Utrecht University

A Data-Driven Decision Model for Machine Learning Model Selection

Author:
Lex Steffens

1st supervisor:
Dr. Slinger Jansen
2nd supervisor:
Prof. dr. Fabiano Dalpiaz
External supervisors:
Dr. Siamak Farshidi
Elena Baninemeh

August 30, 2024

Abstract

Context: Machine learning models are readily accessible and extensively utilized due to their practical utility in predictive modeling tasks. Despite the consistent performance of individual models, selecting the appropriate model for a specific applied machine learning problem remains a significant challenge for research modelers. Various features, such as model trainability and stakeholder comprehensibility, must be considered when applying these models. These considerations can critically influence the long-term viability of a machine learning model.
Method: To address this challenge, we present a meta-model for the decision-making process in the context of machine learning model selection. The creation of this decision model adopts a systematic research approach, combining systematic literature review, expert interviews, case studies, and design science to investigate machine learning model selection approaches. The systematic literature review enables us to gather and analyze relevant information from existing literature. The expert interviews allow a critical approach to our collected data. The case studies help us assess the practical applicability of our findings. Design science allows for the finalization of a decision model.
Results: Our study analyzed 43 common models across 72 common features. We provide a comprehensive taxonomy of machine learning paradigms, approaches, and domains. We provide insights into potential model combinations, trends in model selection, evaluation measures, and frequently used datasets for training and evaluating these models. The collected data was incorporated into a decision model, further developed through expert interview feedback. Finally, the decision model was practically evaluated through eight case studies.
Contribution: Our study presents a data-driven decision model that could aid research modelers in machine learning model selection. We highlight the importance of further developing the decision model to improve its accuracy and scope beyond its current state.

Keywords: Machine learning, model selection, decision model, systematic literature review, expert interview, case study, design science

Contents

Introduction

Machine learning, a sub-field of artificial intelligence (AI) [1], has emerged as a pivotal domain within computer science [2], and statistics [3], characterized by its capacity to enable systems to learn and make decisions from data [4]. Unlike traditional programming paradigms, which rely on explicit instructions to perform tasks, machine learning leverages algorithms to identify patterns within data and improve performance on a given task with experience [5]. This paradigm shift has fueled advancements across numerous fields, including natural language processing (NLP) [6], computer vision [7], and predictive analytics [8], underscoring the transformative potential of machine learning technologies.

Central to machine learning are models, which are mathematical constructs designed to represent complex relationships within data. These models serve as the foundation for learning from and making predictions on data [9]. Typically, a machine learning model comprises a set of parameters that are iteratively adjusted during the training phase to minimize error and enhance predictive accuracy [10]. The diversity of machine learning models is broad, encompassing supervised models [11], such as linear regression [12] and support vector machines (SVM) [13], unsupervised models [14, 15] like clustering algorithms [16], and reinforcement learning models [17] that optimize decision-making through interaction with an environment. The nature of the data influences the choice of model [18], the specific task at hand [19], and the desired balance between interpretability and predictive power [20].

Machine learning models are evaluated based on their performance metrics, such as accuracy, precision, recall, and F1-score for classification tasks [21], or mean squared error for regression tasks [22]. The iterative process of training and validating these models is critical to ensuring their robustness and generalizability to new, unseen data [23]. The continued evolution of machine learning models, driven by advancements in computational power and algorithmic innovation, holds promise for even greater capabilities in analyzing and interpreting complex datasets [24].

Machine learning models are readily accessible and extensively utilized due to their practical utility in predictive modeling tasks [25]. Research modelers are academic researchers who apply machine learning models and techniques within their studies. Despite the consistent performance of individual models, selecting the appropriate model for a specific applied machine learning problem remains a significant challenge for research modelers [26]. In applying these models, various features must be considered, such as model trainability [27] and comprehensibility for stakeholders [28]. These considerations can critically influence the long-term viability of a machine learning model.

1.1 Background

The selection and adoption of machine learning models constitute a critical phase in the deployment of machine learning solutions, characterized by a series of intricate and multifaceted challenges. This process involves identifying the most suitable model for a given task and effectively integrating it into a practical application, ensuring optimal performance and reliability. Many complexities and challenges come into play:

Diverse model landscape: The wide variety of machine learning models, each with unique strengths and weaknesses, requires a thorough understanding of their characteristics to choose the most appropriate one for a given task. Supervised, unsupervised, and reinforcement learning models cater to different application needs [29].

Data quality and quantity: Model performance is significantly influenced by data quality and quantity. Data often contains noise, missing values, or imbalances, necessitating advanced preprocessing and feature engineering [30]. Insufficient data can lead to underfitting, hindering the model's ability to generalize [31].

Hyperparameter tuning: Models have hyperparameters that require careful tuning for optimal performance [32]. This involves extensive experimentation and significant computational resources, employing methods like grid search and Bayesian optimization [33].

Overfitting and generalization: A major challenge is ensuring the model generalizes well to new data. Overfitting occurs when a model captures noise in the training data, leading to poor performance on test data [34]. Cross-validation [35], regularization [36], and dropout [37] are common techniques to mitigate overfitting.

Evaluation metrics: Accurate model assessment depends on selecting appropriate evaluation metrics, such as accuracy, precision, and recall for classification tasks or mean squared error for regression tasks. Thorough cross-validation ensures the model meets performance criteria and is robust [23].

Deployment and integration: Deploying a model into production involves challenges like ensuring computational efficiency [38], managing latency [39], and maintaining system compatibility [40]. Continuous monitoring and maintenance are vital to address issues like model drift [41], ensuring ongoing reliability and performance.

Model selection is selecting or adopting one or multiple models among a vast set of candidate models for a predictive modeling task [26]. There are many concerns and parameters regarding model selection, like model performance [42] and complexity [43]. Some research modelers use an automated or dynamic machine learning selection method, which can support complex decision-making [44, 45]. We propose creating a decision model [46] that helps select the adequate machine learning model for a given problem. This decision model would support research modelers in their specified machine learning fields.

Our decision model can be applied in a variety of different use cases. In practice, it provides a user with a list of recommended models based on their suggested features. We designed the decision model so that anyone can select one or multiple machine learning-related features from a list. Based on this selection, the decision model will automatically generate a list of machine learning models it deems relevant to the selected features. From this generated list, the user can select a single machine learning model to use. Furthermore, the user could go down the provided list to compare several of the suggested machine learning models. This versatility aids research modelers in using the decision model for various machine learning model selection tasks.

Machine learning is a wide domain with various models, each with unique features and criteria for evaluation [47]. These models are applied in various practical fields and many scientific papers covering research projects [48]. A model selection challenge arises when individuals responsible for making machine learning model selection decisions, such as data scientists, are unfamiliar with all available models and their variations [49]. This lack of familiarity can make determining which model to choose difficult. Additionally, understanding how different models can work together and impact each other's performance is another challenge when selecting models for a machine learning-based system [50].

In summary, the selection and adoption of machine learning models are complex and multifaceted, requiring expertise in model characteristics, data quality, hyperparameter tuning, generalization, evaluation metrics, and deployment strategies. This iterative process is essential for successfully implementing reliable machine learning solutions.

1.2 Objectives and challenges

With the development of a decision model come several possible hurdles. The data to be fitted into the decision model could be insufficient, requiring further literature study collection [51]. The model might underfit the decision parameters and not supply specific recommended machine learning models [52]. Contrarily, the decision model might overfit and return only a handful of options at all times [53]. The decision model itself might not be robust for future advancements in machine learning. To remedy this, the model requires a certain degree of flexibility [54]. Further questions can be raised about the data-gathering process involved, such as whether the decision model data aligns with the views of experts [55].

The field of machine learning-based research encompasses various machine learning models, with examples like support vector machines [13, 56], random forests (RF) [57], naïve Bayes [58], and deep learning models like generative adversarial networks (GAN) [59], self-organising maps (SOM) [60], and multilayer perceptrons (MLP) [61]. A thorough examination of these models and their characteristics provides a comprehensive understanding of their advantages and limitations, offering valuable insights for future research and development.

Machine learning models are complicated algorithms with various features we can define [62]. A single machine learning model often supports many features [63], and the diversity between models can lead to a sizeable list of alternative solutions for a single machine learning problem [64]. Features provide one method of presenting solutions for machine learning model selection.

Combining different machine learning models in a research project is a common practice aimed at leveraging the strengths of various models to achieve more accurate predictions, enhance model robustness, and improve overall performance [65]. There are several approaches to combining different machine learning models with the intent of improving performance and robustness [66, 67]. Combining multiple machine learning models in a research project involves thoughtful selection and evaluation. Analyzing and documenting common combinations can aid in machine learning model selection [68].

Selecting the most suitable machine learning model within a decision model can be challenging due to the vast array of models and approaches available [69]. The lack of a transparent classification scheme further complicates the model selection process [70]. Research modelers often struggle to navigate the multitude of available models, leading to uncertainty and a lack of confidence in selecting the optimal model for their specific requirements [26]. Overcoming these challenges is crucial for developing effective solutions in machine learning model recommendation tasks, underscoring the need for continued research to enhance decision models and development processes.

For specific domains, there is no consensus on machine learning model selection. Research modelers may perform a domain-specific performance comparison on several machine learning models, noting the strengths and weaknesses of each model [71, 72]. Some of these comparisons ultimately conclude that no single best model applies to their specific study domain [73, 74]. Such examples highlight the complexity of using machine learning model selection in particular domains.

A single model can have multiple variations. Considering support vector machines, [13, 56], support vector regression (SVR) is a variant of the same model that is suited for regression tasks rather than classification [75]. Support Vector Machines can be further distinguished between linear SVM and non-linear SVM variants [76]. These variants grant extended depth to a research modeler's potential decisions, which should support vector machines as the recommended machine learning model.

1.3 Research process

While machine learning models and their application decision models have gained significant attention [77], existing research in this field is often scattered across diverse sources, hindering comprehensive understanding [78]. Moreover, the multitude of machine learning models, concepts, datasets, and evaluation measures utilized in this research domain can overwhelm a research modeler [79]. To address these issues, we conducted a systematic literature review (SLR) following the guidelines of Kitchenham [80], Xiao [81], and Okoli [82] to consolidate and analyze the information, providing a more comprehensive understanding of the field. Additionally, we developed a decision model based on the data collected from the literature review, serving as a valuable tool for selecting machine learning models. We expanded the decision model to the needs of domain experts by conducting expert interviews. To evaluate the effectiveness of the decision model, we conducted academic case studies following the guidelines outlined by Yin [83].

Collecting and updating data systematically presents a variety of hurdles. Studies often rely on heterogeneous data sources, leading to an imbalance in direct data and results for a research project [84]. Obtaining high-quality and relevant data is crucial for the effectiveness of a study, which is not always directly available due to various research constraints [85]. One way to mitigate this is through data fusion, which combines information from multiple sources to create a unified dataset that provides an accurate representation [86]. Data integration combines data residing in different sources, often merging data from various databases into a single coherent repository [87]. Data standardization consists of establishing rules and structures for data across different sources to ensure compatibility within a single research project [88].

The decision model is designed based on a multi-criteria decision-making (MCDM) model [89]. This provides a structured approach to evaluating and selecting the best alternative decisions based on multiple and often conflicting criteria. This structure effectively compares several machine learning models

and their features. Research modelers provide a selection of requirements for the decision model, and the final provided solution is a subset of machine learning models that meet all requirements.

The decision model was expanded through expert interviews. We contacted various research modelers based on their relevant written papers and inquired how to improve our decision model. Emphasis is put on the scalability, correctness, and flexibility of the decision model, as well as potential overhauls.

The decision model was finalized through design science, resulting in the creation of an artifact. This artifact is our machine learning decision model, evaluated in applicability and utility. Based on this, a usable machine learning model selection format was developed.

The decision model was evaluated through several case studies. For select papers on machine learning, we compared the decision model the research modelers used with our decision model. We investigated what machine learning models would be given as the final result based on our decision model, given the research modeler's requirements. This provided valuable insight into the scalability, correctness, and flexibility of our decision model.

1.4 Related work

When discussing how to select the adequate machine learning model for a given problem, several existing research papers shed perspectives with methods highlighting both the technical and social aspects of machine learning model selection.

Pengzhi Li et al. [90] explained the principle of conventional autoencoders and investigated the primary development process of an autoencoder. They proposed a taxonomy of autoencoder models according to their structures and principles.

Ahmed Tealab et al. [91] studied the advances in time series forecasting models [92] using artificial neural network methodologies [93] in a systematic literature review. They concluded that many studies presented the application of neural network models. Still, few proposed new neural network models for forecasting that considered theoretical support and a systematic procedure in constructing the model.

Mensah Kwabena Patrick et al. [94] provided a comprehensive review of the state-of-the-art architectures, tools, and methodologies in existing implementations of capsule networks. They highlighted the successes, failures and opportunities for further research to motivate researchers and industry players to exploit the full potential of this new field.

Sana Aroussi et al. [95] presented an overview of user Quality of Experience (QoE) and network Quality of Service (QoS) correlation models based on machine learning models. They proposed categorizing correlation models, and their survey provided research modelers with the latest trends and findings within this field.

Jie M. Zhang et al. [96] presented a comprehensive survey of techniques for testing machine learning models. The paper also analyzed trends concerning datasets, research trends, and research focus, concluding with research challenges and promising research directions in machine learning testing.

Zhengjing Ma et al. [97] presented a comprehensive survey of relevant research on machine learning applied in landslide prevention, mainly focusing on landslide detection based on images, susceptibility assessment, and the development of landslide warning systems.

Neha Agarwal et al. [98] presented a systematic literature review on Web Service Clustering (WSC) approaches to enhance service discovery, selection, and recommendation. The SLR also presented various mandatory and optional steps of WSC, evaluation, measures, and datasets.

Amy X. Zhang et al. [99] conducted an online survey with 183 participants who work in various aspects of data science to investigate how data science workers collaborate. They found that data science teams are highly collaborative and work with various stakeholders and tools during the six common steps of a data science workflow.

Suejb Memeti et al. [100] undertook a systematic literature review to aggregate, analyze, and classify the existing software optimization methods for parallel computing systems. They reviewed approaches that use machine learning or meta-heuristics for software optimization at compile-time and run-time.

Victoria Zakharova et al. [101] aimed to fill the research gap in understanding how data specialists evaluate machine learning models and how they communicate results to third parties. A qualitative research design was suggested to explore this, and semi-structured interviews with machine learning practitioners were conducted.

Lei Cui et al. [102] provided a comprehensive survey of the advances in detecting false data attacks using machine learning models. They further investigated the potential research directions considering the deficiencies of current machine learning based mechanisms.

Zuhaira Muhammad Zain et al. [103] performed a comprehensive systematic literature review and meta-analysis on the application of deep learning in software defect prediction. They provided a reference point for future research which could be used to improve research quality in this domain.

Balaji T.K. et al. [104] provided a comprehensive survey of multiple applications of social media analysis using robust machine learning models. They summarized the challenges and benefits of machine learning usage in social media analysis.

Naira Kaieski et al. [105] performed a systematic literature review that aimed to provide an updated computational perspective of how machine learning has been applied to analyze the vital signs of adult hospitalized patients. Their findings demonstrate that many researchers are exploring the use of machine learning models in tasks related to improving the health outcomes of hospitalized patients in distinct units.

Manal Binkhonain et al. [106] reported a systematic review of machine learning models for identifying and classifying non-functional requirements. Their review finds that while machine learning models have potential in the classification and identification of non-functional requirements, they face some open challenges that will affect their performance and practical application.

Table 1: Our study compared to related work studies.

Ref.	Year	Type	RM	#Pub	DM	Tr.	DS	Cat.	MC	#M	#F	#E
Our study	2024	Aca	SLR/EI/CS/DS	275	Yes	Yes	Yes	Yes	Yes	43	72	18
[90]	2023	Aca	Survey	N / A	No	No	No	No	No	18	0	0
[91]	2018	Aca	SLR	17	No	No	No	No	No	12	0	0
[94]	2022	Aca	Survey	N / A	No	No	Yes	No	No	1	0	0
[95]	2014	Aca	Survey	N / A	No	No	No	Yes	No	5	10	5
[96]	2020	Aca	Survey	138	No	Yes	Yes	Yes	No	0	0	0
[97]	2021	Aca	Survey	N / A	No	No	No	Yes	No	19	0	0
[98]	2022	Aca	SLR	84	No	No	Yes	Yes	No	25	18	20
[99]	2020	Aca	Survey	N / A	No	No	No	No	No	0	0	0
[100]	2019	Aca	SRL	57	No	No	No	Yes	No	26	10	0
[101]	2021	Aca	Interview	N / A	No	No	No	No	No	0	0	0
[102]	2020	Aca	Survey	N / A	No	No	Yes	Yes	No	26	6	0
[103]	2023	Aca	SLR	63	No	No	Yes	Yes	No	31	0	15
[104]	2021	Aca	Survey	N / A	No	No	Yes	Yes	No	19	0	24
[105]	2020	Aca	SLR	78	No	No	No	Yes	No	4	0	0
[106]	2019	Aca	Review	N / A	No	No	No	Yes	No	16	0	7

A comprehensive contextualization of our study's position within the existing body of literature can be observed in Table 1, identified through our systematic literature review. Our review encompassed 275 publications (#Pub), making our research extensive within this domain. The table comprises various columns, each showing distinct data. Through our SLR process, we curated only academic (Aca) literature related to machine learning model selection spanning from 2014 to 2023.

The research methods (RM) employed in the selected studies include SLR, Survey, Review, and Interview. Our study incorporates SLR, expert interviews, case studies, and design science, offering a holistic perspective on machine learning model selection. This is noticeably more complex compared to the other SLR studies, as they did not include expert interviews to improve their findings or case studies to evaluate their findings, instead only reporting on the outcomes of the SLR process.

In comparison to the reviewed SLRs, our study stands out for its emphasis on creating a decision model

(DM), which the other studies did not mention. This decision model could be a valuable tool for research modelers, guiding informed decisions and identifying suitable models or model combinations for specific tasks. Regarding observed trends (Tr.) in machine learning model usage, one other study [96] reported on this aspect. Furthermore, six other studies [94, 96, 98, 102, 103, 104] provided valuable insights into open-access datasets (DS) that are commonly used for model evaluation, serving as valuable resources for the research community. Additionally, our study categorized (Cat.) models, in line with ten other studies [95, 96, 97, 98, 100, 102, 103, 104, 105, 106] in the field. However, none of the other papers reported on model combinations (MC), making it challenging to effectively compare our model combination data.

The table highlights the magnitude of the analysis conducted in each study. Our study encompasses a significant number of models (#M), features (#F), and evaluation measures (#E), compared to other studies. Where most studies directly addressed models, only some accounted for features or evaluation measures.

1.5 The role of large language models in machine learning model selection

In recent years, large language models (LLM) [107] have revolutionized the field of artificial intelligence with their ability to understand and generate human-like text, with models like GPT [108] and Gemini [109] being widely adopted for aid in machine learning model selection [110]. While our decision model incorporates an MDCM approach based on defining its features a priori, large language models operate on a conversational level with their user [111]. Large language models have proven to outperform traditional feature engineering methods in several domains [112, 113]. Despite these capabilities, large language models present several challenges that must be considered in machine learning model selection.

While large language models have demonstrated substantial proficiency in various tasks, their reliability can be inconsistent. Large language models operate as black boxes, meaning their decision-making processes are not transparent [114]. This opacity can be problematic, especially in applications requiring high-accuracy data. Furthermore, the datasets used to train these large language models are often proprietary and not publicly disclosed, leading to potential biases and lack of reproducibility [115]. The creation of a custom decision model can mitigate both of these challenges. Our decision model is fully transparent, as the inner workings of the model are only controlled by what features the user provides. Furthermore, our decision model does not require external datasets, as its data was entirely collected through our study's procedures.

Large language models can come with significant computational costs, as running these models requires substantial hardware resources and energy consumption, making them expensive to deploy and maintain [116]. Furthermore, state-of-the-art large language models are generally paired with expensive subscription services for practical use in research [117]. These financial burdens can be particularly challenging for academic and scientific projects, which often operate under tight budget constraints [118]. Despite these limitations, the necessity to employ effective machine learning models remains. Developing a custom model selection approach can be an affordable alternative. By creating a personal artifact, organizations can reduce dependency on expensive large language model infrastructures and subscription fees, leading to significant cost savings. Considering this, our decision model can be an affordable alternative for machine learning model selection.

Since large language models are trained on several datasets simultaneously, data privacy and security arise as potential issues for model selection and training [119]. Using custom datasets can improve data privacy and security. Organizations can ensure that sensitive information is handled appropriately, adhering to privacy regulations and standards. Custom decision models can be designed with specific security protocols, reducing the risk of data breaches associated with third-party data usage.

Large language models require extensive retraining to adapt to new tasks or domains [120]. Custom models can be designed to scale efficiently and be flexible enough to adapt to evolving requirements, providing long-term value. Unlike a pre-trained large language model, a custom decision model can be incrementally updated with new data, maintaining its relevance over time. This scalability is crucial for the machine learning model selection domain, as it is ever expanding with new models and feature requirements [44].

While large language models will continue to be used in the machine learning model selection domain, the aforementioned considerations can lead a research modeler to use a decision model instead.

1.6 Overview

This study presents the development of a decision model. To provide a comprehensive overview, we will discuss utilizing various research methods, including systematic literature review, expert interviews, design science, and case studies. Additionally, it proposes using a decision model based on the collected data to guide research modelers in making informed decisions for their machine learning applications.

Chapter 2 defines the problem statement and research questions and outlines the research methods employed, including systematic literature review, expert interviews, case studies, and design science.

Chapter 3 outlines the methodology used in the systematic literature review, covering the review protocol, paper collection procedures, inclusion/exclusion criteria, quality assessment techniques, data extraction methods, synthesis processes, and systematic search approach.

Chapter 4 presents the findings and analysis of the SLR, exploring various aspects of machine learning model selection, such as models and their characteristics, feature engineering techniques, model combinations, emerging trends, evaluation measures, and available datasets.

Chapter 5 covers the interviews conducted with domain experts, highlighting the various remarks they gave on the data gathered, the structure of said data, and their thoughts on the concept of the decision model itself.

Chapter 6 focuses on the practical utilization of the collected data, addressing project-specific concerns through the introduced decision model. This meta-model is a framework for effective decision-making suited for model selection.

Chapter 7 includes insightful academic case studies that provide practical coverage and validate the conducted research to enrich the evaluation of findings.

Chapter 8 critically examines the outcomes of the systematic literature review, expert interviews, and case studies, answering the research questions, discussing lessons learned, the implications of the findings, and addressing potential threats to the study's validity.

Chapter 9 summarizes the study's contributions and highlights avenues for future research, providing a cohesive closure to the research on a data-driven decision model for machine learning model selection.

Research approach

This study adopted a systematic approach, combining SLR, expert interviews, case studies, and design science to investigate machine learning model selection approaches. The SLR enabled us to gather and analyze relevant information from existing literature. The expert interviews allowed a more critical approach to our collected data. The case studies aided us in assessing the practical applicability of our findings. Design science allowed for the finalization of a functional decision model.

2.1 Problem statement

Machine learning models are vastly applied within academic research through scientific disciplines and practical applications [77, 121]. Their capacity to automatically learn and adapt from data has revolutionized many fields [122]. Because of this, machine learning model selection stands as a cornerstone of many projects, playing a vital role in their success [123]. Selecting the suitable machine learning model significantly influences the quality of predictions [9], the computational efficiency [38], and the interpretability of results [124]. Therefore, the model selection process requires careful consideration of various features, including the nature of the problem [19], the available data [18], and the desired performance metrics [21], making it an indispensable component of modern scientific research and technological advancements.

Developing an effective machine learning model selection decision model relies on accurately identifying the needs of the research project. However, machine learning model selection lacks consensus and comprehensive analysis of optimal approaches [125]. This scattered knowledge makes it challenging for research modelers to choose suitable models for specific scenarios [78]. Understanding prevailing trends [26], emerging patterns [126], and appropriate evaluation measures [47] for machine learning model selection approaches further complicates the development of effective systems. Furthermore, selecting representative datasets for training and evaluation is not straightforward [127]. Consequently, in the realm of machine learning model approaches, the following research challenges have been identified:

Scattered knowledge: The concepts, models, and characteristics of machine learning model approaches are dispersed across diverse academic literature, hindering informed decision-making for developing model selection systems [128]. Systematically consolidating and categorizing existing models is demanding. Research modelers require a comprehensive taxonomy and categorization of machine learning models to make better choices [129].

Model combinations and integration: Combining and integrating models in machine learning is challenging [130]. Finding effective model combinations to improve the decision model accuracy requires investigating compatibility and synergy between models [131].

Trends and emerging patterns: Understanding prevailing trends and emerging patterns in machine learning model selection approaches is crucial [44]. Research modelers need to analyze a large volume of research papers to identify such patterns and tailor their efforts accordingly [31].

Selecting assessment criteria: Choosing appropriate evaluation measures for assessing machine learning model approaches is challenging [132]. Research modelers must identify measures tailored to each approach to evaluate their performance accurately [133].

Selecting datasets: Selecting suitable datasets for training and evaluating machine learning models is complex [134]. Research modelers must analyze and choose representative datasets encompassing various project needs to develop robust selection models [135].

Decision-making process: A comprehensive decision model encompassing various machine learning model concepts and guidelines for selecting model combinations and conducting systematic evaluations is missing from the existing literature [136]. Based on the MCDM structure [89], such a model could aid research modelers in navigating the complexities of machine learning models and streamlining their model selection decision-making processes.

2.2 Research questions

This study features a main research question with six additional research questions regarding machine learning model selection. The MRQ is as follows:

MRQ: How can research modelers be supported in their machine learning model selection process?

The main research question provides the final result of our study. We intended to create a decision model that aids research modelers in their machine learning model selection. Based on the identified research challenges in machine learning model selection, the following research questions are formulated:

RQ1: Which machine learning models should be included in the machine learning model selection decision model?

Through our SLR, we acquired a list of various machine learning models featured in papers. The models found here were the candidates for our decision model.

RQ2: Which characteristics and features should be included in the machine learning model selection decision model?

Through our SLR, we acquired various features related to machine learning models. These features provided a basis for the feature criteria in our decision model.

RQ3: Which training and evaluation datasets should be included in the machine learning model selection decision model?

Through our SLR, we acquired a list of datasets featured in papers. The datasets found here formed a basis for the candidates in our decision model.

RQ4: Which evaluation measures and methods should be included in the machine learning model selection decision model?

We acquired various evaluation measures related to machine learning models through our SLR. The evaluation models found here provided a basis for the evaluation measures featured in our decision model.

RQ5: How can a decision model be developed to support research modelers in selecting machine learning models?

Through a combination of expert interviews and design science, we encountered many concepts that could improve the development of our decision model. We iterated on these concepts, which shaped the decision model's result.

RQ6: How should the machine learning model selection decision model be evaluated?

Through a combination of case studies and design science, several concepts were identified that could enhance the evaluation of our decision model. These concepts underwent iterative refinement, which influenced the final outcome of the decision model.

2.3 Research methods

This section describes the research methods to answer the different research questions.

Table 2: Research questions and their proposed methods.

Research questions	Literature research	Expert interview	Case Study	Design science
MRQ:	X	X	X	X
RQ1:	X			
RQ2:	X			
RQ3:	X			
RQ4:	X			
RQ5:		X		X
RQ6:			X	X

We used a mixed research method [137, 138] to tackle the research questions, combining literature research, expert interviews, case studies, and design science. Table 2 provides an overview.

Literature research: The Systematic Literature Review (SLR) allows us to understand machine learning model approaches comprehensively. The SLR followed guidelines by Kitchenham [80], Xiao [81], and Okoli [82] to identify models, their definitions, model combinations, supported features, potential evaluation measures, and relevant concepts from existing literature.

An SLR is a comprehensive and structured approach to gathering and assessing existing knowledge within a specific domain. We systematically identified, evaluated, and analyzed relevant academic literature while following our predefined protocol described in 3.1.

The process starts with formulating several research questions and establishing the relevant criteria to apply to study selection. Searches are conducted across various scientific databases using predefined keywords as filters. The identified papers are then screened based on their titles and abstracts to assess whether they are relevant enough for a full-text analysis. If so, the full text is analyzed, including the research questions, quality assessment, and data extraction to study findings. Ultimately, the results are synthesized to provide a comprehensive overview of the perceived current state of the knowledge in the domain. Our SLR emphasizes the model categories, model features, model combinations, model trends, evaluation measures, and datasets.

