaningful visual parts but instead appear random to the human eye. Neural networks are optimized for task performance, not human interpretability. Often, learning performance degrades with an increased explainability (Rudin, 2019).

The same goes for lexical challenges. Although earlier layers may offer more interpretable patterns, they carry limited information about the final classification outcome. Highlighting features from these layers may expose the building blocks the model uses and how the lexical input is broken into subproblems, but not how these components

are recombined to form a classification. As a result, such representations are not only weakly informative but potentially misleading. They may appear meaningful, yet offer no insight into why a document is marked relevant. To surface decision-relevant signals, we must operate further towards the end of the pipeline, even if that means forgoing interpretability of steps. Figure 10.8 visualizes this gradient.

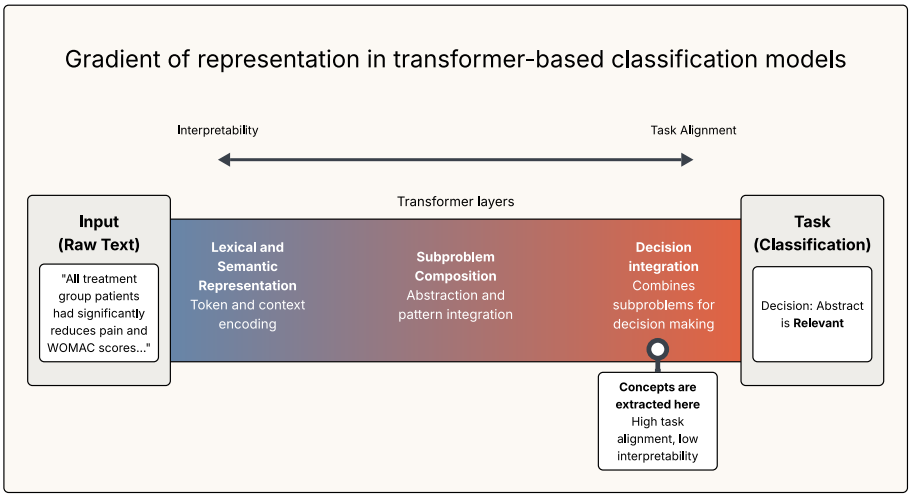


Figure 10.8 Abstract visualization of the interpretability-task alignment gradient found in a transformer-based classification pipeline.

This pattern holds for ASReview and CLARIFY. While earlier transformer layers may contain interpretable, meaningful patterns, the deeper layers, those near the classifier, reflect highly task-specific transformations. There is no reason to assume that latent units in these layers correspond to human-interpretable concepts, even when the training task is structured around inclusion criteria.

This also motivates the decision not to pursue manual labeling of concepts. The pipeline produces alignments between sentences and abstract latent units that influence classification, but does not explain *why*. Any interpretation beyond this point risks reflecting human pattern-seeking rather than grounded evidence. This cognitive bias must be acknowledged when interacting with model explanations.

10.5.3 Findings in Relation to the Study Objectives

Our findings can be summarized along four main objectives. First, we examined whether latent concepts could be extracted directly from ASReview's transformer's hidden embedding layer without retraining. The results show that this is feasible: concepts can be surfaced and presented as sentence-level highlights accessible to users. These reflect decision-relevant internal signals rather than surface-level lexical features. While the extracted concepts encode information used by the model for its final predictions, further work is needed to determine whether they form coherent, user-understandable units.

Second, we explored whether the discovered concepts align with the inclusion criteria. Cosine similarity analyses revealed associations between concepts and criteria, suggesting that the model's latent space captures aspects of these criteria. However, this evidence is correlational and does not imply that the criteria are explicitly encoded as separable concepts. Risks of cognitive bias and interpretability limitations (see Section 5.2) further constrain the strength of this claim.

Third, we considered the practical usefulness of the explanations for reviewers. While we demonstrate sentence-level alignment with concept activation, their effectiveness in practice remains inconclusive. A controlled user study would be needed to assess their impact on reviewer performance. Importantly, high concept importance should not be taken as evidence of causal contribution to inclusion decisions, and any implementation must make this distinction explicit to avoid misinterpretation.

Finally, we assessed the pipeline's practicality for on-the-fly use in ASReview. Results indicate that the method produces concept-level explanations with low latency on standard laptop hardware. By reusing the fixed NMF basis, Sobol importances, and ASReview's feature extractor, and only recomputing sentence-level occlusions as needed, the approach enables ad-hoc execution without retraining. The design is model-agnostic and suitable for integration into active learning workflows.

Looking ahead, concept labeling remains an open challenge. Assigning coherent labels without introducing bias is difficult, but one promising direction is to leverage similarity between concepts and sentences. Because active learning keeps the reviewer in the loop, the system could accept reviewer-proposed labels or criteria and return similarity scores to each concept, optionally with representative sentences. Our results suggest this interaction is viable. The approach is simple to implement, adds minimal computational cost, and enables less biased, user-steerable exploration of the concept space to improve understanding.

10.6 Conclusion

This study introduced CLARIFY, a post-hoc, concept-based explanation pipeline for active learning screening in systematic reviews. Using embeddings from the final hidden layer, the method factorizes the representation space with NMF to discover latent concepts. It integrates without retraining and reuses model elements to minimize computational cost. The pipeline was evaluated on three `SYNERGY` datasets, producing ranked concept importance, concept–criterion similarity measures, and sentence-level highlights. All code and results are openly available as J. Teijema (2025).

CLARIFY demonstrates that post-hoc, concept-based explanations can be integrated into active learning screening without retraining or heavy computation, while preserving model-agnosticism, with about 20 seconds per highlighted abstract in our setup. By surfacing decision-relevant signals, the method moves transformer-based screening models toward greater transparency and interactivity. Practical reviewer benefit requires validation in controlled studies. We see this work as a step toward explainable systematic review tools while employing black-box machine learning models, by (re)enabling accountability in AI-assisted screening.

Chapter 11

Concluding Remarks

This dissertation has presented the complete execution of an ADS project within an academic environment. It included exploratory research, data preparation, software development, large-scale experimentation, real-world validation, and interpretability. With this, the project aimed to improve the use of active learning in the systematic review screening phase, and to do so in a way that reflects both the practical orientation of ADS and expected academic standards.

In this work, the friction between academic and industrial norms played a central role. This final chapter reflects on the academic positioning of Applied Data Science and the four themes proposed in the introduction.

11.1 The Positioning of Applied Data Science

Chapter 2 describes how the increasing influence of industry in data science creates a tension with how projects are traditionally approached in an academic setting. However, as seen in this dissertation, the growing overlap between industrial and academic data science should not be seen as harmful. Industrial techniques applied in this contribution introduced an emphasis on usability and implementation, qualities that support ADS but academic projects often lack. At the same time, academia contributes to expectations of transparency and openness.

Another contribution of this work is demonstrating how operational reflective skills can come together in academic ADS. Throughout this project, operational skills, such as the development of robust software workflows and large-scale simulations, were applied not only as the main applied output but also as the basis for reflective academic inquiry.

11.2 Reflecting on the Themes

The dissertation engaged with four themes representing the challenges of academic ADS: Human-Centered Design, Software Usability, Reproducibility and Evidence, and FAIR data.

11.2.1 Human-Centered Design in Academic ADS

The principles of human-centered design (HCD) can be used in academic ADS projects by relating research outcomes to the problems faced by domain practitioners. The ADS project of this dissertation began with an exploration phase. This provided an understanding of the practical context and allowed for relevant problem identification. This approach mirrors industry practice.

Methods and simulations in this dissertation were developed for academic purposes. Yet, the research focused on practical problems originating from the exploration phase. Results were interpreted relative to these problems. Recommendations and model choices were provided with consideration to how they inform human decision-making in practice. Technical tools were applied as a means to support action, rather than ends in themselves. Adopting the principles of HCD strengthened the value of the research, in applied understanding without abandoning academic standards.

Practically, these principles can be found in this dissertation as user-focused communication and human-centered advice. Chapter 8 shows this by explaining model selection strategies in accessible language. Furthermore, the recommendation sections of chapter 3 and chapter 7 translate raw findings into actionable, human-centered advice.

11.2.2 Software Usability

The adoption of usability principles in research software depends on technical capacity and research conditions. While most applied data scientists possess the expertise required to create maintainable systems, academic incentives often prioritize investments in publication over software quality. As a consequence, software tends to be developed solely for immediate demonstration, resulting in tools that are difficult to reuse.

Investment in software usability is a matter of priority. Usability improves when software is recognized as a primary research output. In this dissertation, the treatment of software as a distinct contribution justified the allocation of resources to usability. This investment enabled collaborators to apply and extend the developed tools. Consequently, the utility of the software increased and provided a return on the effort invested. The resulting software promoted research communication, facilitated collaborations, and improved visibility of the work.

In this work, chapter 6 is the primary example of academic software with a strong focus on usability. Its impact can be read in section 6.7. The software is actively maintained, having seen multiple versions created based on both new features and user feedback. Software outputs from other chapters, while smaller in scope, adhere to similar standards. All released code is tested and accompanied by complete documentation.

11.2.3 Reproducibility and Evidence

In ADS, the problem of proof is not solved through theory, nor through full replication. Modern machine learning models exceed the scope of classical proof, and their

scale often renders empirical proof through full replication infeasible. The work in this dissertation operates under these same constraints. Theoretical guarantees are unavailable, and full empirical validation of the simulations is prohibited by computational costs.

The computationally heavy chapters of this dissertation, such as chapters 7, 9, and 10, addressed the challenge by making every step of the research process open to evaluation through replication data repositories. Data, preprocessing steps, simulation settings, outputs, and analysis scripts are openly available. While full replication still requires substantial computational resources, partial verification is feasible. Components can be rerun and artifacts inspected.

The credibility of the work rests on transparency rather than on the ability to reproduce every computation. The work remains resource-intensive, yet accountability is preserved because the components are open to inspection.

11.2.4 FAIR Data

The benefits of FAIR data practices are widely acknowledged, yet a gap persists between the ideal and actual uptake. In this dissertation, a conscious effort was made to publish all data, models, and software in accordance with FAIR principles.

This work shows that investing in FAIR data yields academic returns, as evidenced by the popularity of the SYNERGY dataset and the ASReview Makita software. The open availability of the project's resources led directly to their visibility and reuse. By valuing data and software as scientific contributions rather than supplementary material, the constraints often cited in the literature were effectively mitigated.

This experience supports the argument for the formal recognition of digital artifacts within PhD trajectories. When the production of FAIR data and software output is rewarded as a valid scholarly work, the incentives for high-quality operational output increase. This recognition can serve as a structural solution that enables other researchers to adopt FAIR practices and capitalize on the resulting benefits.

11.3 Future Outlook

This dissertation shows that Applied Data Science can be carried out in an academic environment while drawing on industry elements. The contribution to the ADS domain lies both in the operational results produced and the reflective academic study.

Academic change is slow regarding reward structures. However, Applied Data Science is a young academic field. Its development as a valid academic domain is largely occurring in the current decade. Currently, while the field still has high plasticity, an opportunity exists to establish norms in which usable software is valued, FAIR data is the standard, and the effort required to achieve it is acknowledged. Prioritizing usability, transparency, and practical relevance ensures that academic data science remains a driver of innovation rather than a follower of industry.

References

Adam, G. P., Wallace, B. C., & Trikalinos, T. A. (2021). Semi-automated tools for systematic searches. In *Methods in molecular biology* (pp. 17–40). Springer US. Retrieved from https://doi.org/10.1007/978-1-0716-1566-9_2 doi: 10.1007/978-1-0716-1566-9_2

Ahmed, N., Das, A., Martin, K., & Banerjee, K. (2024). *The narrow depth and breadth of corporate responsible ai research.* Retrieved from https://arxiv.org/abs/2405.12193

Alwosheel, A., Van Cranenburgh, S., & Chorus, C. G. (2018). Is your dataset big enough? sample size requirements when using artificial neural networks for discrete choice analysis. *Journal of choice modelling*, *28*, 167–182.

Ambert, K. H., Cohen, A. M., Burns, G. A. P. C., Boudreau, E., & Sonmez, K. (2013). Virk: An active learning-based system for bootstrapping knowledge base development in the neurosciences [Journal Article]. *Frontiers in Neuroinformatics*, *7*(DEC). doi: 10.3389/fninf.2013.00038

Andrews, M. (2025). The immortal science of ml: Machine learning and the theory-free ideal: M. andrews. *Erkenntnis*, 1–23.

Aragon, C., Guha, S., Kogan, M., Muller, M., & Neff, G. (2022). *Human-centered data science: an introduction.* MIT Press.

ASReview LAB developers. (2023, 4). *Asreview lab v1.2 - a tool for ai-assisted systematic reviews.* Zenodo. doi: 10.5281/zenodo.7821585

ASReview LAB developers. (2025). *Asreview makita: A workflow generator for simulation studies using the command line interface of asreview lab (v1.1).* Zenodo. Retrieved from https://doi.org/10.5281/zenodo.15720003 doi: 10.5281/zenodo.15720003

Baumer, E. P. (2017). Toward human-centered algorithm design. *Big Data & Society*, *4*(2), 2053951717718854.

Beller, E., Clark, J., Tsafnat, G., Adams, C., Diehl, H., Lund, H., … founding members of the, I. g. (2018). Making progress with the automation of systematic reviews: principles of the international collaboration for the automation of systematic reviews (icasr) [Journal Article]. *Syst Rev*, *7*(1), 77. (Beller, Elaine Clark, Justin Tsafnat, Guy Adams, Clive Diehl, Heinz Lund, Hans Ouzzani, Mourad Thayer, Kristina Thomas, James Turner, Tari Xia, Jun Robinson, Karen Glasziou, Paul eng Letter England 2018/05/21 Syst Rev. 2018 May 19;7(1):77. doi: 10.1186/s13643-018-0740-7.) doi: 10.1186/s13643-018-0740-7

Beltagy, I., Lo, K., & Cohan, A. (2019). Scibert: A pretrained language model for scientific text. *arXiv preprint arXiv:1903.10676*.

