AI Model Lifecycle Management: Systematic Mapping Study and Solution for AI Democratisation

THESIS
submitted in partial fulfillment of the requirements of the degree of
MASTER OF SCIENCE

in

Embedded System

by

Yuanhao Xie

TU Delft Delft University of Technology
Software Engineering Research Group
Department of Software Technology
Faculty EEMCS, Delft University of Technology
Delft, the Netherlands
www.ewi.tudelft.nl

ING
ING Bank Personeel B.V.
Acanthus, Bijmerdreef 24
Amsterdam, the Netherlands
www.ing.nl

Abstract

The development of artificial intelligence (AI) has made various industries eager to realise and obtain the benefits of AI. There is an increasing amount of research surrounding AI, most of which is centred on the development of new AI algorithms and techniques, thereby, however, ignoring an increasing set of practical problems related to AI mode lifecycle management, examples include versioning control of the data and models, difficulties in model deployment, transparency, model reproducibility, fairness and balance in data, and ethics. In contrast to this extensive list, research on the AI model lifecycle management is limited, and there is currently no comprehensive study. To address this gap, we researched the life cycle management of AI models. The research consisted of two parts. First, we conducted a systematic mapping study, which consists of the classification and counting of published papers in this field. By summarising the current situation of this field through quantitative and qualitative research, we obtained an overview, identified research gaps, and provided suggestions for future research. We then selected one of the specific themes, AI democratisation. Using this theme, we researched relevant literature, highlighted the importance of model documentation and the research gap, carried out research on the existing model documentation framework, improved the current framework, and introduced a solution for promoting AI democratisation. Specifically, we proposed a tool to automatically generate model documentation. Finally, we conducted a user study on the tool as a means of gathering suggestions for future improvements.

Purpose: Improve the life cycle management of AI from a theoretical and practical perspective through conducting a comprehensive study of the life cycle management of artificial intelligence models; a solution to a specific topic/research gap (i,e., the democratisation of AI).
Method: We carried out systematic mapping research on the life cycle management of AI models. Regarding AI democratisation, we researched literature pertaining to the democratisation of existing AI, compared with existing solutions, and improved AI from the perspective of generating model documents.
Results and Conclusion: The systemic map was obtained by categorising and counting related publications, improving the existing model document framework, and proposing a solution to automatically generate model documents, thereby improving AI democratisation.

Thesis Committee:

University supervisor:	Dr. Jan S. Rellermeyer, Faculty EEMCS, TU Delft
	Dr. Luís Cruz, Faculty EEMCS, TU Delft
Committee Member:	Prof. Dr. Arie van Deursen, Faculty EEMCS, TU Delft
	Dr. Lydia Chen, Faculty EEMCS, TU Delft

Acknowledgements

I would like to express sincere appreciation to my supervisors, Dr. Jan S. Rellermeyer and Dr. Luís Cruz for their guidance, support, and help during the project. I also want to acknowledge the support from my parents Jianguo Xie and Yingjie Sun. I would like to thank Hennie Huijgen, Elvan Kula, Wim Spaargaren, Martijn Steenbergen, Jerry Brons for their willing help and suggestion with this thesis.

Table of contents

I. Introduction

The term ‘Artificial intelligence’ was first proposed in 1956 by John McCarthy, Marvin Minsky, Nathan Rochester, and Claude Shannon. AI was then defined as "an attempt will be made to find how to make machines use language, form abstractions and concepts, solve kinds of problems now reserved for humans, and improve themselves" and "For the present purpose the artificial intelligence problem is taken to be that of making a machine behave in ways that would be called intelligent if a human were so behaving." [1] The AI at that time was classified as Symbolic AI. While the initial results impressed the scientific audience at that time, there were also critical voices that foreshadowed discussions that have continued over the decades. For example, Symbolic AI is reviewed as too much "disembodied abstractness", and its process is non-symbolic [3][5]. Alternative approaches include Connectionist AI and AI of Actionism. Connectionist AI refers to the realisation of intelligent behaviour by imitating the connection of neurons. Its origin can be traced to the research of Warren McCulloch and Walter Pitts on signals in the brain neural network [2]. Kunihiko Fukushima later developed the first true multilayer neural network in 1975. AI of Actionism is based on cybernetics. Unlike Symbolic AI and Connectionist AI, AI of Actionism studies using external behaviourism instead of intrinsic human thinking. Reinforcement learning is a representative algorithm of Actionism AI. With rapid development, artificial intelligence can no longer be defined simply from the perspective of these three categories. Rather, it is more a collection of various fields [5]. When we talk about artificial intelligence in the present day, we are generally referring to machine learning (ML) [6], which technically is just one branch of AI. In addition to ML, AI includes elements including Expert Systems, Multi-Agent Systems, Recommender Systems, and Robotics and perception. The AI mentioned in this article is general AI instead of ML.