The synthesis of this collective data aids in further the development of the decision model and potential future research by offering insights for research modelers within the domain. Based on the SLR findings, we developed a machine learning model selection decision model.

Expert interview: The expert interview allowed us to shape and guide the development of our decision model. The expert interview follows the guidelines of Meuser [139] regarding the definition of experts. Expert interviews are an essential knowledge-acquisition technique in qualitative research.

The first step to conducting successful expert interviews is identifying the right domain experts. During the conducting of the expert interview, it is necessary to follow a predefined policy as a systematic approach to ask questions for specific information related to the aims of our study, which should align with the research questions. Each interview series would follow a semi-structured interview policy lasting 30 to 60 minutes. Knowledge was captured incrementally, while the interviewer behaved as neutral as possible.

Expert interviews have proven to be a reliable research method [140]; therefore, we further developed our machine learning model selection decision model based on these expert interviews.

Case study: To evaluate the practical applicability of the decision model in real-world scenarios, we conducted case studies following the guidelines of Yin [83]. A case study is an empirical research method investigating a phenomenon within a specific context in the domain of interest. The researcher can freely perform empirical research in any way if it can describe, evaluate, and explain their hypothesis.

Planning and scoping a case study research project that addresses research questions directly can be challenging. The large amount of data needs to be processed correctly. The unit of analysis explains what exemplifies a case, with a complete set of data for one study of the unit of analysis constituting a single case [141]. It is typically defined based on the research questions and the expected level to address said research questions [83]. The number of cases to be studied depends on the focus of the research question, where single issues provide in-depth investigation and full description.

To evaluate our decision model, we selected multiple cases in the context of different research papers to explore the research questions and theoretical evolution more broadly and prevent additional biases. These case studies assessed whether the proposed decision model would effectively assist research modelers in selecting machine learning models for their projects.

Design science: Design science is an iterative process, broadly considered a problem-solving process [142]. It attempts to produce generalizable knowledge about design processes and design decisions.

The design process can be considered similar to a theory, as a set of hypotheses that can eventually be proven directly by the artifact they describe. The feasibility of a design can be supported through a scientific theory so long as the design comprises all principles of said theory. Research investigations often follow a repetitive cycle, and design science similarly tends to iterate on itself through various

gradual changes. Design science can be applied directly to creating a model or taxonomy, where the resulting artifact should exist so that it is robust to any potential changes or alterations in its contents.

Design science enabled us to build the machine learning selection decision model based on the data found in the SLR and provided the foundation for its evaluation.

We addressed all research questions using this mixed research method, including literature research, expert interviews, case studies, and design science. We contributed meaningful insights and practical solutions to advance machine learning model selection decision models.

Systematic literature review methodology

This section contains the overview of the systematic literature review (SLR) and the results of each phase.

In this study, we followed the procedures and guidelines outlined by Kitchenham [80], Xiao [81], and Okoli [82] to address some of the research questions highlighted in Chapter 2. Accordingly, we adopted the following review protocol to collect and extract data from relevant studies systematically. This protocol is displayed in Figure 3.1, and is based on the protocol set by Farshidi [143, 144]. following steps were taken to conduct the SLR:

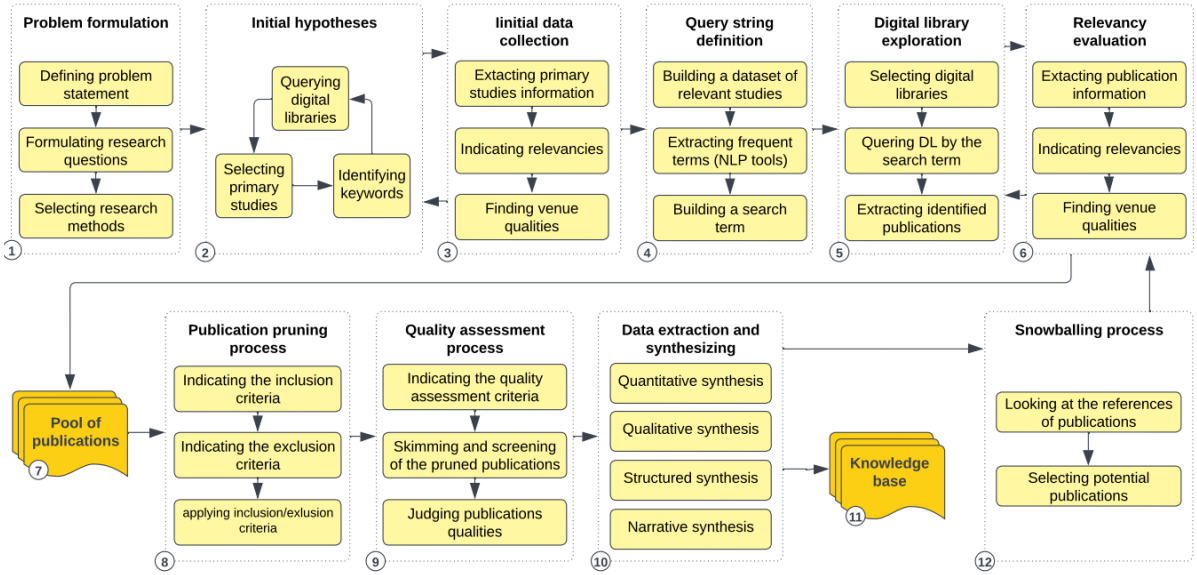


Figure 3.1: Illustrates the review protocol employed in this study, following the prescribed procedures and guidelines outlined by Kitchenham, Xiao, and Okoli. The review protocol consists of 12 elements systematically executed to collect and extract data from relevant studies. These steps ensure a rigorous investigation and adherence to scientific standards in the research process.

(1) Problem formulation: In this research phase, we followed the prescribed procedures and guidelines of Xiao [81] to define the problem statement and research questions. By identifying the research methods, including using an SLR, we ensured that our study addressed a subset of research questions suitable for an SLR. This systematic approach allowed us to conduct a rigorous investigation.

(2) Initial hypotheses: During the initial stage, we considered a set of keywords to search for primary studies that could address our research questions. These keywords formed the basis for identifying potential seed papers, which served as the starting point for our literature review. This method enabled us to explore relevant publications systematically.

(3) Initial data collection: We manually collected a comprehensive set of characteristics for primary studies, including source, URL, title, keywords, abstract, venue, venue quality, type of publication, number of citations, publication year, and relevancy level. This meticulous process ensured that our review focused on essential information and facilitated the establishment of inclusion/exclusion criteria.

(4) Query string definition: By analyzing primary studies' keywords, abstracts, and titles, we constructed a search query based on frequent terms found in highly relevant and high-quality papers. This approach helped refine our search and ensure the inclusion of relevant publications.

(5) Digital library exploration: We thoroughly explored digital libraries such as ACM, ScienceDirect, and Elsevier, using the generated search query to query these databases. This systematic exploration of reputable sources ensured the comprehensive coverage of relevant publications.

(6) Relevancy evaluation: We assessed the characteristics of the resulting publications and added them to our collection while estimating their relevancy based on their alignment with our research questions and challenges. This evaluation process ensured the inclusion of highly relevant publications in our review.

(7) The pool of publications: The collected papers and their associated characteristics formed the pool of publications that served as the foundation for our subsequent review.

(8) Publication pruning process: We rigorously applied inclusion/exclusion criteria to evaluate the pool of publications, eliminating irrelevant material and refining the selection to include the most relevant and high-quality studies. This process enhanced the quality and focus of our review.

(9) Quality assessment process: We assessed the quality of the remaining publications based on established criteria, including the clarity of research questions and findings. This evaluation ensured that only high-quality studies were included in our review, enhancing the reliability of our findings.

(10) Data extraction and synthesizing: Through systematic data extraction, we obtained relevant information from the selected publications, synthesizing the findings to identify key insights. This rigorous process facilitated the identification and summarization of critical information.

(11) Knowledge base: The final set of selected highly relevant and high-quality publications and their characteristics formed a comprehensive knowledge base. Additionally, the extracted data provided a mapping that connected specific findings to their respective sources. This knowledge base is a valuable resource for future research, offering a consolidated summary of the essential conclusions and enabling further analysis.

By meticulously following this systematic review protocol, we followed rigorous and scientific standards in collecting and analyzing the relevant literature on machine learning models. This approach ensured the validity and reliability of our study, allowing us to address the research questions identified in our study effectively.

3.1 Paper collection

During the automatic search phase of our systematic literature review, we implemented a robust search strategy to retrieve pertinent and high-quality publications from scientific search engines. We extracted keywords from an initial set of publications obtained through the manual search process to formulate our search query. These keywords were identified based on the frequent terms used by research modelers in highly relevant and high-quality papers. We further refined the keyword selection using a topic modeling tool, Sketch Engine [145], which helped identify pertinent additional terms. We were unsatisfied with the number of relevant publications we found during this SLR search query, so we decided to reformulate our search query based on the results of the first phase. We redid this phase with our new search query, and in total, we identified 128 highly relevant and high-quality publications during this initial phase of the SLR.

The search query was carefully constructed to target publications addressing machine learning model selection in software engineering. It was formulated using the logical operators ”AND” and ”OR” to combine the selected keywords. The final search query in this SLR was as follows.

("machine learning" OR "ml") AND ("model" OR "technique" OR "algorithm" OR "framework") AND ("software engineering")

The search query was employed during the automatic search phase, and the resulting publications 3009 results underwent a rigorous screening process based on our predefined inclusion/exclusion criteria. This ensured that only relevant and high-quality publications were included in our data extraction and analysis. The effectiveness of the search query was assessed by comparing the search results with those obtained from the manual search to ensure consistency and comprehensiveness.

3.2 Inclusion/exclusion criteria

Inclusion/exclusion criteria are essential guidelines used to determine the relevance and eligibility of studies for inclusion in a systematic literature review or meta-analysis [146]. These criteria ensure that the selected studies are high-quality and directly address the research question under investigation. Inclusion criteria specify the characteristics or attributes a study must possess to be considered for inclusion in the review. We employed rigorous inclusion and exclusion criteria during this study phase to filter out irrelevant and low-quality publications. Our criteria encompassed several factors, including the quality of the publication venue, the publication year, the number of citations, and the relevancy of the publication to our research topic. These criteria were carefully defined and consistently applied to ensure that only high-quality and relevant publications were included in our review. By adhering to these criteria, we evaluated publications that provided valuable insights and contributed significantly to our research topic. After applying our predefined inclusion/exclusion criteria, we identified and selected 524 publications from the initial pool of 3009.

3.3 Quality assessment

During the SLR, we comprehensively assessed the quality of the selected publications after applying the inclusion/exclusion criteria. Several factors were taken into consideration to evaluate the quality and suitability of the publications for our research:

Research method: We evaluated whether the chosen research method was appropriate for addressing the research question. The clarity and transparency of the research methodology were also assessed.

Research type: We considered whether the publication presented original research, a review article, a case study, or a meta-analysis. The relevance and scope of the research in the field of machine learning were also taken into account.

Data collection method: We evaluated the appropriateness of the data collection method concerning the research question. The adequacy and clarity of the reported data collection process were also assessed.

Evaluation method: We assessed whether the chosen evaluation method was suitable for addressing the research question. The transparency and statistical significance of the reported results were considered.

Problem statement: We evaluated whether the publication identified the research problem and provided sufficient background information. The clarity and definition of the research question were also taken into account.

Research questions: We assessed the relevance, clarity, and definition of the research questions concerning the research problem.

Research challenges: We considered whether the publication identified and acknowledged the challenges and limitations associated with the research.

Statement of findings: We evaluated whether the publication reported the research results and whether the findings were relevant to the research problem and questions.

Real-world use cases: We assessed whether the publication provided real-world use cases or applications for the proposed method or model.

Based on the assessments of the above factors, we evaluated the quality of the publications. Through this evaluation process, 275 publications were selected from the initial pool of 524. These selected publications demonstrated high quality and relevance to our research question, meeting the predefined inclusion/exclusion criteria. The consensus ensured a rigorous and reliable selection of publications for further analysis and data extraction in the SLR.

3.4 Data extraction and synthesizing

During the data extraction and synthesis phase of the SLR, our primary objective was to address the identified research questions and gain insights into the foundational models commonly employed by research modelers in their machine learning model selection approaches. We aimed to understand the features of these models and the evaluation measures utilized by research modelers to assess their

approaches. Furthermore, we explored the potential combinations of models that research modelers incorporated into their research papers.

We extracted relevant data from the papers included in our review to achieve these objectives. From our perspective, evaluation measures encompass a range of measurements and key performance indicators (KPIs) used to evaluate the performance of the models [147]. Features can encompass any characteristics of models that authors highlight to demonstrate specific functionalities. These features play a role in a research modeler's selection of models. Examples of features include a model's data compatibility [18] or their prediction capabilities [9].

In this context, 'models' refer to mathematical, algorithmic models or processes that can be applied in various domains. For instance, support vector machines [13] [56] and long short-term memory (LSTM) [148] are examples of models commonly utilized in machine learning.

By extracting and analyzing this data, we aimed to comprehensively understand the existing literature, including popular open-access datasets used for training and evaluating the models. This knowledge empowered us to contribute insights and recommendations to the academic community, supporting them in selecting appropriate models and approaches for their machine learning research endeavors.

3.5 Search process

In this study, we followed the review protocol presented in this section to gather relevant studies. The search process involved an automated search phase, which utilized renowned digital libraries such as ACM DL, IEEE Xplore, ScienceDirect, and Springer, and a manual search phase using Google Scholar. Google Scholar was excluded from the automated search due to its tendency to generate numerous irrelevant studies [149]. Furthermore, Google Scholar significantly overlaps the other digital libraries considered in this SLR [150]. Figure 3.1 provides a comprehensive overview of the sequential phases of the search process, outlining the number of studies encompassed within each stage.

Table 3: Presents an overview of the systematic search process employed to identify relevant publications on machine learning model selection. The search process involved both manual and automatic searches, incorporating specific inclusion/exclusion criteria to ensure the retrieval of high-quality results. The search query used in the automatic search was carefully designed to retrieve relevant publications from scientific search engines, while the manual search involved screening articles from selected venues. The final set of articles obtained from both searches was then subjected to comprehensive analysis and synthesis to provide valuable insights into the current state of machine learning model selection research.

Publication	Phase 1	Phase 2	Phase 3	Phase 4
Google Scholar	25	24	24	16
ACM DL	741	157	157	61
IEEE Xplore	132	104	104	55
ScienceDirect	330	129	129	78
Springer	1781	110	110	65
Total	3009	524	524	275

Table 3 provides numeric insights into the search process conducted in four phases: Phase 1, Phase 2, Phase 3, and Phase 4.

Phase 1 (Pool of publications): We initially performed a manual search, resulting in 25 relevant publications through Google Scholar. Additionally, automated searches from ACM DL, IEEE Xplore, ScienceDirect, and Springer contributed to the pool of publications. The complete result was 25 Google Scholar, 741 ACM DL, 132 IEEE Xplore, 330 ScienceDirect, and 1781 Springer publications.

Phase 2 (Publication pruning process): In this phase, the inclusion/exclusion criteria were applied to the collected publications, ensuring the selection of high-quality and relevant studies. The numbers were reduced to 24 in Google Scholar, 157 in ACM DL, 104 in IEEE Xplore, 129 in ScienceDirect, and 110 in Springer publications, respectively.

Phase 3 (Quality assessment process): Quality assessment was conducted for the publications based on several criteria, resulting in a final selection of 524 publications from all sources.

Phase 4 (Data extraction and synthesizing): During this phase, data extraction and synthesis were performed to gain insights into foundational machine learning models, features, evaluation measures, and potential combinations of models used by research modelers. By carefully applying the review protocol, we retrieved 275 high-quality studies for our comprehensive analysis and synthesis in this systematic literature review, with 16 publications in Google Scholar, 61 publications in ACM DL, 55 publications in IEEE Xplore, 78 publications in ScienceDirect, and 65 publications in Springer.

Findings and analysis

In this section, we analyze the SLR results and provide an overview of the collected data, which was analyzed to address the research questions identified in our study.

4.1 Models

The SLR conducted in our study has revealed a diverse array of models employed in machine learning. These models encompass several approaches, each characterized by unique characteristics and methodologies. For a comprehensive understanding of these models, including their definitions and descriptions, please refer to the Appendix A.

We have examined their underlying principles and methodologies to categorize these models effectively. The Appendix A provides detailed definitions and explanations of the identified categories, offering comprehensive insights into each category and its specific characteristics.

Among the identified categories, prominent ones include supervised, unsupervised, and reinforcement learning. These categories encompass a broad range of models applied in machine learning. However, it is essential to note that these three specific categories represent only a subset of the diverse range of models identified in our SLR.

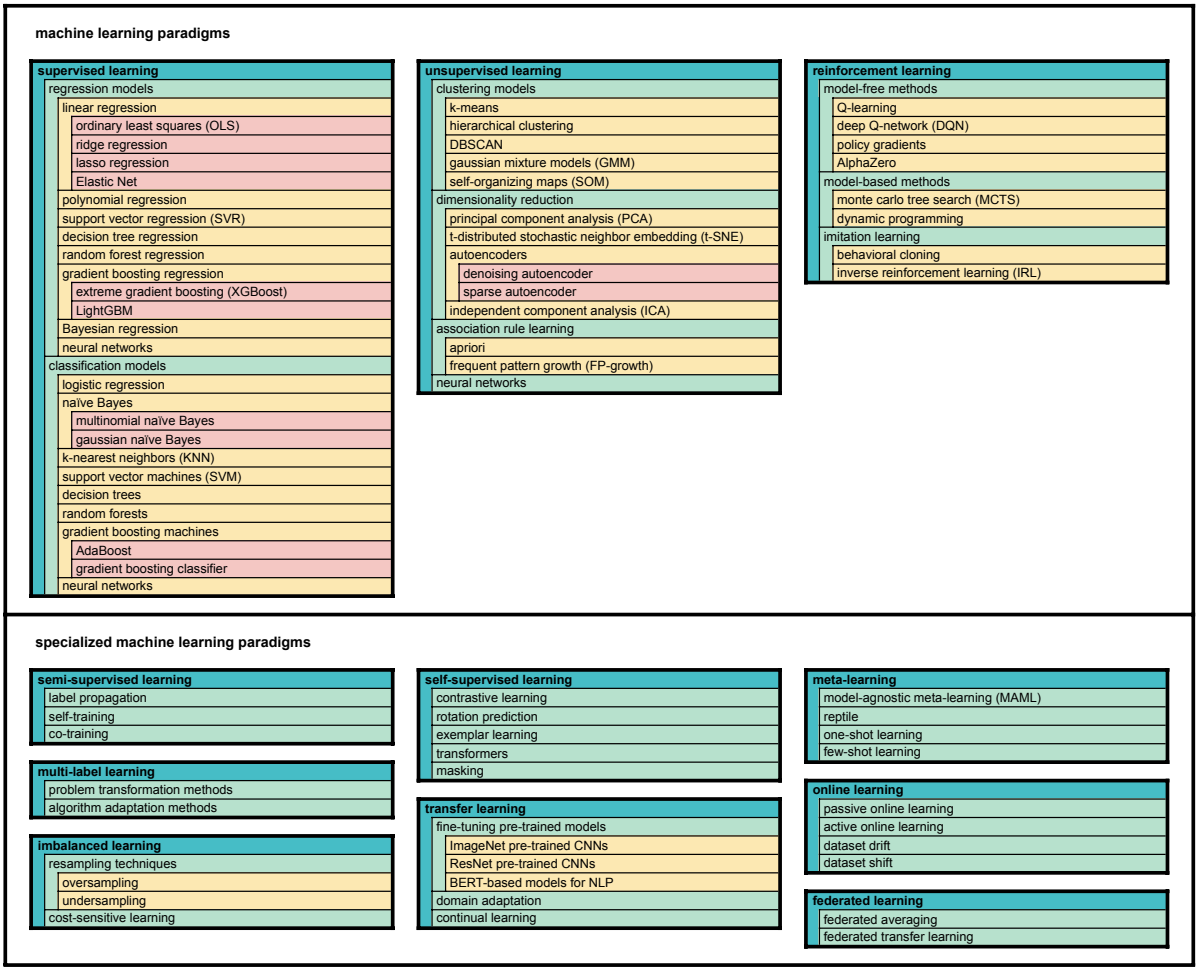


Figure 4.1: Shows the first half of the taxonomy for machine learning models. For access to the complete list of models and related mapping, please refer to the Appendix A.

Figure 4.1 and Figure 4.2 present our developed taxonomy based on the SLR on machine learning model selection. The taxonomy was set up using machine learning and model descriptions from the

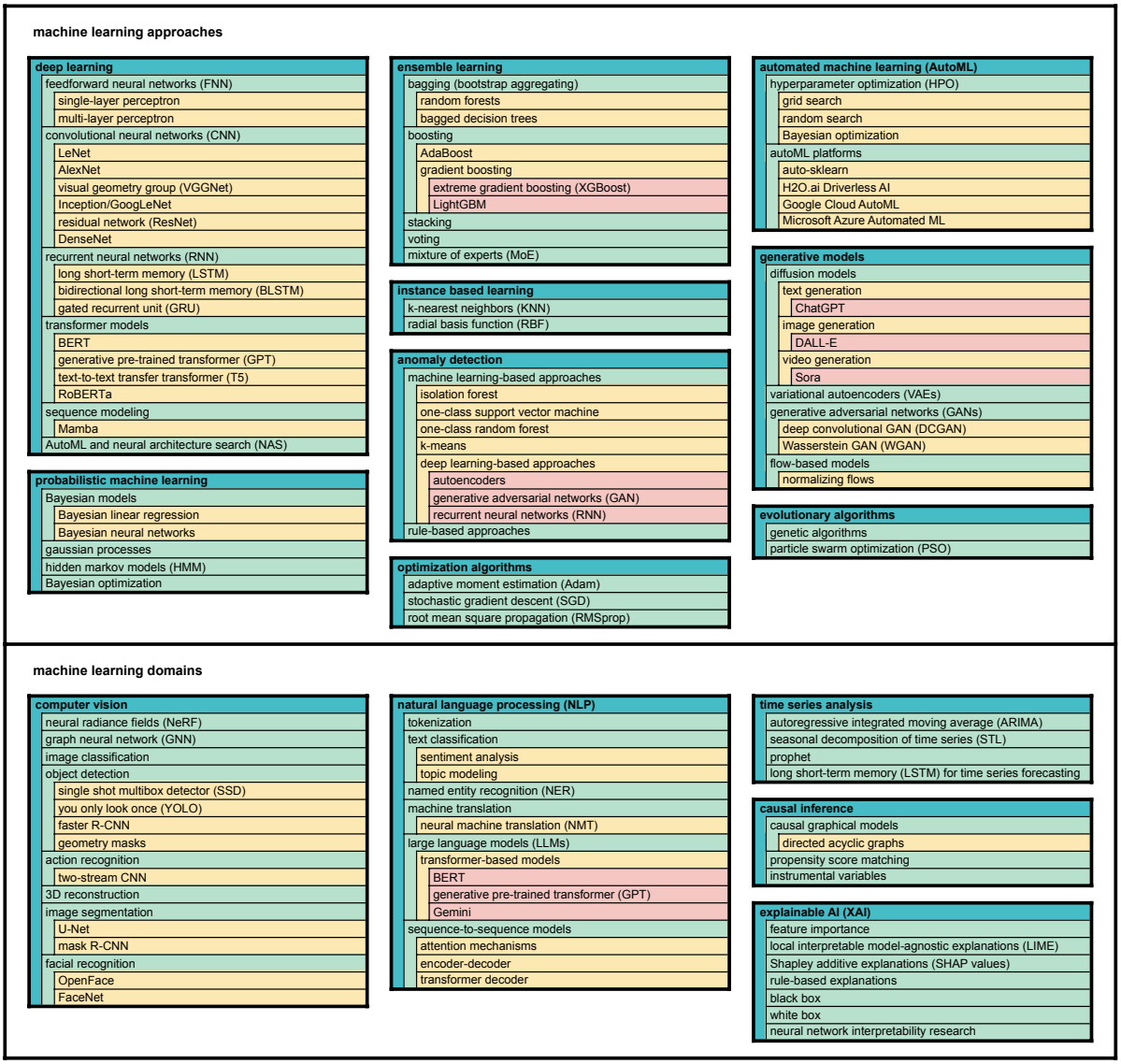


Figure 4.2: Shows the second half of the taxonomy for machine learning models. For access to the complete list of models and related mapping, please refer to the Appendix A.

various papers in the SLR. The four colors represent layers of depth within the taxonomy, with models categorized into designated groups. The taxonomy is divided into four main groups: "machine learning paradigms, specialized machine learning paradigms, machine learning approaches, and machine learning domains". Models in machine learning can often be categorized into multiple groups [151], and this taxonomy highlights their versatility and diverse functionalities.

Following the completion of the data extraction and synthesis phase of the SLR, a selection of 43 commonly used models, from a total of 548, were identified, each mentioned in at least ten publications. We abbreviate these models for the remainder of this study. Their full model names and used abbreviations can be viewed in the Appendix B.

Figure 4.3 shows how our identified models can be categorized in the taxonomy. For example, gradient boosting (GB) [152], a widely recognized model within the field with over 26 citations in our review, exhibits characteristics that align with various categories. Gradient boosting falls under supervised learning, where it iteratively combines multiple weak learners to create a strong predictive model [153]. Gradient boosting can focus on misclassification instances for classification and predict high errors for regression [154]. Gradient boosting also belongs to ensemble learning, as it combines the predictions of multiple weak learner models to produce more robust and accurate final predictions [155].

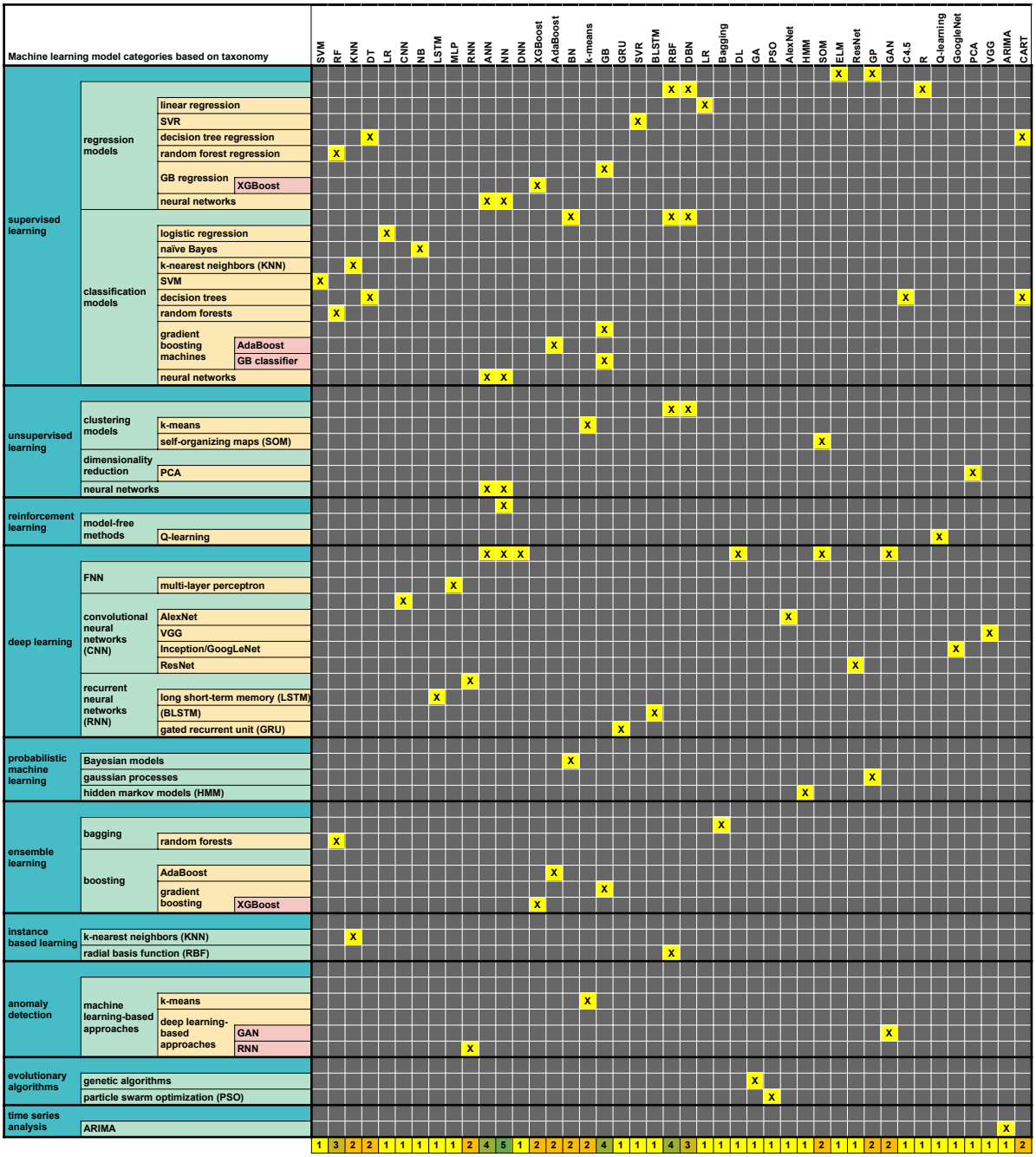


Figure 4.3: Illustrates a mapping of the taxonomy's categories to models in machine learning. The table presents the comprehensive mapping of several category layers to our 43 corresponding models. For access to the complete list of models and related mapping, please refer to the Appendix A.