Best, A., Terpstra, J. L., Moor, G., Riley, B., Norman, C. D., & Glasgow, R. E. (2009). Building knowledge integration systems for evidence-informed decisions.

Journal of health organization and management, *23*(6), 627–641.

Bianco, G. D., Duarte, D., & Gonçalves, M. A. (2023). Reducing the user labeling effort in effective high recall tasks by fine-tuning active learning. *Journal of Intelligent Information Systems*, 1–20.

Bischof, L. (2025, October 21). *AsReviewSLR - EULAR Working Group on AI-Driven Screening.* OSF. Retrieved from `https://doi.org/10.17605/OSF.IO/T8KNJ` doi: 10.17605/OSF.IO/T8KNJ

Boetje, J., & van de Schoot, R. (2024). The safe procedure: a practical stopping heuristic for active learning-based screening in systematic reviews and meta-analyses. *Systematic Reviews*, *13*(1), 81.

Bojanowski, P., Grave, E., Joulin, A., & Mikolov, T. (2016). Enriching word vectors with subword information. *arXiv preprint arXiv:1607.04606*.

Bornmann, L., Haunschild, R., & Mutz, R. (2021). Growth rates of modern science: a latent piecewise growth curve approach to model publication numbers from established and new literature databases. *Humanities and Social Sciences Communications*, *8*(1), 1–15.

Bornmann, L., & Mutz, R. (2015). Growth rates of modern science: A bibliometric analysis based on the number of publications and cited references. *Journal of the association for information science and technology*, *66*(11), 2215–2222.

Boutsidis, C., & Gallopoulos, E. (2008). Svd based initialization: A head start for nonnegative matrix factorization. *Pattern recognition*, *41*(4), 1350–1362.

Bravo, À., Patel, P., & Atanasov, P. (2023). Msr63 implementing simple active learning (al) boosters considerably improves the early identification of relevant studies in the systematic literature review (slr) process. *Value in Health*, *26*(12), S404–S405.

Bron, M. P., van der Heijden, P. G., Feelders, A. J., & Siebes, A. P. (2024). Using chao's estimator as a stopping criterion for technology-assisted review. *arXiv preprint arXiv:2404.01176*.

Brouwer, M., Ferdinands, G., van den Brand, S. A. G. E., de Boer, J., de Bruin, J., Schulte-Frankenfeld, P. M., … Bockting, C. (2023, Jun). *Systematic review data from "psychological theories of depressive relapse and recurrence" (brouwer et al., 2019).* OSF. Retrieved from `osf.io/r45yz` doi: 10.17605/OSF.IO/R45YZ

Brouwer, M., & van de Schoot, R. (2023, Jun). *Results reanalyzing meta-analysis depression data without hard-to-find papers.* OSF. Retrieved from `osf.io/qrdwj` doi: 10.17605/OSF.IO/QRDWJ

Brouwer, M. E., Williams, A. D., Kennis, M., Fu, Z., Klein, N. S., Cuijpers, P., & Bockting, C. L. (2019). Psychological theories of depressive relapse and recurrence: A systematic review and meta-analysis of prospective studies. *Clinical psychology review*, *74*, 101773.

Byrne, F., Hofstee, L., Teijema, J., De Bruin, J., & van de Schoot, R. (2024). Impact of active learning model and prior knowledge on discovery time of elusive relevant papers: a simulation study. *Systematic Reviews*, *13*(1), 175.

Callaghan, M. W., & Mueller-Hansen, F. (2020). Statistical stopping criteria for automated screening in systematic reviews [Journal Article]. *SYSTEMATIC REVIEWS*, *9*(1). doi: 10.1186/s13643-020-01521-4

Campos, D. G., Fütterer, T., Gfrörer, T., Lavelle-Hill, R., Murayama, K., König, L., … Scherer, R. (2024). Screening smarter, not harder: A comparative analysis of machine learning screening algorithms and heuristic stopping criteria for systematic reviews in educational research. *Educational Psychology Review*,

36(1), 19.

Carammia, M., Iacus, S. M., & Porro, G. (2024). *Rethinking scale: The efficacy of fine-tuned open-source llms in large-scale reproducible social science research.* Retrieved from https://arxiv.org/abs/2411.00890

Carey, N., Harte, M., & McCullagh, L. (2021). The use of a text-mining screening tool for systematic review of treatments for relapsed/refractory diffuse large b-cell lymphoma. *International Journal of Technology Assessment in Health Care*, *37*(S1), 2–2.

Carvallo, A., & Parra, D. (n.d.). Comparing word embeddings for document screening based on active learning [Conference Proceedings]. In (Vol. 2414, pp. 100–107).

Carvallo, A., Parra, D., Lobel, H., & Soto, A. (2020). Automatic document screening of medical literature using word and text embeddings in an active learning setting [Journal Article]. *SCIENTOMETRICS*, *125*(3), 3047–3084. doi: 10.1007/s11192-020-03648-6

Chen, T., & Guestrin, C. (2016). Xgboost: A scalable tree boosting system. In *Proceedings of the 22nd acm sigkdd international conference on knowledge discovery and data mining* (pp. 785–794).

Chen, Y., Mani, S., & Xu, H. (2012). Applying active learning to assertion classification of concepts in clinical text. *Journal of biomedical informatics*, *45*(2), 265–272.

Christiansen, B., Neuhaus, U., Schulz, M., Hargreaves, A., Di Marco, A., Proietti, G., … others (2022). A manifesto for applied data science-reasoning from a business perspective. In *itadata* (pp. 123–131).

Cleveland, W. S. (2001). Data science: an action plan for expanding the technical areas of the field of statistics. *International statistical review*, *69*(1), 21–26.

Cohen, A. M., Hersh, W. R., Peterson, K., & Yen, P.-Y. (2006). Reducing workload in systematic review preparation using automated citation classification. *Journal of the American Medical Informatics Association*, *13*(2), 206–219.

Collobert, R., & Weston, J. (2008). A unified architecture for natural language processing: Deep neural networks with multitask learning. In *Proceedings of the 25th international conference on machine learning* (pp. 160–167).

Cormack, G. V., & Grossman, M. R. (n.d.-a). Scalability of continuous active learning for reliable high-recall text classification [Conference Proceedings]. In *Cikm'16: Acm conference on information and knowledge management* (pp. 1039–1048). Indianapolis Indiana USA: ACM. doi: 10.1145/2983323.2983776

Cormack, G. V., & Grossman, M. R. (n.d.-b). Technology-assisted review in empirical medicine: Waterloo participation in clef ehealth 2017 [Conference Proceedings]. In (Vol. 1866).

Cormack, G. V., & Grossman, M. R. (n.d.-c). Technology-assisted review in empirical medicine: Waterloo participation in clef ehealth 2018 [Conference Proceedings]. In (Vol. 2125).

Cormack, G. V., & Grossman, M. R. (2015). Autonomy and reliability of continuous active learning for technology-assisted review [Journal Article]. *arXiv:1504.06868 [cs]*.

Cottier, B., Rahman, R., Fattorini, L., Maslej, N., & Owen, D. (2024). *The rising costs of training frontier ai models.*

Cowie, K., Rahmatullah, A., Hardy, N., Holub, K., & Kallmes, K. (2022). Web-based software tools for systematic literature review in medicine: Systematic search and feature analysis. *JMIR Medical Informatics*, *10*(5), e33219. Retrieved from

https://doi.org/10.2196/33219 doi: 10.2196/33219

Cumpston, M., Li, T., Page, M. J., Chandler, J., Welch, V. A., Higgins, J. P., & Thomas, J. (2019). Updated guidance for trusted systematic reviews: a new edition of the cochrane handbook for systematic reviews of interventions. *The Cochrane database of systematic reviews*, *2019*(10).

Davenport, T., & Malone, K. (2021). Deployment as a critical business data science discipline. *Harvard Data Science Review*, *3*(1).

de Boer, J., Hofstee, L., Hindriks, S., & van de Schoot, R. (2021, April). *Systematic reviews at utrecht university and umc utrecht 2020.* Zenodo. Retrieved from https://doi.org/10.5281/zenodo.4725568 doi: 10.5281/zenodo.4725568

De Bruin, J. (2023, 1). Scitree - like tree, but optimized for science. (If you use this software, please cite it as below.) doi: 10.5281/zenodo.7500084

de Bruin, J., Lombaers, P., Kaandorp, C., Teijema, J. J., van der Kuil, T., Yazan, B., … van de Schoot, R. (2025). Asreview lab v2: Open-source text screening with multiple agents and oracles. *Available at SSRN 5136987*.

den Boer, R., Hofstee, L., Leenaars, C., Bagheri, A., Teijema, J., & van de Schoot, R. (2024). Advancing multilingual abstract classification: A comparative analysis of feature extraction models in systematic reviews.

developers, A. L. (2022, June). *Asreview wordcloud: A tool to create a visual impression of the verbal content within a systematic review dataset.* Zenodo. Retrieved from https://doi.org/10.5281/zenodo.6625855 doi: 10.5281/zenodo.6625855

developers, A. L. (2025a, February). *Asreview datatools.* Zenodo. Retrieved from https://doi.org/10.5281/zenodo.14825952 doi: 10.5281/zenodo.14825952

developers, A. L. (2025b, June). *Asreview dory - new and exciting models for asreview.* Zenodo. Retrieved from https://doi.org/10.5281/zenodo.15649248 doi: 10.5281/zenodo.15649248

developers, A. L. (2025c, June). *Asreview insights - insights and plotting tool for the asreview project.* Zenodo. Retrieved from https://doi.org/10.5281/zenodo.15639008 doi: 10.5281/zenodo.15639008

de Vries, W., van Cranenburgh, A., Bisazza, A., Caselli, T., van Noord, G., & Nissim, M. (2019). *Bertje: A dutch bert model.* Retrieved from https://arxiv.org/abs/1912.09582

Di Nunzio, G. M. (n.d.). Finding all the needles in a haystack: A system to estimate the costs of e-discovery and systematic reviews [Conference Proceedings]. In (Vol. 2167, p. 106).

Donners, A. A., Rademaker, C. M., Bevers, L. A., Huitema, A. D., Schutgens, R. E., Egberts, T. C., & Fischer, K. (2021). Pharmacokinetics and associated efficacy of emicizumab in humans: a systematic review. *Clinical Pharmacokinetics*, *60*(11), 1395–1406.

Drozdz, J. A., & Ladomery, M. R. (2024). The peer review process: past, present, and future. *British Journal of Biomedical Science*, *81*, 12054.

Elmore, R., Schmidt, L., Lam, J., Howard, B. E., Tandon, A., Norman, C., … Shah, R. R. (2020). Risk and protective factors in the covid-19 pandemic: A rapid evidence map [Journal Article]. *FRONTIERS IN PUBLIC HEALTH*, *8*. doi: 10.3389/fpubh.2020.582205

Feng, F., Yang, Y., Cer, D., Arivazhagan, N., & Wang, W. (2020). Language-agnostic BERT sentence embedding. *CoRR*, *abs/2007.01852*. Retrieved from https://arxiv.org/abs/2007.01852

Feng, F., Yang, Y., Cer, D., Arivazhagan, N., & Wang, W. (2022). *Language-agnostic bert sentence embedding.* Retrieved from https://arxiv.org/abs/2007.01852

Ferdinands, G. (2021). Ai-assisted systematic reviewing: selecting studies to compare bayesian versus frequentist sem for small sample sizes. *Multivariate Behavioral Research*, *56*(1), 153–154.

Ferdinands, G., Schram, R., de Bruin, J., Bagheri, A., Oberski, D. L., Tummers, L., … van de Schoot, R. (2023). Performance of active learning models for screening prioritization in systematic reviews: a simulation study into the average time to discover relevant records. *Systematic Reviews*, *12*(1), 100.

Ferdinands, G., Teijema, J., De Bruin, J., Brouwer, M., & Van de Schoot, R. (2022, July). *Scripts and output for the simulation study determining the time to discovery for the depression data.* Zenodo. Retrieved from https://doi.org/10.5281/zenodo.6839002 doi: 10.5281/zenodo.6839002

Fortunato, S., Bergstrom, C. T., Börner, K., Evans, J. A., Helbing, D., Milojević, S., … others (2018). Science of science. *Science*, *359*(6379), eaao0185.

Fu, Z., Brouwer, M., Kennis, M., Williams, A., Cuijpers, P., & Bockting, C. (2021). Psychological factors for the onset of depression: a meta-analysis of prospective studies. *BMJ open*, *11*(7), e050129.

Gama, J., Žliobaitė, I., Bifet, A., Pechenizkiy, M., & Bouchachia, A. (2014). A survey on concept drift adaptation. *ACM computing surveys (CSUR)*, *46*(4), 1–37.

Gates, A., Johnson, C., & Hartling, L. (2018). Technology-assisted title and abstract screening for systematic reviews: a retrospective evaluation of the abstrackr machine learning tool. *Systematic reviews*, *7*, 1–9.

Gelsleichter, Y., Banzi, R., Naudet, F., Vinatier, C., Kertész, I., & Varga, M. (2025). Survey about barriers and solutions for enhancing computational reproducibility in scientific research [version 1; peer review: awaiting peer review]. *F1000Research*, *14*(1278). doi: 10.12688/f1000research.172013.1

Ghai, B., Liao, Q. V., Zhang, Y., Bellamy, R. K. E., & Mueller, K. (2020). Explainable active learning (XAL): an empirical study of how local explanations impact annotator experience. *CoRR*, *abs/2001.09219*. Retrieved from https://arxiv.org/abs/2001.09219

Gibney, E. (2022). Is ai fuelling a reproducibility crisis in science. *Nature*, *608*(7922), 250–1.

Gibney, E. (2024). These ai firms publish the world's most highly cited work. *Nature*, *632*(8025), 487–487.

Glorot, X., Bordes, A., & Bengio, Y. (2011). Deep sparse rectifier neural networks. In *Proceedings of the fourteenth international conference on artificial intelligence and statistics* (pp. 315–323).