Due to its fast, efficient, and accurate problem-solving ability, AI has been applied in various industries. In medical care, for example, ML can assist doctors in making better diagnoses. AI is also useful in the transportation field, as technologies like autonomous driving and automatic obstacle avoidance can better coordinate flights as well as reduce congestion and the number of traffic accidents. In manufacturing, robots liberated the human labour force. Biometric identifiers, such as face, iris, and fingerprint recognition, are of great significance in the security field. Since it is able to perform more complex calculations on a large amount of data, ML can help financial institutions perform risk assessment, detect money laundering and fraudulent transactions, and improve the user experience.

Despite its advantages, AI’s rapid development and wide application have brought many problems, including difficulty in deployment, data security, model transparency, and model reproducibility. These problems are visible in each of the aforementioned AI practical applications. For instance, ML requires a large amount of data sharing and integration. In the healthcare field, data sharing and integration promote AI feasibility but also create a number of privacy issues. The non-transparent deep neural network brings the safety problems in auxiliary diagnosis. Similarly, the automatic obstacle avoidance technique raises safety concerns when it improves traditional transportation. In addition to the previously introduced questions regarding trustworthiness, there are many problems with AI development, deployment, and management. For example, ML development relies on large amounts of data. The quality of data is directly related to the performance of the model, but how can developers of AI solutions ensure the fairness and balance of the data to prevent model biases[7,8].

The changes in data directly affect the model, but how to conduct the versioning control to ensure model reproducibility. The deployment of ML models is non-trivial. The most intuitive question: How do we ensure the robustness of the model when the data quality changes. All of these questions concerning AI model lifecycle management are examples of serious practical non-trivial problems, often hidden behind the success stories of prosperity in the field of AI.

Although there is a considerable amount of research and applications on new ML algorithms and AI technologies, research on the issues of AI life cycle management is sparse and there is a lack of comprehensive study about this topic. To address these shortcomings, we researched how to improve the life cycle management of AI models. This research is part of the Fintech Research AI (AFR), a collaboration between Delft University of Technology and ING Bank that ''seeks to develop new AI-driven theories, methods, and tools in large scale data and software analytics'' [9]. The core of the AFR consists of 10 research tracks including AI model life cycle management [9].

Taking into consideration the discussion mentioned earlier in this chapter, we divided the work into two different projects as a way to improve the life cycle management of AI models both theoretically and practically:

- A Systematic mapping study on AI model lifecycle management
- The prototype of a documentation tool to improve the transparent model reporting, thereby making AI more accessible and maintainable to traditional software developers.

Using these central objectives, this paper is divided into three sections. This first section provides background information on AI and the motivation for this study. The subsequent sections describe the two projects listed above. Each of these two sections has its own introduction, research question, methodology, results, discussion, and conclusion. The motivation of the second section, a solution for AI democratisation, is based on a portion of the findings produced by the mapping study.

The abstract of the systematic mapping study provides an overview of the entire mapping study. This is followed by the introduction, which explains the motivation for this study as well as related work and research questions. To answer our research question, we lay out our methodology in parts 1 chapter 2 by describing the four parts of the mapping study: conducting the search, screening paper, keywording, and data extraction. The schematic map, which shows the results of the mapping study, is discussed in Chapter 3. The final chapter consists of the conclusion of the mapping study.