4.2 Features

Our study investigated the features supported and incorporated by machine learning models, emphasizing their significance in the field. We identified a total of 273 distinct features. We recognized 72 well-defined and established features within the field, highlighting their relevance and impact in machine learning research. For a comprehensive understanding of these features, please consult the Appendix A, where detailed definitions and explanations are provided.

To effectively organize and comprehend these features, we categorized them into 19 categories based on their context, domain, and applications. Machine learning models possess the versatility to support a wide range of features, each tailored to specific use cases and applications. Some common features

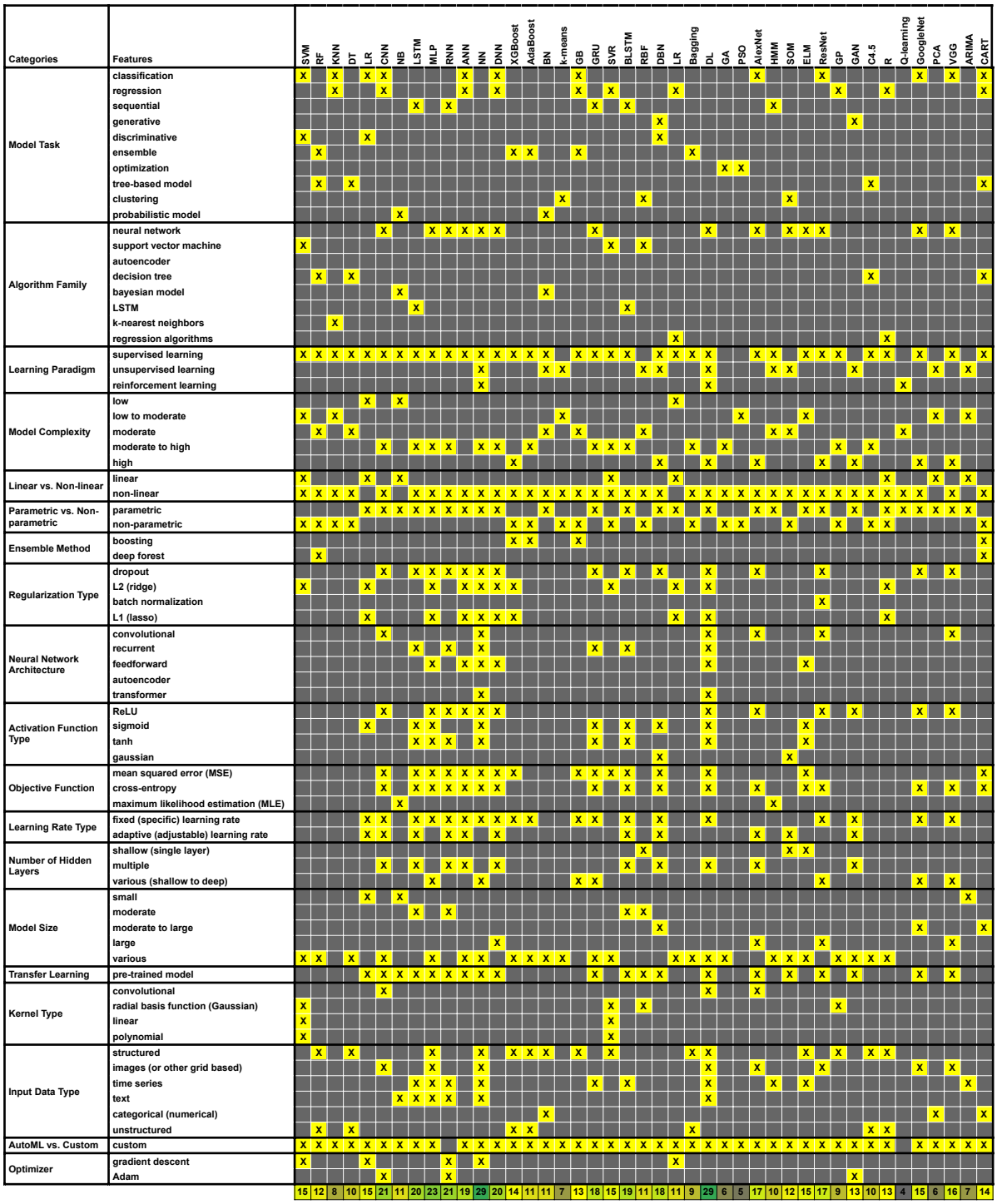


Figure 4.4: Illustrates a mapping of features to models in machine learning. The table presents the comprehensive mapping of 72 distinct features to the corresponding models, grouped into 19 categories. For detailed definitions and explanations of the features, please refer to the Appendix A.

include classification [70], neural network [93], supervised learning [11], non-linear [156], parametric [157], dropout [158], convolutional [159], ReLU [160], MSE [161], fixed learning rate [162], pre-trained model [163], and structured data [164].

Figure 4.4 illustrates the mapping of features to models in machine learning based on study descriptions. Each cell represents whether the corresponding feature commonly refers to the associated model. This list of features was based on our initial findings in the SLR through the 524 papers we reviewed and was later refined to a more comprehensive and complete list. The mapping of the features is based on

the documentation provided by the SLR. For instance, the neural network (NN) [93] is a broad model encompassing many features like supervised, unsupervised, and reinforcement learning. A more nuanced model like naïve Bayes [58] slots into fewer features overall.

This mapping process involved determining which models are most suited to addressing specific features in a given problem. It necessitates a comprehensive analysis of the problem's characteristics and an understanding of the available models' capabilities, strengths, and weaknesses. For instance, in the domain of clustering [16], clustering models such as k-means [165] and self-organising maps [60] have proven to be effective in handling features related to image clustering of data. Conversely, for time series data problems [92], models like ARIMA [166] or long short-term memory [148] have proven to be more suitable choices.

4.3 Model combinations

After conducting our SLR, it was evident that some model combinations were integrated to address the considerations of research modelers, including feature requirements and evaluation measures. The selected publications proposed viable combinations of models based on the authors' research and assessed the outcomes resulting from these combinations.

A combination matrix resembling a symmetric adjacency matrix was constructed to thoroughly examine the various model combinations, treating the models as nodes and the combinations as edges in a graph representation [167]. The upper or lower triangular matrix was utilized to depict unique combinations. Figure 4.5 visually presents this combination matrix, encompassing the 43 selected models. The diagonal cells of the matrix indicate the number of publications discussing each model independently. For instance, our analysis identified 149 papers on support vector machines [13, 56] and 125 papers on random forests [57].

Within the matrix, the cells represent the number of papers discussing the combinations of the corresponding columns and rows. For example, there were 88 papers discussing the combination of support vector machines and random forests. In comparison, 79 papers addressed the combination of support vector machines and k-nearest neighbors (KNN) [168].

The matrix's color coding indicates the number of research articles associated with each combination. Bright yellow cells signify a higher volume of research conducted in the literature, while darker cells denote lower volumes. Additionally, gray cells indicate areas without evidence regarding valid combinations based on the authors' perspectives. However, it is crucial to note that these gray cells represent potential areas warranting further investigation, offering research modelers opportunities to explore the feasibility of such combinations.

Overall, the combination matrix serves as an extensive overview of the model combinations in machine learning model selection research, shedding light on their frequency in the literature. It can be considered a valuable resource for research modelers seeking to identify existing combinations and areas requiring further exploration.

4.4 Model trends

In recent studies, machine learning models have witnessed significant advancements across various domains, leading to notable trends in their development and application [77]. It is worth investigating these trends in recent years and beyond. We refer to a wide range of machine learning models research modelers can employ in machine learning tasks.

To gain insights into the usage patterns of these models, we organized the 43 selected models based on the publication years of the studies that referenced them. While these publications range from 2013 to 2023, we only found relevant data from 2017 and beyond. As such, the model trends displayed only cover 2017 to 2023. Figure 4.6 provides an overview of these trends.

Among the selected models, support vector machines [13, 56], random forests [57], k-nearest neighbors [168], decision trees [169]. logistic regression (LR) [170] emerged as the top five most frequently mentioned models, appearing in over 500 papers. It is important to note that while some models recently gained substantial attention, such as residual network (ResNet) [163] and VGG16/19 (VGG) [171], our study encompasses models from various periods. For our displayed overview, there was an outlier for a paper

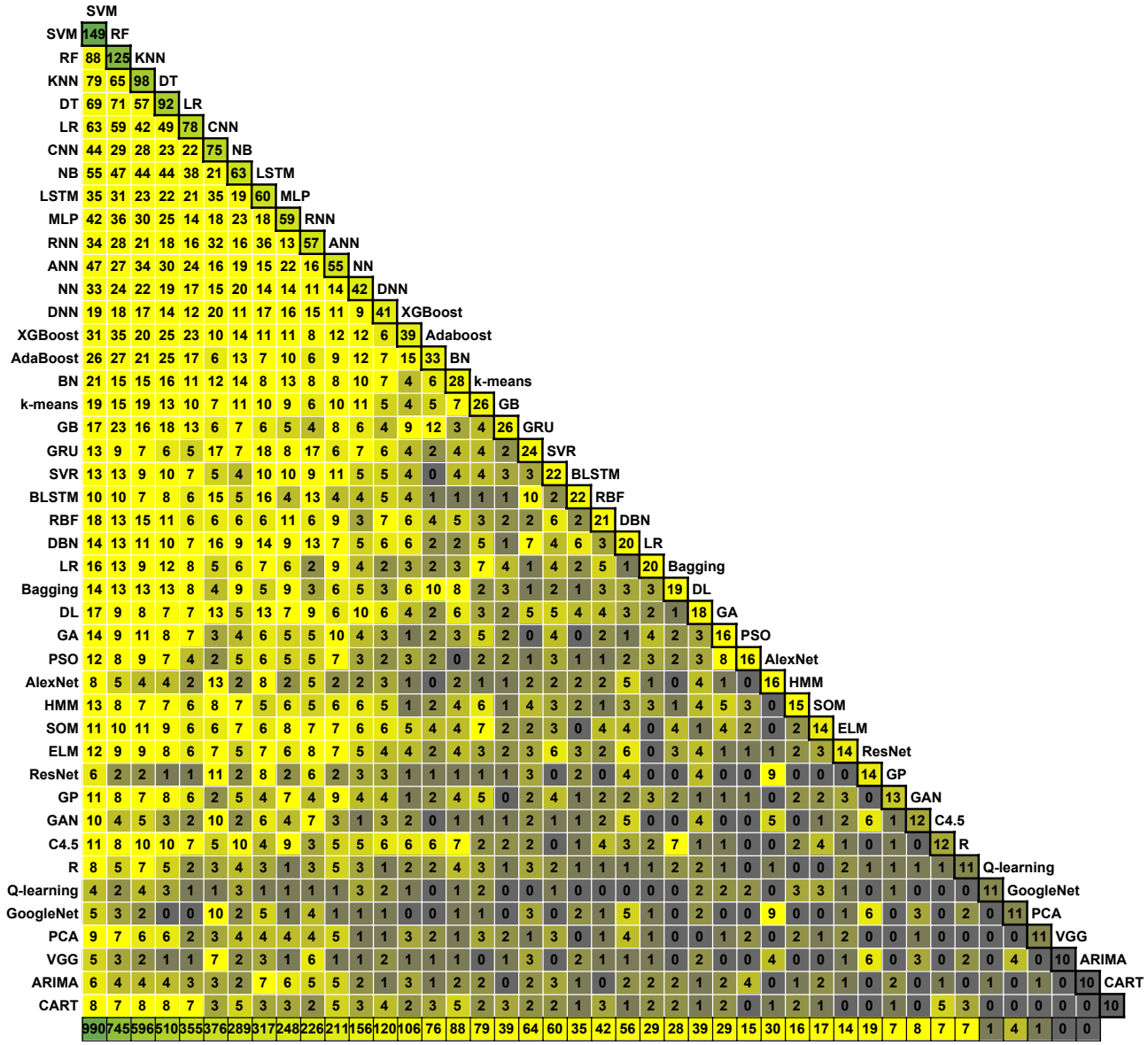


Figure 4.5: Shows a matrix representation of model combinations in machine learning research. The matrix illustrates the combinations of 43 selected models, where each cell indicates the number of publications discussing the corresponding model combination. The diagonal cells represent the number of publications discussing each model individually. Bright yellow cells indicate a higher research volume, while darker cells indicate lower volumes. Gray cells represent areas where no evidence was found for valid combinations. The last row of the matrix represents the frequency of publications in which the models on the diagonal cells were considered in combination with others. For instance, we identified 990 mentions for publications that included support vector machines as one of their design decisions in combination with other models. The combination matrix provides insights into the frequency and popularity of model combinations, aiding research modelers in identifying existing combinations and areas for further investigation. For access to the complete list of model combinations and related mapping, please refer to the Appendix A.

in 2014 that mentioned the use of recurrent neural networks (RNN) [172]. These trends shed light on the popularity and usage patterns of different models in machine learning. By identifying frequently mentioned models and observing shifts in their prevalence over time, research modelers can stay informed about the evolving landscape of machine learning and make informed decisions when selecting models for their specific applications.

4.5 Evaluation measures

In machine learning-based projects, comprehensive evaluation measures are crucial. Evaluation measures quantitatively gauge the quality of model outputs [173]. These measures play a critical role in ensuring accurate and reliable results for a study.

	SVM	RF	KNN	DT	LR	CNN	NB	LSTM	MLP	RNN	ANN	NN	DNN	XGBoost	AdaBoost	BN	k-means	GB	GRU	SVR	BLSTM	RBF	DBN	LR	Bagging	DL	GA	PSO	AlexNet	HMM	SOM	ELM	ResNet	GP	GAN	C4.5	R	Q-learning	GoogleNet	PCA	VGG	ARIMA	CART
2023	12	9	7	7	4	8	6	9	6	7	4	5	3	7	6	3	1	4	4	2	3	0	3	0	4	4	0	1	3	0	2	3	6	1	4	2	2	0	2	1	5	2	0
2022	40	38	26	25	20	20	15	16	17	11	17	12	13	15	10	6	8	6	7	4	6	6	5	5	4	5	4	4	3	7	4	5	3	2	2	2	4	2	2	1	0	2	2
2021	47	46	31	34	34	23	20	17	17	16	15	9	12	14	10	8	9	13	5	8	9	6	4	9	3	3	4	4	7	4	1	2	3	6	3	4	2	3	3	4	3	2	6
2020	28	19	21	14	12	15	12	13	13	13	12	10	9	2	5	6	4	2	3	5	4	6	4	3	5	6	4	4	2	3	3	2	1	1	3	2	2	4	2	3	2	3	1
2019	12	5	6	8	6	7	4	2	2	5	5	4	3	1	2	2	0	0	2	1	0	3	2	2	1	0	2	1	1	0	1	2	1	2	0	1	1	0	1	1	0	0	1
2018	7	5	4	2	2	2	5	3	4	3	2	2	1	0	0	1	2	1	2	2	0	0	2	0	1	0	2	2	0	1	2	0	0	1	0	1	0	0	1	1	0	0	0
2017	3	3	3	2	0	0	1	0	0	1	0	0	0	0	0	2	2	0	1	0	0	0	0	1	1	0	0	0	0	0	1	0	0	0	0	0	0	2	0	0	0	1	0
	149	125	98	92	78	75	63	60	59	57	55	42	41	39	33	28	26	26	24	22	22	21	20	20	19	18	16	16	16	15	14	14	14	13	12	12	11	11	11	11	10	10	10

Figure 4.6: Illustrates the trend of models mentioned in machine learning research over publication years, highlighting the popularity and emergence of various models. For access to the complete list of model trends mapping, please refer to the Appendix A.

While accuracy is a commonly employed evaluation measure, it may not adequately represent the model's performance, especially in imbalanced classes [174]. Alternative measures such as precision, recall, and F1-score are used to evaluate model performance, mainly when dealing with imbalanced data [175]. Additionally, evaluation measures like the area under the curve (AUC) are frequently used to assess binary classifiers [176]. These measures provide insights into the model's ability to differentiate between positive and negative instances, mainly when the costs of false positives and false negatives differ [177].

For ranking problems, evaluation measures such as mean absolute percentage error (MAPE) can be employed [178]. These measures evaluate the quality of the ranked lists generated by the model and estimate its effectiveness in predicting relevant instances. When evaluating regression models, measures such as root mean squared error (RMSE) are used to quantify the discrepancy between predicted values and actual values of the target variable [179].

The selection of appropriate evaluation measures is crucial to ensure the accuracy and reliability of machine learning models. Selecting an adequate evaluation measure depends on the specific problem domain, data type, and project objectives. These factors are pivotal in selecting the most appropriate evaluation measures. Figure 4.7 presents the evaluation measures identified in at least ten publications.

Precision, recall, F1-Score, accuracy, and area under the curve are among the top five evaluation measures identified in the SLR. For detailed explanations of the specified evaluation measures, please refer to the Appendix A.

4.6 Datasets

Datasets are fundamental to machine learning and data science research, as they provide the raw material for training and testing models and enable the development of solutions to complex problems [180]. Datasets exist in various forms and sizes, ranging from small, well-curated collections to large, unstructured datasets with millions of records [181]. The quality of datasets is crucial [180], as high-quality data ensures the accuracy and reliability of models, while low-quality data can introduce biases and inaccuracies. Data quality encompasses completeness, accuracy, consistency, and relevance.

The size and complexity of a dataset pose challenges in terms of storage, processing, and analysis [182]. Large datasets require specialized tools and infrastructure to handle the volume and velocity of data [183]. Additionally, complex datasets, such as graphs, images, and text, may require specialized techniques and models for extracting meaningful information and patterns [184].

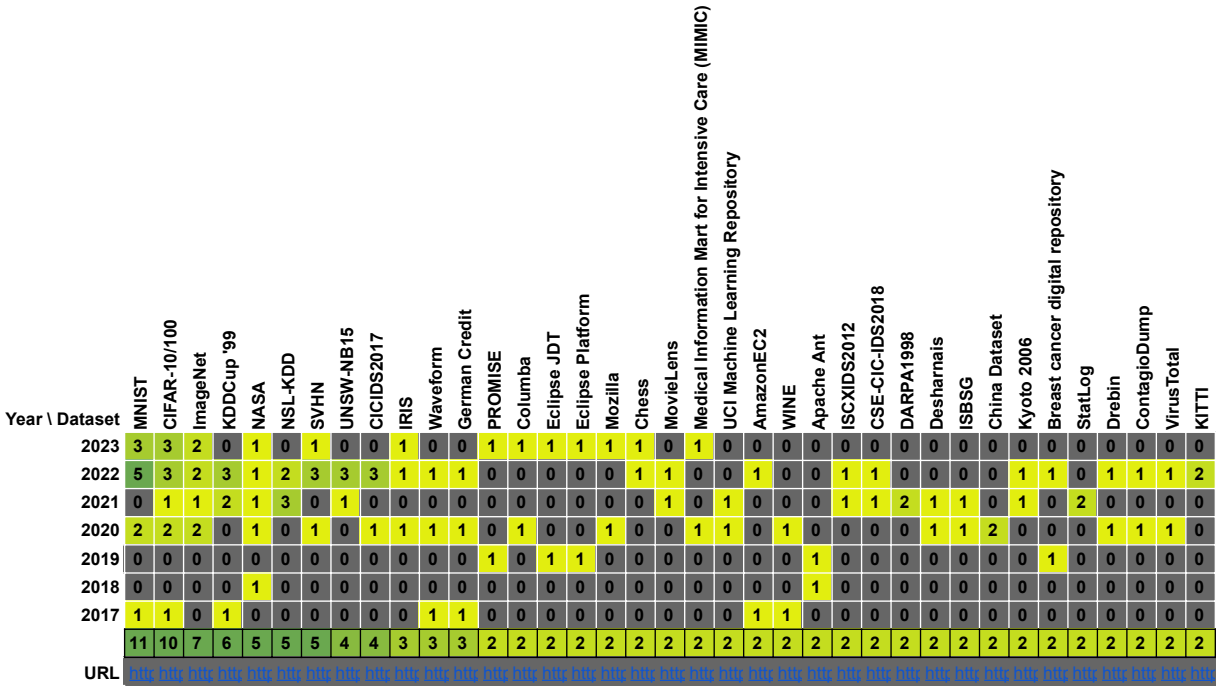
Furthermore, the availability of datasets is a vital consideration in advancing machine learning research and applications [185]. Open datasets that are freely accessible and well-documented foster collaboration and innovation, while proprietary datasets may restrict access and impede progress [186]. Data sharing and ethical considerations in data use are increasingly recognized, leading to efforts to promote open access and responsible data practices [187].

In this study, we identified 37 datasets that research modelers have utilized in the context of machine learning that were mentioned in at least two publications. Figure 4.8 provides an overview of these datasets and their frequency of usage from 2013 to 2023. Notably, MNIST, CIFAR-10/100, ImageNet,

Evaluation measures	
122	Recall
120	Accuracy
101	F1-Score
95	Precision
67	Area Under the ROC Curve (AUC)
40	Specificity
39	Mean Absolute Error (MAE)
38	Root Mean Squared Error (RMSE)
27	False Positive Rate (FPR)
24	Mean Squared Error (MSE)
21	R-Squared
15	Confusion Matrix
14	Error Rate
13	K-Fold Cross-Validation
13	Matthew Correlation Coëfficient (MCC)
12	Mean Absolute Percentage Error (MAPE)
10	False Negative Rate (FNR)
10	10-Fold Cross-Validation

Figure 4.7: Shows an overview of evaluation measures used in machine learning, including performance metrics such as accuracy, precision, recall, F1-score, area under the curve, and other evaluation techniques. For access to the complete list of identified evaluation measures, please refer to the Appendix A.

KDDCup '99, and NASA emerged as the top five datasets commonly used in evaluating machine learning models. These datasets have been utilized individually in 5 or more publications, highlighting their significance and wide adoption.



Year \ Dataset	MNIST	CIFAR-10/100	ImageNet	KDDCup '99	NASA	NSL-KDD	SVHN	UNSW-NB15	CICIDS2017	IRIS	Waveform	German Credit	PROMISE	Columba	Eclipse JDT	Eclipse Platform	Mozilla	Chess	MovieLens	Medical Information Mart for Intensive Care (MIMIC)	UCI Machine Learning Repository	AmazonEC2	WINE	Apache Ant	ISCXIDS2012	CSE-CIC-IDS2018	DARPA1998	Desharnais	ISBSG	China Dataset	Kyoto 2006	Breast cancer digital repository	StatLog	Drebin	ContagioDump	VirusTotal	KITTI
2023	3	3	2	0	1	0	1	0	0	1	0	0	1	1	1	1	1	1	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
2022	5	3	2	3	1	2	3	3	3	1	1	1	0	0	0	0	0	1	1	0	0	1	0	0	1	1	0	0	0	0	1	1	0	1	1	1	2
2021	0	1	1	2	1	3	0	1	0	0	0	0	0	0	0	0	0	0	1	0	1	0	0	0	1	1	2	1	1	0	1	0	2	0	0	0	0
2020	2	2	2	0	1	0	1	0	1	1	1	1	0	1	0	0	1	0	0	1	1	0	1	0	0	0	0	1	1	2	0	0	0	1	1	1	0
2019	0	0	0	0	0	0	0	0	0	0	0	0	1	0	1	1	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	1	0	0	0	0	0
2018	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0
2017	1	1	0	1	0	0	0	0	0	0	1	1	0	0	0	0	0	0	0	0	0	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	11	10	7	6	5	5	5	4	4	3	3	3	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2
URL	http	http	http	http	http	http	http	http	http	http	http	http	http	http	http	http	http	http	http	http	http	http	http	http	http	http	http	http	http	http	http	http	http	http	http	http	http

Figure 4.8: Shows datasets commonly used for machine learning models. The table includes the names of the datasets and their corresponding URLs. For access to the complete list of identified datasets, please refer to the Appendix A.

Expert interviews

The expert interview phase further strengthened the data collected from the SLR. We interviewed with the intent to add to or alter our established data pool while also perceiving other types of feedback regarding our data and the decision model itself.

5.1 Experts

A total of 11 experts were interviewed for this phase. The interviews took roughly 60 minutes each, with most of the time spent on our ten core expert interview questions. We will cover each question and discuss the data we extracted from the expert interviews related to them. Participants were asked to consent before participation. Our consent form can be read in the Appendix C.

Q1: What is your practical or scientific research background?

All of our experts have worked for several years with machine learning models. Table 4 highlights some background information on our participating experts, displaying for each participant their academic degree, academic domain, and current position as a machine learning expert. For example, the table shows that eight experts are Master's graduates, and three experts are PhD graduates. Experts shared that they noticed certain trends within machine learning usage, such as how decision-making has expanded far beyond handcrafted features and heuristics and how many see promise for further growth of machine learning use within their respective fields.

Table 4: Overview of each participant's academic degree, academic domain, and current position as a machine learning expert.

Participant	Degree	Academic domain	Current position
1	PhD	Radiophysics	Machine learning scientist lead
2	Master	Information studies: data science	Machine learning engineer
3	Master	Computer science: data science in engineering	Machine learning engineer
4	Master	Artificial intelligence	Machine learning/MLOps engineer
5	Master	Computing science	Senior machine learning engineer
6	PhD	Computer science	Machine learning/data scientist
7	Master	Electrical engineering and computer science	Machine learning/MLOps engineer
8	Master	Computer science	Research engineer
9	Master	Bioinformatics and systems biology	Data scientist
10	PhD	Computer vision and machine learning	Senior artificial intelligence architect
11	Master	Computer/information technology administration and management	Senior machine learning engineer

Q2: How do you typically select a machine learning model or a combination of them?

The most commonly replied practice was to select machine learning models depending on a task or use case. Experts noted how industrial and academic use of machine learning contrasted with certain practical concerns. Industrial use is often occupied with a model's computational resource intensity, stating that model speed is generally preferred over model accuracy. Company restrictions were also mentioned, as certain machine learning models are preferred based on whatever cloud provider a company is subscribed to. One participant mentioned: *"Five years ago, it was common to train your own model and use an existing architecture. Nowadays, you have powerful existing models, and you don't even need to fine-tune them"*, further highlighting how the industrial pipeline has become more streamlined. Academic use generally favors the highest possible accuracy metric, thus being more open to using slower algorithms. Furthermore, academic users are interested in the established nature of machine learning models in their academic fields.

Q3: What factors or features come into play during the decision-making process?

The most commonly mentioned feature that comes into play is the perceived performance of a machine learning model. One participant mentioned how language models are often unreliable in accurately understanding features: *"A model will say that a 100kg of feathers weighs less than 100kg of steel. Such mistakes are common in many language models"*. Another common feature is the perceived model computation speed, with one participant stating: *"In robotics, memory constraints are prevalent. Since*

they have limited RAM and circuitry, quick light-weight models are preferred". Other notable features are the amount of available data, the model scalability, the quality of the available data, the ability of the model to iterate, how well-suited the model is for hyperparameter tuning and the practical complexity of the model. Model explainability was deemed an important feature, with one participant stating: *"The simpler to understand the model is preferred, even if the performance is worse because they are easier to maintain and transfer to a client"*. Furthermore, experts mentioned how the number of model features affects selection. Some experts exhaustively attempt feature combinations, increasing their effort for each additional feature.

Q4: What are some machine learning models that you are familiar with?

Deep learning models like convolutional neural networks, recurrent neural networks, and NLP-related models like GPT [108], were common mentions. The overall knowledge of machine learning differed per participant. Though some participants were familiar with machine learning models in various domains, most were only comfortable discussing models within their subdomain, such as models in computer vision, NLP, or time series analysis.