Gomes, D. G., Pottier, P., Crystal-Ornelas, R., Hudgins, E. J., Foroughirad, V., Sánchez-Reyes, L. L., … others (2022). Why don't we share data and code? perceived barriers and benefits to public archiving practices. *Proceedings of the Royal Society B*, *289*(1987), 20221113.

Goodfellow, I., Bengio, Y., Courville, A., & Bengio, Y. (2016). *Deep learning* (Vol. 1) (No. 2). MIT press Cambridge.

Gregor, K., Meyer, B. F., Gaida, T., Justo Vasquez, V., Bett-Williams, K., Forrest, M., … Rammig, A. (2025). Best practices in software development for robust and reproducible geoscientific models based on insights from the global carbon project models. *EGUsphere*, *2025*, 1–44. Retrieved from https://egusphere.copernicus.org/preprints/2025/egusphere-2025-1733/ doi: 10

.5194/egusphere-2025-1733

Gundersen, O. E., Gil, Y., & Aha, D. W. (2018). On reproducible ai: Towards reproducible research, open science, and digital scholarship in ai publications. *AI magazine*, *39*(3), 56–68.

Guo, F., Luo, Y., Yang, L., & Zhang, Y. (2023). Scimine: An efficient systematic prioritization model based on richer semantic information. In *Proceedings of the 46th international acm sigir conference on research and development in information retrieval* (pp. 205–215).

Hagendorff, T., & Meding, K. (2021, September). Ethical considerations and statistical analysis of industry involvement in machine learning research. *AI; SOCIETY*, *38*(1), 35–45. Retrieved from http://dx.doi.org/10.1007/s00146-021-01284-z doi: 10.1007/s00146-021-01284-z

Halfpenny, N., Alleman, C., Eaton, J., & van Vliet, M. (2019). Pns335: Using machine learning for efficiency improvements in systematic literature reviews of clinical efficacy and safety [Journal Article]. *Value in Health*, *22*((Halfpenny N.; Eaton J.) Pharmerit International, York, United Kingdom), S821. doi: 10.1016/j.jval.2019.09.2235

Hall, T., Beecham, S., Bowes, D., Gray, D., & Counsell, S. (2012). A systematic literature review on fault prediction performance in software engineering. *IEEE Transactions on Software Engineering*, *38*(6), 1276-1304. doi: 10.1109/TSE.2011.103

Hamel, C., Kelly, S. E., Thavorn, K., Rice, D. B., Wells, G. A., & Hutton, B. (2020). An evaluation of distillersr's machine learning-based prioritization tool for title/abstract screening – impact on reviewer-relevant outcomes [Journal Article]. *BMC Medical Research Methodology*, *20*(1). doi: 10.1186/s12874-020-01129-1

Harmsen, W., de Groot, J., Harkema, A., van Dusseldorp, I., de Bruin, J., van den Brand, S., & van de Schoot, R. (2024). Machine learning to optimize literature screening in medical guideline development. *Systematic Reviews*, *13*(1), 177.

Hashimoto, K., Kontonatsios, G., Miwa, M., & Ananiadou, S. (2016). Topic detection using paragraph vectors to support active learning in systematic reviews [Journal Article]. *JOURNAL OF BIOMEDICAL INFORMATICS*, *62*, 59–65. doi: 10.1016/j.jbi.2016.06.001

Hastie, T., Tibshirani, R., Friedman, J., et al. (2009). *The elements of statistical learning*. Springer series in statistics New-York.

Herrmann, M., Lange, F. J. D., Eggensperger, K., Casalicchio, G., Wever, M., Feurer, M., … Bischl, B. (2024). Position: Why we must rethink empirical research in machine learning. *arXiv preprint arXiv:2405.02200*.

Higgins, J. P., Thomas, J., Chandler, J., Cumpston, M., Li, T., Page, M. J., & Welch, V. A. (2019). *Cochrane handbook for systematic reviews of interventions*. John Wiley & Sons.

Hou, J., Wang, X., Dubois, J.-J., Rice, R. B., Haddock, A., & Wang, Y. (2022). Extreme systematic reviews: A large literature screening dataset to support environmental policymaking. In *Proceedings of the 31st acm international conference on information & knowledge management* (pp. 4029–4033).

Howard, B. E., Phillips, J., Tandon, A., Maharana, A., Elmore, R., Mav, D., … Shah, R. R. (2020). Swift-active screener: Accelerated document screening through active learning and integrated recall estimation [Journal Article]. *Environment International*, *138*, 105623. doi: 10.1016/j.envint.2020.105623

Hughes, M., Li, I., Kotoulas, S., & Suzumura, T. (2017). Medical text classification using convolutional neural networks. In *Informatics for health: connected citizen-led wellness and population health* (pp. 246–250). IOS Press.

Hunter-Zinck, H., De Siqueira, A. F., Vásquez, V. N., Barnes, R., & Martinez, C. C. (2021). *Ten simple rules on writing clean and reliable open-source scientific software* (Vol. 17) (No. 11). Public Library of Science San Francisco, CA USA.

Irizarry, R. (2020). *The role of academia în data science education. harvard data science review, 2 (1)*.

ISO. (2010). *Ergonomics of human-system interaction – part 210: Human-centred design for interactive systems* (No. 9241-210). Geneva, Switzerland: ISO.

ISO. (2018). *Ergonomics of human-system interaction – part 11: Usability: Definitions and concepts* (No. 9241-11). Geneva, Switzerland: ISO.

Jacobsen, A., de Miranda Azevedo, R., Juty, N., Batista, D., Coles, S., Cornet, R., … others (2020). *Fair principles: interpretations and implementation considerations* (Vol. 2) (No. 1-2). MIT Press One Rogers Street, Cambridge, MA 02142-1209, USA journals-info ….

Jaderberg, M., Simonyan, K., Vedaldi, A., & Zisserman, A. (2016). Reading text in the wild with convolutional neural networks. *International journal of computer vision*, *116*(1), 1–20.

Jain, S., & Wallace, B. C. (2019). Attention is not explanation. *arXiv preprint arXiv:1902.10186*.

Jentzsch, S., & Turan, C. (2022). Gender bias in bert - measuring and analysing biases through sentiment rating in a realistic downstream classification task. In *Proceedings of the 4th workshop on gender bias in natural language processing (gebnlp)* (p. 184–199). Association for Computational Linguistics. Retrieved from http://dx.doi.org/10.18653/v1/2022.gebnlp-1.20 doi: 10.18653/v1/2022.gebnlp-1.20

Jeyaraman, M., Muthu, S., & Ganie, P. A. (2021). Does the source of mesenchymal stem cell have an effect in the management of osteoarthritis of the knee? meta-analysis of randomized controlled trials. *Cartilage*, *13*(1_suppl), 1532S–1547S.

Jimenez, R. C., Lee, T., Rosillo, N., Cordova, R., Cree, I. A., Gonzalez, A., & Ruiz, B. I. I. (2022). Machine learning computational tools to assist the performance of systematic reviews: A mapping review. *BMC Medical Research Methodology*, *22*(1). Retrieved from https://doi.org/10.1186/s12874-022-01805-4 doi: 10.1186/s12874-022-01805-4

Johnson, J. A., Rodeberg, N. T., & Wightman, R. M. (2016). Failure of standard training sets in the analysis of fast-scan cyclic voltammetry data. *ACS chemical neuroscience*, *7*(3), 349–359.

Jourdan, F., Picard, A., Fel, T., Risser, L., Loubes, J. M., & Asher, N. (2023). Cockatiel: Continuous concept ranked attribution with interpretable elements for explaining neural net classifiers on nlp tasks. *arXiv preprint arXiv:2305.06754*.

Jurowetzki, R., Hain, D., Mateos-Garcia, J., & Stathoulopoulos, K. (2021). The privatization of ai research (-ers): Causes and potential consequences–from university-industry interaction to public research brain-drain? *arXiv preprint arXiv:2102.01648*.

Kanoulas, E., Li, D., Azzopardi, L., & Spijker, R. (n.d.-a). Clef 2017 technologically assisted reviews in empirical medicine overview [Conference Proceedings].

Kanoulas, E., Li, D., Azzopardi, L., & Spijker, R. (n.d.-b). Clef 2018 technologically assisted reviews in empirical medicine overview [Conference Proceedings]. In

Clef 2018 technologically assisted reviews in empirical medicine overview.

Kanoulas, E., Li, D., Azzopardi, L., & Spijker, R. (n.d.-c). Clef 2019 technology assisted reviews in empirical medicine overview [Conference Proceedings]. In (Vol. 2380).

Kapoor, S., & Narayanan, A. (2023). Leakage and the reproducibility crisis in machine-learning-based science. *Patterns*, *4*(9).

KDD25 ADS chairs. (2025). *Applied data science (ads) track: Call for papers – kdd 2025.* Retrieved 2025-12-01, from https://kdd2025.kdd.org/applied-data-science-ads-track-call-for-papers/

Kennis, M., Gerritsen, L., van Dalen, M., Williams, A., Cuijpers, P., & Bockting, C. (2020). Prospective biomarkers of major depressive disorder: a systematic review and meta-analysis. *Molecular psychiatry*, *25*(2), 321–338.

Keren, L. S., Liberzon, A., & Lazebnik, T. (2023). A computational framework for physics-informed symbolic regression with straightforward integration of domain knowledge. *Scientific Reports*, *13*(1), 1249.

Khalil, H., Ameen, D., & Zarnegar, A. (2022). Tools to support the automation of systematic reviews: a scoping review. *Journal of Clinical Epidemiology*, *144*, 22–42. Retrieved from https://doi.org/10.1016/j.jclinepi.2021.12.005 doi: 10.1016/j.jclinepi.2021.12.005

Kreuzberger, D., Kühl, N., & Hirschl, S. (2022). Machine learning operations (mlops): Overview. *Definition, and Architecture. arXiv*, *20222205*.

Lamprecht, A.-L., Garcia, L., Kuzak, M., Martinez, C., Arcila, R., Martin Del Pico, E., … others (2020). Towards fair principles for research software. *Data Science*, *3*(1), 37–59.

Landhuis, E. (2016). Scientific literature: Information overload. *Nature*, *535*(7612), 457–458.

Laskar, M. T. R., Bari, M. S., Rahman, M., Bhuiyan, M. A. H., Joty, S., & Huang, J. X. (2023). *A systematic study and comprehensive evaluation of chatgpt on benchmark datasets.* Retrieved from https://arxiv.org/abs/2305.18486

Le, Q., & Mikolov, T. (2014). Distributed representations of sentences and documents. In *International conference on machine learning* (pp. 1188–1196).

Lee, D., & Seung, H. S. (2000). Algorithms for non-negative matrix factorization. *Advances in neural information processing systems*, *13*.

Lee, D. D., & Seung, H. S. (1999). Learning the parts of objects by non-negative matrix factorization. *nature*, *401*(6755), 788–791.

Lee, S., Shakir, A., Koenig, D., & Lipp, J. (2024). *Open source strikes bread - new fluffy embeddings model.* Retrieved from https://www.mixedbread.ai/blog/mxbai-embed-large-v1

Leenaars, C. H., Drinkenburg, W. P., Nolten, C., Dematteis, M., Joosten, R. N., Feenstra, M. G., & De Vries, R. B. (2019). Sleep and microdialysis: an experiment and a systematic review of histamine and several amino acids. *Journal of Circadian Rhythms*, *17*.

Leung, C. K., Pasi, G., & Wang, L. (2023). Theoretical and practical data science and analytics: challenges and solutions. *International Journal of Data Science and Analytics*, *16*(4), 403–406.

Li, D., Zafeiriadis, P., & Kanoulas, E. (n.d.). Aps: An active pubmed search system for technology assisted reviews [Conference Proceedings]. In (pp. 2137–2140). doi: 10.1145/3397271.3401401

Li, Z., Gurgel, H., Dessay, N., Hu, L., Xu, L., & Gong, P. (2020). Semi-supervised

text classification framework: An overview of dengue landscape factors and satellite earth observation [Journal Article]. *INTERNATIONAL JOURNAL OF ENVIRONMENTAL RESEARCH AND PUBLIC HEALTH*, *17*(12). doi: 10.3390/ijerph17124509

Liu, J., Timsina, P., & El-Gayar, O. (2018). A comparative analysis of semi-supervised learning: The case of article selection for medical systematic reviews [Journal Article]. *INFORMATION SYSTEMS FRONTIERS*, *20*(2, SI), 195–207. doi: 10.1007/s10796-016-9724-0

Liu, O. L., Lee, H.-S., Hofstetter, C., & Linn, M. C. (2008). Assessing knowledge integration in science: Construct, measures, and evidence. *Educational Assessment*, *13*(1), 33–55.

Lombaers, P., de Bruin, J., & van de Schoot, R. (2024). Reproducibility and data storage for active learning-aided systematic reviews. *Applied Sciences*, *14*(9). Retrieved from https://www.mdpi.com/2076-3417/14/9/3842 doi: 10.3390/app14093842

Lundberg, S. M., & Lee, S.-I. (2017). A unified approach to interpreting model predictions. Curran Associates, Inc. Retrieved from http://papers.nips.cc/paper/7062-a-unified-approach-to-interpreting-model-predictions.pdf

Maitner, B., Santos Andrade, P. E., Lei, L., Kass, J., Owens, H. L., Barbosa, G. C., … others (2024). Code sharing in ecology and evolution increases citation rates but remains uncommon. *Ecology and Evolution*, *14*(8), e70030.

Management of JADS. (2023). *Journal of applied data sciences has been accepted for inclusion in scopus.* Retrieved from https://bright-journal.org/Journal/index.php/JADS/announcement/view/1

Markus, A. F., Kors, J. A., & Rijnbeek, P. R. (2021). The role of explainability in creating trustworthy artificial intelligence for health care: a comprehensive survey of the terminology, design choices, and evaluation strategies. *Journal of biomedical informatics*, *113*, 103655.