Q5: What machine learning models from our commonly used model list are you familiar with?

The individual results of the participants can be observed in the Appendix D, with 'Y' indicating familiarity and 'N' indicating unfamiliarity.

At least two participants recognized every machine learning model in our list, which we deemed an acceptable number for a model to be considered well-known. The only exception was the extreme learning machine (ELM) [188], with only one participant familiarizing it. Based on this, the relevancy of the extreme learning machine was questioned.

Q6: What features in our machine learning feature list are you familiar with?

The individual results of the participants can be observed in the Appendix D, with 'Y' indicating familiarity and 'N' indicating unfamiliarity.

Some features were recognized by less than two participants, which led us to reconsider these features for our decision model. Some features were removed from the decision model, while others were deemed integral to machine learning. Figure 5.1 shows our reasoning per reconsidered machine learning model or feature. The figure shows that we decided to keep the extreme learning machine as a model due to its relevance to computational efficiency. It further shows we decided to include SARSA [189], inductive logic programming [190], RAS-CO [191], contractive regularization [192], competitive layer [193], matern [194], and auto-associative [195] as features for our decision model. We also changed the category name for "model type" to "model task".

Category	Model/Feature	Included	Reasoning
Model	ELM	Y	Relevant in computational efficiency
Model Type	inductive logic programming	N	An algorithm family
Algorithm family	SARSA	Y	An On-Policy Learning paradigm
Algorithm family	inductive logic programming	Y	Relevant in interpretability in learned models
Algorithm family	analytic hierarchy process	N	A decision-making methodology
Algorithm family	post-decision state learning	N	Not considered a well-established term
Algorithm family	MALMOS	N	An operationalizing methodology
Algorithm family	compound covariate predictor	N	An approach in predictive modeling
Algorithm family	rough set	N	A mathematical framework
Ensemble methods	RAS-CO	Y	A novel ensemble method
Regularization type	contractive regularization	Y	Relevant in ensuring robust model predictions
Neural network architecture	competitive layer	Y	Relevant in SOM models
Neural network architecture	fishblock	N	Too unknown overall
Neural network architecture	contradiction layer	N	Too unknown overall
Kernel type	matern	Y	Commonly applied in Gaussian processes
Kernel type	auto-associative	Y	Crucial to autoencoder learning

Figure 5.1: Showcases our summarized reasoning on why a model or feature should or should not be included in our final feature list.

Q7: What features in our machine learning list would you modify or alter in definition?

Participants suggested several additions and alterations to our feature list, all of which were considered and weighed against each other. Some of the more general additions include large language models (LLM) [107], multimodal models [196], and machine learning optimizers [197]. Furthermore, participants mentioned how our feature list was already quite exhaustive in its current state, thus not needing much addition.

Q8: What are your thoughts on our machine learning taxonomy?

Participants suggested several additions and alterations to our taxonomy, most of which were machine learning models or paradigm-related. While much of the individual placement of specific models remains unaltered, some changes were made to the naming and ordering of the top-layer categories. One common notion was that deep learning was easily misread as a separate machine learning paradigm adjacent to supervised or unsupervised learning. Thus, deep learning and neural network models were relocated accordingly. Another notion was that some spacing and coloring decisions hurt the overall readability of the taxonomy. Finally, some participants mentioned how the taxonomy might be "too complete" and wondered whether some data should be omitted. All of these notions were taken into account for our final taxonomy design. With these changes, we increased our total number of machine learning models to 571. Our features were also altered, and a category for optimizers was added. The total number of features the decision model now includes is 273, distributed over 19 categories.

Q9: How would you prefer a decision model to support you with machine learning model selection?

Most participants had a clear preference for one of two distinct options. The first option is a large language model or GPT-related approach to a decision model, where a user converses with an AI language model about their machine learning model task to make decisions. One participant in favor of this approach stated: *"Given the state of the art GPT, the ideal decision model would be an AI assistant that talks you through the whole model selection process. Humans can get tired and are prone to forget things, so an embedded domain expert could assist you in selecting and reminding you of what steps to take"*. The second option concerns a method akin to our created decision model, which allows users to select specific feature requirements for their machine learning model task, resulting in a transparent and structured approach to decision-making. One participant states: *"I am not a fan of a language model approach because I cannot know what is going on inside the black box. From an academic point of view, where you have to provide reasoning to publish papers, a decision model should be as white box as possible"*. Our decision model implementation is discussed in further detail in Chapter 6. Various specific needs were suggested; these include having hyperparameters outlined per model, providing a list of data suited per model, selecting models based on evaluation metrics, featuring models in performance rankings, providing open source data on the decision model, considering the hardware requirements for a model, providing a list of common model combinations, presenting the specific influence of each neural network layer in a model, automatically presenting fine-tuning comparisons between models, automatically performing exhaustive feature selection (to remove bias set by the initial model), and the inclusion of several plotting and visualization metrics.

The overall reception for the use of a machine learning model selection decision model was positive across all participants, with many stating they look forward to the results of our study and further developments within the field. One participant suggested our decision model could find good use in practice: *"In the industry, as models get more complicated, oftentimes the process of model selection becomes a case of just throwing a bunch of models at a wall to see what sticks. Rather than setting up an initial hypothesis, practitioners just try several models to see which one produces the best results. A decision model could speed up this process considerably"*. Another participant sees academic potential, stating: *"One problem that I often face as a researcher is understanding and learning from previous data. If I have a neural network that performs well on a particular dataset, I want to know what layer combinations positively impact this result. Compared to AutoML solutions that provide a model without further context, the decision model could offer insight into what particular features are at play"*.

Q10: Do you have any final remarks about the study or the interview?

Criticism was given regarding the repetition in the models covered during the interview. Furthermore, some participants argued whether the decision model would be better off excluding some of the machine learning models they consider outdated.

5.2 Interview evaluation

We conducted a substantial number of expert interviews, which significantly improved our decision model. Based on the expert interview feedback, we subjectively altered some aspects of our data, and we are confident in the design science we used to justify these alterations.

Decision model

This section describes how research modelers make decisions during machine learning model selection and how they can incorporate these decisions into a decision model. It illustrates a systematic approach to machine learning model selection based on academic literature.

6.1 Research challenges

Research modelers face the challenge of selecting the most suitable combination of models to apply to their predictive modeling task [198]. In this section, we present a meta-model for the decision-making process in the context of machine learning model selection. Adopting this meta-model is based on the principles outlined in the ISO/IEC/IEEE standard 42010 [199], which provides a framework for conceptual modeling of architecture description. This process requires a systematic approach to ensure that the chosen models effectively capture the research modeler's goals. Figure 6.1 displays this process in detail.

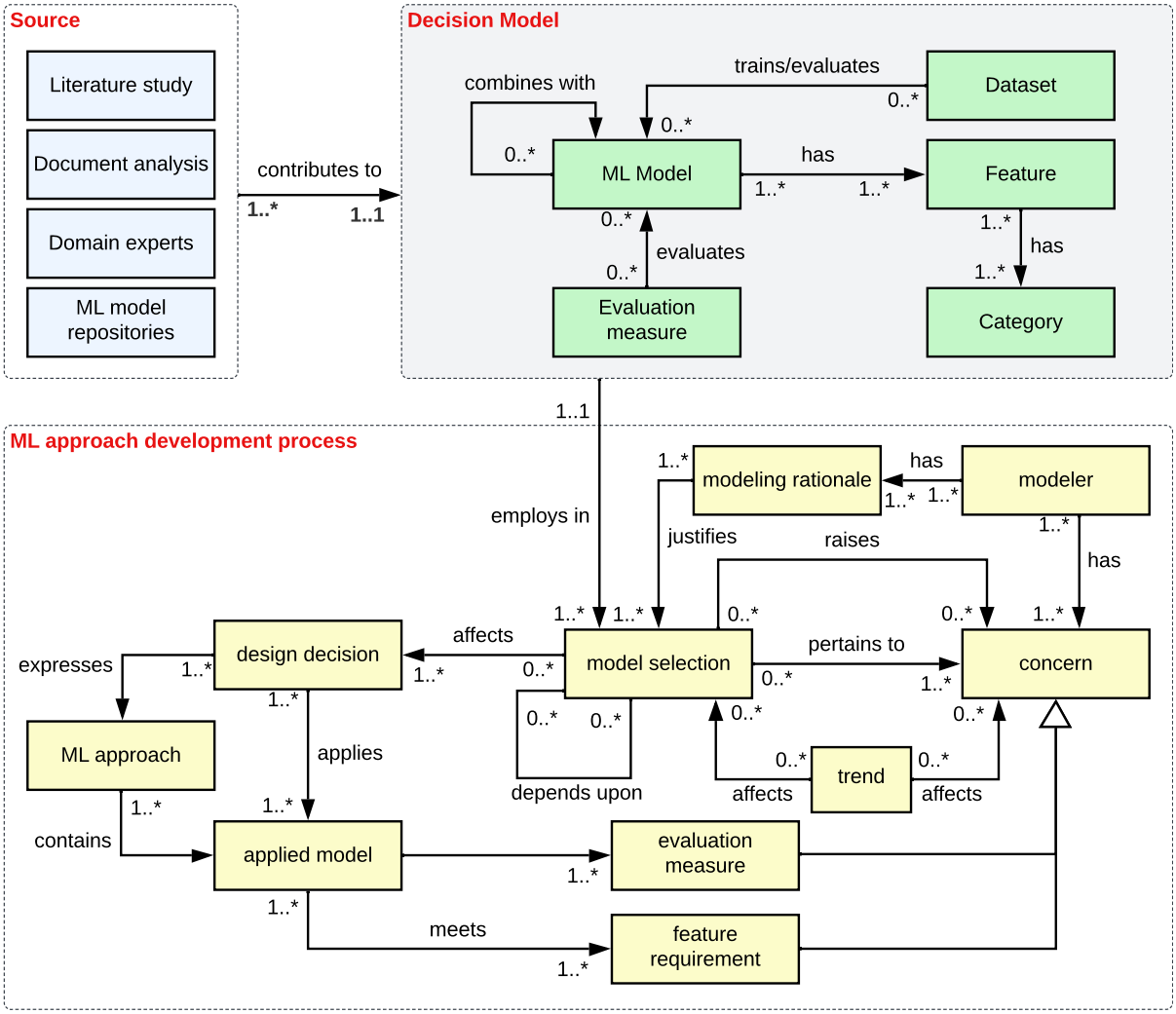


Figure 6.1: Illustrates the role of the decision model in the scope of a meta-model.

Consider a scenario where research modelers encounter this challenge and describe the machine learning model selection decision-making process as follows:

Goal and concerns: The research modelers aim to build a machine learning model selection approach

through a decision model system. They desire to accurately determine project goals' underlying purposes or objectives, enabling personalized and precise responses. The research modelers have concerns regarding functional requirements, and they aim to achieve an acceptable level of quality based on their evaluation measures.

Identification of models and features: To address this problem, the research modelers consider various models that can capture project goals in the scientific context. Based on their concerns, they identify essential features, such as the project task or project data type. They explore the available model paradigms and approaches, such as supervised, unsupervised, reinforcement, deep, and ensemble learning. The research modelers also consider the recent trends in employing models for machine learning model selection.

Evaluation of models: The research modelers review the descriptions and capabilities of several models that align with machine learning. They analyze each model's strengths, limitations, and applicability to the decision-modeling problem. They consider the model's ability to handle data and output efficiency. This evaluation allows them to shortlist a set of candidate models that have the potential to address the machine learning model selection challenges effectively.

In-depth analysis: The research modelers conduct a more detailed analysis of the shortlisted models. They examine the associated techniques for each model to ensure their suitability in the model selection system. They assess training data requirements, model complexity, interpretability, and scalability. Additionally, they explore the possibility of combining models to identify compatible combinations or evaluate the existing literature on such combinations. If necessary, further study may be conducted to assess the feasibility of model combinations. This step helps them identify the optimal variety of models that best capture research project goals and address their concerns.

6.2 Decision-making process structure

The decision-making process for machine learning model selection can be formally described using a structured approach, as outlined in Algorithm 1:

Algorithm 1 Machine learning model selection process.

Input: Project goals, functional requirements
Output: Optimal selection of (combined) machine learning models

procedure ModelSelection
    Identify project goals and objectives
    Identify functional requirements
    Identify potential models and their features
    for each model m in Models **do**
        Evaluate m based on predefined criteria (strengths, limitations, applicability)
    end for
    Shortlist candidate models based on evaluation
    for each shortlisted model m' **do**
        Conduct in-depth analysis (training data requirements, complexity, interpretability, scalability)
        Explore potential combinations of m' with other models
    end for
    Assess feasibility of model combinations
    Evaluate performance of feasible combinations
    Return Optimal selection of (combined) machine learning models
end procedure

This structured algorithm provides a formal framework for the decision-making process, ensuring clarity and consistency in selecting machine learning models.

6.3 A decision model for machine learning model selection

The decision model's sources are directly responsible for the contributions to the decision model. The literature study, document analysis, domain experts, and machine learning model repositories shape the

contents of the decision model on a one-to-many basis. The decision model contains a list of several machine learning models. Each machine learning model has a list of features, and each feature has an assigned category. Evaluation measures impact the machine learning models, and datasets both train and evaluate machine learning models. Machine learning models can also be combined with other machine learning models to create machine learning model combinations.

Decision theories have wide-ranging applications in various fields, including e-learning [200] and software production [201, 202]. In the literature, decision-making is commonly defined as a process involving problem identification, data collection, defining alternatives, and selecting feasible solutions with ranked preferences [203]. However, decision-makers approach decision problems differently, as they have their priorities, tacit knowledge, and decision-making policies [204]. These differences in judgment necessitate addressing them in decision models, which is a primary focus in multiple-criteria decision-making (MCDM).

MCDM problems involve evaluating a set of alternatives and considering decision criteria [**?**]. The challenge lies in selecting the most suitable alternatives based on decision-makers' preferences and requirements [205]. It is important to note that MCDM problems do not have a single optimal solution, and decision-makers' preferences play a vital role in differentiating between solutions [205]. In this study, we approach the problem of model selection as an MCDM problem within the context of a decision model for machine learning model selection.

Let $Models = m1, m2, \ldots, m\|Models\|$ be a set of models found in the literature (decision space), such as *SVM*, *DT*, and *BERT*. Let $Features = f1, f2, \ldots, f\|Features\|$ be a set of features associated with the models, such as classification, supervised learning, and image data. Each model $m \in Models$ supports a subset of the set *Features*. The objective is to identify the most suitable models, or a combination of models, represented by the set $Solutions \subset Models$, that address the features of research modelers denoted as *Features*, where $Features \subseteq \{Must\ have\ features \cup Should\ have\ features \cup Could\ have\ Features \cup Will\ not\ have\ features\}$, based on the MoSCoW principle [206]. Accordingly, research modelers can adopt a systematic strategy that allows for selecting combinations of models by employing an MCDM approach. This approach involves taking *Models* and their associated *Features* as input and applying a weighting method to prioritize the *Features* based on the preferences of decision-makers. An aggregation method ranks the *Models* and proposes fitting Solutions. Consequently, the MCDM approach can be formally expressed as follows:

$$MCDM : Models \times Features \rightarrow Solutions$$

The MCDM decision model framework is valuable for research modelers working on recommender systems [207]. This approach could help research modelers explore options systematically and choose the best combination of models to select an effective machine learning model approach. The decision model suggests five steps for selecting a variety of models:

(1) Models: In this phase, research modelers should gain insights into best practices and well-known models employed by other research modelers in machine learning model selection. The Appendix A can be used to understand the definitions of models. The Appendix A can also help in becoming familiar with the categories used to classify these models. Figure 4.1 and Figure 4.2 illustrate the categorization of models in this study, and Figure 4.6 presents the trends observed considering the machine learning model used in research.

(2) Feature requirements elicitation: In this step, research modelers need to fully understand the core aspects of the machine learning model selection problem they are studying. They should carefully analyze their specific scenario to identify the key characteristics required in the models they seek, which may involve using a combination of models. For instance, research modelers might consider regression [12] and structured data [164] as essential feature requirements for model selection problems. Research modelers can refer to A to better understand feature definitions and model characteristics. This will help them select the most suitable features for their machine learning model selection problem.

(3) Finding feasible solutions: In this step, research modelers should identify models that can feasibly fulfill all of their feature requirements. 4.4 can be used to determine which models support specific features. For example, the table shows that publications explicitly mentioned support vector machines as a suitable model for classification [70], non-parametric [208], and L2 regularization [209] use. Based on these findings, if a machine learning project requires these three feature requirements,

support vector machines could be selected as one of the potential solutions. If the number of feature requirements increases, the selection problem can be converted into a set covering problem to identify the smallest sub-collection of models that collectively satisfy all feature requirements.

(4) Selecting feasible combinations: In this phase, research modelers need to assess whether the identified models can be integrated or combined. 4.5 provides information on the feasibility of combining models based on the reviewed articles in this study. If the table does not indicate a potential combination, it does not necessarily imply that the combination is impossible. This means no evidence supports its feasibility, and research modelers should investigate the combination independently.

5) Performance analysis: After identifying a set of feasible combinations, research modelers should address their remaining concerns regarding evaluation measures. 4.7 and the Appendix A can be used to understand the typical concerns other research modelers in the field employ regarding this domain. Additionally, 4.8 provides insights into frequently used datasets across the domain and applications. Research modelers can then build their solutions and pipelines using off-the-shelf models from various libraries, such as TensorFlow [40] and Scikit-learn [210]. These solutions can be evaluated using desired datasets to assess whether they meet all the specified concerns. This phase of the decision model differs from the previous four phases, as it requires significant ad-hoc efforts in developing, training, and evaluating the models.

By employing this decision-making process, research modelers can develop a machine learning model approach that accurately captures and understands their user requirements and project goals. This enables personalized and precise responses, enhancing the overall user experience and utility.

Case Studies

In this section, we present an evaluation of our proposed decision model through several scientific case studies [143]. Our case study method is not to be confused with an industrial case study method, which is a detailed examination of a specific instance within an industrial setting [211]. The papers we selected represent the outcome of our study as we performed our decision-making process in the context of those scientific projects. Eight domain-relevant papers are covered to understand the applicability of the decision model to the participant's projects. For each paper, we utilized our decision model based on what we believed was parallel to decision-making within the paper. Firstly, based on the paper, we decided what we deem as features, divided into Must have, Should have, Could have, and Won't have features, based on the MoSCoW principle [206]. Secondly, we determined what models are used in the paper and for what purpose. Subsequently, we entered our features into the decision model, generating a list of alternative solutions. These alternative solutions should theoretically serve as machine learning models that could adequately replace the models used in the study with relative success.

After this preparation phase, we presented our findings to the original author of the paper during our case study interview, asking them several questions: *"Do you agree with our selected features?", "Do you agree with our selected models?", "Do you feel our alternative solutions make sense and fit the study domain?", "Are you interested in using our decision model?"*. After repeating this process for all eight case studies, we compiled our gathered data and evaluated how accurate our prepared data was for each question. We compared the results our decision model offered with what the original authors had considered. Finally, we summarized the findings as the overall results of our case studies.

In Figure 7.1, we provide a summarized overview of the case studies conducted in this research. This includes details about each case study's specific domains, the study's publication, the study domain, the models and feature requirements identified by the case study, and the alternative solutions provided by our decision model.

7.1 Case study 1

This study is called: *"Protein Family Classification from Scratch: A CNN Based Deep Learning Approach"*. Next-generation sequencing techniques enable the generation and identification of sequenced proteins and their biological families and functions. Despite this, many proteins remain uncharacterized in bioinformatics. Traditional classification models focus on N-Gram features from sequences, neglecting motif information and the affinity between motifs and adjacent amino acids. Previous clustering-based algorithms used domain knowledge and extensive data samples to define protein features and annotate families. The paper aims to introduce a convolutional neural network-based approach for amino acid representation learning using limited characterized proteins, incorporating amino acid location information [212]. They evaluate the method on reviewed protein sequences from the UniProt database and also validate the model using typically ignored unreviewed protein records.

7.1.1 Feature requirements

For the feature selection, we decided on two features in total. **'Must have: classification model'**, was decided based on the study's preference to classify amino acids through representation learning. The participant agreed with this reasoning. **'Must have: convolutional neural network'**, was decided based on the study's convolutional neural network-based solution. The participant noted that while an amino CNN structure was suitable at the time, it was more of a personal preference than a necessity. They further noted that domain-specific features like amino acid compatibility, detailed gene features, and protein prediction are relevant.

7.1.2 Models

We identified the sole model used as a convolutional neural network model. The participant addressed that they used a one-dimensional convolutional neural network structure (1D-CNN) [213], different from a normal 2D convolution and particularly suited to amino acid data strings.

7.1.3 Results

Our decision model produced a list of alternative solutions, the top 10 being: deep convolutional neural network (DCNN) [7], dense CNN [214], gate CNN [215], temporal convolution networks (TCN) [216], generic CNN (GCNN) [217], tuned dedicated CNN (TDCNN) [218], attention-based time-incremental CNN (ATI-CNN) [219], residual network (ResNet) [163], visual geometry group [171], ResNet50/152V2 [220].

The participant noted that the list of alternative solutions is too broad for the amino acid domain. As examples, they mentioned how Unet [221] is suited to semantic segmentation and AlexNet [7] is suited to image classification. They also noted that they are unfamiliar with some of the models listed.

7.1.4 Analysis

The participant noted how the broadness of the alternative solutions hinders the potential use of our decision model. They also noted how it could be a useful tool if tuned for more specific domains like the amino acid protein sequences in their studies. They also mentioned the need to see how their data modality fits into a specific model.

Case study 1 showcases how our decision model is very knowledgeable in a broad sense but possibly not specialized enough to be useful to certain domain experts. Furthermore, the list of alternative solutions showcases how the decision model underfits on a lack of specific features.

7.2 Case study 2

This study is called: *"Runtime Adaptation in Wireless Sensor Nodes Using Structured Learning"*. Markov decision processes (MDPs) [222] enable dynamic adaptation and self-optimization of cyber-physical systems at runtime. Recently, reinforcement learning techniques, which simplify Markov decision process components to reduce computational requirements, have been popular. This paper argues that advancements in compact MDP models (CMMs) challenge this trend, especially in designing wireless sensor network nodes. A novel compact MDP model-based approach for self-aware wireless sensor nodes is introduced and compared to Q-learning [223], a common reinforcement learning technique [224]. It is shown that reinforcement learning models do not effectively serve a specific class of sensor nodes. Through simulations and a prototype implementation, it is demonstrated that compact MDP models offer significantly better runtime adaptation performance than Q-learning with similar resource requirements.

7.2.1 Feature requirements

For the feature selection, we decided on two features in total. **'Must have: reinforcement learning'**, was agreed on the study's research domain being reinforcement learning. **'Could have: MDP'**, was chosen as a likely feature due to the paper's artifact being a specific type of Markov decision process, namely a compact MDP model. The participant noted that their research approach was not based on feature selection. While Q-learning became popular during their research, they recognized themselves as Markov decision process domain experts. Consequently, they sought out Markov decision process-related problems to tackle. They would define a discrete state space, specifically using system knowledge for the domain of a problem. Subsequently, they progressed to methods where the system would adjust the state space based on runtime observations. The participants noted that they believed this research methodology was prevalent in academic research. They also recognized that this research method was not mentioned in their paper.

7.2.2 Models

We identified the sole model as a compact MDP model. The participants provided additional context to their reasoning behind this model choice. They were dealing with an application where the Markov decision process and reinforcement learning models were beneficial tools for solving problems. However, these solutions required too much computational resources. They investigated low memory and CPU-related solutions, such as implementing compression mechanisms. They created an artifact that could compact representation through compression while maintaining the Markov decision process and reinforcement learning compatibility. This is where they coined the term "compact MDP model".

7.2.3 Results

Our decision model produced a list of alternative solutions, the top 10 being: deep learning (DL) [6], neural network (NN) [93], Q-learning [223], Deep Q-learning [225], Deep Q-network [225], dueling DQL [226], double DQL [227], double-dueling deep Q-learning [228], post-decision state learning [229], utility-based learning (UL) [230].

The participants noted how they are familiar with roughly half of the models listed. They mention that all models appear relevant to their study's domain. They say reading through alternative reinforcement learning solutions like multi-armed bandit (MAB) [231], value iteration [232], and temporal difference learning [233], while searching for possible candidate models for their study. They note, however, how none of the alternative solutions listed would directly address the core issue of their study, which is the resource constraint problem during computation.

7.2.4 Analysis

The participant noted that our decision model sounds useful conceptually. They are interested in seeing how usable the decision model is practically and how accurate its results are in a variety of domains. They also noted how the model could aid in proposing domain-specific models they may not be aware of. Our decision model's extensive survey of data could speed up this learning process.

Case study 2 showcases how our decision model can provide accurate alternative solutions for a given machine learning domain.

7.3 Case study 3

This study is called *"Identifying click baits using various machine learning and deep learning techniques"*. It mentions how most readers today prefer online news for instant updates and personalized recommendations. However, this format also brings issues like clickbait social media posts designed to attract attention rather than inform. This paper aims to develop a system to predict the likelihood of social media posts (tweets) related to news articles being clickbaits [234]. GloVe embeddings are used to represent text data numerically, and various features such as Word Mover's Distances, subjectivity, and polarity of tweets are engineered. Several machine learning models are trained for classification, including logistic regression [170], random forest [57], XGBoost [235], and LightGBM [236]. Additionally, deep learning models like deep neural networks [93] and long short-term memory [148] are implemented to enhance the predictive system.

7.3.1 Feature requirements

For the feature selection, we decided on two features in total. **'Must have: classification model'**, was decided due to the domain being a classification machine learning problem. **'Should have: text data compatibility'** was chosen as a relevant feature because of the clickbait identification problem. The participant noted that he agreed that any model for this domain should be a classification model that is compatible with text data.

7.3.2 Models

We identified logistic regression as the main model used as their final choice. Random forest, XGBoost, LightGBM, deep neural network, and long short-term memory were other models used. The participants agreed with all of these model choices, with logistic regression being the final solution for their research.

7.3.3 Results

Our decision model produced the following list of alternative solutions: bidirectional encoder representations from transformers (BERT) [112], multinomial naïve Bayes (MNB) [237], discriminative multinomial naïve Bayes (DMNB) [238], multilayer feedforward ANN [5], attention-based bi-directional LSTM (ABLSTM) [239], multitask-clinical BERT (MT-Clinical BERT) [240], BEHRT [241], CheXbert [242], HyperNet [243], forwarding neural network [244], deep learning for code clones (DLC) [245].

The participants agreed that BERT, MNB, DMNV, multilayer feedforward ANN, and attention-based bi-directional LSTM are suitable alternative solutions. However, they noted being apprehensive about using domain-specific models like multitask-clinical BERT.

7.3.4 Analysis

The participant noted being interested in using the decision model and trying out its suggested alternative solutions.

Case study 3 showcases a positive response to the use of our decision model and provides an example where the decision model does not underfit to return a list of models that is considered too broad.

7.4 Case study 4

This study is called: *"DeepVulSeeker: A novel vulnerability identification framework via code graph structure and pre-training mechanism"*. Software vulnerabilities can lead to system crashes, privacy leaks, or even physical damage. Identifying these vulnerabilities promptly in extensive codebases is crucial for patching them. Current models, including classic and deep-learning-based approaches, have significant drawbacks and fail to meet industry demands. To address these issues, the paper proposes DeepVulSeeker, an automated vulnerability identification framework utilizing code graph structures and semantic features through graph representation self-attention and pre-training mechanisms [246]. Experiments demonstrate that DeepVulSeeker achieves 0.99 accuracy on traditional CWE datasets and outperforms existing models on two complex datasets. Additionally, case studies show its ability to understand the implications of vulnerability. DeepVulSeeker is fully implemented and open-sourced for future research.

7.4.1 Feature requirements

For the feature selection, we decided on three features in total. **'Must have: classification model'**, was decided on due to the nature of the study being a classification problem. **'Must have: convolutional neural network'**, was chosen because the body of data needed to be convoluted into multiple layers. **'Should have: multiple layers'** was chosen as a relevant feature due to the nature of the data as well. The participant agreed that our features fit, noting that using a a convolutional neural network was necessary because there was no data body. The participant noted they had different perspectives on what features fit the entirety of their study. Taking this overarching approach, they named abstract syntax tree [247], control flow graph [248], data flow graph [249], and embedded models [250] as their four features.

7.4.2 Models

We identified the model used as their artifact, DeepVulSeeker, which involves a graph representation self-attention (GRSA) model, convolutional neural network, and multilayer perceptron. The participant agreed with our selected models and added that self-attention [251] and pre-training [163] are additional model factors that could be considered part of the model artifact. They further mentioned that newer models in this domain are more likely to use large language model-related models.

7.4.3 Results

Our decision model produced a list of alternative solutions, the top 10 being: deep convolutional neural network (DCNN) [7], gate CNN [215], generic CNN (GCNN) [217], tuned dedicated CNN (TDCNN) [218], attention-based time-incremental CNN (ATI-CNN) [219], LeNet [252], AlexNet [7], R-CNN [253], Faster R-CNN [254], Mask R-CNN [255].

The participants questioned using the provided alternative models, as they had tested some of them in their study domain before, with underwhelming results. They further noted how it can be unclear how useful an alternative solution model is before testing it. This process potentially wastes time, and they believed it beneficial if a model's performance could be considered beforehand. They further mentioned that AlexNet and graph neural networks (GNN) [256] could be added to the list of alternative solutions. They especially considered graph neural networks superior to most of the alternative suggestions.

7.4.4 Analysis

The participant was interested in using our decision model in different research domains. They emphasized how selecting a model from a predefined list of potential models could reduce the time needed to find an appropriate one. This illustrates how the decision model can increase productivity by offering a detailed list of machine learning models that fit a specific domain. However, it also showcases how model unfamiliarity may undermine the decision model's use. Furthermore, the participant mentioned some models they deemed adequate additions to the alternative solutions provided for their domain, highlighting how further decision model improvements could be beneficial.

Case study 4 showcases how the machine learning model selection process can be sped up with a predetermined list of models suited to a problem domain. It further illustrates the potential of updating such a model in line with current research developments.

7.5 Case study 5

This study is called *"Smart contract vulnerability detection based on semantic graph and residual graph convolutional networks with edge attention"*. Smart contracts facilitate credible transactions without third parties but are vulnerable to exploitation and cannot be modified once deployed. Ensuring their security is crucial due to the rapid increase in smart contracts. Deep learning offers a promising solution for detecting vulnerabilities, but current models fail to capture the syntax and semantic information embedded in smart contracts. This paper addresses function-level vulnerability detection by constructing a novel semantic graph (SG) for each function and using graph convolutional networks (GCNs) with residual blocks and edge attention [257]. The proposed model involves three stages: creating SGs with rich syntax and semantic information, learning code content and semantic features using an EA-RGCN model, and classifying functions as vulnerable. Experiments on real-world smart contract datasets show that the semantic graph and EA-RGCN model significantly improve accuracy, precision, recall, and F1-score detection performance.

7.5.1 Feature requirements

For the feature selection, we decided on three features in total. **'Must have: convolutional neural network'**, was decided based on the study's incorporation of the convolutional neural network as its central model structure. **'Should have: residual network architecture'** was decided based on the relevance of the residual network architecture within the proposed artifact. **'Should have: graph architecture'**, was similarly deemed relevant based on the structure of the proposed artifact. The participant agreed with our reasoning that the convolutional neural network is the central structure and that the residual network and graph architectures are also relevant.

7.5.2 Models

We identified the main model as EA-RGCN, which consists of a word2vec component [258], edge attention (EA) [259], and a residual graph convolutional network (RGCN) [260]. The participant agreed that this structure summarizes the primary model of the study.

7.5.3 Results

Our decision model produced the following list of alternative solutions: residual network (ResNet) [163], ResNet50/152V2 [220], Hourglass [261], dilated residual network [262], ResNeXt [263], deep residual networks [264], UNet [221], SegNet [265], directed graph convolutional neural network [266].

The participant noted their mixed feelings, deeming only some models a good alternative. They felt a dilated residual network and a directed graph convolutional neural network might be adequate alternative solutions to their research.

7.5.4 Analysis

The participant showed interest in using our decision model in future research related to and unrelated to convolutional neural network structures. However, they noted learning how the decision model works internally before deciding whether to use it in a research domain.

Case study 5 showcases how the decision model can provide accurate alternative solutions for several specific machine learning model architectures combined. Although not all alternative solutions were deemed equally relevant, none seemed to contradict the domain directly.

7.6 Case study 6

This study is called *"Digital forensic framework for smart contract vulnerabilities using ensemble models"*. Forensic identification of vulnerabilities in Ethereum-based smart contracts is crucial as they manage significant cryptocurrency assets. These smart contracts can have vulnerabilities, including denial of service (DoS), access control issues, arithmetic overflow, bad randomness, re-entrance, and unchecked low-level calls. This paper presents a novel vulnerability detection system using natural language processing and machine learning [267]. The model targets Ethereum-based smart contracts and uses benchmark data with different vulnerability types. Unlike classical methods that use the SolMatrix tool and face issues with feature variations and data imbalance, this methodology assesses the probability of vulnerabilities in Ethereum Solidity Smart Contracts. The proposed model's performance, evaluated in terms of accuracy, F-measure, and area under the curve, shows that with SMOTE sampled data, the random forest algorithm achieved nearly 90% accuracy, an area under the curve of about 0.7, and an average F-measure of 0.86.

7.6.1 Feature requirements

For the feature selection, we decided on two features in total. **'Must have: classification model'**, was decided based on the detection task the domain demands. **'Should have: text data compatibility'**, was deemed a relevant feature due to the nature of smart contracts data being text. The participant agreed with both of these features, sharing no further remarks.

7.6.2 Models

We identified several models within the study. Related models are a continuous bag of words and a continuous skip-gram. The main machine learning model is an ensemble structure consisting of a decision tree, random forest, bagging [268], AdaBoost [269], and gradient boost model [152]. The participant agreed with the identified models and the structure we described for their function.

7.6.3 Results

Our decision model produced the following list of alternative solutions: bidirectional encoder representations from transformers (BERT) [112], multinomial naïve Bayes (MNB) [237], discriminative multinomial naïve Bayes (DMNB) [238], multilayer feedforward ANN [5], attention-based bi-directional LSTM (ABLSTM) [239], multitask-clinical BERT (MT-Clinical BERT) [240], BEHRT [241], CheXbert [242], HyperNet [243], forwarding neural network [244], deep learning for code clones (DLC) [245].

The participant noted that our alternative solutions were relevant to the problem domain. They considered all alternative solutions as possible avenues for better results in the study.

7.6.4 Analysis

The participant mentioned being interested in the workings of our decision model and expressed interest in using it in future research.

Case Study 6 showcases high-quality alternative solutions provided by the decision model, as the participants rated all machine learning models as relevant and interesting avenues for their research.

7.7 Case study 7

This study is called *"TP-Detect: trigram-pixel based vulnerability detection for Ethereum smart contracts"*. Smart contracts on the Ethereum blockchain are immutable programs that execute when predetermined conditions are met but are vulnerable to coding errors. This paper attempts to classify these vulnerabilities using feature extraction and machine learning [270]. A dataset was constructed from pixel values of images and trigram feature extraction. It was trained using various machine learning models,

including multilabel k-nearest neighbors (MLkNN) [271], binary relevance k-nearest neighbors (BRkNN) [272], random forest, and naïve Bayes. The naïve Bayes model performed the best, achieving F1-scores of 99.38% and 99.44% using binary relevance and classifier chain, respectively. The random forest model also showed strong performance with F1-scores of 96.71% and 96.61%. In contrast, multilabel k-nearest neighbors and binary relevance k-nearest neighbors had lower F1 scores of 88.19% and 89.71%, respectively. This indicates that the created dataset is more effective than models using opcode characteristics or image-based detection.

7.7.1 Feature requirements

For the feature selection, we decided on four features in total. **'Must have: classification model'**, was decided on due to the inherent nature of the classifier task in vulnerability detection. **'Could have: KNN model'** was selected as a likely feature due to the prevalent k-nearest neighbors structure in the compared machine learning models. **'Could have: naïve Bayes model'**, was deemed a likely model feature due to a naïve Bayes model being used for machine learning comparison. **'Could have: Deep Forest model'**, was deemed a likely model due to specific variants like random forest being used as a machine learning comparison model. The participants agreed with our selected features and deemed them relevant to finding adequate alternative solutions. They furthermore did not suggest any additional features that we could consider related.

7.7.2 Models

We identified the models used as k-nearest neighbors, multilabel k-nearest neighbors, binary relevance k-nearest neighbors, random forest, and naïve Bayes, with naïve Bayes performing the best on their artificial dataset. The participant agreed to our machine learning model descriptions, leaving no further comments.

7.7.3 Results

Our decision model produced the following list of alternative solutions: multinomial naïve Bayes (MNB) [237], discriminative multinomial naïve Bayes (DMNB) [238], adaptive k-nearest neighbor [273], k-star classifier [274], classification and regression tree (CART) [275], k-star [276], local nearest neighbor (LNN) [277], weighted random forest (WRF) [278], augmented naïve Bayes [279], weighted KNN [280], quantum nearest neighbors [281], Ruan-Xue-Liu-Tan-Li (RXLTL) [282], Ibk [100].

The participant noted that our alternative solutions were relevant to the problem domain. They considered all alternative solutions as possible avenues for better results in the study.

7.7.4 Analysis

The participant mentioned being interested in the workings of our decision model and expressed interest in using it in future research.

Case Study 7 showcases high-quality alternative solutions provided by the decision model. The participants rated all machine learning models as relevant and interesting avenues for their research. It highlights how the decision model can produce lists related to specific feature requirements, like several variants of the k-nearest neighbor structure.

7.8 Case study 8

This study is called *"Block-gram: Mining knowledgeable features for efficiently smart contract vulnerability detection"*. Smart contracts on the blockchain, such as those used for decentralized applications on Ethereum, require effective vulnerability detection to prevent significant economic losses. Since repairing and updating smart contracts is challenging, detecting vulnerabilities before deployment is crucial. Due to extensive feature requirements and inefficiency, traditional methods like code analysis and learning-based techniques are deemed too time-consuming for large-scale contracts. This paper proposes improving detection efficiency by reducing feature dimensions with expert knowledge [283]. The authors introduce block-gram, a feature extraction model that forms low-dimensional knowledge-based features from bytecode by converting runtime code into opcode sequences and mining scalable

block-gram features. Combined with SHAP values for interpretability, these features are evaluated on a dataset of 33,885 contracts using seven learning algorithms. Results show that the new method speeds up detection by 25× to 650× compared to N-gram features while enhancing model interpretability.

7.8.1 Feature requirements

For the feature selection, we decided on a single feature. **'Must have: dimensionality reduction'**, was decided based on the main topic of the study, which is to improve detection efficiency by reducing the dimensions of features. The participant agreed with our selected feature, stating dimensionality reduction [284] is a significant topic for machine learning in general.

7.8.2 Models

We identified their artifact as the block-gram model. This model was tested with machine learning models, including XGBoost, random forest, k-nearest neighbors, logistic regression, decision tree, naïve Bayes, and long short-term memory. The participant agreed with our selected models and our description of using the Block-gram model.

7.8.3 Results

Our decision model produced the following list of alternative solutions: partial least squares (PLS) [285], linear discriminant analysis (LDA) [286], principal component analysis (PCA) [287], RPCA [288], multivariant discriminant analysis (MDA) [289], PLSA [290], CTM [291], Nnge [100], t-SNE [292], uMAP [293], canonical correlation analysis (CCA) [294].

The participant noted how a small feature space was crucial to their research. However, they noted that they were not experts on dimensionality reduction models, having relied on expert knowledge during their study. Therefore, they found themselves unfit to comment critically on our list of alternative solutions.

7.8.4 Analysis

The participant noted being interested in seeing the inner workings of the decision model. They were interested in using the decision model in future research as well.

Case study 8 was adequately performed as the participants agreed to our selected features and models. However, their lack of expertise regarding the dimensionality reduction domain made them unfit to critically assess whether our alternative solutions would benefit their current or future research.

7.9 Evaluation

After completing all case studies, we compiled the summarized feedback into Figure 7.2. For each question asked for every case study, we narrowed down whether the feedback was positive, mixed, or negative. Positive feedback means the participant agreed with our statements and offered no direct criticism. Mixed feedback means the participant agreed with only some of our statements and offered criticism. Negative feedback means the participant rejected our statements.

When examining the results of our case studies, several statements can be made. The feature selection phase received five positive responses and three mixed responses. Considering this data, we can conclude the feature selection phase was performed with decent accuracy. The model selection phase received seven positive responses and one negative response. Considering this data, we can conclude that the model selection process was performed with high accuracy, though not without mistakes, as the single negative response should not be ignored. The alternative solutions provided by our decision model received two positive responses, five mixed responses, and one negative response. Considering this data, we can conclude that while the decision model occasionally provides accurate alternative solutions, it often delivers mixed-quality results. Standout issues here are a situation where the decision model underfits and produces too many alternative solutions, resulting in bloated decision data, or a situation where the decision model produces too many irrelevant alternative solutions, resulting in low-quality decision data. Finally, when considering the interest in the decision model, we see five positive responses and three mixed responses.

Case study 1	
Study name	Protein Family Classification from Scratch: A CNN Based Deep Learning Approach
Publication	IEEE/ACM Trans. Comput. Biol. Bioinformatics
Domain	protein family classification
Selected features	Must have: classification model, convolutional neural network
Selected models	CNN
Alternative solutions	Deep convolutional neural network (DCNN), dense CNN, gate CNN,...
Case study 2	
Study name	Runtime Adaptation in Wireless Sensor Nodes Using Structured Learning
Publication	ACM Trans. Cyber-Phys. Syst.
Domain	runtime adaptation
Selected features	Must have: reinforcement learning; Could have: MDP
Selected models	compact MDP model (CMM)
Alternative solutions	deep learning (DL), neural network (NN), Q-learning,...
Case study 3	
Study name	Identifying click baits using various machine learning and deep learning techniques
Publication	International Journal of Information Technology
Domain	identifying click baits
Selected features	Must have: classification model; Should have: text data compatibility
Selected models	Logistic regression, Random forest, XGBoost,...
Alternative solutions	BERT, multinomial naïve bayes (MNB), discriminative multinomial naïve bayes (DMNV),...
Case study 4	
Study name	DeepVulSeeker: A novel vulnerability identification framework via code graph structure and pre-training mechanism
Publication	Future Generation Computer Systems
Domain	smart contract vulnerability detection
Selected features	Must have: classification model, convolutional neural network; Should have: multiple layers
Selected models	CNN, MLP
Alternative solutions	deep convolutional neural network (DCNN), gate CNN, generic CNN (GCNN),...
Case study 5	
Study name	Smart contract vulnerability detection based on semantic graph and residual graph convolutional networks with edge attention
Publication	Journal of Systems and Software
Domain	smart contract vulnerability detection
Selected features	Must have: convolutional neural network; Should have: residual network architecture, graph architecture
Selected models	word2vec, Residual graph convolutional network (RGCN), Edge attention (EA)
Alternative solutions	Residual network (ResNet), ResNet50/152V2, Hourglass,...
Case study 6	
Study name	Digital forensic framework for smart contract vulnerabilities using ensemble models
Publication	Multimedia Tools and Applications
Domain	smart contract vulnerability detection
Selected features	Must have: classification model; Should have: text data compatibility
Selected models	Ensemble model of Decision tree, Random forest, Bagging,...
Alternative solutions	BERT, multinomial naïve bayes (MNB), discriminative multinomial naïve bayes (DMNV),...
Case study 7	
Study name	TP-Detect: trigram-pixel based vulnerability detection for Ethereum smart contracts
Publication	Multimedia Tools and Applications
Domain	smart contract vulnerability detection
Selected features	Must have: classification model; Could have: KNN model, Naïve bayes model, Deep forest model
Selected models	KNN, Multilabel KNN, Binary relevance KNN,...
Alternative solutions	multinomial naïve bayes (MNB), discriminative multinomial naïve bayes (DMNV), adaptive KNN,...
Case study 8	
Study name	Block-gram: Mining knowledgeable features for efficiently smart contract vulnerability detection
Publication	Digital Communications and Networks
Domain	smart contract vulnerability detection
Selected features	Must have: dimensionality reduction
Selected models	Block-gram, XGBoost, Random forest,...
Alternative solutions	partial least squares (PLS), linear discriminant analysis (LDA), principal component analysis (PCA),...

Figure 7.1: A summarized overview of the data prepared for all case studies.

Considering this data, we can conclude that overall interest in our decision model is present but sometimes accompanied by skepticism. Skeptic remarks mainly regarded the actual use of the decision model in practice and whether or not it would be suited for use in a wide variety of machine learning domains.

Case study	1	2	3	4	5	6	7	8
Feature selection	Mixed	Mixed	Positive	Mixed	Positive	Positive	Positive	Positive
Model selection	Negative	Positive	Positive	Positive	Positive	Positive	Positive	Positive
Alternative solutions	Negative	Mixed	Mixed	Mixed	Mixed	Positive	Positive	Mixed
Decision model	Mixed	Positive	Positive	Mixed	Mixed	Positive	Positive	Positive

Figure 7.2: A summary of the overall feedback of each participant to each of our questions.

Discussion

This section comprehensively evaluates the study's implications in machine learning model selection. We cover the evaluations of our results related to our research questions. We mention crucial outcomes during our systematic literature review. We cover worthwhile notions from both the expert interviews and the case studies. We critically examine the validity and methodology of the results, emphasizing various potential threats to validity. The outcomes and accompanying limitations of the study are discussed as well. Finally, the practical implications of the findings are explored, and possible avenues of future research are considered.

8.1 Research questions

Regarding our first research question: *"Which machine learning models should be included in the machine learning model selection decision model?"*, the SLR provided a comprehensive list of 548 total machine learning models, which the expert interview feedback further increased to a total of 571 machine learning models. For the analysis in this paper, we focused only on the machine learning models featured in at least ten papers. The five most common models were support vector machines, random forests, k-nearest neighbors, decision trees, and logistic regression. We also considered what models tend to be used in a single research project to enhance their utility for the machine learning decision model. The five models combined with another model most often are support vector machines, random forests, k-nearest neighbors, decision trees, and convolutional neural networks. We also considered model trends and how certain models have been used more frequently in recent years. The most frequently used models are roughly the same over the years: support vector machines, random forests, k-nearest neighbors, decision trees, and logistic regression. In recent years, we have noticed a rising trend in the use of residual networks and VGG16/19. This data shows us the most relevant machine learning models to add to our machine learning decision model.

Regarding our second research question: *"Which characteristics and features should be included in the machine learning model selection decision model?"*, The SLR provided substantial insight into how machine learning models can be categorized and what features can be assigned. We were able to set up a comprehensive taxonomy for machine learning models, which categorizes each model into at least one of 25 main categories. Some of the most associated main categories are supervised, unsupervised, and reinforcement learning. These categories encompass a broad range of models and techniques applied in machine learning, serving as essential machine learning paradigms. The categories are further divided into subcategories, characterizing each machine learning model more specifically. We created a comprehensive list of features that can be associated with each machine learning model. 273 features were identified based on the SLR, which we divided into 19 categories based on their context, domain, and applications. These features aid the decision model, providing a set of parameters based on which specific models can be selected.

Regarding our third research question: *"Which training and evaluation datasets should be included in the machine learning model selection decision model?"*, The SLR provided a list of commonly utilized datasets across the various papers. We identified a total of 37 datasets that were published in at least two publications. Notably, MNIST, CIFAR-10/100, ImageNet, KDDCup '99, and NASA emerged as the top five datasets commonly used in evaluating machine learning models. Combined, these datasets have been utilized in over 30 publications, highlighting their significance and wide adoption. This information regarding the datasets provides our decision model with a sufficient list of machine learning datasets on which to rely.

Regarding our fourth research question: *"Which evaluation measures and methods should be included in the machine learning model selection decision model?"*, The SLR provided a comprehensive list of the most commonly used evaluation measures for machine learning. Among all the evaluated measures, we focused on the evaluation measures that were mentioned in at least ten papers. Precision, recall, F1-Score, accuracy, and area under the curve are among the top five evaluation measures identified in the SLR. The most relevant evaluation measures aid as additional data in our machine learning decision model.

Regarding our fifth research question: *"How can a decision model be developed to support research modelers in selecting machine learning models?"*, this was answered through our expert interviews and design science methods. We successfully built our decision model based on the MCDM framework. The model and feature list that make up our decision model were reevaluated based on the feedback received from the case studies. The number of models increased, and the feature list was adjusted to cover more avenues of machine learning. Furthermore, the taxonomy was revised to communicate better how machine learning paradigms, approaches, and domains relate to each other. Additional data was added to the taxonomy, and readability was also improved.

Our sixth research question, *"How should the machine learning model selection decision model be evaluated?"*, was answered through our case study and design science. While the feature selection and model selection phases were rated positively overall, more skepticism arose regarding the decision model and its alternative solutions. In future iterations of the decision model, we must critically review whether the decision model provides a list of alternative solutions that are accurately within the scope of a research project. Furthermore, the alternative solutions should be highly relevant to the selected features. Newer iterations of our decision model can be reviewed with similar case studies, possibly at a larger scale.

Our main research question, *"How can research modelers be supported in their machine learning model selection process?"* can be answered by summarizing our overall findings. The SLR provided a large amount of data, which can be processed into our comprehensive taxonomy, model, feature, evaluation method, and dataset-related tables. The expert interviews improved this data to a level suitable for scientific use. Our design science developed the decision model on a solid foundation and valuable data. The decision model was verified in our case studies, and it was shown to provide reliable and useful model suggestions related to specific domains.

8.2 Threats to validity

Validity evaluation is a well-established pillar in empirical studies, encompassing systematic literature reviews, expert interviews, and case study research [295]. This paper's validity assessment covers various dimensions, including SLR outcomes, expert interview participants, case study participants, construct validity, internal validity, external validity, conclusion validity, study limitations, and decision model expansions and alterations. These validity assessments may impact this study's findings and their meaning.

8.2.1 SLR outcomes

Throughout the systematic literature review, We collected 548 models, out of which 379 were singletons, representing 69% of the total models. This observation indicates that many research modelers develop and use unique models tailored to their research questions. However, relying heavily on singletons can restrict the generalizability of research outcomes and impede meaningful comparisons between separate approaches. Encouraging the adoption of common models or establishing standards of model evaluation could significantly enhance the reproducibility and comparability of machine learning research [296].

In some instances, the methodology for combining models was not clearly described in the publications. This lack of transparency challenges understanding the underlying techniques and logic used and evaluating their effectiveness. Explicitly providing descriptions of model combination techniques and the reasons behind their selection is crucial to increase transparency and facilitate the replication and extension of research findings [297].

Regarding dataset usage, we observed that not all datasets are publicly accessible. This presents research modelers with potential issues in replicating and validating reported results. Consequently, the ability to objectively compare and benchmark different models becomes hampered, impeding the identification of state-of-the-art techniques and areas for improvement [298]. The consequences of this issue extend further, as the duplication effort in collecting and preparing new datasets consumes resources and consequently decelerates research progress. While the prominence of data science within organizations has given rise to teams of research modelers collaborating, the collaborative practices employed still vary according to the kinds of tools used [99]. To mitigate these challenges, a culture of openness and collaboration across the larger scientific research community is crucial.

8.2.2 Expert interviews participants

The expert interview participants were willing and supportive in offering feedback on our questions. Communication with all participants progressed positively, and no concerns regarding ambiguous consent to their participation were raised. The expert interview participants were generally interested in using a final version of our decision model. Some participants were also interested in using our taxonomy and feature data. The participants mentioned how our model and feature databases might be bloated to some degree, suggesting the potential removal of outdated and obscure models and features. While the time needed for each interview varied, all participants declared they had no additional questions at the end, suggesting that communication of all questions was clear to them.

8.2.3 Case study participants

The case study interview participants were willing and supportive of our using their paper for a case study. Communication with all participants progressed positively, and no concerns regarding ambiguous consent to their participation were raised. The case study participants were interested in using our decision model and the reasoning behind our selected models and features for their paper. Some participants showed interest in reading our final paper as well.

Participants noted how the decision model can recommend many alternative solutions, which they describe as a positive or negative trait. While some participants took part in the interview through a video call, some opted to answer questions via email instead. This can arguably increase the ambiguity of some answers these participants provided. Nevertheless, participants noted having no additional questions once the interviews finished, suggesting communication of all questions was clear to them. Another potential validity constraint is how many of our case studies occurred within the smart contract vulnerability detection domain. Additional case studies in distinct domains would provide a broader incorporation of our decision model, increasing the robustness of the case study results.

8.2.4 Construct validity

Construct validity pertains to selecting appropriate operational measures for the concepts under the systematic literature review study. Several potential threats to construct validity need to be addressed. The inclusion and exclusion criteria for each phase in the SLR are clearly defined in several subsections in Chapter 3, minimizing the risk of inappropriate selections. In this research, we developed a meta-model (shown in Figure 6.1) based on the ISO/IEC/IEEE standard 42010 [199] framework to represent the decision-making process for machine learning model selection. We formulated comprehensive research questions using the meta-model's essential elements, ensuring an exhaustive coverage of pertinent publications on machine learning model selection approaches.

8.2.5 Internal validity

Internal validity threats pertain to verifying cause-effect relationships within the study's scope to ensure robustness. We employed a rigorous quasi-gold standard (QGS) [299] to minimize selection bias in paper inclusion. By combining manual and automated search strategies, the QGS gave us an accurate evaluation of sensitivity and precision. Our search spanned four popular online digital libraries, widely regarded to encompass many high-quality publications relevant to machine learning model selection. The review process and data collection were handled by a single researcher, who validated their findings through expert interviews and case studies. To avoid significant sample bias, the expert interview and case study participants were individuals selected only from people outside of our research team. For the expert interviews, we contacted various industry and research modeling professionals. These professionals had various domains of expertise, and no participant was professionally connected to another. For the case studies, various research modelers were contacted based on their relevant case study material. Detailed questions were asked to minimize potential subjective quality assessment during user analysis. Additionally, all meetings with participants were recorded as evidence for transparency.

8.2.6 External validity

External validity pertains to the generalizability of the study's findings in real-world applications. The study considered publications discussing machine learning model selection approaches across multiple

years. Although potential exclusions and studies' inaccessibility may impact the generalizability of SLR and case study results, the prospected proportion of inaccessible studies is not expected to significantly affect overall findings. Furthermore, the knowledge extracted from this research can be applied to support the development of new theories, methods, and heuristics for machine learning model selection challenges, thus benefiting academia in this field.

8.2.7 Conclusion validity

Conclusion validity pertains to the accuracy and consistency of the study's conclusions. It ensures that the study's methods, including data collection and analysis, can be replicated to yield consistent results. To ensure reproducibility, each step encompassing the systematic literature review is documented appropriately in Chapter 3. Furthermore, the design decisions and exclusions of data are covered and motivated in Chapter 4 and Chapter 5. The data collected during the study is made publicly available in the Appendix A, facilitating potential replications of our expert interviews and case studies. Furthermore, all data steps of the SLR are recorded in the Appendix A for easy referencing and result validation. Though there is always potential for bias in our selection process, potential retreads of our data selection render this data transparent.

8.2.8 Study limitations

While we put effort into maintaining scientific validity wherever possible, certain avenues of research still bring probable bias with them. For example, we limited our expert interview and case study participant groups to English-speaking people, which excludes certain groups in the scientific field. During our data evaluation, there were several discussions regarding machine learning models and machine learning feature-related semantics. Sometimes, participants and the interviewer used identical terminology to refer to different concepts. To progress our data collection on these concepts, we had to come to an agreement on what a concept or term entailed specifically. Furthermore, while we did not consciously discriminate against participants based on anything other than their machine learning expertise, our sample size limits the diversity of domain-specific experts included in our expert interviews and case studies.

8.2.9 Decision model expansions and alterations

While our current decision model is an attempt to cover the machine learning model selection domain exhaustively, it is ultimately an artifact of its time. We have designed our data collection process and decision model creation to be repeatable for future iterations. While we have received positive remarks regarding our current decision model results from industry and research modeling experts, it cannot be understated how volatile and ever-expanding the machine learning domain is. The expert interviews provided several insights into current and upcoming machine learning techniques and domains, rendering our confidence in the data coverage of our decision model to a sufficient level.

Conclusion and future work

9.1 Conclusion

In this paper, we have comprehensively investigated creating a decision model for machine learning selection. Our main objective was to address the challenge machine learning research modelers face in selecting the most effective combinations of machine learning models for their research domain.

To ensure the credibility and reliability of our findings, we conducted a systematic literature review, conducted 11 expert interviews, and performed eight case studies. We aimed to design and improve our decision model and test its accuracy and practical use. We meticulously considered various dimensions of validity, including construct validity, internal validity, external validity, and conclusion validity.