Marshall, I. J., & Wallace, B. C. (2019). Toward systematic review automation: a practical guide to using machine learning tools in research synthesis [Journal Article]. *Syst Rev*, *8*(1), 163. doi: 10.1186/s13643-019-1074-9

Mauricio, D., & Gonzalez, N. (n.d.). *Optimización de estrategias de búsquedas científicas médicas utilizando técnicas de inteligencia artificial* (Doctoral dissertation). doi: 10.11144/javeriana.10554.58492

McIntosh, S., Kamei, Y., Adams, B., & Hassan, A. E. (2016). An empirical study of the impact of modern code review practices on software quality. *Empirical Software Engineering*, *21*(5), 2146–2189.

Menon, J., Struijs, F., & Whaley, P. (2022). The methodological rigour of systematic reviews in environmental health. *Critical Reviews in Toxicology*, *52*(3), 167–187.

Miwa, M., Thomas, J., O'Mara-Eves, A., & Ananiadou, S. (2014). Reducing systematic review workload through certainty-based screening [Journal Article]. *Journal of Biomedical Informatics*, *51*, 242–253. doi: 10.1016/j.jbi.2014.06.005

Moher, D., Shamseer, L., Clarke, M., Ghersi, D., Liberati, A., Petticrew, M., … Group, P.-P. (2015). Preferred reporting items for systematic review and meta-analysis protocols (prisma-p) 2015 statement. *Systematic reviews*, *4*(1), 1.

Molinari, A., & Kanoulas, E. (2022). Transferring knowledge between topics in systematic reviews. *Intelligent Systems with Applications*, *16*, 200150.

Mons, B., Neylon, C., Velterop, J., Dumontier, M., da Silva Santos, L. O. B., & Wilkinson, M. D. (2017). Cloudy, increasingly fair; revisiting the fair data

guiding principles for the european open science cloud. *Information Services and Use*, *37*(1), 49–56.

Montavon, G., Orr, G., & Müller, K.-R. (2012). *Neural networks: tricks of the trade* (Vol. 7700). springer.

Montévil, M. (2021). Computational empiricism: the reigning épistémè of the sciences. *Philosophy World Democracy*.

Muthu, S. (2022). The efficiency of machine learning-assisted platform for article screening in systematic reviews in orthopaedics. *International Orthopaedics*, 1–6.

Muthu, S., & Ramakrishnan, E. (2021). Fragility analysis of statistically significant outcomes of randomized control trials in spine surgery: a systematic review. *Spine*, *46*(3), 198–208.

Naseem, U., Razzak, I., Khan, S. K., & Prasad, M. (2021). A comprehensive survey on word representation models: From classical to state-of-the-art word representation language models. *Transactions on Asian and Low-Resource Language Information Processing*, *20*(5), 1–35.

Naur, P. (1974). Concise survey of computer methods. *(No Title)*.

Nedelcu, A., Oerther, B., Engel, H., Sigle, A., Schmucker, C., Schoots, I. G., … Benndorf, M. (2023). A machine learning framework reduces the manual workload for systematic reviews of the diagnostic performance of prostate magnetic resonance imaging. *European Urology Open Science*, *56*, 11–14.

Neeleman, R., Leenaars, C. H., Oud, M., Weijdema, F., & van de Schoot, R. (2024). Addressing the challenges of reconstructing systematic reviews datasets: a case study and a noisy label filter procedure. *Systematic Reviews*, *13*(1), 69.

Norman, C., Leeflang, M., & Névéol, A. (n.d.). Limsi@clef ehealth 2018 task 2: Technology assisted reviews by stacking active and static learning [Conference Proceedings]. In (Vol. 2125).

Norman, C. R., Leeflang, M. M. G., Porcher, R., & Neveol, A. (2019). Measuring the impact of screening automation on meta-analyses of diagnostic test accuracy [Journal Article]. *SYSTEMATIC REVIEWS*, *8*(1). doi: 10.1186/s13643-019-1162-x

O'Connor, A. M., Tsafnat, G., Gilbert, S. B., Thayer, K. A., Shemilt, I., Thomas, J., … Wolfe, M. S. (2019). Still moving toward automation of the systematic review process: a summary of discussions at the third meeting of the international collaboration for automation of systematic reviews (icasr) [Journal Article]. *Syst Rev*, *8*(1), 57. doi: 10.1186/s13643-019-0975-y

Olorisade, B. K., Brereton, P., & Andras, P. (2017). Reproducibility of studies on text mining for citation screening in systematic reviews: Evaluation and checklist [Journal Article]. *J Biomed Inform*, *73*, 1–13. (Olorisade, Babatunde Kazeem Brereton, Pearl Andras, Peter eng 2017/07/18 J Biomed Inform. 2017 Sep;73:1-13. doi: 10.1016/j.jbi.2017.07.010. Epub 2017 Jul 12.) doi: 10.1016/j.jbi.2017.07.010

Olorisade, B. K., De Quincey, E., Andras, P., & Brereton, P. (n.d.). A critical analysis of studies that address the use of text mining for citation screening in systematic reviews [Conference Proceedings]. In (Vol. 01-03-June-2016). doi: 10.1145/2915970.2915982

Orel, E., Ciglenecki, I., Thiabaud, A., Temerev, A., Calmy, A., Keiser, O., & Merzouki, A. (2023). An automated literature review tool (literev) for streamlining and accelerating research using natural language processing and machine learning:

descriptive performance evaluation study. *Journal of medical Internet research*, *25*, e39736.

Oud, M., Arntz, A., Hermens, M. L., Verhoef, R., & Kendall, T. (2018). Specialized psychotherapies for adults with borderline personality disorder: A systematic review and meta-analysis. *Australian & New Zealand Journal of Psychiatry*, *52*(10), 949–961.

Oude Wolcherink, M., Pouwels, X., van Dijk, S., Doggen, C., & Koffijberg, H. (2023). Can artificial intelligence separate the wheat from the chaff in systematic reviews of health economic articles? *Expert Review of Pharmacoeconomics & Outcomes Research*, *23*(9), 1049–1056.

Owens, B. (2024). Rage against machine learning driven by profit. *Nature*, S6–S9. Retrieved from https://doi.org/10.1038/d41586-024-02985-3 doi: 10.1038/d41586-024-02985-3

O'Mara-Eves, A., Thomas, J., McNaught, J., Miwa, M., & Ananiadou, S. (2015). Using text mining for study identification in systematic reviews: a systematic review of current approaches [Journal Article]. *Systematic reviews*, *4*(1), 1–22.

Page, M. J., Moher, D., Bossuyt, P. M., Boutron, I., Hoffmann, T. C., Mulrow, C. D., … others (2021). Prisma 2020 explanation and elaboration: updated guidance and exemplars for reporting systematic reviews. *bmj*, *372*.

Parker, R., Graff, D., Kong, J., Chen, K., & Maeda, K. (2011). English gigaword. *Linguistic Data Consortium*.

Pei, G. (2025). Open science falling behind in the era of artificial intelligence. *Frontiers in Research Metrics and Analytics*, *10*, 1595824.

Pellegrini, M., & Marsili, F. (2021). Evaluating software tools to conduct systematic reviews: a feature analysis and user survey. *Form@re - Open Journal per la formazione in rete*, *21*(2), 124–140. Retrieved from https://doi.org/10.36253/form-11343 doi: 10.36253/form-11343

Pijls, B. G. (2023). Machine learning assisted systematic reviewing in orthopaedics. *Journal of Orthopaedics*.

Price, D. J. D. S. (1963). *Little science, big science*. Columbia university press.

Przybyła, P., Brockmeier, A. J., Kontonatsios, G., Le Pogam, M.-A., McNaught, J., von Elm, E., … Ananiadou, S. (2018). Prioritising references for systematic reviews with robotanalyst: a user study. *Research synthesis methods*, *9*(3), 470–488.

Pytlak, R., Bukhvalova, B., Cichosz, P., Fajdek, B., Grahek-Ogden, D., Jastrzebski, B., … Waszkowski, R. (2023). Machine learning based system for the automation of systematic literature reviews. In *2023 ieee international conference on bioinformatics and biomedicine (bibm)* (pp. 4389–4397).

Rahimi, A., & Recht, B. (2017). *Reflections on random kitchen sinks*. Retrieved 2025-12-03, from https://archives.argmin.net/2017/12/05/kitchen-sinks/

Rathbone, J., Hoffmann, T., & Glasziou, P. (2015). Faster title and abstract screening? evaluating abstrackr, a semi-automated online screening program for systematic reviewers [Journal Article]. *Syst Rev*, *4*, 80. (Rathbone, John Hoffmann, Tammy Glasziou, Paul eng Research Support, Non-U.S. Gov't England Syst Rev. 2015 Jun 15;4:80. doi: 10.1186/s13643-015-0067-6.) doi: 10.1186/s13643-015-0067-6

Reimers, N., & Gurevych, I. (2019, 11). Sentence-bert: Sentence embeddings using siamese bert-networks. In *Proceedings of the 2019 conference on empirical methods in natural language processing*. Association for Computational Linguistics. Retrieved from http://arxiv.org/abs/1908.10084

Ribeiro, M. T., Singh, S., & Guestrin, C. (2016). "why should I trust you?": Explaining the predictions of any classifier. *CoRR*, *abs/1602.04938*. Retrieved from http://arxiv.org/abs/1602.04938

Robledo, S., Aguirre, A. M. G., Hughes, M., & Eggers, F. (2021). "hasta la vista, baby" – will machine learning terminate human literature reviews in entrepreneurship? *Journal of Small Business Management*, 1–30. Retrieved from https://doi.org/10.1080/00472778.2021.1955125 doi: 10.1080/00472778.2021.1955125

Rolnick, D., & Tegmark, M. (2017). The power of deeper networks for expressing natural functions. *arXiv preprint arXiv:1705.05502*.

Romanov, S., Siqueira, A. S., de Bruin, J., Teijema, J., Hofstee, L., & van de Schoot, R. (2024, 01). Optimizing ASReview simulations: A generic multiprocessing solution for 'light-data' and 'heavy-data' users. *Data Intelligence*, 1-19. Retrieved from https://doi.org/10.1162/dint_a_00244 doi: 10.1162/dint_a_00244

Ros, R., Bjarnason, E., & Runeson, P. (2017). A machine learning approach for semi-automated search and selection in literature studies. In *Proceedings of the 21st international conference on evaluation and assessment in software engineering* (pp. 118–127).

Rudin, C. (2019). Stop explaining black box machine learning models for high stakes decisions and use interpretable models instead. *Nature machine intelligence*, *1*(5), 206–215.

Ruijer, E., Hassink, W., & Brinkhuis, M. J. S. (n.d.). Towards a framework for digital governance curricula.
doi: 10.1177/15701255251406770

Sadowski, C., Söderberg, E., Church, L., Sipko, M., & Bacchelli, A. (2018). Modern code review: a case study at google. In *Proceedings of the 40th international conference on software engineering: Software engineering in practice* (pp. 181–190).

Saeidmehr, A., Steel, P. D. G., & Samavati, F. F. (2024). Systematic review using a spiral approach with machine learning. *Systematic Reviews*, *13*(1), 32.

Samuel, S., Löffler, F., & König-Ries, B. (2020). Machine learning pipelines: provenance, reproducibility and fair data principles. In *International provenance and annotation workshop* (pp. 226–230).

Savchenko, E., & Lazebnik, T. (2022). Computer aided functional style identification and correction in modern russian texts. *Journal of Data, Information and Management*, *4*(1), 25–32.

Scott, A. M., Forbes, C., Clark, J., Carter, M., Glasziou, P., & Munn, Z. (2021). Systematic review automation tools improve efficiency but lack of knowledge impedes their adoption: a survey. *Journal of Clinical Epidemiology*, *138*, 80–94. Retrieved from https://doi.org/10.1016/j.jclinepi.2021.06.030 doi: 10.1016/j.jclinepi.2021.06.030

Sebastiani, F. (2002). Machine learning in automated text categorization. *ACM computing surveys (CSUR)*, *34*(1), 1–47.

Semmelrock, H., Ross-Hellauer, T., Kopeinik, S., Theiler, D., Haberl, A., Thalmann, S., & Kowald, D. (2025). Reproducibility in machine-learning-based research: Overview, barriers, and drivers. *AI Magazine*, *46*(2), e70002.

Sep, M. S., Vellinga, M., Sarabdjitsingh, R. A., & Joëls, M. (2021). The rodent object-in-context task: A systematic review and meta-analysis of important variables. *PloS one*, *16*(7), e0249102.