Drawing inspiration from the ISO/IEC/IEEE standard 42010 [199], we devised a meta-model to represent the decision-making process for machine learning model selection. By formulating comprehensive research questions, we ensured the inclusion of relevant studies and achieved far-reaching coverage of pertinent publications.

Our study offers a holistic understanding of machine learning models. The SLR analyzed over 500 papers from recent years, identifying 571 distinct models and 273 features. This data collection provides valuable insight into the various machine learning models and approaches, contributing to the advancement within the field.

We performed 11 expert interviews to enhance further and improve our decision model. These interviews aided our design science and proved to be beneficial to the further development of our decision model, model data, feature data, and taxonomy.

Building on the findings of the SLR, we proposed a decision model to guide research modelers in selecting the most suitable machine learning models for their research tasks. The decision model considers several features and provides rankings based on relevancy and importance. The decision model enhances the abilities of research modelers by providing an effective and efficient alternative to existing machine learning model selection approaches.

We demonstrated the applicability of the decision model through eight case studies, highlighting its practical use-case scenarios. The decision model aids research modelers in identifying potential solutions for their research problem, providing many accurate and relevant results.

9.2 Future work

We consider our decision model a robust answer to machine learning model selection and envision our decision model to be updated and improved upon further in the future. Our decision model data could be updated by integrating newer machine learning models and features or by adding legacy models and features that have acquired new relevance. Our decision model could be expanded by performing studies similar to our data-driven decision model approach, improving the relevant data to a higher accuracy. Large language models could play an important role in extracting data from future publications, collecting new features and models for our decision model, and allowing the data relevancy to remain intact. Furthermore, our decision model could be enhanced in ways outside the scope of this study, catering to specific user needs or substantially altering its inner workings based on new developments.

Acknowledgement

We extend our sincere gratitude to the domain experts who actively participated in our expert interviews. Their valuable insights and expertise have significantly enriched the quality of this study. We further extend this gratitude to our case study authors. Their valuable feedback and insights, as well as their cooperation and willingness to share their publications, further improved the robustness of the decision model's accuracy. Finally, we extend our appreciation to Ms. Baninemeh, who helped expand our scope during the case study recruitment process and significantly improved the overall feedback we received during this phase.

Appendices

Data

This appendix provides a link to all additional data acquired during this study.

For our full database related to this study, please refer to:

`https://data.mendeley.com/datasets/drh9669vc3/1`

Abbreviations

Model	Abbreviation	Model	Abbreviation	Model	Abbreviation
support vector machine (SVM)	SVM	**bayesian network**	BN	**self-organising map (SOM)**	SOM
random forest (RF)	RF	**k-means**	k-means	**extreme learning machine (ELM)**	ELM
k-nearest neighbours (KNN)	KNN	**gradient boosting (GB)**	GB	**residual network (ResNet)**	ResNet
decision tree (DT)	DT	**gated recurrent unit (GRU)**	GRU	**gaussian process (GP)**	GP
logistic regression (LR)	LR	**support vector regression (SVR)**	SVR	**generative adversarial network (GAN)**	GAN
convolutional neural network (CNN)	CNN	**bidirectional LSTM (BLSTM)**	BLSTM	**C4.5**	C4.5
naïve bayes (NB)	NB	**radial basis function (RBF)**	RBF	**regression**	R
long short-term memory (LSTM)	LSTM	**deep belief network (DBN)**	DBN	**Q-learning**	Q-learning
multilayer perceptron (MLP)	MLP	**linear regression**	LR	**GoogleNet**	GoogleNet
recurrent neural network (RNN)	RNN	**Bagging**	Bagging	**principal component analysis (PCA)**	PCA
artificial neural network (ANN)	ANN	**deep learning (DL)**	DL	**VGG16/19**	VGG
neural network (NN)	NN	**genetic algorithms**	GA	**ARIMA**	ARIMA
deep neural network (DNN)	DNN	**particle swarm optimization (PSO)**	PSO	**classification and regression tree (CART)**	CART
XGBoost	XGBoost	**AlexNet**	AlexNet		
Adaboost	AdaBoost	**hidden markov model (HMM)**	HMM		

Figure B.1: Provides the full names of our 43 most commonly selected models, along with their used abbreviations.

Expert interview consent form

This appendix provides the participant consent form related to the expert interviews.

Consent form version 1.0 (01-02-2024)

Information about our research project

-The project is called: 'A Data-Driven Decision Model for Machine Learning Model Selection'. The main research question is *"How can research modelers be supported in their machine learning model selection process?"*. The project involves the creation of a selection model to aid research modelers in selecting machine learning models for their projects. This selection model was created through data collection and design science. The expert interview phase collects additional data to improve the model further. The final project result will be a published model detailed in a written thesis.

-The main researchers are:

Lex Steffens Game and Media Technology master student at Utrecht University
a.j.steffens@students.uu.nl

Dr. Siamak Farshidi Researcher and lecturer at the Department of Information and Computer Science at Utrecht University
s.farshidi@uu.nl

Dr. Slinger Jansen Assistant professor at the Department of Information and Computer Science at Utrecht University
slinger.jansen@uu.nl

Dr. Fabiano Dalpiaz Professor of Software Production in the Department of Information and Computing Sciences at Utrecht University
f.dalpiaz@uu.nl

Information about participation

-Your role in the participation: You will be interviewed by researcher Lex Steffens on some topics related to machine learning model selection. We expect some of your prior knowledge on machine learning to aid our data collection. The interview will take roughly 45 to 60 minutes. The interviews will be conducted through Microsoft Teams via an invitation (unless you specifically request another platform).

Information about privacy

Data that will be collected: We will record the interview and possibly transcribe its content. The interview will act as evidence for the project; however, it will not be published alongside the project, and your name will not be published either.

-Data we will share with you: Upon completion of the research project, we will share the publications with you. Your interview recording will not be shared with you unless you specifically request it.

Data subject's rights

Participation is voluntary. You will not be compensated in any way for participating in our project. You have the right to withdraw from participation at any time during our proceedings. You are free to refuse participation, and this will not negatively impact you or your relationship with us.

-How to contact us: Should you want to exercise any of the aforementioned rights, or want to inquire about anything else, you can send your message to Lex Steffens at a.j.steffens@students.uu.nl

Expert interview responses

Model recognized by expert	1	2	3	4	5	6	7	8	9	10	11
SVM	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
RF	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
KNN	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
DT	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
LR	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
CNN	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
NB	Y	Y	Y	Y	Y	Y	Y	Y	N	Y	Y
LSTM	Y	Y	Y	Y	Y	Y	Y	Y	N	Y	Y
MLP	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	N
RNN	Y	Y	Y	Y	Y	Y	Y	Y	N	Y	Y
ANN	Y	Y	Y	Y	N	Y	Y	Y	N	Y	Y
NN	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
DNN	Y	Y	Y	Y	Y	Y	Y	Y	N	Y	Y
XGBoost	Y	Y	Y	Y	Y	Y	Y	Y	Y	N	Y
AdaBoost	Y	Y	Y	Y	Y	Y	N	Y	N	Y	Y
BN	Y	N	Y	Y	Y	Y	N	N	N	N	Y
k-means	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
GB	Y	N	Y	Y	Y	Y	Y	Y	Y	Y	Y
GRU	Y	N	Y	Y	N	N	N	N	N	N	N
SVR	Y	N	Y	Y	N	Y	Y	Y	Y	Y	Y
BLSTM	Y	N	Y	Y	N	Y	Y	Y	N	N	N
RBF	Y	N	N	N	N	Y	N	Y	N	Y	Y
DBN	Y	Y	Y	N	Y	Y	N	N	N	N	N
LR	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
Bagging	Y	Y	Y	Y	Y	N	Y	Y	N	Y	Y
DL	Y	Y	Y	Y	Y	Y	Y	Y	N	Y	Y
GA	Y	N	Y	Y	Y	Y	Y	Y	N	Y	N
PSO	Y	N	N	N	N	Y	N	Y	N	N	N
AlexNet	Y	Y	Y	Y	N	Y	Y	Y	N	Y	N
HMM	Y	N	Y	Y	N	Y	Y	Y	Y	Y	Y
SOM	N	N	N	Y	N	Y	Y	N	N	Y	N
ELM	N	N	N	N	N	Y	N	N	N	N	N
ResNet	Y	N	N	Y	Y	Y	Y	Y	N	Y	N
GP	Y	N	N	N	N	Y	N	Y	N	N	Y
GAN	Y	Y	Y	Y	Y	N	Y	Y	Y	Y	N
C4.5	N	N	Y	Y	N	N	N	Y	N	N	N
R	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
Q-learning	Y	N	Y	N	Y	Y	Y	Y	N	N	N
GoogleNet	Y	N	N	N	Y	Y	N	Y	N	Y	Y
PCA	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
VGG	Y	N	N	Y	N	N	Y	Y	N	Y	Y
ARIMA	N	Y	N	Y	N	Y	Y	Y	Y	N	N
CART	Y	N	Y	N	N	Y	Y	Y	N	N	N

Figure D.1: Showcases the familiarity of each participant with all of our main machine learning models.

Feature recognized by expert	1	2	3	4	5	6	7	8	9	10	11
Model Type											
classification	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
regression	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
sequential	Y	Y	Y	Y	Y	Y	Y	Y	N	Y	Y
generative	Y	Y	Y	Y	N	Y	Y	Y	Y	Y	N
discriminative	Y	N	Y	Y	Y	Y	Y	Y	N	Y	N
ensemble	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
optimization	Y	Y	Y	Y	N	Y	Y	Y	Y	Y	Y
tree-based model	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
clustering	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
probabilistic model	Y	N	Y	Y	Y	Y	Y	Y	Y	Y	Y
object detection	Y	Y	Y	Y	N	Y	Y	Y	Y	Y	Y
evolutionary algorithms	N	N	Y	Y	N	Y	Y	Y	N	Y	N
instance based learning	Y	N	Y	Y	Y	Y	Y	N	N	Y	Y
decision making	N	N	Y	Y	N	Y	Y	N	Y	Y	Y
processing for NLP	Y	Y	Y	Y	N	Y	Y	Y	Y	Y	Y
prediction mechanism	Y	Y	N	Y	N	Y	Y	N	Y	Y	Y
dimensionality reduction	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
pretrained transformer for NLP	Y	Y	Y	Y	N	Y	Y	Y	N	Y	N
fuzzy inference	N	N	N	Y	Y	Y	N	N	N	N	N
time series forecasting	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
feature extraction	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
topic modeling	Y	Y	N	Y	N	Y	N	Y	N	Y	N
ranking	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
feature selection	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
association rule mining	N	N	Y	N	Y	Y	N	N	N	N	N
representation learning	Y	N	N	N	N	Y	Y	Y	N	Y	N
statistical	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	N
inductive logic programming	N	N	N	N	N	Y	N	N	N	N	N
metric learning	Y	Y	N	Y	N	Y	Y	Y	N	Y	Y
feature representation	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
collaborative machine learning	Y	Y	N	N	Y	Y	N	N	N	N	Y
instance segmentation	Y	Y	Y	Y	N	Y	Y	N	N	Y	N

Feature recognized by expert	1	2	3	4	5	6	7	8	9	10	11
Algorithm Family											
neural network	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
support vector machine	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
autoencoder	Y	Y	Y	Y	N	Y	Y	Y	N	Y	N
decision tree	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
bayesian model	Y	N	Y	Y	Y	Y	Y	Y	Y	Y	Y
boosting	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
LSTM	Y	Y	Y	Y	Y	Y	Y	Y	N	Y	Y
k-nearest neighbors	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
regression algorithms	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
Q-learning	Y	N	Y	N	Y	Y	Y	Y	N	N	N
extreme learning machine	Y	N	N	Y	N	Y	N	N	N	N	N
rule-based	Y	N	N	N	Y	Y	Y	N	Y	Y	N
fuzzy logic	Y	N	Y	Y	Y	Y	N	N	N	N	N
generative adversarial network	Y	Y	Y	Y	Y	Y	Y	Y	N	Y	N
BERT	Y	Y	Y	Y	N	N	Y	Y	N	Y	N
generalized linear model (GLM)	Y	N	N	N	N	Y	Y	Y	N	N	N
genetic algorithms	Y	N	Y	Y	N	Y	Y	Y	N	Y	N
naïve bayes	Y	N	Y	Y	Y	Y	Y	Y	Y	Y	Y
hidden markov model	Y	N	Y	Y	N	Y	Y	Y	Y	Y	Y
deep belief network	Y	N	N	N	N	Y	Y	Y	N	N	N
perceptron	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
attention mechanisms	Y	Y	Y	Y	Y	Y	Y	Y	N	Y	N
ARIMA	N	Y	N	Y	N	Y	Y	Y	Y	N	N
bagging	Y	Y	Y	Y	Y	N	Y	Y	N	Y	Y
k-means	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
probabilistic graphical model	Y	N	N	N	N	Y	N	N	N	N	N
particle swarm optimization	N	N	N	N	N	Y	N	Y	N	N	N
frequent itemset mining	Y	N	N	N	Y	Y	N	Y	N	N	N
gaussian process	Y	N	Y	Y	N	Y	Y	Y	N	N	Y
linear discriminant analysis	Y	Y	N	Y	N	Y	Y	Y	N	Y	Y
radial basis function	Y	N	Y	N	N	Y	Y	Y	N	Y	Y
principal component analysis	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
SARSA	N	N	N	N	N	N	N	N	N	N	N
policy gradient	Y	N	N	N	N	Y	N	Y	N	N	N
partial least squares	Y	Y	Y	N	Y	Y	N	N	Y	N	N
markov decision process	Y	N	Y	Y	Y	Y	Y	Y	Y	Y	N
federated learning	Y	Y	N	Y	N	N	Y	Y	N	N	N
inductive logic programming	N	N	N	N	N	Y	N	N	N	N	N
dictionary learning	Y	N	N	Y	N	Y	Y	N	N	Y	N
hierarchical agglomerative clustering	Y	N	N	N	Y	N	N	N	N	Y	N
expectation-maximization	Y	N	Y	Y	Y	Y	Y	Y	N	Y	N
analytic hierarchy process	N	N	N	N	N	Y	N	N	N	N	N
post-decision state learning	N	N	N	N	N	Y	N	N	N	N	N
utility-based learning	N	N	N	N	N	Y	N	N	N	N	Y
MALMOS	Y	N	N	N	N	N	N	N	N	N	N
temporal difference learning	Y	N	N	N	N	Y	N	N	N	N	N
compound covariate predictor	N	N	N	N	N	Y	N	N	N	N	N
mean shift	Y	N	N	N	N	Y	Y	Y	N	Y	N
density-based spatial clustering	Y	N	Y	N	N	Y	Y	Y	Y	Y	Y
dynamic programming	Y	N	Y	Y	Y	Y	Y	Y	Y	Y	Y
bag of words	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
gaussian mixture model	Y	N	Y	Y	N	Y	Y	Y	Y	Y	Y
rough set	N	N	N	N	N	Y	N	N	N	N	N
hashing	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
convolutional neural network	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
recurrent neural network	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y

Figure D.2: Showcases the familiarity of each participant with all of our machine learning features.

Feature recognized by expert	1	2	3	4	5	6	7	8	9	10	11
Learning Paradigm											
supervised learning	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
unsupervised learning	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
reinforcement learning	Y	Y	Y	Y	Y	Y	Y	Y	Y	N	Y
semi-supervised learning	Y	Y	Y	Y	N	Y	Y	Y	N	Y	Y
self-supervised learning	Y	Y	Y	Y	N	Y	Y	Y	N	Y	Y
online learning	Y	Y	Y	Y	Y	Y	Y	Y	Y	N	Y
Model Complexity											
low	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
low to moderate	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
moderate	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
moderate to high	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
high	Y	Y	Y	Y	Y	Y	Y	Y	N	Y	Y
very high	Y	Y	Y	Y	Y	Y	Y	Y	N	Y	N
various	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
Linearity											
linear	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
non-linear	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
Parametric vs. Non-parametric											
parametric	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
non-parametric	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
Ensemble Methods											
boosting	Y	Y	Y	Y	Y	Y	Y	Y	N	Y	Y
deep forest	Y	N	Y	Y	N	Y	Y	Y	N	Y	Y
bagging	Y	Y	Y	Y	Y	Y	Y	Y	N	Y	Y
RAS-CO	N	N	N	N	N	Y	N	N	N	N	N
random subspace	Y	N	N	Y	N	Y	N	N	N	Y	N
random committee	Y	N	N	N	N	Y	Y	N	N	Y	N
rotation forest	N	N	N	N	Y	Y	N	Y	N	N	N
stacking	Y	Y	N	Y	N	N	Y	Y	Y	Y	Y
co-training	Y	N	N	N	N	N	Y	N	N	Y	N
mixture of experts	Y	N	Y	Y	N	N	Y	Y	N	Y	N
Regularization Type											
dropout	Y	Y	Y	Y	Y	Y	Y	Y	N	Y	Y
L2 (ridge)	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
batch normalization	Y	Y	Y	Y	N	Y	Y	Y	N	Y	Y
L1 (lasso)	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
gradient penalty	Y	N	N	Y	N	Y	N	Y	Y	Y	Y
reduced error pruning	Y	N	N	N	N	Y	Y	N	N	N	N
Wasserstein	Y	N	N	Y	N	N	N	Y	N	Y	N
contractive regularization	Y	N	N	N	N	N	N	N	N	N	N
layer normalization	Y	Y	N	Y	N	Y	Y	Y	N	Y	Y
instance normalization	Y	N	N	Y	N	Y	Y	Y	N	Y	Y
freezing parameters	Y	N	N	Y	Y	Y	Y	Y	N	Y	N
weight decay	Y	N	Y	Y	N	Y	Y	Y	N	Y	N

Feature recognized by expert	1	2	3	4	5	6	7	8	9	10	11
Neural Network Architecture											
convolutional	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
recurrent	Y	Y	Y	Y	Y	Y	Y	Y	N	Y	Y
feedforward	Y	Y	Y	Y	Y	Y	Y	Y	N	Y	Y
autoencoder	Y	Y	Y	Y	N	Y	Y	Y	N	Y	Y
transformer	Y	Y	Y	Y	Y	Y	Y	Y	N	Y	Y
residual network (skip connections)	Y	N	Y	Y	Y	Y	Y	Y	N	Y	N
fully connected	Y	Y	Y	Y	N	Y	Y	Y	N	Y	Y
generative adversarial	Y	Y	Y	Y	Y	Y	Y	Y	N	Y	N
boltzman machines	Y	N	Y	Y	N	Y	N	Y	N	N	Y
deep Q-network	Y	N	Y	N	Y	Y	Y	Y	N	N	N
feature pyramid	Y	N	N	Y	N	Y	N	N	N	Y	N
bidirectional	Y	Y	Y	Y	N	Y	Y	Y	N	Y	N
graph	Y	N	Y	Y	N	Y	N	Y	N	Y	Y
region proposal	Y	N	N	Y	N	Y	Y	Y	N	Y	N
mechanism	Y	Y	Y	Y	Y	Y	Y	Y	N	Y	N
grid-like structure	Y	Y	N	Y	N	Y	Y	N	N	Y	N
siamese	Y	N	Y	Y	N	N	Y	Y	N	Y	N
dilated convolutions	Y	N	N	Y	N	Y	N	Y	N	Y	N
autoregressive	Y	Y	Y	Y	N	Y	Y	Y	N	Y	N
inception	Y	N	N	Y	N	Y	Y	Y	N	Y	N
sparse	Y	Y	Y	Y	Y	Y	Y	Y	N	Y	Y
mask head	Y	Y	N	N	N	N	Y	Y	N	Y	N
subnetworks	Y	Y	N	Y	N	Y	Y	Y	N	Y	Y
gated recurrent unit	Y	N	Y	Y	N	Y	N	N	N	N	Y
actor-critic	Y	N	Y	N	N	N	Y	Y	N	N	N
dense blocks	Y	N	Y	Y	N	Y	Y	N	N	Y	N
probabilistic	Y	N	Y	Y	N	Y	Y	Y	N	Y	N
radial basis function	Y	N	N	N	N	Y	N	Y	N	Y	Y
spikes	N	N	N	N	N	Y	Y	Y	N	N	N
fuzzy logic	N	N	Y	Y	N	Y	N	N	N	N	N
capsule layers	N	N	N	N	N	Y	N	Y	N	N	N
hierarchical attention	N	N	N	Y	N	Y	Y	Y	N	Y	N
competitive layer	N	N	N	N	N	Y	N	N	N	N	N
triplet	Y	N	Y	N	N	Y	Y	N	N	N	N
inference composition	N	N	N	N	N	Y	N	N	N	Y	N
generative pre-trained transformer	Y	Y	Y	Y	N	Y	Y	Y	N	Y	N
hypernetwork	N	N	N	N	N	Y	N	Y	N	N	N
micro neural network	Y	N	N	N	N	Y	Y	N	N	Y	N
keypoint association network	Y	N	N	N	N	Y	N	Y	N	N	N
dual path	Y	N	N	N	N	Y	N	N	N	Y	N
fishblock	N	N	N	N	N	N	N	N	N	N	N
recursive	Y	N	N	Y	N	Y	Y	Y	N	Y	Y
shallow	Y	Y	N	Y	Y	Y	Y	Y	N	Y	Y
functional linkages	N	N	N	N	N	Y	N	N	N	Y	N
contradiction layer	N	N	N	N	N	Y	N	N	N	N	N
multi-head	Y	Y	Y	N	N	Y	Y	Y	N	Y	N
conditional random field	Y	N	N	N	N	Y	N	N	N	Y	N
multifilter	N	Y	N	N	N	Y	Y	N	N	Y	Y
modular	Y	N	N	Y	N	Y	N	Y	N	Y	N
fuser	Y	N	N	Y	N	Y	N	Y	N	Y	N
two-stream	Y	N	N	N	N	Y	N	N	N	Y	N

Figure D.3: Showcases the familiarity of each participant with all of our machine learning features.

Feature recognized by expert	1	2	3	4	5	6	7	8	9	10	11
Activation Function Type											
ReLU	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
sigmoid	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
tanh	Y	Y	Y	Y	Y	Y	Y	Y	N	Y	Y
gaussian	Y	Y	Y	Y	N	Y	Y	Y	N	N	Y
radial basis function (RBF)	Y	N	N	N	N	Y	Y	Y	N	Y	Y
gated linear unit (GLU)	Y	N	Y	Y	N	Y	N	Y	N	N	N
logistic	Y	Y	Y	Y	N	Y	Y	Y	N	Y	Y
stochastic units	Y	Y	N	Y	N	Y	Y	N	N	N	Y
PReLU	Y	N	N	N	N	Y	N	N	N	Y	Y
ReLU 6	Y	N	N	N	N	Y	N	Y	N	Y	Y
Objective Function											
MSE	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
cross-entropy	Y	Y	Y	Y	N	Y	Y	Y	Y	Y	Y
maximum likelihood estimation (MLE)	Y	Y	Y	Y	N	Y	Y	Y	N	Y	Y
hinge loss	Y	N	Y	Y	N	Y	Y	N	N	Y	Y
minimize classification error	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
maximize reward	Y	Y	Y	Y	N	Y	Y	Y	N	N	Y
information gain	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
joint probability distribution	Y	N	Y	Y	N	Y	Y	Y	N	Y	Y
Gini impurity	Y	Y	N	Y	Y	Y	N	Y	Y	N	Y
binary cross-entropy (log loss)	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
masked language model (MLM)	Y	N	Y	Y	N	Y	N	Y	N	Y	N
focal loss	Y	N	N	Y	N	Y	N	Y	N	Y	N
reconstruction error	Y	Y	Y	Y	N	Y	Y	Y	N	Y	N
minimize sum of distances	Y	N	Y	Y	Y	Y	Y	Y	Y	Y	N
temporal difference error	Y	N	N	N	N	Y	Y	N	N	N	Y
adversarial loss	Y	Y	Y	Y	N	Y	Y	Y	N	Y	N
task-specific objectives	Y	N	N	N	N	Y	Y	Y	N	Y	N
maximum a posteriori (MAP) estimation	Y	N	N	Y	N	Y	Y	Y	N	Y	N
minimize weighted sum	Y	N	N	Y	N	Y	Y	Y	Y	Y	Y
preserve topological relationships	N	N	N	N	N	Y	Y	Y	Y	N	N
similarity measure	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
gain ratio	Y	N	Y	Y	N	Y	N	N	N	N	Y
deviance	Y	N	N	Y	N	Y	N	Y	N	N	Y
variance maximization	Y	N	N	Y	N	Y	Y	Y	Y	Y	Y
KL divergence	Y	N	Y	Y	N	Y	Y	Y	Y	Y	N
minimize quantization error	Y	N	N	N	N	Y	Y	Y	N	Y	Y
minimize regression error	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
induce logic rules	N	N	N	Y	N	Y	N	N	N	N	N
dictionary elements	Y	N	N	Y	N	Y	N	N	N	Y	N
evolve population of solutions	N	N	N	N	N	Y	Y	N	N	Y	N
gradient penalty	Y	N	N	Y	N	Y	Y	Y	Y	Y	N
minimax	Y	N	Y	N	N	Y	Y	Y	N	Y	Y
replaced token detection	N	N	N	Y	N	Y	N	Y	N	N	N
maximum entropy	Y	N	Y	Y	N	Y	Y	Y	N	Y	Y
Learning Rate Type											
fixed (specific) learning rate	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
adaptive (adjustable) learning rate	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
Number of Hidden Layers											
shallow (single layer)	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
multiple	Y	Y	Y	Y	Y	Y	Y	Y	N	Y	Y
deep	Y	Y	Y	Y	Y	Y	Y	Y	N	Y	Y
various (shallow to deep)	Y	Y	Y	Y	Y	Y	Y	Y	N	Y	Y
dynamically grows	Y	N	N	N	N	Y	N	Y	N	N	N

Feature recognized by expert	1	2	3	4	5	6	7	8	9	10	11
Model Size											
small	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
small to moderate	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
moderate	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
moderate to large	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
large	Y	Y	Y	Y	Y	Y	Y	Y	N	Y	Y
various	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
dynamically grows	Y	N	N	N	N	Y	N	Y	N	N	N
Transfer Learning											
common pre-trained model	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
uncommon pre-trained model	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
fine-tuning	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
Kernel Type											
convolutional	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
radial basis function	Y	N	N	N	N	Y	Y	Y	N	Y	Y
linear	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
polynomial	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
gaussian	Y	Y	Y	Y	N	Y	Y	Y	Y	Y	Y
matern	N	N	N	N	N	Y	N	N	N	N	N
auto-associative	N	N	N	N	N	Y	N	N	N	N	N
Input Data Type											
various	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
structured	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
images (or other grid based)	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
time series	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
text	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
categorical (numerical)	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
unstructured	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
state action pairs and rewards	Y	N	Y	Y	N	Y	Y	N	N	N	Y
audio	Y	Y	Y	Y	N	Y	Y	N	Y	N	N
transactional data	Y	N	N	Y	Y	Y	N	N	N	Y	N
AutoML vs. Custom											
AutoML	Y	N	N	Y	Y	Y	Y	Y	Y	N	N
custom	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y

Figure D.4: Showcases the familiarity of each participant with all of our machine learning features.

References

[1] Fabiano Dalpiaz and Nan Niu. Requirements engineering in the days of artificial intelligence. *IEEE software*, 37(4):7–10, 2020.

[2] Pedro Domingos. A few useful things to know about machine learning. *Communications of the ACM*, 55(10):78–87, 2012.

[3] Peter L Bartlett, Andrea Montanari, and Alexander Rakhlin. Deep learning: a statistical viewpoint. *Acta numerica*, 30:87–201, 2021.

[4] Shai Shalev-Shwartz and Shai Ben-David. *Understanding machine learning: From theory to algorithms*. Cambridge university press, 2014.

[5] David E Rumelhart, Geoffrey E Hinton, and Ronald J Williams. Learning representations by back-propagating errors. *nature*, 323(6088):533–536, 1986.

[6] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. *Advances in neural information processing systems*, 30, 2017.

[7] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton. Imagenet classification with deep convolutional neural networks. *Advances in neural information processing systems*, 25, 2012.

[8] Rich Caruana and Alexandru Niculescu-Mizil. An empirical comparison of supervised learning algorithms. In *Proceedings of the 23rd international conference on Machine learning*, pages 161–168, 2006.

[9] Isabelle Guyon and André Elisseeff. An introduction to variable and feature selection. *Journal of machine learning research*, 3(Mar):1157–1182, 2003.

[10] Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014.

[11] Yoshua Bengio et al. Learning deep architectures for ai. *Foundations and trends® in Machine Learning*, 2(1):1–127, 2009.

[12] Stuart Lloyd. Least squares quantization in pcm. *IEEE transactions on information theory*, 28(2):129–137, 1982.

[13] Congying Xia, Chenwei Zhang, Xiaohui Yan, Yi Chang, and Philip S Yu. Zero-shot user intent detection via capsule neural networks. *arXiv preprint arXiv:1809.00385*, 2018.

[14] Yujia Li, Oriol Vinyals, Chris Dyer, Razvan Pascanu, and Peter Battaglia. Learning deep generative models of graphs. *arXiv preprint arXiv:1803.03324*, 2018.

[15] Lex Poon, Siamak Farshidi, Na Li, and Zhiming Zhao. Unsupervised anomaly detection in data quality control. In *2021 IEEE International Conference on Big Data (Big Data)*, pages 2327–2336. IEEE, 2021.

[16] M Emre Celebi, Hassan A Kingravi, and Patricio A Vela. A comparative study of efficient initialization methods for the k-means clustering algorithm. *Expert systems with applications*, 40(1):200–210, 2013.

[17] David Silver, Thomas Hubert, Julian Schrittwieser, Ioannis Antonoglou, Matthew Lai, Arthur Guez, Marc Lanctot, Laurent Sifre, Dharshan Kumaran, Thore Graepel, et al. Mastering chess and shogi by self-play with a general reinforcement learning algorithm. *arXiv preprint arXiv:1712.01815*, 2017.

[18] Mehdi Mohammadi, Ala Al-Fuqaha, Sameh Sorour, and Mohsen Guizani. Deep learning for iot big data and streaming analytics: A survey. *IEEE Communications Surveys & Tutorials*, 20(4):2923–2960, 2018.

[19] Yoshua Bengio. Practical recommendations for gradient-based training of deep architectures. In *Neural networks: Tricks of the trade: Second edition*, pages 437–478. Springer, 2012.

[20] Molnar Christoph. *Interpretable machine learning: A guide for making black box models explainable*. Leanpub, 2020.

[21] José Hernández-Orallo, Peter Flach, and César Ferri Ramírez. A unified view of performance metrics: Translating threshold choice into expected classification loss. *Journal of Machine Learning Research*, 13:2813–2869, 2012.

[22] Arthur E Hoerl and Robert W Kennard. Ridge regression: Biased estimation for nonorthogonal problems. *Technometrics*, 12(1):55–67, 1970.

[23] Ted Dunning and Ellen Friedman. *Practical machine learning: a new look at anomaly detection*. " O'Reilly Media, Inc.", 2014.

[24] Omar Y Al-Jarrah, Paul D Yoo, Sami Muhaidat, George K Karagiannidis, and Kamal Taha. Efficient machine learning for big data: A review. *Big Data Research*, 2(3):87–93, 2015.

[25] M.I. Jordan and T.M. Mitchell. Machine learning: Trends, perspectives, and prospects. *Science*, 349(6245):255–260, 2015.

[26] Sebastian Raschka. Model evaluation, model selection, and algorithm selection in machine learning. *University of Wisconsin-Madison, Department of Statistics*, 2018.

[27] Chiyuan Zhang, Samy Bengio, Moritz Hardt, Benjamin Recht, and Oriol Vinyals. Understanding deep learning (still) requires rethinking generalization. *Communications of the ACM*, 64(3):107–115, 2021.

[28] Sendhil Mullainathan and Jann Spiess. Machine learning: an applied econometric approach. *Journal of Economic Perspectives*, 31(2):87–106, 2017.

[29] Michele Sebag. A tour of machine learning: An ai perspective. *AI Communications*, 27(1):11–23, 2014.

[30] Abhinav Jain, Hima Patel, Lokesh Nagalapatti, Nitin Gupta, Sameep Mehta, Shanmukha Guttula, Shashank Mujumdar, Shazia Afzal, Ruhi Sharma Mittal, and Vitobha Munigala. Overview and importance of data quality for machine learning tasks. In *Proceedings of the 26th ACM SIGKDD international conference on knowledge discovery & data mining*, pages 3561–3562, 2020.

[31] Alon Halevy, Peter Norvig, and Fernando Pereira. The unreasonable effectiveness of data. *IEEE intelligent systems*, 24(2):8–12, 2009.

[32] Tong Yu and Hong Zhu. Hyper-parameter optimization: A review of algorithms and applications. *arXiv preprint arXiv:2003.05689*, 2020.

[33] Hussain Alibrahim and Simone A Ludwig. Hyperparameter optimization: Comparing genetic algorithm against grid search and bayesian optimization. In *2021 IEEE Congress on Evolutionary Computation (CEC)*, pages 1551–1559. IEEE, 2021.

[34] Lutz Prechelt. Early stopping-but when? In *Neural Networks: Tricks of the trade*, pages 55–69. Springer, 2002.

[35] Daniel Berrar et al. Cross-validation., 2019.

[36] Bangti Jin and Peter Maass. Sparsity regularization for parameter identification problems. *Inverse Problems*, 28(12):123001, 2012.

[37] Nitish Srivastava, Geoffrey Hinton, Alex Krizhevsky, Ilya Sutskever, and Ruslan Salakhutdinov. Dropout: a simple way to prevent neural networks from overfitting. *The journal of machine learning research*, 15(1):1929–1958, 2014.

[38] Geoffrey Hinton, Oriol Vinyals, and Jeff Dean. Distilling the knowledge in a neural network. *arXiv preprint arXiv:1503.02531*, 2015.

[39] Kaiyuan Guo, Shulin Zeng, Jincheng Yu, Yu Wang, and Huazhong Yang. [dl] a survey of fpga-based neural network inference accelerators. *ACM Transactions on Reconfigurable Technology and Systems (TRETS)*, 12(1):1–26, 2019.

[40] TensorFlow Team. Tensorflow: Large-scale machine learning on heterogeneous systems, 2015. *URL http://tensorflow. org/. Software available from tensorflow. org*, 2015.

[41] Masashi Sugiyama and Motoaki Kawanabe. *Machine learning in non-stationary environments: Introduction to covariate shift adaptation*. MIT press, 2012.

[42] Peter Flach. Performance evaluation in machine learning: the good, the bad, the ugly, and the way forward. In *Proceedings of the AAAI conference on artificial intelligence*, volume 33, pages 9808–9814, 2019.

[43] Michael J Kearns. *The computational complexity of machine learning*. MIT press, 1990.

[44] Weipeng Zhou and Gang Luo. Parameter sensitivity analysis for the progressive sampling-based bayesian optimization method for automated machine learning model selection. In *VLDB Workshop on Data Management and Analytics for Medicine and Healthcare*, pages 213–227. Springer, 2020.

[45] Yarens J Cruz, Marcelino Rivas, Ramón Quiza, Rodolfo E Haber, Fernando Castaño, and Alberto Villalonga. A two-step machine learning approach for dynamic model selection: A case study on a micro milling process. *Computers in Industry*, 143:103764, 2022.

[46] Gary A Klein and Roberta Calderwood. Decision models: Some lessons from the field. *IEEE Transactions on systems, man, and cybernetics*, 21(5):1018–1026, 1991.

[47] José Hernández-Orallo, Peter Flach, and César Ferri Ramírez. A unified view of performance metrics: Translating threshold choice into expected classification loss. *Journal of Machine Learning Research*, 13:2813–2869, 2012.

[48] Eric Mjolsness and Dennis DeCoste. Machine learning for science: state of the art and future prospects. *science*, 293(5537):2051–2055, 2001.

[49] Robert Philipp, Andreas Mladenow, Christine Strauss, and Alexander Völz. Machine learning as a service: Challenges in research and applications. In *Proceedings of the 22nd International Conference on Information Integration and Web-based Applications & Services*, pages 396–406, 2020.

[50] Omer Sagi and Lior Rokach. Ensemble learning: A survey. *Wiley interdisciplinary reviews: data mining and knowledge discovery*, 8(4):e1249, 2018.

[51] David Nijssen and Andreas H Schumann. Dealing with insufficient data: metadata modelling and stochastic exploration of the decision space. 2012.

[52] Wallace E Larimore. Accuracy confidence bands including the bias of model under-fitting. In *The Modeling of Uncertainty in Control Systems: Proceedings of the 1992 Santa Barbara Workshop*, pages 275–287. Springer, 2005.

[53] Douglas M Hawkins. The problem of overfitting. *Journal of chemical information and computer sciences*, 44(1):1–12, 2004.

[54] Tom Chávez and Ross D Shachter. Decision flexibility. *arXiv preprint arXiv:1302.4937*, 2013.

[55] Abigail R Colson and Roger M Cooke. Expert elicitation: using the classical model to validate experts' judgments. *Review of Environmental Economics and Policy*, 2018.

[56] Ze Hu, Zhan Zhang, Haiqin Yang, Qing Chen, and Decheng Zuo. A deep learning approach for predicting the quality of online health expert question-answering services. *Journal of biomedical informatics*, 71:241–253, 2017.

[57] Steven J Rigatti. Random forest. *Journal of Insurance Medicine*, 47(1):31–39, 2017.

[58] Geoffrey I Webb, Eamonn Keogh, and Risto Miikkulainen. Naïve bayes. *Encyclopedia of machine learning*, 15(1):713–714, 2010.

[59] Antonia Creswell, Tom White, Vincent Dumoulin, Kai Arulkumaran, Biswa Sengupta, and Anil A Bharath. Generative adversarial networks: An overview. *IEEE signal processing magazine*, 35(1):53–65, 2018.

[60] Andrew James Smith. Applications of the self-organising map to reinforcement learning. *Neural networks*, 15(8-9):1107–1124, 2002.

[61] Leonardo Noriega. Multilayer perceptron tutorial. *School of Computing. Staffordshire University*, 4(5):444, 2005.

[62] Jafar Alzubi, Anand Nayyar, and Akshi Kumar. Machine learning from theory to algorithms: an overview. In *Journal of physics: conference series*, volume 1142, page 012012. IOP Publishing, 2018.

[63] Avrim L Blum and Pat Langley. Selection of relevant features and examples in machine learning. *Artificial intelligence*, 97(1-2):245–271, 1997.

[64] Fannia Pacheco, Ernesto Exposito, Mathieu Gineste, Cedric Baudoin, and Jose Aguilar. Towards the deployment of machine learning solutions in network traffic classification: A systematic survey. *IEEE Communications Surveys & Tutorials*, 21(2):1988–2014, 2018.

[65] Sotiris B Kotsiantis, Ioannis D Zaharakis, and Panayiotis E Pintelas. Machine learning: a review of classification and combining techniques. *Artificial Intelligence Review*, 26:159–190, 2006.

[66] Sunita B Aher and LMRJ Lobo. Combination of machine learning algorithms for recommendation of courses in e-learning system based on historical data. *Knowledge-Based Systems*, 51:1–14, 2013.

[67] Patrick Bajari, Denis Nekipelov, Stephen P Ryan, and Miaoyu Yang. Demand estimation with machine learning and model combination. Technical report, National Bureau of Economic Research, 2015.

[68] Babatunde K Olorisade, Pearl Brereton, and Peter Andras. Reproducibility in machine learning-based studies: An example of text mining. 2017.

[69] Frank Hutter, Lars Kotthoff, and Joaquin Vanschoren. *Automated machine learning: methods, systems, challenges*. Springer Nature, 2019.

[70] Berk Ustun and Cynthia Rudin. Methods and models for interpretable linear classification. *arXiv preprint arXiv:1405.4047*, 2014.

[71] Raisa Abedin Disha and Sajjad Waheed. Performance analysis of machine learning models for intrusion detection system using gini impurity-based weighted random forest (giwrf) feature selection technique. *Cybersecurity*, 5(1):1, 2022.

[72] Akshit Kamboj, Priyanshu Kumar, Amit Kumar Bairwa, and Sandeep Joshi. Detection of malware in downloaded files using various machine learning models. *Egyptian Informatics Journal*, 24(1):81–94, 2023.

[73] Elena Dumitrescu, Sullivan Hué, Christophe Hurlin, and Sessi Tokpavi. Machine learning for credit scoring: Improving logistic regression with non-linear decision-tree effects. *European Journal of Operational Research*, 297(3):1178–1192, 2022.

[74] Jason H Moore, Folkert W Asselbergs, and Scott M Williams. Bioinformatics challenges for genome-wide association studies. *Bioinformatics*, 26(4):445–455, 2010.

[75] Mariette Awad, Rahul Khanna, Mariette Awad, and Rahul Khanna. Support vector regression. *Efficient learning machines: Theories, concepts, and applications for engineers and system designers*, pages 67–80, 2015.

[76] Sourish Ghosh, Anasuya Dasgupta, and Aleena Swetapadma. A study on support vector machine based linear and non-linear pattern classification. In *2019 International Conference on Intelligent Sustainable Systems (ICISS)*, pages 24–28. IEEE, 2019.

[77] Michael Frank, Dimitris Drikakis, and Vassilis Charissis. Machine-learning methods for computational science and engineering. *Computation*, 8(1):15, 2020.

[78] Finale Doshi-Velez and Been Kim. Towards a rigorous science of interpretable machine learning. *arXiv preprint arXiv:1702.08608*, 2017.

[79] Thomas Serre. Deep learning: the good, the bad, and the ugly. *Annual review of vision science*, 5(1):399–426, 2019.

[80] Barbara Kitchenham, O Pearl Brereton, David Budgen, Mark Turner, John Bailey, and Stephen Linkman. Systematic literature reviews in software engineering–a systematic literature review. *Information and software technology*, 51(1):7–15, 2009.

[81] Yu Xiao and Maria Watson. Guidance on conducting a systematic literature review. *Journal of planning education and research*, 39(1):93–112, 2019.

[82] Chitu Okoli and Kira Schabram. A guide to conducting a systematic literature review of information systems research. 2015.

[83] Robert K Yin. *Case study research: Design and methods*, volume 5. sage, 2009.

[84] Bartosz Krawczyk. Learning from imbalanced data: open challenges and future directions. *Progress in artificial intelligence*, 5(4):221–232, 2016.

[85] Richard Y Wang, Henry B Kon, and Stuart E Madnick. Data quality requirements analysis and modeling. In *Proceedings of IEEE 9th International Conference on Data Engineering*, pages 670–677. IEEE, 1993.

[86] Jens Bleiholder and Felix Naumann. Data fusion. *ACM computing surveys (CSUR)*, 41(1):1–41, 2009.

[87] Yifeng Li and Alioune Ngom. Data integration in machine learning. In *2015 IEEE International Conference on Bioinformatics and Biomedicine (BIBM)*, pages 1665–1671. IEEE, 2015.

[88] Michal S Gal and Daniel L Rubinfeld. Data standardization. *NYUL Rev.*, 94:737, 2019.

[89] Siamak Farshidi. *Multi-criteria decision-making in software production.* PhD thesis, Utrecht University, 2020.

[90] Pengzhi Li, Yan Pei, and Jianqiang Li. A comprehensive survey on design and application of autoencoder in deep learning. *Applied Soft Computing*, 138:110176, 2023.

[91] Ahmed Tealab. Time series forecasting using artificial neural networks methodologies: A systematic review. *Future Computing and Informatics Journal*, 3(2):334–340, 2018.

[92] Douglas C Montgomery, Cheryl L Jennings, and Murat Kulahci. *Introduction to time series analysis and forecasting.* John Wiley & Sons, 2015.

[93] Martin Anthony, Peter L Bartlett, Peter L Bartlett, et al. *Neural network learning: Theoretical foundations*, volume 9. cambridge university press Cambridge, 1999.

[94] Mensah Kwabena Patrick, Adebayo Felix Adekoya, Ayidzoe Abra Mighty, and Baagyire Y Edward. Capsule networks–a survey. *Journal of King Saud University-computer and information sciences*, 34(1):1295–1310, 2022.

[95] Sana Aroussi and Abdelhamid Mellouk. Survey on machine learning-based qoe-qos correlation models. In *2014 International Conference on Computing, Management and Telecommunications (ComManTel)*, pages 200–204. IEEE, 2014.

[96] Jie M Zhang, Mark Harman, Lei Ma, and Yang Liu. Machine learning testing: Survey, landscapes and horizons. *IEEE Transactions on Software Engineering*, 48(1):1–36, 2020.

[97] Zhengjing Ma, Gang Mei, and Francesco Piccialli. Machine learning for landslides prevention: a survey. *Neural Computing and Applications*, 33:10881–10907, 2021.

[98] Neha Agarwal, Geeta Sikka, and Lalit Kumar Awasthi. A systematic literature review on web service clustering approaches to enhance service discovery, selection and recommendation. *Computer Science Review*, 45:100498, 2022.

[99] Amy X Zhang, Michael Muller, and Dakuo Wang. How do data science workers collaborate? roles, workflows, and tools. *Proceedings of the ACM on Human-Computer Interaction*, 4(CSCW1):1–23, 2020.

[100] Suejb Memeti, Sabri Pllana, Alécio Binotto, Joanna Kołodziej, and Ivona Brandic. Using meta-heuristics and machine learning for software optimization of parallel computing systems: a systematic literature review. *Computing*, 101:893–936, 2019.

[101] Victoria Zakharova and Alena Suvorova. Social aspects of machine learning model evaluation: Model interpretation and justification from ml-practitioners' perspective. In *IMS*, pages 230–234, 2021.

[102] Lei Cui, Youyang Qu, Longxiang Gao, Gang Xie, and Shui Yu. Detecting false data attacks using machine learning techniques in smart grid: A survey. *Journal of Network and Computer Applications*, 170:102808, 2020.

[103] Zuhaira Muhammad Zain, Sapiah Sakri, and Nurul Halimatul Asmak Ismail. Application of deep learning in software defect prediction: systematic literature review and meta-analysis. *Information and Software Technology*, 158:107175, 2023.

[104] TK Balaji, Chandra Sekhara Rao Annavarapu, and Annushree Bablani. Machine learning algorithms for social media analysis: A survey. *Computer Science Review*, 40:100395, 2021.

[105] Naira Kaieski, Cristiano Andre Da Costa, Rodrigo da Rosa Righi, Priscila Schmidt Lora, and Bjoern Eskofier. Application of artificial intelligence methods in vital signs analysis of hospitalized patients: A systematic literature review. *Applied Soft Computing*, 96:106612, 2020.

[106] Manal Binkhonain and Liping Zhao. A review of machine learning algorithms for identification and classification of non-functional requirements. *Expert Systems with Applications: X*, 1:100001, 2019.

[107] Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, et al. Language models are few-shot learners. *Advances in neural information processing systems*, 33:1877–1901, 2020.

[108] Alec Radford, Karthik Narasimhan, Tim Salimans, Ilya Sutskever, et al. Improving language understanding by generative pre-training. 2018.

[109] Gemini Team, Rohan Anil, Sebastian Borgeaud, Yonghui Wu, Jean-Baptiste Alayrac, Jiahui Yu, Radu Soricut, Johan Schalkwyk, Andrew M Dai, Anja Hauth, et al. Gemini: a family of highly capable multimodal models. *arXiv preprint arXiv:2312.11805*, 2023.

[110] Fabio Petroni, Tim Rocktäschel, Patrick Lewis, Anton Bakhtin, Yuxiang Wu, Alexander H Miller, and Sebastian Riedel. Language models as knowledge bases? *arXiv preprint arXiv:1909.01066*, 2019.

[111] Sofus Macskassy and Suzanne Stevenson. A conversational agent. *Master Essay, Rutgers University*, 1996.

[112] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*, 2018.

[113] Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, and Peter J Liu. Exploring the limits of transfer learning with a unified text-to-text transformer. *Journal of machine learning research*, 21(140):1–67, 2020.

[114] Warren J Von Eschenbach. Transparency and the black box problem: Why we do not trust ai. *Philosophy & Technology*, 34(4):1607–1622, 2021.

[115] Yang Liu, Jiahuan Cao, Chongyu Liu, Kai Ding, and Lianwen Jin. Datasets for large language models: A comprehensive survey. *arXiv preprint arXiv:2402.18041*, 2024.

[116] Dieuwertje Luitse and Wiebke Denkena. The great transformer: Examining the role of large language models in the political economy of ai. *Big Data & Society*, 8(2):20539517211047734, 2021.

[117] Lingjiao Chen, Matei Zaharia, and James Zou. Frugalgpt: How to use large language models while reducing cost and improving performance. *arXiv preprint arXiv:2305.05176*, 2023.

[118] Brian A Jacob and Lars Lefgren. The impact of research grant funding on scientific productivity. *Journal of public economics*, 95(9-10):1168–1177, 2011.

[119] Runhua Xu, Nathalie Baracaldo, and James Joshi. Privacy-preserving machine learning: Methods, challenges and directions. *arXiv preprint arXiv:2108.04417*, 2021.

[120] Shervin Minaee, Tomas Mikolov, Narjes Nikzad, Meysam Chenaghlu, Richard Socher, Xavier Amatriain, and Jianfeng Gao. Large language models: A survey. *arXiv preprint arXiv:2402.06196*, 2024.

[121] Thorsten Wuest, Daniel Weimer, Christopher Irgens, and Klaus-Dieter Thoben. Machine learning in manufacturing: advantages, challenges, and applications. *Production & Manufacturing Research*, 4(1):23–45, 2016.

[122] Rahul Rai, Manoj Kumar Tiwari, Dmitry Ivanov, and Alexandre Dolgui. Machine learning in manufacturing and industry 4.0 applications, 2021.

[123] David Bollier, Charles M Firestone, et al. *The promise and peril of big data*. Aspen Institute, Communications and Society Program Washington, DC, 2010.

[124] Scott M Lundberg and Su-In Lee. A unified approach to interpreting model predictions. *Advances in neural information processing systems*, 30, 2017.

[125] Sebastian Raschka. Model evaluation, model selection, and algorithm selection in machine learning. *arXiv preprint arXiv:1811.12808*, 2018.

[126] Shai Shalev-Shwartz and Shai Ben-David. *Understanding machine learning: From theory to algorithms*. Cambridge university press, 2014.

[127] Arun Kumar, Matthias Boehm, and Jun Yang. Data management in machine learning: Challenges, techniques, and systems. In *Proceedings of the 2017 ACM International Conference on Management of Data*, pages 1717–1722, 2017.

[128] Siamak Farshidi, Xiaofeng Liao, Na Li, Doron Goldfarb, Barbara Magagna, Markus Stocker, Keith Jeffery, Peter Thijsse, Christian Pichot, Andreas Petzold, et al. Knowledge sharing and discovery across heterogeneous research infrastructures. *Open Research Europe*, 1:68, 2021.

[129] Pooja Rani, Rajneesh Kumar, Anurag Jain, and Rohit Lamba. Taxonomy of machine learning algorithms and its applications. *Journal of Computational and Theoretical Nanoscience*, 17(6):2508–2513, 2020.

[130] Sheng-Xiang Lv, Lu Peng, Huanling Hu, and Lin Wang. Effective machine learning model combination based on selective ensemble strategy for time series forecasting. *Information Sciences*, 612:994–1023, 2022.

[131] Mengye Ren, Eleni Triantafillou, Sachin Ravi, Jake Snell, Kevin Swersky, Joshua B Tenenbaum, Hugo Larochelle, and Richard S Zemel. Meta-learning for semi-supervised few-shot classification. *arXiv preprint arXiv:1803.00676*, 2018.

[132] Rahman Ali, Sungyoung Lee, and Tae Choong Chung. Accurate multi-criteria decision making methodology for recommending machine learning algorithm. *Expert Systems with Applications*, 71:257–278, 2017.

[133] Peter Flach and Meelis Kull. Precision-recall-gain curves: Pr analysis done right. *Advances in neural information processing systems*, 28, 2015.

[134] Mebus Geyh. Selection of suitable data sets improves th/u dates of dirty material. *Geochronometria*, 30(1):69–77, 2008.

[135] Dodo Zaenal Abidin, Siti Nurmaini, Reza Firsandava Malik, Errissya Rasywir, Yovi Pratama, et al. Rssi data preparation for machine learning. In *2020 International Conference on Informatics, Multimedia, Cyber and Information System (ICIMCIS)*, pages 284–289. IEEE, 2020.

[136] Joaquin Vanschoren. Meta-learning: A survey. *arXiv preprint arXiv:1810.03548*, 2018.

[137] Slinger Jansen. Applied multi-case research in a mixed-method research project: Customer configuration updating improvement. In *Information systems research methods, epistemology, and applications*, pages 120–139. IGI Global, 2009.

[138] R Burke Johnson and Anthony J Onwuegbuzie. Mixed methods research: A research paradigm whose time has come. *Educational researcher*, 33(7):14–26, 2004.

[139] Michael Meuser and Ulrike Nagel. The expert interview and changes in knowledge production. In *Interviewing experts*, pages 17–42. Springer, 2009.

[140] Alexander Bogner, Beate Littig, and Wolfgang Menz. Introduction: Expert interviews—an introduction to a new methodological debate. In *Interviewing experts*, pages 1–13. Springer, 2009.

[141] Peta Darke, Graeme Shanks, and Marianne Broadbent. Successfully completing case study research: combining rigour, relevance and pragmatism. *Information systems journal*, 8(4):273–289, 1998.

[142] David Fortus, Joseph Krajcik, Ralph Charles Dershimer, Ronald W Marx, and Rachel Mamlok-Naaman. Design-based science and real-world problem-solving. *International Journal of Science Education*, 27(7):855–879, 2005.

[143] Siamak Farshidi, Kiyan Rezaee, Sara Mazaheri, Amir Hossein Rahimi, Ali Dadashzadeh, Morteza Ziabakhsh, Sadegh Eskandari, and Slinger Jansen. Understanding user intent modeling for conversational recommender systems: a systematic literature review. *User Modeling and User-Adapted Interaction*, pages 1–64, 2024.

[144] Angel Temelko, Fang Hou, Siamak Farshidi, and Slinger Jansen. Systematic literature review of the trust reinforcement mechanisms exist in package ecosystems. *arXiv preprint arXiv:2407.02522*, 2024.

[145] Adam Kilgarriff, Vít Baisa, Jan Bušta, Miloš Jakubíček, Vojtěch Kovář, Jan Michelfeit, Pavel Rychlỳ, and Vít Suchomel. The sketch engine: ten years on. *Lexicography*, 1(1):7–36, 2014.

[146] Julian PT Higgins, Sally Green, et al. Cochrane handbook for systematic reviews of interventions. 2008.

[147] Jia Shi, Gang He, and Xinwen Liu. Anomaly detection for key performance indicators through machine learning. In *2018 International Conference on Network Infrastructure and Digital Content (IC-NIDC)*, pages 1–5. IEEE, 2018.

[148] Jürgen Schmidhuber, Sepp Hochreiter, et al. Long short-term memory. *Neural Comput*, 9(8):1735–1780, 1997.

[149] Lauren Z Atkinson and Andrea Cipriani. How to carry out a literature search for a systematic review: a practical guide. *BJPsych Advances*, 24(2):74–82, 2018.

[150] Robert Tomaszewski. A study of citations to stem databases: Acm digital library, engineering village, ieee xplore, and mathscinet. *Scientometrics*, 126(2):1797–1811, 2021.

[151] Iqbal H Sarker. Machine learning: Algorithms, real-world applications and research directions. *SN computer science*, 2(3):160, 2021.

[152] Alexey Natekin and Alois Knoll. Gradient boosting machines, a tutorial. *Frontiers in neurorobotics*, 7:21, 2013.

[153] Yu Shi, Jian Li, and Zhize Li. Gradient boosting with piece-wise linear regression trees. *arXiv preprint arXiv:1802.05640*, 2018.

[154] Richard Zemel and Toniann Pitassi. A gradient-based boosting algorithm for regression problems. *Advances in neural information processing systems*, 13, 2000.

[155] Senyan Yang, Jianping Wu, Yiman Du, Yingqi He, and Xu Chen. Ensemble learning for short-term traffic prediction based on gradient boosting machine. *Journal of Sensors*, 2017(1):7074143, 2017.

[156] Sam T Roweis and Lawrence K Saul. Nonlinear dimensionality reduction by locally linear embedding. *science*, 290(5500):2323–2326, 2000.

[157] Christopher JC Burges. A tutorial on support vector machines for pattern recognition. *Data mining and knowledge discovery*, 2(2):121–167, 1998.

[158] Nitish Srivastava, Geoffrey Hinton, Alex Krizhevsky, Ilya Sutskever, and Ruslan Salakhutdinov. Dropout: a simple way to prevent neural networks from overfitting. *The journal of machine learning research*, 15(1):1929–1958, 2014.