Settles, B. (2009). *Active learning literature survey* (Tech. Rep.).
Settles, B. (2012). *Active learning* (Vol. 6) (No. 1). Springer Cham. doi: 10.1007/978-3-031-01560-1
Sheikholeslami, S. (2019). *Ablation programming for machine learning.*
Sparck Jones, K. (1972). A statistical interpretation of term specificity and its application in retrieval. *Journal of documentation*, *28*(1), 11–21.
Spruit, M. (2021). *Translational data science in population health.* Universiteit Leiden. Retrieved from https://scholarlypublications.universiteitleiden.nl/access/item%3A3567345/view (Inaugural Lecture)
Srivastava, N., Hinton, G., Krizhevsky, A., Sutskever, I., & Salakhutdinov, R. (2014). Dropout: a simple way to prevent neural networks from overfitting. *The journal of machine learning research*, *15*(1), 1929–1958.
Stoel, L., Mourits, G., & van de Schoot, R. (2023). *Procedure and results for the initial selection of software for systematically screening large amounts of textual data implementing active learning.* OSF. Retrieved from osf.io/g3nkz doi: 10.17605/OSF.IO/G3NKZ
Streiber, A. M., Hoepel, S. J., Blok, E., Van Rooij, F. J., Neitzel, J., Labrecque, J., … Bos, D. (2025). Improving reproducibility of data analysis and code in medical research: 5 recommendations to get started. *BMJ open*, *15*(10), e104691.
Sun, C., Qiu, X., Xu, Y., & Huang, X. (2020). *How to fine-tune bert for text classification?*
Teijema, J. (2021a, July). *Asreview cnn 17 layer model plugin.* Zenodo. Retrieved from https://doi.org/10.5281/zenodo.5084887 doi: 10.5281/zenodo.5084887
Teijema, J. (2021b, July). *A code repository for: 'a comparison of performance between optimal neural networks and classical algorithms in active learning based text classification'.* Zenodo. Retrieved from https://doi.org/10.5281/zenodo.5084912 doi: 10.5281/zenodo.5084912
Teijema, J. (2023). *jteijema/code-for-simulation-based-active-learning-for-systematic-reviews: v1.0.* Zenodo. doi: 10.5281/zenodo.8095084
Teijema, J. (2025, August). *jteijema/asr-xai: v0.1.* Zenodo. Retrieved from https://doi.org/10.5281/zenodo.16797395 doi: 10.5281/zenodo.16797395
Teijema, J., Van de Schoot, R., & Bagheri, A. (2022, July). *A code repository for: Active learning-based systematic reviewing using switching classification models: the case of the onset, maintenance, and relapse of depressive disorders.* Zenodo. Retrieved from https://doi.org/10.5281/zenodo.6799806 doi: 10.5281/zenodo.6799806
Teijema, J. J. (2023a, 5). *jteijema/asreview-simulation-project: v1.1.4.* Zenodo. doi: 10.5281/zenodo.7993561
Teijema, J. J. (2023b). *Simulation Data for: Large-Scale Simulation Study of Active Learning models for Systematic Reviews.* DataverseNL. Retrieved from https://doi.org/10.34894/NYFSJY doi: 10.34894/NYFSJY
Teijema, J. J. (2024, March). *jteijema/synergy-simulations-website: Release on Zenodo.* Zenodo. Retrieved from doi.org/10.5281/zenodo.13169790 doi: 10.5281/zenodo.13169790
Teijema, J. J., de Bruin, J., Bagheri, A., & van de Schoot, R. (2025). Large-scale simulation study of active learning models for systematic reviews. *International Journal of Data Science and Analytics*, 1–22. doi: https://doi.org/10.1007/s41060-025-00777-0
Teijema, J. J., Hofstee, L., Brouwer, M., de Bruin, J., Ferdinands, G., de Boer, J., …

others (2023). Active learning-based systematic reviewing using switching classification models: the case of the onset, maintenance, and relapse of depressive disorders. *Frontiers in Research Metrics and Analytics*, *8*, 1178181.

Teijema, J. J., Ribeiro, G., Seuren, S., Anadria, D., Bagheri, A., & van de Schoot, R. (2025). Simulation-based active learning for systematic reviews: A scoping review of literature. *Journal of Information Science*.

Teijema, J. J., van den Brand, S. A. G. E., Bagheri, A., & van de Schoot, R. (2023, 5). *Simulation-based active learning for systematic reviews: A systematic review of the literature - repository.* OSF. Retrieved from `osf.io/t9hgm` doi: 10.17605/OSF.IO/T9HGM

Teijema, J. J., Van de Schoot, R., Ferdinands, G., Lombaers, P., & De Bruin, J. (2023, 5). Asreview makita: a workflow generator for simulation studies using the command line interface of asreview lab.
doi: 10.5281/zenodo.7550649

Tenopir, C., Rice, N. M., Allard, S., Baird, L., Borycz, J., Christian, L., … Sandusky, R. J. (2020). Data sharing, management, use, and reuse: Practices and perceptions of scientists worldwide. *PloS one*, *15*(3), e0229003.

Thomas, J., McNaught, J., & Ananiadou, S. (2011). Applications of text mining within systematic reviews [Journal Article]. *Res Synth Methods*, *2*(1), 1–14. doi: 10.1002/jrsm.27

Thomas, J., Noel-Storr, A., Marshall, I., Wallace, B., McDonald, S., Mavergames, C., … Turner, T. (2017). Living systematic reviews: 2. combining human and machine effort [Journal Article]. *Journal of clinical epidemiology*, *91*, 31–37.

Timsina, P., El-Gayar, O., & Liu, J. (n.d.). Active learning for the automation of medical systematic review creation [Conference Proceedings].

Tiwana, A. (2004). An empirical study of the effect of knowledge integration on software development performance. *Information and Software Technology*, *46*(13), 899–906.

Tsafnat, G., Glasziou, P., Choong, M. K., Dunn, A., Galgani, F., & Coiera, E. (2014). Systematic review automation technologies. *Systematic reviews*, *3*(1), 1–15.

Tsou, A. Y., Treadwell, J. R., Erinoff, E., & Schoelles, K. (2020). Machine learning for screening prioritization in systematic reviews: comparative performance of abstrackr and eppi-reviewer [Journal Article]. *Syst Rev*, *9*(1), 73. doi: 10.1186/s13643-020-01324-7

van den Brand, S., Hofstee, L., Teijema, J., Melnikov, V., Brouwer, M., & Van de Schoot, R. (2021, December). *Scripts for post-processing mega-meta screening results.* Zenodo. Retrieved from `https://doi.org/10.5281/zenodo.5752358` doi: 10.5281/zenodo.5752358

van de Schoot, R., de Bruin, J., Schram, R., Zahedi, P., de Boer, J., Weijdema, F., … Oberski, D. L. (2021). An open source machine learning framework for efficient and transparent systematic reviews [Journal Article]. *NATURE MACHINE INTELLIGENCE*, *3*(2), 125-133. doi: 10.1038/s42256-020-00287-7

van de Schoot, R., Messina Coimbra, B., Evenhuis, T., Lombaers, P., Weijdema, F., de Bruin, L., … others (2025). The hunt for the last relevant paper: Blending the best of humans and ai. *European Journal of Psychotraumatology*, *16*(1), 2546214.

Van Dinter, R., Tekinerdogan, B., & Catal, C. (2021). Automation of systematic literature reviews: A systematic literature review [Journal Article]. *Information and Software Technology*, *136*. doi: 10.1016/j.infsof.2021.106589

Varghese, A., Hong, T., Hunter, C., Agyeman-Badu, G., & Cawley, M. (2019). Active learning in automated text classification: a case study exploring bias in predicted model performance metrics [Journal Article]. *Environment Systems and Decisions*, *39*(3), 269–280. doi: 10.1007/s10669-019-09717-3

Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., … Polosukhin, I. (2017). Attention is all you need. *Advances in neural information processing systems*, *30*.

Vilone, G., & Longo, L. (2021). Notions of explainability and evaluation approaches for explainable artificial intelligence. *Information Fusion*, *76*, 89–106.

Wagner, G., Lukyanenko, R., & Paré, G. (2022). Artificial intelligence and the conduct of literature reviews. *Journal of Information Technology*, *37*(2), 209–226.

Wallace, B. (2012). *Abstrackr: Software for semi-automatic citation screening.* Agency for Healthcare Research and Quality, Rockville, MD. Retrieved from `https://effectivehealthcare.ahrq.gov/products/abstractr/abstract`

Wallace, B. C., Small, K., Brodley, C. E., Lau, J., & Trikalinos, T. A. (n.d.-a). Deploying an interactive machine learning system in an evidence-based practice center: abstrackr [Conference Proceedings]. In *the 2nd acm sighit symposium* (p. 819). Miami, Florida, USA: ACM Press. doi: 10.1145/2110363.2110464

Wallace, B. C., Small, K., Brodley, C. E., Lau, J., & Trikalinos, T. A. (n.d.-b). Modeling annotation time to reduce workload in comparative effectiveness reviews [Conference Proceedings]. In (pp. 28–35). doi: 10.1145/1882992.1882999

Wallace, B. C., Small, K., Brodley, C. E., & Trikalinos, T. A. (n.d.-a). Active learning for biomedical citation screening [Conference Proceedings]. In (pp. 173–181). doi: 10.1145/1835804.1835829

Wallace, B. C., Small, K., Brodley, C. E., & Trikalinos, T. A. (n.d.-b). Who should label what? instance allocation in multiple expert active learning [Conference Proceedings]. In (pp. 176–187). doi: 10.1137/1.9781611972818.16

Wallace, B. C., Trikalinos, T. A., Lau, J., Brodley, C., & Schmid, C. H. (2010). Semi-automated screening of biomedical citations for systematic reviews [Journal Article]. *BMC Bioinformatics*, *11*(1), 55. doi: 10.1186/1471-2105-11-55

Wang, L. L., & Lo, K. (2021). Text mining approaches for dealing with the rapidly expanding literature on covid-19. *Briefings in Bioinformatics*, *22*(2), 781–799.

Wang, W., Wei, F., Dong, L., Bao, H., Yang, N., & Zhou, M. (2020). *Minilm: Deep self-attention distillation for task-agnostic compression of pre-trained transformers.*

Wang, Z., Nayfeh, T., Tetzlaff, J., O'Blenis, P., & Murad, M. H. (2020). Error rates of human reviewers during abstract screening in systematic reviews. *PloS one*, *15*(1), e0227742.

Wilkinson, M. D., Dumontier, M., Aalbersberg, I. J., Appleton, G., Axton, M., Baak, A., … others (2016). The fair guiding principles for scientific data management and stewardship. *Scientific data*, *3*(1), 1–9.

Yamada, T., Yoneoka, D., Hiraike, Y., Hino, K., Toyoshiba, H., Shishido, A., … Yamauchi, T. (2020). Deep neural network for reducing the screening workload in systematic reviews for clinical guidelines: Algorithm validation study [Journal Article]. *Journal of Medical Internet Research*, *22*(12), e22422. doi: 10.2196/22422

Yan, L., Sha, L., Zhao, L., Li, Y., Martinez-Maldonado, R., Chen, G., … Gašević, D. (2023, August). Practical and ethical challenges of large language models

in education: A systematic scoping review. *British Journal of Educational Technology*, *55*(1), 90–112. Retrieved from http://dx.doi.org/10.1111/bjet.13370 doi: 10.1111/bjet.13370

Yu, Z., Carver, J. C., Rothermel, G., & Menzies, T. (2022). Assessing expert system-assisted literature reviews with a case study. *Expert Systems with Applications*, *200*, 116958.

Yu, Z., Kraft, N. A., & Menzies, T. (2018). Finding better active learners for faster literature reviews [Journal Article]. *EMPIRICAL SOFTWARE ENGINEERING*, *23*(6), 3161–3186. doi: 10.1007/s10664-017-9587-0

Yu, Z., & Menzies, T. (n.d.). Data balancing for technologically assisted reviews: Undersampling or reweighting [Conference Proceedings]. In (Vol. 1866).

Yu, Z., & Menzies, T. (2019). Fast(2): An intelligent assistant for finding relevant papers [Journal Article]. *EXPERT SYSTEMS WITH APPLICATIONS*, *120*, 57–71. doi: 10.1016/j.eswa.2018.11.021

Zhang, H., Cormack, G. V., Grossman, M. R., & Smucker, M. D. (2020). Evaluating sentence-level relevance feedback for high-recall information retrieval [Journal Article]. *Information Retrieval Journal*, *23*(1). doi: 10.1007/s10791-019-09361-0

Zou, J., & Kanoulas, E. (2020). Towards question-based high-recall information retrieval [Journal Article]. *ACM Transactions on Information Systems*, *38*(3). doi: 10.1145/3388640

Appendices

Appendix A

Introduction: List of Applied Data Science Masters

Institution	Programme	First graduates	Country	Link
University of Gothenburg / Chalmers University of Technology	MSc Applied Data Science	2019	Sweden	https://www.gu.se/sites/default/files/2025-10/Bed%C3%B6marutl%C3%A5tande%20-%20Master%20in%20Applied%20Data%20Science.pdf
San José State University	MS Applied Data Intelligence / Applied Data Science	2020	USA	https://www.sjsu.edu/professional/programs/applied-data-intelligence-ms.php
USC (University of Southern California)	MS Applied Data Science	2020	USA	https://datascience.usc.edu/academics/master-of-science-in-applied-data-science/
Utrecht University	MSc Applied Data Science	2021	Netherlands	https://www.uu.nl/en/news/new-masters-programme-applied-data-science
University of Michigan (UMSI)	Master of Applied Data Science (MADS)	2021	USA	https://www.si.umich.edu/about-umsi/news/umsis-online-mads-degree-graduate-first-students-august
Malmö University	MSc Computer Science: Applied Data Science	2023	Sweden	https://nordmedianetwork.org/resources/degree-programmes/computer-science-applied-data-science/
HAN University of Applied Sciences	Master Applied Data Science	2025	Netherlands	https://www.han.nl/nieuws/2022/11/nieuwe-master-applied-data-science-bij-de-han/
University of Johannesburg	Master of Applied Data Science (CW)	2029	South Africa	https://www.uj.ac.za/university-courses/master-of-applied-data-science-course-work/

Table A.1 A non-exhaustive list of explicitly branded applied data science master's programmes and their first graduating cohorts.

Appendix B

Chapter 3: Simulation-based Active Learning for Systematic Reviews: A Scoping Review of Literature

B.1 Inclusion Table

Table B.1

Title	Year	Reference
Modeling annotation time to reduce workload in comparative effectiveness reviews	2010	B. C. Wallace, Small, Brodley, Lau, and Trikalinos (n.d.-b)
Active learning for biomedical citation screening	2010	B. C. Wallace, Small, Brodley, and Trikalinos (n.d.-a)
Semi-automated screening of biomedical citations for systematic reviews	2010	B. C. Wallace et al. (2010)
Who should label what? instance allocation in multiple expert active learning	2011	B. C. Wallace, Small, Brodley, and Trikalinos (n.d.-b)
Deploying an interactive machine learning system in an evidence-based practice center: abstrackr	2012	B. C. Wallace, Small, Brodley, Lau, and Trikalinos (n.d.-a)
Virk: an active learning-based system for bootstrapping knowledge base development in the neurosciences	2013	Ambert et al. (2013)
Reducing systematic review workload through certainty-based screening	2014	Miwa et al. (2014)
Autonomy and Reliability of Continuous Active Learning for Technology-Assisted Review	2015	Cormack and Grossman (2015)
Active learning for the automation of medical systematic review creation	2015	Timsina, El-Gayar, and Liu (n.d.)
Faster title and abstract screening? Evaluating Abstrackr, a semi-automated online screening program for systematic reviewers	2015	Rathbone et al. (2015)
Topic detection using paragraph vectors to support active learning in systematic reviews	2016	Hashimoto et al. (2016)
Scalability of Continuous Active Learning for Reliable High-Recall Text Classification	2016	Cormack and Grossman (n.d.-a)
Technology-assisted review in empirical medicine: Waterloo participation in CLEF eHealth 2017	2017	Cormack and Grossman (n.d.-b)
Data balancing for technologically assisted reviews: Undersampling or reweighting	2017	Yu and Menzies (n.d.)
A machine learning approach for semi-automated search and selection in literature studies	2017	Ros, Bjarnason, and Runeson (2017)
CLEF 2017 technologically assisted reviews in empirical medicine overview	2017	Kanoulas, Li, Azzopardi, and Spijker (n.d.-a)
Finding better active learners for faster literature reviews	2018	Yu, Kraft, and Menzies (2018)
LIMSI@CLEF eHealth 2018 Task 2: Technology assisted reviews by stacking active and static learning	2018	C. Norman, Leeflang, and Névéol (n.d.)