[159] C-C Jay Kuo. Understanding convolutional neural networks with a mathematical model. *Journal of Visual Communication and Image Representation*, 41:406–413, 2016.

[160] Xavier Glorot, Antoine Bordes, and Yoshua Bengio. Deep sparse rectifier neural networks. In *Proceedings of the fourteenth international conference on artificial intelligence and statistics*, pages 315–323. JMLR Workshop and Conference Proceedings, 2011.

[161] Zhou Wang and Alan C Bovik. Mean squared error: Love it or leave it? a new look at signal fidelity measures. *IEEE signal processing magazine*, 26(1):98–117, 2009.

[162] Leslie N Smith. Cyclical learning rates for training neural networks. In *2017 IEEE winter conference on applications of computer vision (WACV)*, pages 464–472. IEEE, 2017.

[163] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778, 2016.

[164] Usama Fayyad, Gregory Piatetsky-Shapiro, and Padhraic Smyth. From data mining to knowledge discovery in databases. *AI magazine*, 17(3):37–37, 1996.

[165] Mohiuddin Ahmed, Raihan Seraj, and Syed Mohammed Shamsul Islam. The k-means algorithm: A comprehensive survey and performance evaluation. *Electronics*, 9(8):1295, 2020.

[166] Robert H Shumway and Stoffer. Arima models. *Time series analysis and its applications: with R examples*, pages 75–163, 2017.

[167] Kuzman Ganchev, Joao Graca, John Blitzer, and Ben Taskar. Multi-view learning over structured and non-identical outputs. *arXiv preprint arXiv:1206.3256*, 2012.

[168] Oliver Kramer. K-nearest neighbors. *Dimensionality reduction with unsupervised nearest neighbors*, pages 13–23, 2013.

[169] Yan-Yan Song and LU Ying. Decision tree methods: applications for classification and prediction. *Shanghai archives of psychiatry*, 27(2):130, 2015.

[170] Michael P LaValley. Logistic regression. *Circulation*, 117(18):2395–2399, 2008.

[171] Karen Simonyan and Andrew Zisserman. Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556*, 2014.

[172] Larry R Medsker and LC Jain. Recurrent neural networks. *Design and Applications*, 5(64-67):2, 2001.

[173] Munazza Zaib, Wei Emma Zhang, Quan Z Sheng, Adnan Mahmood, and Yang Zhang. Conversational question answering: A survey. *Knowledge and Information Systems*, 64(12):3151–3195, 2022.

[174] Marco Tulio Ribeiro, Tongshuang Wu, Carlos Guestrin, and Sameer Singh. Beyond accuracy: Behavioral testing of nlp models with checklist. *arXiv preprint arXiv:2005.04118*, 2020.

[175] Jin Huang and Charles X Ling. Constructing new and better evaluation measures for machine learning. In *IJCAI*, pages 859–864, 2007.

[176] Andrew P Bradley. The use of the area under the roc curve in the evaluation of machine learning algorithms. *Pattern recognition*, 30(7):1145–1159, 1997.

[177] Charles Elkan. The foundations of cost-sensitive learning. In *International joint conference on artificial intelligence*, volume 17, pages 973–978. Lawrence Erlbaum Associates Ltd, 2001.

[178] Chris Burges, Tal Shaked, Erin Renshaw, Ari Lazier, Matt Deeds, Nicole Hamilton, and Greg Hullender. Learning to rank using gradient descent. In *Proceedings of the 22nd international conference on Machine learning*, pages 89–96, 2005.

[179] Songwon Seo. *A review and comparison of methods for detecting outliers in univariate data sets*. PhD thesis, University of Pittsburgh, 2006.

[180] Rongqi Pan, Mojtaba Bagherzadeh, Taher A Ghaleb, and Lionel Briand. Test case selection and prioritization using machine learning: a systematic literature review. *Empirical Software Engineering*, 27(2):29, 2022.

[181] Jinhui Yuan, Jianmin Li, and Bo Zhang. Learning concepts from large scale imbalanced data sets using support cluster machines. In *Proceedings of the 14th ACM international conference on Multimedia*, pages 441–450, 2006.

[182] Roberto V Zicari. Big data: Challenges and opportunities. *Big data computing*, 564:103, 2014.

[183] Liang Zhao, Sherif Sakr, Anna Liu, and Athman Bouguettaya. *Cloud data management*. Springer, 2014.

[184] Ziwei Zhang, Peng Cui, and Wenwu Zhu. Deep learning on graphs: A survey. *IEEE Transactions on Knowledge and Data Engineering*, 34(1):249–270, 2020.

[185] Alexander J Ratner, Christopher M De Sa, Sen Wu, Daniel Selsam, and Christopher Ré. Data programming: Creating large training sets, quickly. *Advances in neural information processing systems*, 29, 2016.

[186] Sebastian Vollmer, Bilal A Mateen, Gergo Bohner, Franz J Király, Rayid Ghani, Pall Jonsson, Sarah Cumbers, Adrian Jonas, Katherine SL McAllister, Puja Myles, et al. Machine learning and artificial intelligence research for patient benefit: 20 critical questions on transparency, replicability, ethics, and effectiveness. *bmj*, 368, 2020.

[187] Mark A Rothstein. Ethical issues in big data health research. *Journal of Law, Medicine & Ethics*, 43(2):425–429, 2015.

[188] Guang-Bin Huang, Qin-Yu Zhu, and Chee-Kheong Siew. Extreme learning machine: a new learning scheme of feedforward neural networks. In *2004 IEEE international joint conference on neural networks (IEEE Cat. No. 04CH37541)*, volume 2, pages 985–990. Ieee, 2004.

[189] Richard S Sutton. Learning to predict by the methods of temporal differences. *Machine learning*, 3:9–44, 1988.

[190] Stephen Muggleton and Luc De Raedt. Inductive logic programming: Theory and methods. *The Journal of Logic Programming*, 19:629–679, 1994.

[191] Jiao Wang, Si-wei Luo, and Xian-hua Zeng. A random subspace method for co-training. In *2008 IEEE International joint conference on neural networks (IEEE World Congress on Computational Intelligence)*, pages 195–200. IEEE, 2008.

[192] Salah Rifai, Pascal Vincent, Xavier Muller, Xavier Glorot, and Yoshua Bengio. Contractive auto-encoders: Explicit invariance during feature extraction. In *Proceedings of the 28th international conference on international conference on machine learning*, pages 833–840, 2011.

[193] Teuvo Kohonen. The self-organizing map. *Proceedings of the IEEE*, 78(9):1464–1480, 1990.

[194] Viacheslav Borovitskiy, Alexander Terenin, Peter Mostowsky, et al. Matérn gaussian processes on riemannian manifolds. *Advances in Neural Information Processing Systems*, 33:12426–12437, 2020.

[195] Bernhard Schölkopf, Alexander Smola, and Klaus-Robert Müller. Nonlinear component analysis as a kernel eigenvalue problem. *Neural computation*, 10(5):1299–1319, 1998.

[196] Kelvin Xu, Jimmy Ba, Ryan Kiros, Kyunghyun Cho, Aaron Courville, Ruslan Salakhudinov, Rich Zemel, and Yoshua Bengio. Show, attend and tell: Neural image caption generation with visual attention. In *International conference on machine learning*, pages 2048–2057. PMLR, 2015.

[197] Sebastian Ruder. An overview of gradient descent optimization algorithms. *arXiv preprint arXiv:1609.04747*, 2016.

[198] Isabelle Guyon, Amir Saffari, Gideon Dror, and Gavin Cawley. Model selection: beyond the bayesian/frequentist divide. *Journal of Machine Learning Research*, 11(1), 2010.

[199] ISO/IEC/IEEE. Systems and software engineering–architecture description. *ISO/IEC/IEEE 42010: 2011 (E)(Revision of ISO/IEC 42010: 2007 and IEEE Std 1471-2000)*, 2011:1–46, 2011.

[200] Rakesh Garg, Ramesh Kumar, and Sandhya Garg. Madm-based parametric selection and ranking of e-learning websites using fuzzy copras. *IEEE Transactions on Education*, 62(1):11–18, 2018.

[201] Lai Xu and Sjaak Brinkkemper. Concepts of product software. *European Journal of Information Systems*, 16:531–541, 2007.

[202] Siamak Farshidi, Izaak Beer Kwantes, and Slinger Jansen. Business process modeling language selection for research modelers. *Software and Systems Modeling*, 23(1):137–162, 2024.

[203] Dinora R Fitzgerald, Susan Mohammed, and Gül Okudan Kremer. Differences in the way we decide: The effect of decision style diversity on process conflict in design teams. *Personality and Individual Differences*, 104:339–344, 2017.

[204] Michael Doumpos and Evangelos Grigoroudis. *Multicriteria decision aid and artificial intelligence: links, theory and applications*. John Wiley & Sons, 2013.

[205] Mrinmoy Majumder and Mrinmoy Majumder. Multi criteria decision making. *Impact of urbanization on water shortage in face of climatic aberrations*, pages 35–47, 2015.

[206] Tatiana Kravchenko, Tatiana Bogdanova, and Timofey Shevgunov. Ranking requirements using moscow methodology in practice. In *Computer Science On-line Conference*, pages 188–199. Springer, 2022.

[207] Elena Baninemeh, Siamak Farshidi, and Slinger Jansen. A decision model for decentralized autonomous organization platform selection: Three industry case studies. *Blockchain: Research and Applications*, 4(2):100127, 2023.

[208] Edward L Kaplan and Paul Meier. Nonparametric estimation from incomplete observations. *Journal of the American statistical association*, 53(282):457–481, 1958.

[209] Jerome Friedman, Trevor Hastie, and Rob Tibshirani. Regularization paths for generalized linear models via coordinate descent. *Journal of statistical software*, 33(1):1, 2010.

[210] Gaël Varoquaux, Lars Buitinck, Gilles Louppe, Olivier Grisel, Fabian Pedregosa, and Andreas Mueller. Scikit-learn: Machine learning without learning the machinery. *GetMobile: Mobile Computing and Communications*, 19(1):29–33, 2015.

[211] Robert K Yin. *Case study research: Design and methods*, volume 5. sage, 2009.

[212] Da Zhang and Mansur R Kabuka. Protein family classification from scratch: a cnn based deep learning approach. *IEEE/ACM transactions on computational biology and bioinformatics*, 18(5):1996–2007, 2020.

[213] Bendong Zhao, Huanzhang Lu, Shangfeng Chen, Junliang Liu, and Dongya Wu. Convolutional neural networks for time series classification. *Journal of systems engineering and electronics*, 28(1):162–169, 2017.

[214] Gao Huang, Zhuang Liu, Laurens Van Der Maaten, and Kilian Q Weinberger. Densely connected convolutional networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 4700–4708, 2017.

[215] Yann N Dauphin, Angela Fan, Michael Auli, and David Grangier. Language modeling with gated convolutional networks. In *International conference on machine learning*, pages 933–941. PMLR, 2017.

[216] Shaojie Bai, J Zico Kolter, and Vladlen Koltun. An empirical evaluation of generic convolutional and recurrent networks for sequence modeling. *arXiv preprint arXiv:1803.01271*, 2018.

[217] Christian Szegedy, Wei Liu, Yangqing Jia, Pierre Sermanet, Scott Reed, Dragomir Anguelov, Dumitru Erhan, Vincent Vanhoucke, and Andrew Rabinovich. Going deeper with convolutions. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1–9, 2015.

[218] Barret Zoph, Vijay Vasudevan, Jonathon Shlens, and Quoc V Le. Learning transferable architectures for scalable image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 8697–8710, 2018.

[219] Huan Song, Deepta Rajan, Jayaraman Thiagarajan, and Andreas Spanias. Attend and diagnose: Clinical time series analysis using attention models. In *Proceedings of the AAAI conference on artificial intelligence*, volume 32, 2018.

[220] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Identity mappings in deep residual networks. In *Computer Vision–ECCV 2016: 14th European Conference, Amsterdam, The Netherlands, October 11–14, 2016, Proceedings, Part IV 14*, pages 630–645. Springer, 2016.

[221] Olaf Ronneberger, Philipp Fischer, and Thomas Brox. U-net: Convolutional networks for biomedical image segmentation. In *Medical image computing and computer-assisted intervention–MICCAI 2015: 18th international conference, Munich, Germany, October 5-9, 2015, proceedings, part III 18*, pages 234–241. Springer, 2015.

[222] Eitan Altman. *Constrained Markov decision processes*. Routledge, 2021.

[223] Long-Ji Lin. Self-improving reactive agents based on reinforcement learning, planning and teaching. *Machine learning*, 8:293–321, 1992.

[224] Adrian Sapio, Shuvra S Bhattacharyya, and Marilyn Wolf. Runtime adaptation in wireless sensor nodes using structured learning. *ACM Transactions on Cyber-Physical Systems*, 4(4):1–28, 2020.

[225] Volodymyr Mnih, Koray Kavukcuoglu, David Silver, Alex Graves, Ioannis Antonoglou, Daan Wierstra, and Martin Riedmiller. Playing atari with deep reinforcement learning. *arXiv preprint arXiv:1312.5602*, 2013.

[226] Ziyu Wang, Tom Schaul, Matteo Hessel, Hado Hasselt, Marc Lanctot, and Nando Freitas. Dueling network architectures for deep reinforcement learning. In *International conference on machine learning*, pages 1995–2003. PMLR, 2016.

[227] Hado Van Hasselt, Arthur Guez, and David Silver. Deep reinforcement learning with double q-learning. In *Proceedings of the AAAI conference on artificial intelligence*, volume 30, 2016.

[228] Ali Shakarami, Mostafa Ghobaei-Arani, and Ali Shahidinejad. A survey on the computation offloading approaches in mobile edge computing: A machine learning-based perspective. *Computer Networks*, 182:107496, 2020.

[229] Navindra Persaud, Peter McLeod, and Alan Cowey. Post-decision wagering objectively measures awareness. *Nature neuroscience*, 10(2):257–261, 2007.

[230] W Bradley Knox, Stephane Hatgis-Kessell, Serena Booth, Scott Niekum, Peter Stone, and Alessandro Allievi. Models of human preference for learning reward functions. *arXiv preprint arXiv:2206.02231*, 2022.

[231] Peter Auer, Nicolo Cesa-Bianchi, and Paul Fischer. Finite-time analysis of the multiarmed bandit problem. *Machine learning*, 47:235–256, 2002.

[232] Anagha Kulkarni, Siddharth Srivastava, and Subbarao Kambhampati. A unified framework for planning in adversarial and cooperative environments. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 33, pages 2479–2487, 2019.

[233] Christopher John Cornish Hellaby Watkins. Learning from delayed rewards. 1989.

[234] Sohom Ghosh. Identifying click baits using various machine learning and deep learning techniques. *International Journal of Information Technology*, 13(3):1235–1242, 2021.

[235] Tianqi Chen and Carlos Guestrin. Xgboost: A scalable tree boosting system. In *Proceedings of the 22nd acm sigkdd international conference on knowledge discovery and data mining*, pages 785–794, 2016.

[236] Guolin Ke, Qi Meng, Thomas Finley, Taifeng Wang, Wei Chen, Weidong Ma, Qiwei Ye, and Tie-Yan Liu. Lightgbm: A highly efficient gradient boosting decision tree. *Advances in neural information processing systems*, 30, 2017.

[237] Arthur Asuncion, Max Welling, Padhraic Smyth, and Yee Whye Teh. On smoothing and inference for topic models. *arXiv preprint arXiv:1205.2662*, 2012.

[238] Manal Binkhonain and Liping Zhao. A review of machine learning algorithms for identification and classification of non-functional requirements. *Expert Systems with Applications: X*, 1:100001, 2019.

[239] Peng Zhou, Wei Shi, Jun Tian, Zhenyu Qi, Bingchen Li, Hongwei Hao, and Bo Xu. Attention-based bidirectional long short-term memory networks for relation classification. In *Proceedings of the 54th annual meeting of the association for computational linguistics (volume 2: Short papers)*, pages 207–212, 2016.

[240] Kexin Huang, Jaan Altosaar, and Rajesh Ranganath. Clinicalbert: Modeling clinical notes and predicting hospital readmission. *arXiv preprint arXiv:1904.05342*, 2019.

[241] Elias Hossain, Rajib Rana, Niall Higgins, Jeffrey Soar, Prabal Datta Barua, Anthony R Pisani, and Kathryn Turner. Natural language processing in electronic health records in relation to healthcare decision-making: a systematic review. *Computers in biology and medicine*, 155:106649, 2023.

[242] Akshay Smit, Saahil Jain, Pranav Rajpurkar, Anuj Pareek, Andrew Y Ng, and Matthew P Lungren. Chexbert: combining automatic labelers and expert annotations for accurate radiology report labeling using bert. *arXiv preprint arXiv:2004.09167*, 2020.

[243] Tao Kong, Anbang Yao, Yurong Chen, and Fuchun Sun. Hypernet: Towards accurate region proposal generation and joint object detection. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 845–853, 2016.

[244] George Bebis and Michael Georgiopoulos. Feed-forward neural networks. *Ieee Potentials*, 13(4):27–31, 1994.

[245] Maggie Lei, Hao Li, Ji Li, Namrata Aundhkar, and Dae-Kyoo Kim. Deep learning application on code clone detection: A review of current knowledge. *Journal of Systems and Software*, 184:111141, 2022.

[246] Jin Wang, Hui Xiao, Shuwen Zhong, and Yinhao Xiao. Deepvulseeker: A novel vulnerability identification framework via code graph structure and pre-training mechanism. *Future Generation Computer Systems*, 148:15–26, 2023.

[247] Ira D Baxter, Andrew Yahin, Leonardo Moura, Marcelo Sant'Anna, and Lorraine Bier. Clone detection using abstract syntax trees. In *Proceedings. International Conference on Software Maintenance (Cat. No. 98CB36272)*, pages 368–377. IEEE, 1998.

[248] Frances E Allen. Control flow analysis. *ACM Sigplan Notices*, 5(7):1–19, 1970.

[249] Kavi, Buckles, and Bhat. A formal definition of data flow graph models. *IEEE Transactions on computers*, 100(11):940–948, 1986.

[250] RCH Cheng and TC Iles. Embedded models in three-parameter distributions and their estimation. *Journal of the Royal Statistical Society: Series B (Methodological)*, 52(1):135–149, 1990.

[251] Junhyun Lee, Inyeop Lee, and Jaewoo Kang. Self-attention graph pooling. In *International conference on machine learning*, pages 3734–3743. pmlr, 2019.

[252] Mohammed Kayed, Ahmed Anter, and Hadeer Mohamed. Classification of garments from fashion mnist dataset using cnn lenet-5 architecture. In *2020 international conference on innovative trends in communication and computer engineering (ITCE)*, pages 238–243. IEEE, 2020.

[253] Puja Bharati and Ankita Pramanik. Deep learning techniques—r-cnn to mask r-cnn: a survey. *Computational Intelligence in Pattern Recognition: Proceedings of CIPR 2019*, pages 657–668, 2020.

[254] Madhusri Maity, Sriparna Banerjee, and Sheli Sinha Chaudhuri. Faster r-cnn and yolo based vehicle detection: A survey. In *2021 5th international conference on computing methodologies and communication (ICCMC)*, pages 1442–1447. IEEE, 2021.

[255] Kaiming He, Georgia Gkioxari, Piotr Dollár, and Ross Girshick. Mask r-cnn. In *Proceedings of the IEEE international conference on computer vision*, pages 2961–2969, 2017.

[256] Zonghan Wu, Shirui Pan, Fengwen Chen, Guodong Long, Chengqi Zhang, and S Yu Philip. A comprehensive survey on graph neural networks. *IEEE transactions on neural networks and learning systems*, 32(1):4–24, 2020.

[257] Da Chen, Lin Feng, Yuqi Fan, Siyuan Shang, and Zhenchun Wei. Smart contract vulnerability detection based on semantic graph and residual graph convolutional networks with edge attention. *Journal of Systems and Software*, 202:111705, 2023.

[258] Kenneth Ward Church. Word2vec. *Natural Language Engineering*, 23(1):155–162, 2017.

[259] Zhijie Zhang, Huazhu Fu, Hang Dai, Jianbing Shen, Yanwei Pang, and Ling Shao. Et-net: A generic edge-attention guidance network for medical image segmentation. In *Medical Image Computing and Computer Assisted Intervention–MICCAI 2019: 22nd International Conference, Shenzhen, China, October 13–17, 2019, Proceedings, Part I 22*, pages 442–450. Springer, 2019.

[260] Chao Qi, Jianming Zhang, Hongjie Jia, Qirong Mao, Liangjun Wang, and Heping Song. Deep face clustering using residual graph convolutional network. *Knowledge-Based Systems*, 211:106561, 2021.

[261] Iaroslav Melekhov, Juha Ylioinas, Juho Kannala, and Esa Rahtu. Image-based localization using hourglass networks. In *Proceedings of the IEEE international conference on computer vision workshops*, pages 879–886, 2017.

[262] Fisher Yu, Vladlen Koltun, and Thomas Funkhouser. Dilated residual networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 472–480, 2017.

[263] Tianyan Zhou, Yong Zhao, and Jian Wu. Resnext and res2net structures for speaker verification. In *2021 IEEE Spoken Language Technology Workshop (SLT)*, pages 301–307. IEEE, 2021.

[264] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778, 2016.

[265] Vijay Badrinarayanan, Alex Kendall, and Roberto Cipolla. Segnet: A deep convolutional encoder-decoder architecture for image segmentation. *IEEE transactions on pattern analysis and machine intelligence*, 39(12):2481–2495, 2017.

[266] Zekun Tong, Yuxuan Liang, Changsheng Sun, David S Rosenblum, and Andrew Lim. Directed graph convolutional network. *arXiv preprint arXiv:2004.13970*, 2020.

[267] Lohith JJ, Kunwar Singh, and Bharatesh Chakravarthi. Digital forensic framework for smart contract vulnerabilities using ensemble models. *Multimedia Tools and Applications*, pages 1–44, 2023.

[268] Leo Breiman. Bagging predictors. *Machine learning*, 24:123–140, 1996.

[269] Cao Ying, Miao Qi-Guang, Liu Jia-Chen, and Gao Lin. Advance and prospects of adaboost algorithm. *Acta Automatica Sinica*, 39(6):745–758, 2013.

[270] Pooja Srinivasan et al. Tp-detect: trigram-pixel based vulnerability detection for ethereum smart contracts. *Multimedia Tools and Applications*, 82(23):36379–36393, 2023.

[271] Min-Ling Zhang and Zhi-Hua Zhou. A k-nearest neighbor based algorithm for multi-label classification. In *2005 IEEE international conference on granular computing*, volume 2, pages 718–721. IEEE, 2005.

[272] Niloofar Rastin, Mansoor Zolghadri Jahromi, and Mohammad Taheri. A generalized weighted distance k-nearest neighbor for multi-label problems. *Pattern Recognition*, 114:107526, 2021.

[273] Shiliang Sun and Rongqing Huang. An adaptive k-nearest neighbor algorithm. In *2010 seventh international conference on fuzzy systems and knowledge discovery*, volume 1, pages 91–94. IEEE, 2010.

[274] Deeman Y Mahmood and Mohammed A Hussein. Intrusion detection system based on k-star classifier and feature set reduction. *International Organization of Scientific Research Journal of Computer Engineering (IOSR-JCE) Vol*, 15(5):107–112, 2013.

[275] Roger J Lewis. An introduction to classification and regression tree (cart) analysis. In *Annual meeting of the society for academic emergency medicine in San Francisco, California*, volume 14. Citeseer, 2000.

[276] Nai-Wen Chang, Wei-Hao Deng, and Sun-Yuan Hsieh. Conditional diagnosability of (n, k)-star networks under the comparison diagnosis model. *IEEE Transactions on Reliability*, 64(1):132–143, 2014.

[277] Nitin Bhatia et al. Survey of nearest neighbor techniques. *arXiv preprint arXiv:1007.0085*, 2010.

[278] Stacey J Winham, Robert R Freimuth, and Joanna M Biernacka. A weighted random forests approach to improve predictive performance. *Statistical Analysis and Data Mining: The ASA Data Science Journal*, 6(6):496–505, 2013.

[279] Huajie Zhang and Charles X Ling. An improved learning algorithm for augmented naive bayes. In *Pacific-Asia Conference on Knowledge Discovery and Data Mining*, pages 581–586. Springer, 2001.

[280] Halil Yigit. A weighting approach for knn classifier. In *2013 international conference on electronics, computer and computation (ICECCO)*, pages 228–231. IEEE, 2013.

[281] Nathan Wiebe, Ashish Kapoor, and Krysta M Svore. Quantum nearest-neighbor algorithms for machine learning. *Quantum information and computation*, 15(3-4):318–358, 2015.

[282] Farid Ablayev, Marat Ablayev, Joshua Zhexue Huang, Kamil Khadiev, Nailya Salikhova, and Dingming Wu. On quantum methods for machine learning problems part ii: Quantum classification algorithms. *Big Data Mining and Analytics*, 3(1):56–67, 2019.

[283] Xueshuo Xie, Haolong Wang, Zhaolong Jian, Yaozheng Fang, Zichun Wang, and Tao Li. Blockgram: Mining knowledgeable features for efficiently smart contract vulnerability detection. *Digital Communications and Networks*, 2023.

[284] Laurens Van Der Maaten, Eric Postma, Jaap Van den Herik, et al. Dimensionality reduction: a comparative. *J Mach Learn Res*, 10(66-71), 2009.

[285] Jaesung Cha. Partial least squares. *Adv. Methods Mark. Res*, 407:52–78, 1994.

[286] Petros Xanthopoulos, Panos M Pardalos, Theodore B Trafalis, Petros Xanthopoulos, Panos M Pardalos, and Theodore B Trafalis. Linear discriminant analysis. *Robust data mining*, pages 27–33, 2013.

[287] Rasmus Bro and Age K Smilde. Principal component analysis. *Analytical methods*, 6(9):2812–2831, 2014.

[288] Emmanuel J Candès, Xiaodong Li, Yi Ma, and John Wright. Robust principal component analysis? *Journal of the ACM (JACM)*, 58(3):1–37, 2011.

[289] Candido Peres and Mário Antão. The use of multivariate discriminant analysis to predict corporate bankruptcy: A review. *Aestimatio: The IEB International Journal of Finance*, (14):108–131, 2017.

[290] Thomas Hofmann. Unsupervised learning by probabilistic latent semantic analysis. *Machine learning*, 42:177–196, 2001.

[291] David Blei and John Lafferty. Correlated topic models. *Advances in neural information processing systems*, 18:147, 2006.

[292] Laurens Van der Maaten and Geoffrey Hinton. Visualizing data using t-sne. *Journal of machine learning research*, 9(11), 2008.

[293] Leland McInnes, John Healy, and James Melville. Umap: Uniform manifold approximation and projection for dimension reduction. *arXiv preprint arXiv:1802.03426*, 2018.

[294] Galen Andrew, Raman Arora, Jeff Bilmes, and Karen Livescu. Deep canonical correlation analysis. In *International conference on machine learning*, pages 1247–1255. PMLR, 2013.

[295] Xin Zhou, Yuqin Jin, He Zhang, Shanshan Li, and Xin Huang. A map of threats to validity of systematic literature reviews in software engineering. In *2016 23rd Asia-Pacific Software Engineering Conference (APSEC)*, pages 153–160. IEEE, 2016.

[296] Saleema Amershi, Andrew Begel, Christian Bird, Robert DeLine, Harald Gall, Ece Kamar, Nachiappan Nagappan, Besmira Nushi, and Thomas Zimmermann. Software engineering for machine learning: A case study. In *2019 IEEE/ACM 41st International Conference on Software Engineering: Software Engineering in Practice (ICSE-SEIP)*, pages 291–300. IEEE, 2019.

[297] Hiroshi Kuwajima, Hirotoshi Yasuoka, and Toshihiro Nakae. Engineering problems in machine learning systems. *Machine Learning*, 109(5):1103–1126, 2020.

[298] David Pujol, Ryan McKenna, Satya Kuppam, Michael Hay, Ashwin Machanavajjhala, and Gerome Miklau. Fair decision making using privacy-protected data. In *Proceedings of the 2020 Conference on Fairness, Accountability, and Transparency*, pages 189–199, 2020.

[299] He Zhang, Muhammad Ali Babar, and Paolo Tell. Identifying relevant studies in software engineering. *Information and Software Technology*, 53(6):625–637, 2011.