Continued on next page

Table B.1 – continued from previous page

Title	Year	Reference
A comparative analysis of semi-supervised learning: The case of article selection for medical systematic reviews	2018	J. Liu, Timsina, and El-Gayar (2018)
Prioritising references for systematic reviews with RobotAnalyst: A user study	2018	Przybyła et al. (2018)
Technology-assisted review in empirical medicine: Waterloo participation in CLEF eHealth 2018	2018	Cormack and Grossman (n.d.-c)
CLEF 2018 technologically assisted reviews in empirical medicine overview	2018	Kanoulas, Li, Azzopardi, and Spijker (n.d.-b)
Finding all the needles in a haystack: A System to estimate the costs of e-discovery and systematic reviews	2018	Di Nunzio (n.d.)
CLEF 2019 technology assisted reviews in empirical medicine overview	2019	Kanoulas, Li, Azzopardi, and Spijker (n.d.-c)
Measuring the impact of screening automation on meta-analyses of diagnostic test accuracy	2019	C. R. Norman et al. (2019)
FAST(2): An intelligent assistant for finding relevant papers	2019	Yu and Menzies (2019)
Comparing word embeddings for document screening based on active learning	2019	Carvallo and Parra (n.d.)
Active learning in automated text classification: a case study exploring bias in predicted model performance metrics	2019	Varghese, Hong, Hunter, Agyeman-Badu, and Cawley (2019)
PNS335: Using Machine Learning for Efficiency Improvements in Systematic Literature Reviews of Clinical Efficacy and Safety	2019	Halfpenny, Alleman, Eaton, and van Vliet (2019)
Towards Question-based High-recall Information Retrieval	2020	Zou and Kanoulas (2020)
Evaluating sentence-level relevance feedback for high-recall information retrieval	2020	Zhang, Cormack, Grossman, and Smucker (2020)
Statistical stopping criteria for automated screening in systematic reviews	2020	Callaghan and Mueller-Hansen (2020)
Deep Neural Network for Reducing the Screening Workload in Systematic Reviews for Clinical Guidelines: Algorithm Validation Study	2020	Yamada et al. (2020)
Automatic document screening of medical literature using word and text embeddings in an active learning setting	2020	Carvallo et al. (2020)
APS: An Active PubMed Search System for Technology Assisted Reviews	2020	D. Li, Zafeiriadis, and Kanoulas (n.d.)
Risk and Protective Factors in the COVID-19 Pandemic: A Rapid Evidence Map	2020	Elmore et al. (2020)
Machine learning for screening prioritization in systematic reviews: comparative performance of Abstrackr and EPPI-Reviewer	2020	Tsou et al. (2020)

Continued on next page

Table B.1 – continued from previous page

Title	Year	Reference
An evaluation of DistillerSR's machine learning-based prioritization tool for title/abstract screening – impact on reviewer-relevant outcomes	2020	Hamel et al. (2020)
SWIFT-Active Screener: Accelerated document screening through active learning and integrated recall estimation	2020	Howard et al. (2020)
Semi-Supervised Text Classification Framework: An Overview of Dengue Landscape Factors and Satellite Earth Observation	2020	Z. Li et al. (2020)
AI-Assisted systematic reviewing: selecting studies to compare Bayesian versus Frequentist SEM for small sample sizes	2020	Ferdinands (2021)
An open source machine learning framework for efficient and transparent systematic reviews	2021	van de Schoot et al. (2021)
The Use Of A Text-Mining Screening Tool For Systematic Review Of Treatments For Relapsed/Refractory Diffuse Large B-Cell Lymphoma	2021	Carey et al. (2021)
Transferring knowledge between topics in systematic reviews	2022	Molinari and Kanoulas (2022)
Assessing expert system-assisted literature reviews with a case study	2022	Yu et al. (2022)
Extreme Systematic Reviews: A Large Literature Screening Dataset to Support Environmental Policymaking	2022	Hou et al. (2022)
The efficiency of machine learning-assisted platform for article screening in systematic reviews in orthopaedics	2023	Muthu (2022)
Reducing the user labeling effort in effective high recall tasks by fine-tuning active learning	2023	Bianco, Duarte, and Gonçalves (2023)
A Machine Learning Framework Reduces the Manual Workload for systematic reviews of the Diagnostic Performance of Prostate Magnetic Resonance Imaging	2023	Nedelcu et al. (2023)
Performance of active learning models for screening prioritization study into the Average Time to Discover relevant records	2023	Ferdinands et al. (2023)
An Automated Literature Review Tool (LiteRev) for Streamlining and Accelerating Research Using Natural Language Processing and Machine Learning: Descriptive Performance Evaluation Study	2023	Orel et al. (2023)
Machine Learning assisted systematic reviewing in orthopaedics	2023	Pijls (2023)

Continued on next page

Table B.1 – continued from previous page

Title	Year	Reference
Implementing Simple Active Learning (AL) Boosters considerably improves the early identification of relevant studies in the Systematic Literature Review (SLR) process	2023	Bravo, Patel, and Atanasov (2023)
Can artificial intelligence separate the wheat from the chaff in systematic reviews of health economic articles?	2023	Oude Wolcherink et al. (2023)
Machine learning based system for the automation of systematic literature reviews	2023	Pytlak et al. (2023)
SciMine: An Efficient Systematic Prioritization Model Based on Richer Semantic Information	2023	Guo, Luo, Yang, and Zhang (2023)
Active learning-based systematic reviewing using switching classification models: the case of the onset, maintenance, and relapse of depressive disorders	2023	chapter 4
Impact of Active learning model and prior knowledge on discovery time of elusive relevant papers: a simulation study	2024	Byrne et al. (2024)
Screening Smarter, Not Harder: A Comparative Analysis of Machine Learning Screening Algorithms and Heuristic Stopping Criteria for SR in Educational Research	2024	Campos et al. (2024)
Systematic review using a spiral approach with machine learning	2024	Saeidmehr, Steel, and Samavati (2024)

Table B.1 List of studies selected for inclusion in the literature review, sorted on publication year. The table includes the title, year of publication, and reference for each study.

B.2 Dataset Table

Table B.2 Summary of the largest available datasets utilized for systematic review screening phase automation research.

Collection Name	Author	Field	Datasets	Description	Times seen
CLEF eHealth 2019 Cormack and Grossman (n.d.-b, n.d.-c); Kanoulas et al. (n.d.-c)	E. Kanoulas, D. Li, L. Azzopardi and R. Spijker	Medicine	111	111 systematic reviews from CLEF eHealth collected over three years. The 2017 and 2018 collections contribute 80 reviews on Diagnostic Test Accuracy, all conducted by Cochrane researchers. In 2019, an additional 31 reviews were incorporated, encompassing not just Diagnostic Test Accuracy but also Intervention, Prognosis, and Qualitative systematic reviews.	16
Cohen Cohen et al. (2006)	Cohen, A.M., Hersh, W. R., Peterson, K., & Yen, P.-Y.	Medicine	15	15 systematic drug class reviews carried out by three institutions: the Oregon Evidence-Based Practice Center (EPC), the Southern California EPC, and the Research Triangle Institute/University of North Carolina (RTI/UNC) EPC.	5

Continued on next page

Collection Name	Author	Field	Datasets	Description	Times seen
Hamel Hamel et al. (2020)	Hamel, Kelly, Thavorn, Rice, Wells, Hutton	Medicine	10	10 previously conducted systematic reviews, focusing on responses during title/abstract screening and the finalized lists of included studies. These reviews were carried out by research teams with a background in knowledge synthesis reviews. The research was a collaborative effort between the Ottawa Hospital Research Institute and the University of Ottawa Heart Institute in Ottawa.	1
RobotAnalyst database Przybyła et al. (2018)	Przybyla, Brockmeier, Austin, Kontonatsios, Le Pogam, McNaught, Elm, Nolan, Ananiadou	Medicine	22	Used for testing RobotAnalyst. 22 collections were screened completely, i.e., a reviewer made a relevance decision for every reference. All collections were used with no post hoc selection. The smallest dataset encompassed 86 references, while the largest approached 5,000. The proportion of pertinent references ranged from 0.28% (6 out of 2,148) to 30%.	1

Continued on next page

Collection Name	Author	Field	Datasets	Description	Times seen
AHRQ evidence reports Tsou et al. (2020)	Amy Y. Tsou, Jonathan R. Treadwell, Eileen Erinoff and Karen Schoelles	Medicine	9	Evidence reviews assessing specific medical interventions conducted after 2010, with a clear citation screening threshold. Large reviews, from AHRQ EPC publications (2012–2018), contain over 1,000 citations each and employ dual screening. Smaller reviews, mainly ECRI "Emerging Technology" reports (200–999 citations), use single screening by one analyst.	1
SYNERGY dataset (Chapter 5)	De Bruin, Ma, Ferdinands, Teijema, Van de Schoot	Medicine, Psychology, Computer science, Biology, Chemistry, Mathematics	26	SYNERGY is a free and open dataset on study selection in systematic reviews, comprising 169,288 academic works from 26 systematic reviews. Only 2,834 (1.67%) are included. Useful for IR with sparse labels and for NLP/ML/network analysis due to rich metadata. Total of 82,668,134 trainable data points.	–

Appendix C

Chapter 7: Large-Scale Simulation Study of Active Learning Models for Systematic Reviews

C.1 Summary of datasets used in simulations

Dataset	Topic	Total	Include	Ratio	Sim. S1	Sim. S2
Appenzeller-Herzog_2019	Medicine	2.873	26	0.9%	338	92
Bos_2018	Medicine	4.878	10	0.2%	130	92
Brouwer_2019	Medicine, Psychology	38.114	62	0.2%	806	92
Chou_2003	Medicine	1.908	15	0.8%	195	92
Chou_2004	Medicine	1.630	9	0.6%	117	92
Donners_2021	Medicine	258	15	5.8%	195	92
Hall_2012	Computer science	8.793	104	1.2%	1.352	92
Jeyaraman_2020	Medicine	1.175	96	8.2%	1.248	92
Leenaars_2019	Chemistry, Medicine, Psychology	5.812	17	0.3%	221	92
Leenaars_2020	Medicine	7.216	583	8.1%	7.579	92
Meijboom_2021	Medicine	882	37	4.2%	481	92
Menon_2022	Medicine	975	74	7.6%	962	92
Moran_2021	Biology, Medicine	5.214	111	2.1%	1.443	92
Muthu_2021	Medicine	2.719	336	12.4%	4.368	92
Nelson_2002	Medicine	366	80	21.9%	1.040	92
Oud_2018	Medicine, Psychology	952	20	2.1%	260	92
Radjenovic_2013	Computer science	5.935	48	0.8%	624	92
Sep_2021	Psychology	271	40	14.8%	520	92
Smid_2020	Computer science, Mathematics	2.627	27	1.0%	351	92
Valk_2021	Medicine, Psychology	725	89	0.8%	1.157	92
van_de_Schoot_2018	Medicine, Psychology	4.544	38	12.3%	494	92
van_der_Waal_2022	Medicine	1.970	33	1.7%	429	92
van_Dis_2020	Medicine, Psychology	9.128	72	0.8%	936	92
Walker_2018	Biology, Medicine	48.375	762	1.6%	65*	92
Wassenaar_2017	Biology, Chemistry, Medicine	7.668	111	1.4%	1.443	92
Wolters_2018	Medicine	4.280	19	0.4%	247	92
Total		**169.288**	**2.834**		**27.001**	**2.392**

Table C.1 Summary of datasets used in the large-scale simulation studies. Each dataset is listed with its main research topic(s), total number of records, and number and proportion of relevant records. The final columns show the number of simulations conducted in Study 1 and Study 2, respectively.

C.2 Feature extractors used in the second phase of the simulations

Name	Description	Pre-trained	Speed	Extracted Features	Study
OneHot	Encodes text in a binary fashion, representing each word as a 1 (present) or 0 (absent) in the vector, ignoring word order and context.	No	Fast	Word presence	2
TF-IDF	Enhances the OneHot approach by weighting words based on their frequency across records, helping to identify more informative words.	No	Fast	Weighted word frequency	1 & 2
Doc2Vec (Vector Size 40)	Generates record embeddings in a 40-dimensional space, capturing the semantic meaning of words in context.	No	Medium	Word co-occurrence, document-level context, semantic meaning	1 & 2
Doc2Vec (Vector Size 120)	Similar to its 40-dimensional counterpart but provides a more detailed representation by using 120 dimensions.	No	Medium	Word co-occurrence, document-level context, semantic meaning	2
Doc2Vec Non-Negative	A 40-vector-wide Doc2Vec embedding scaled between 0 and 1, allowing for use with naïve Bayes.	No	Medium	Word co-occurrence, document-level context, semantic meaning	2
MiniLM (W. Wang et al., 2020)	A small, fast transformer model that retains much of the larger models' ability to capture nuanced language features.	Yes	Medium	Syntax, semantic meaning, context, attention	1
distiluse-base-multilingual-cased-v2 (Reimers & Gurevych, 2019)	A multilingual sentence embedding model designed to represent a wide range of languages.	Yes	Medium	Syntax, language-independent semantic meaning, context, attention	2

Continued on next page

Table C.2 – continued from previous page

Name	Description	Pre-trained	Speed	Extracted Features	Study
FastText (Wiki-news-300d-1M-subword) (Bojanowski et al., 2016)	Employs subword information, allowing embeddings for unseen words by decomposing them into smaller units.	Yes	Medium	Subword information, semantic meaning	2
SpaCy[1]	Uses an NLP framework to provide embeddings particularly suited for syntactic analysis.	Yes	Medium	Syntax, word-level semantic meaning	2
Word2Vec (Google-News-300)[2]	Generates word embeddings based on word co-occurrence in a large corpus of Google News articles.	Yes	Medium	Context, semantic meaning	2
mxbai-embed-large-v1 (S. Lee et al., 2024)	A pretrained sentence-transformer by Mixedbread.ai, optimized for document retrieval tasks.	Yes	Slow	Syntax, semantic meaning, context, attention	2
all-mpnet-base-v2 (head-only)[3]	A transformer-based model optimized for sentence embeddings. "Head-only" (Sun, Qiu, Xu, & Huang, 2020) disregards tokens exceeding the token length.	Yes	Slow	Syntax, semantic meaning, context, attention	1 & 2
all-mpnet-base-v2 (hier. mean)	Hierarchical mean (Sun et al., 2020): tokens exceeding the length limit are embedded separately and averaged into one vector.	Yes	Slow	Syntax, semantic meaning, context, attention	2
LaBSE (Feng et al., 2020)	A multilingual sentence embedding model developed by Google.	Yes	Slow	Syntax, language-independent semantic meaning, context, attention	2

Table C.2 Feature extractors used in the second phase of the simulations, ordered approximately by embedding computation speed.

Appendix D

Chapter 10: CLARIFY: Concept Level Active-Learning Ranker Interpreter for Systematic Reviews

D.1 PseudoCode

```
INPUT
  Dataset of abstracts with titles and labels
  Feature extractor FE
  Classifier CLF, Balancer BAL, Querier QRY
  K = number of NMF concepts
  S = number of Sobol designs

OUTPUT
  Global concept importances
  Per-abstract highlighted sentences with (concept, intensity)

1. EMBED AND NORMALIZE
  FOR each record i IN corpus DO
    SET text_i = CONCAT(title_i, abstract_i)
    OBTAIN x_i = FE.EMBED(text_i)
  ENDFOR
  SET A = STACK(x_i)
  COMPUTE A = NORMALIZE(A, $\theta$_min, $\theta$_max)   // store $\theta$_min
     , $\theta$_max for reuse

2. ACTIVE LEARNING
  INIT cycle with CLF, BAL, QRY on features A
  WHILE stopping criterion NOT met DO
    CALL QRY to obtain next records
    OBTAIN labels for queried records
    FIT CLF on labeled set with BAL
  ENDWHILE
  SET A_pos = SUBSET of A WHERE label_i = 1

3. CONCEPT FACTORIZATION
  CALL NMF.FIT on A_pos with K components
  SET W = concept basis, U = activations for A_pos

4. GLOBAL CONCEPT IMPORTANCE (SOBOL TOTAL-ORDER)
  FOR each positive embedding a_i IN A_pos DO
    OBTAIN Sobol perturbations guided by concept base W
    ESTIMATE classifier variance using JANSEN_TOTAL_ORDER
    ACCUMULATE importance scores
  ENDFOR
  COMPUTE S_global = average accumulated scores
  COMPUTE $\tau$ = MEAN(S_global) + STD(S_global)
  SET C_top = { k | S_global[k] > $\tau$ }

5. SENTENCE-LEVEL OCCLUSION WITH FIXED W
  FOR any positive record i DO
    SET T_full = CONCAT(title_i, abstract_i)
    SET u_full = U[i]

```

```
    SPLIT T_full into components = sentences
    FOR each component_j IN components DO
      FORM T_minus_j by removing component_j from T_full
      OBTAIN a_minus_j = FE.EMBED(T_minus_j)
      COMPUTE a_minus_j = NORMALIZE(a_minus_j, $\theta$_min, $\theta$_max)
      OBTAIN u_minus_j = NMF.TRANSFORM_W(a_minus_j, W)
      COMPUTE $\Delta$u_j = u_full - u_minus_j
      RESTRICT $\Delta$u_j to indices in C_top
      SCALE $\Delta$u_j to [0, 1]
      SET concept_id = ARGMAX($\Delta$ u_j[k])
      SET intensity  = MAX($\Delta$ u_j[k])
      ASSIGN component_j WITH (concept_id, intensity)
    ENDFOR
  ENDFOR

6. OUTPUT
  DISPLAY S_global and C_top
  FOR each abstract_i DO
    DISPLAY components with assigned concept labels and intensities
  ENDFOR
```

Summary

Systematic Reviews are the gold standard for synthesizing scientific evidence. They rely on reliable, transparent, and standardized procedures to assemble research scattered across studies. However, they are characterized by labor intensity, particularly during the screening phase, where thousands of documents must be manually assessed. The exponential growth of scientific publication has made this process increasingly unsustainable.

Applied Data Science, specifically through Active Learning, offers a technological solution. Active Learning is a machine learning approach where the model incrementally learns from the user during the screening process. Unlike static models, it can almost immediately prioritize the most likely relevant records, without requiring heaps of training data. This allows researchers to find relevant evidence earlier and potentially stop screening before assessing every single document.

In this dissertation, I argue that the successful integration of this technique requires more than just algorithmic performance. The development of a tool that aligns with the academic standards requires the operational efficiency of industry tools and the reflective lens required in academia. The main motivation behind this dissertation is to *evaluate and strengthen the application of Active Learning for accelerating the screening phase of systematic reviews, while ensuring the process remains transparent, reproducible, and human-centered.*

In Chapters 1 and 2, I provide the research context and theoretical framework. First, in Chapter 1, I introduce the problem of information overload in science and how active learning serves as a "human-in-the-loop" solution to reduce screening workload without losing control. Second, in Chapter 2, I analyze the field of Applied Data Science (ADS) itself. I contrast the financial incentives of industry with the knowledge incentives of academia. I introduce four recurring themes that guide the rest of the work: Human-Centered Design, Software Usability, Reproducibility, and FAIR Data.

In Chapters 3 and 4, I focus on Exploration. Chapter 3 presents a scoping review of existing simulation studies, identifying a lack of standardization in the field. Chapter 4 conducts exploratory studies on model behavior, specifically testing whether switching between different machine learning models during a review can improve performance.

In Chapters 5 and 6, I focus on Data Preparation and Tooling. We introduce two open-science contributions: SYNERGY (Chapter 5), a gold-standard open dataset of 26 fully labeled systematic reviews, and Makita (Chapter 6), a workflow generator

that automates large-scale, reproducible simulation studies.

In Chapters 7, 8, and 9, I focus on Modeling and Evaluation. Chapter 7 presents a large-scale simulation study to benchmark model performance. Chapter 8 provides practical explanations for users navigating the "maze of models." Chapter 9 validates these findings externally using datasets from a major European non-governmental health organization.

In Chapter 10, I focus on Interpretation. Chapter 10 proposes CLARIFY, a new method for "Explainable AI" (XAI) in systematic reviews. This tool helps users interpret why a model ranked a specific document highly by extracting and visualizing the underlying concepts driving the decision.

Finally, in Chapter 11, I reflect on the broader implications of this work. I revisit the themes from Chapter 2, arguing that Applied Data Science in academia must use both operational skills and reflective skills. I conclude that by adopting industrial standards for usability and data management, academic research can achieve greater impact and reproducibility.

Samenvatting

Systematische reviews zijn de gold-standard voor het verzamelen van wetenschappelijk bewijs. Een systematisch review bestaat uit betrouwbare, transparante en gestandaardiseerde procedures om verspreid onderzoek uit verschillende studies samen te brengen. Ze worden echter gekenmerkt door hun arbeidsintensiteit, met name tijdens de screeningfase, waarin duizenden documenten handmatig moeten worden beoordeeld. Door de exponentiële groei van wetenschappelijke publicaties wordt het behoud van kwaliteit in proces steeds minder houdbaar.

Applied Data Science, en met name active-learning, biedt een technologische oplossing. Active-learning is een machine learning techniek waarbij het model tijdens het screeningproces stapsgewijs van de gebruiker leert. In tegenstelling tot statische modellen kan het direct, zonder stapels training data, prioriteit geven aan de meest waarschijnlijk relevante records, waardoor onderzoekers eerder relevant bewijs kunnen vinden en mogelijk kunnen stoppen met screenen voordat ze alle documenten hebben beoordeeld.

In dit proefschrift stel ik dat een succesvolle implementatie van deze techniek, in gereedschap bedoeld voor de academische setting, meer vereist dan alleen algoritmische prestaties; het vereist een fusie van de operationele efficiëntie die in het bedrijfsleven zo belangrijk is voor Applied Data Science en de reflectieve lens die in de academische wereld nodig is. De belangrijkste motivatie achter dit proefschrift is het *evalueren en versterken van de toepassing van active-learning voor het verbeteren van de screeningfase van systematische reviews, terwijl het proces transparant, reproduceerbaar en mensgericht blijft.*

In de hoofdstukken 1 en 2 geef ik de onderzoekscontext en het theoretisch kader weer. Eerst introduceer ik in hoofdstuk 1 het probleem van informatie-overload in de wetenschap en hoe Active-learning dient als een 'human-in-the-loop'-oplossing om de screeningsarbeid te verminderen zonder de menselijke controle te verliezen. Ten tweede analyseer ik in hoofdstuk 2 het veld van Applied Data Science (ADS) zelf. Ik zet de financiële drijfveren van de industrie af tegen de kennis drijfveef van de academische wereld. Ik introduceer vier thema's die de rest van het werk sturen: mensgericht ontwerp, bruikbaarheid van software, reproduceerbaarheid, en FAIR-data.

In hoofdstukken 3 en 4 richt ik me op exploratie. Hoofdstuk 3 presenteert een scoping review van bestaande simulatiestudies, waarbij een gebrek aan standaardisatie in het vakgebied wordt vastgesteld. Hoofdstuk 4 voert exploratieve studies uit naar

modelgedrag, waarbij specifiek wordt getest of het wisselen tussen verschillende machine learning-modellen tijdens een review de prestaties kan verbeteren.

In de hoofdstukken 5 en 6 richt ik me op data en tooling. Deze hoofdstukken introduceren twee open-science contributies: SYNERGY (hoofdstuk 5), een gold-standard open dataset van 26 volledig gelabelde systematische reviews, en Makita (hoofdstuk 6), een workflowgenerator die grootschalige, reproduceerbare simulatiestudies automatiseert.

In de hoofdstukken 7, 8 en 9 richt ik me op modellering en evaluatie. Hoofdstuk 7 presenteert een grootschalige simulatiestudie om de prestaties van modellen te benchmarken. Hoofdstuk 8 biedt praktische uitleg voor gebruikers die zich een weg banen door het 'doolhof der modellen'. Hoofdstuk 9 valideert bevindingen extern met behulp van datasets van een grote Europese niet-gouvernementele gezondheidsorganisatie.

Ten slotte reflecteer ik in hoofdstuk 11 op de bredere implicaties van dit werk. Ik kom terug op de thema's uit hoofdstuk 2 en stel dat Applied Data Science in de academische wereld operationele vaardigheden moet combineren met kritische reflectie. Ik concludeer dat academisch onderzoek een grotere impact en reproduceerbaarheid kan bereiken door industriële normen voor bruikbaarheid en gegevensbeheer toe te passen.

Curriculum Vitae

Education

2022 -- 2025 PhD in Applied Data Science, Utrecht University, Utrecht, Netherlands

Doctoral research on explainable AI and active learning for systematic reviews, focusing on simulation workflows, FAIR data, and user-centric AI tools.

2020 -- 2021 Master's in Applied Data Science (Cum Laude), Utrecht University, Utrecht, Netherlands

Interdisciplinary program bridging data science with health, geo, social and behavioural sciences. Graduated Cum Laude with a GPA equivalent of 4.0.

2016 -- 2020 Bachelor's in Information Science, Utrecht University, Utrecht, Netherlands

Combined psychology, communication studies, organizational science, and computing science. Thesis on Process Mining and Architecture Mining.

2010 -- 2016 Pre-University Education (Gymnasium), Johan van Oldebarnevelt Gymnasium, Amersfoort, Netherlands

Nature & Technology track with additional courses in Latin and Music history and theory.

Notable Experiences

2026 -- present Administrative Staff — Ministry of Defence, Netherlands

Starting January 2026, Jelle will work as part of the administrative staff developing machine learning tooling to support operational teams.

2024 Industry & Sponsorship Chair, Co-Organizer — BNAIC/BeNeLearn 2024, Utrecht, Netherlands

Planned and executed the leading AI/ML conference in the Benelux region, coordinating keynotes, parallel sessions, and industry partnerships.

2024 Guest Lecturer — Safe Experimentation with Large Language Models, European Central Bank & De Nederlandsche Bank, Frankfurt/Amsterdam

Delivered a guest lecture on ASReview and AI in finance, discussing transparency, ethical AI, and reproducible research for central banks and regulators.

2024 Co-Organizer and Presenter — Kickstart AI, Amsterdam, Netherlands

Organized and presented a workshop on Makita, a workflow generator for reproducible large-scale simulation studies, providing hands-on training.

2023 Invited Speaker — Research Indaba, North-West University, Optentia Research Unit, Cape Town, South Africa

Spoke on AI and GPT technologies, their potential and limitations, and implications for research and practice in the social sciences.

2023 OCRE Grant Recipient & Collaborator — OCRE Project & VSHN AG, Zurich, Switzerland / Utrecht, Netherlands

Awarded an OCRE cloud grant to run large-scale systematic review simulations using commercial cloud platforms. Collaborated with VSHN on Kubernetes-based deployments using Exoscale.

2022 Invited Speaker — University of Oxford, Department of Education, Oxford, UK

Presented on the application of machine learning in systematic reviews, with emphasis on dataset quality and open science practices.

2022 Speaker — Summit AI and Predictive Health (EWUU Alliance), Wageningen campus, Netherlands

Presented use-cases on inclusive training datasets for text mining, demonstrating importance of diverse coverage in AI training data.

2022 Speaker — Netherlands National Open Science Festival, Amsterdam, Netherlands

Presented on hybrid models and dataset benchmarking for systematic reviews at the national Open Science event.

2022 Fellowship Recipient — Hofvijverkring Fellowship, The Hague, Netherlands

Received competitive fellowship to broaden international collaboration opportunities and networks in data science and AI.

2021 Researcher — Utrecht University, ASReview Research Group, Utrecht, Netherlands

Contributed to the development and maintenance of ASReview, open-source software for AI-driven systematic reviews.

List of Publications

Brouwer, M., Hofstee, L., Teijema, J. J., Ferdinands, G., de Boer, J., Weijdema, F., … others (2022). Ai-aided systematic review to create a database with potentially relevant papers on depression, anxiety, and addiction.

Byrne, F., Hofstee, L., Teijema, J. J., de Bruin, J., & van de Schoot, R. (2023). The influence of active learning model and prior knowledge choice on how long it takes to find hard-to-find relevant papers: Examining the variability of the time to discovery and the stability of its rank-orders.

Byrne, F., Hofstee, L., Teijema, J. J., De Bruin, J., & van de Schoot, R. (2024). Impact of active learning model and prior knowledge on discovery time of elusive relevant papers: a simulation study. *Systematic Reviews*, *13*(1), 175.

de Bruin, J., Lombaers, P., Kaandorp, C., Teijema, J. J., van der Kuil, T., Yazan, B., … van de Schoot, R. (2025). Asreview lab v.2: Open-source text screening with multiple agents and a crowd of experts. *Patterns*, *6*(7), 101318. Retrieved from `https://www.sciencedirect.com/science/article/pii/S2666389925001667` doi: https://doi.org/10.1016/j.patter.2025.101318

De Bruin, J., Ma, Y., Ferdinands, G., Teijema, J. J., & Van de Schoot, R. (2023). Synergy-open machine learning dataset on study selection in systematic reviews. *Version V1*.

den Boer, R., Hofstee, L., Leenaars, C., Bagheri, A., Teijema, J. J., & van de Schoot, R. (2024). Advancing multilingual abstract classification: A comparative analysis of feature extraction models in systematic reviews.

Ferdinands, G., Schram, R., de Bruin, J., Bagheri, A., Oberski, D. L., Tummers, L., Teijema, J. J., & Van de Schoot, R. (2023). Performance of active learning models for screening prioritization in systematic reviews: a simulation study into the average time to discover relevant records. *Systematic Reviews*, *12*(1), 100.

Kamsma, T. M., Teijema, J. J., van Roij, R., & Spitoni, C. (2025). Echo state and band-pass networks with aqueous memristors: Leaky reservoir computing with a leaky substrate. *Chaos: An Interdisciplinary Journal of Nonlinear Science*, *35*(9).

Romanov, S., Siqueira, A. S., de Bruin, J., Teijema, J. J., Hofstee, L., & van de Schoot, R. (2024). Optimizing asreview simulations: A generic multiprocessing solution for 'light-data' and 'heavy-data' users. *Data Intelligence*, *6*(2), 320–343.

Teijema, J. J. (2023). *Simulation data for: Large-scale simulation study of active learning models for systematic reviews.*

Teijema, J. J. (2025). *Replication Data for: External validation of machine learning hyperparameters for systematic review screening prioritization.* DataverseNL. Retrieved from https://doi.org/10.34894/YSXMVV doi: 10.34894/YSXMVV

Teijema, J. J., & Bagheri, A. (2025, October). CLARIFY: Concept Level Active-Learning Ranker Interpreter for Systematic Reviews. In *Bnaic/benelearn 2025 conference.*

Teijema, J. J., de Bruin, J., Bagheri, A., & van de Schoot, R. (2025). Large-scale simulation study of active learning models for systematic reviews. *International Journal of Data Science and Analytics*, 1–22. doi: https://doi.org/10.1007/s41060-025-00777-0

Teijema, J. J., Hofstee, L., Brouwer, M., De Bruin, J., Ferdinands, G., De Boer, J., … Bagheri, A. (2023). *Active learning-based systematic reviewing using switching classification models: the case of the onset, maintenance, and relapse of depressive disorders* (No. Volume 8). PsyArXiv.

Teijema, J. J., Ribeiro, G., Seuren, S., Anadria, D., Bagheri, A., & van de Schoot, R. (0). Simulation-based active learning for systematic reviews: A scoping review of literature. *Journal of Information Science*, *0*(0), 01655515251379058. Retrieved from https://doi.org/10.1177/01655515251379058 doi: 10.1177/01655515251379058

Teijema, J. J., Ribeiro, G., Seuren, S., Anadria, D., Bagheri, A., & van de Schoot, R. (2025). Simulation-based active learning for systematic reviews: A scoping review of literature. *Journal of Information Science*.

Teijema, J. J., van de Schoot, R., Ferdinands, G., Lombaers, P., & de Bruin, J. (2024). Makita—a workflow generator for large-scale and reproducible simulation studies mimicking text labeling. *Software Impacts*, *21*, 100663.

van den Brand, S., Hofstee, L., Teijema, J. J., Melnikov, V., Brouwer, M., & Van de Schoot, R. (2021). Scripts for post-processing mega-meta screening results.

Acknowledgements

De uitvoering van een onderzoekstraject als het huidige vindt natuurlijk niet plaats in een vacuüm, maar wordt getekend door de steun van anderen. Ik kan zonder enige twijfel zeggen dat ik zonder onderstaanden geen schijn van kans had om dit vierjarige traject te doorlopen, laat staan zonder permanente academische littekens. In vier jaar tijd hebben mensen die ik daarvoor niet kende zich gevormd tot mijn academische surrogaatfamilie.

Ik prijs mezelf erg gelukkig met mijn sociale kring. Ik voel bewondering en waardering voor de mensen om mij heen, en ben me ervan bewust dat dit een zeldzame luxe is, zowel binnen als buiten de muren van de universiteit.

Rens Ondertussen vijf jaar geleden begon ik mijn masterscriptie onder jouw supervisie. Dit beviel mij klaarblijkelijk zodanig dat ik besloot er een PhD aan vast te knopen. Sindsdien heb je elke dag klaargestaan om mij te helpen groeien tot volwaardig onderzoeker. Ik ben je enorm dankbaar voor je tijd, inzet en inzichten. De afgelopen jaren vormen de basis van mijn carrière, en elk gesprek dat we gevoerd hebben heeft mij geholpen weer een stukje beter te worden in wat ik doe. Ik kijk ontzettend uit naar de volgende stap in mijn carrière, die ik mede aan jou te danken heb, en zal altijd met plezier terugdenken aan de afgelopen jaren.

Ook ben ik je erg dankbaar voor jouw persoonlijke touch in mijn PhD-traject. Horrorverhalen over PhD-trajecten zijn er zat: eindeloze vertraging, uitval, of dramatische supervisie. Problemen waar ik me vanaf het eerste moment geen enkele zorg over heb hoeven maken. Ik ben me erg bewust van het geluk dat ik heb gehad met mijn begeleidingsteam. Hartelijk bedankt voor je onvoorwaardelijke inzet als supervisor de afgelopen jaren.

Robert Robert, naast dat ik je beschouw als een enorme expert in mijn vakgebied, heb ik vooral ontzettend genoten van jouw openhartigheid en benaderbaarheid als leidinggevende. Als ik met een probleem zat, hoefde ik het maar aan jou voor te leggen en dan zat ik met zes oplossingen in plaats van één probleem. Een verbetering, zeker, maar dan resteerde mij enkel nog de keuze voor welk antwoord ik ging. Jouw hulp was altijd vrijblijvend, direct en erg waardevol. Ik heb het mede aan jou te danken wie ik ben geworden als onderzoeker. Ik wil je heel erg bedanken voor jouw inzet als onderdeel van mijn begeleidingsteam in de afgelopen jaren.

Matthieu & Lars Matthieu en Lars, jullie maakten samen mijn begeleidingsteam compleet. Wat een ongelooflijke mazzel heb ik gehad met een team als dit; zo'n sterrencast krijgt niet iedereen. Ook jullie wil ik bedanken voor jullie inzichten, ideeën, en support. Hoogtepunten van de afgelopen vier jaar waren voor mij de momenten waarop ik zoveel mogelijk van mijn team bij elkaar kreeg. (Dat bleek met

jullie agenda's nog vaak een uitdaging, maar de belofte van een goede lunch doet wonderen.) Dit waren voor mij de uitgelezen momenten om mijn mond te houden en te luisteren. Wanneer ik jullie bij elkaar bracht, werd het zonder uitzondering een feest van creativiteit, van diepgaande analyses over mijn werk, van waardevolle nieuwe inzichten, en een frisse kijk op mijn onderzoek. Niets gaf mij zoveel verse moed tijdens mijn PhD-traject als dat. Ik wil jullie van harte bedanken voor jullie inzet in de afgelopen vier jaar en voor jullie rol in mijn team.

Jonathan & Peter Zoals de bovenstaanden beschouwd kunnen worden als de vierkoppige ouders van mijn academische familie, zo zijn jullie mijn oudere broers. Altijd beschikbaar om wat dan ook uit te leggen, ergens over te debatteren, te brainstormen, of nieuwe plannen te maken. Maar tegelijkertijd ook niet bang om me erop te wijzen als ik iets stoms aan het doen ben of om na het werk een biertje mee te drinken. Jonathan en Peter, als ik anderen verhalen vertel over de uniek briljante mensen die je tegen kunt komen in een academische setting, gaat het vaak over jullie.

Reading Committee My sincere thanks to Prof. dr. Daniel Oberski, Prof. dr. Marco Spruit, Prof. dr. Antal van den Bosch, Dr. ing. Georg Krempl, and Dr. Ioanna Lykourentzou for joining my reading Committee. Thank you for lending your expertise and time for the assessment of my dissertation. I look forward to answering your questions during my defense with you.

Hofvijverkring Met gulle hand hebben de leden van de Hofvijverkring mij via hun Fellowship geholpen om mijn onderzoek internationaal uit te voeren. Zij hebben mij in staat gesteld om naar verre oorden te reizen, daar mijn inzichten te delen, en terug te keren met ervaringen die niet alleen mijn onderzoek hebben verrijkt, maar ook mijzelf als wetenschapper en als persoon.

ASReview Crew Elke week op donderdag komen de mensen van het AI Knowledge Discovery Lab bij elkaar om te werken, te overleggen en, misschien wel het belangrijkste, om te lunchen. Voordat ik aan mijn PhD begon, werd ik ervoor gewaarschuwd dat het een eenzaam en vervreemdend traject kan zijn. Het tegendeel bleek waar: er ging geen week voorbij waarin ik niet iemand nieuws ontmoette. Sommigen spreek je eenmaal; anderen spreek je regelmatig, en hoop ik in de toekomst te blijven spreken. Een enorme extended family dus, om de familiemetafoor nog even aan te halen. In het laatste jaar kreeg ik er zelfs een klein PhD-broertje bij. Timo, ik wens je een net zo fijne ervaring toe als ik heb gehad in onze onderzoeksgroep, en ik hoop dat ik nog af en toe stukken toegestuurd krijg voor feedback.

Matt, Lex, Thomas, Tim, Tom, Wøut Mijn jongens wil ik van harte bedanken voor de afgelopen jaren. Niet zozeer voor de diepe wetenschappelijke inzichten die ze me hebben verschaft; die waren eerlijk gezegd vrij schaars. Maar wel voor de ontspanning, de afleiding, en vooral de hoognodige normaliteit. Bedankt voor een luisterend oor als ik weer eens wat te klagen had, het gemoedelijke knikje, en een drankje om het academische leed te verzachten. Jullie weten me altijd weer vakkundig te verlossen van intellectueel gezever en me met beide benen op de grond te zetten.

Mijn ouders & familie Cas, Simone, Fenna, en de rest van mijn lieve familie, ook jullie wil ik bedanken voor jullie support de afgelopen jaren. Bedankt voor het lezen van mijn stukken en voor het luisteren naar mijn verhalen. Ik kan er zonder twijfel van uitgaan dat jullie onvoorwaardelijk voor mij klaarstaan. Dit geeft me de vrijheid om risico's te nemen en nieuwe avonturen aan te gaan, zoals in de afgelopen vier jaar, maar ook in de komende jaren.

Tessa En tot slot, mijn grootste steun: Tessa. Jarenlang kon ik bij jou terecht voor alles wat ik maar wilde. Begreep je elk onderwerp? Misschien niet, maar *by God*, wat deed je je best om het te begrijpen. En tot mijn grote vreugde kon je, nadat ik vier jaar lang tegen je aan had gepraat, aan iemand anders perfect uitleggen waar mijn onderzoek over gaat. Een groter plezier kon je mij niet doen.

Bedankt voor je onuitputtelijke support en begrip. Ik heb genoten van onze avonturen samen tot nu toe en kijk enorm uit naar alles wat ons nog te wachten staat.

ABSTRACT: Systematic reviews are the gold standard for synthesizing scientific evidence, yet the explosion of published research is turning their manual screening phase into an unsustainable bottleneck.

Data Science offers a solution: Active Learning. This machine learning technique incrementally learns from the user to prioritize relevant records, without needing large prelabeled datasets. Tools like ASReview introduce this technology, but its implementation is far from complete, and deploying it effectively in academia requires more than raw algorithmic performance.

This book details an academic Applied Data Science project that improves the effectiveness of Active Learning within ASReview, following the technology as its implementation matures. While industry projects often focus on performance metrics, the academic setting introduces additional demands for explainability and reproducibility. Simultaneously, the applied nature of the work requires delivering actionable advice.

To define how to execute an applied study in this environment, this dissertation engages with four core themes:
Human-Centered Design, Software Usability, Reproducibility and Evidence, and FAIR data.

ISBN: 978-90-393-8059-8