Motivation for the second part of this study is based on results from the mapping study. This study found that, in the limited research on the life cycle of AI, there are few types of research on the subtopic of the democratisation of artificial intelligence. That said, the outcome of research on other subtopics can help improve the democratisation of artificial intelligence. With the massive application of AI in various industries, democratisation has become a problem in urgent demand of a solution. Therefore, the second part of this project addresses how we can use and integrate existing tools and research results of AI. In the first chapter, we introduce research questions and describe the results from literature research on AI democratisation. Then, we explain the implementation of a tool for automatically generating the model documents. We design a user study to assess the viability of the proposed tool, presented in the form of a functioning prototype, by interviewing practitioners and gathering feedback. The user study is explained in Chapter 3, which is followed by a description of the testing results and discussion in Chapter 4. We end by providing our conclusion

II. Systematic Mapping Study on AI model Lifecycle Management

Abstract

Objectives:

Our mappings study mainly aims to (1) get a comprehensive overview of current state of "AI model lifecycle management"; (2) identify research trends and categories in this field; (3) emphasize the promising research direction for the future research

Method:

We conducted a systematic mapping study to broadly exam the research of AI model lifecycle management published from 2005 to 2020

Results:

A total of 264 of the 3884 papers were finally selected. The research of AI model lifecycle management was classified into 6 main categories:

- Trustworthy
- Lifecycle management (from an overall perspective)
- Data management
- Model management
- Production
- Computing System/Architecture

and 31 sub categories.

Conclusion:

The research of artificial intelligence is developing rapidly, but the total number is still limited. The participating universities, companies and researchers are evenly distributed. 40% of the articles proposed solutions to specific problems rather than verifying or evaluating documents. Regarding the research of artificial intelligence life cycle, many topics lack accurate and authoritative definitions. Among all 31 sub-themes, model deployment, AI lifecycle management (overall perspective), security and fairness are the topics that receive the most attention. Many articles affirm the trend of AI democratization, but there is no specific solution.

1. Introduction

1.1 Motivation

With the development of AI, various industries are eager to discover and reap the benefits of AI, and progressively more AI-related research is being carried out. For instance, Chaurasia and Pal [10] used a support vector machine to carry out the analysis and prediction of heart disease. Kumari and Rashid [11] utilised artificial neural networks and decision trees to establish the relationship between diabetes and blood sugar rate. Through textual analysis and sentimental factors, Gao and Lin [12] analysed the relationships of various writing features and defaults of P2P lenders. Dixon, Klabjan and Bang [13] employed deep neural networks to predict the trends of commodity and foreign exchange markets. In the security field, Jin et al. [14] used a decision tree classification algorithm to implement malware detection in the Android system. Cheng et al. [15] proposed a positional approach for high-speed trains based on the least square support vector machine. Liu et al. fused lidar data and visual data to detect and recognise traffic signs, thereby improving the performance of the support driver assistance system. AI is also used in many other fields, such as education, aerospace, agriculture, semiconduction, games and entertainment.

However, the purpose of most of this research is to develop new AI algorithms or techniques to solve an issue in a given field. There are growing numbers of problems related to AI model life cycle management, such as version control of data and models, the difficulties of model deployment, transparency, model reproducibility, and fairness and balance in data and ethics [16,17]. Despite this, there is a limited number of studies on AI model life cycle management, and there is no comprehensive study on the life cycle management of AI models. To use AI more correctly and effectively, in addition to focusing on the self-organising tree algorithm, we must gain insight into all parts of the AI model life cycle and understand the impact of each stage and aspect. Although there has been some research on specific subtopics of AI life cycle management and the development of tools, there is still no comprehensive overview of this field in academia. To fill this gap, we have conducted a systematic mapping study of the life cycle management of AI. This study is concerned with classifying and counting published papers in this field. By summarising the current situation in this field through quantitative and qualitative research, we have developed an overview, identified research gaps and provided suggestions for future research directions.

1.2 AI model life cycle

As mentioned in the previous chapters, AI is a collection of various fields, each of which has a different system structure. In this section, we only explain the concept of the life cycle of one of these fields: machine learning. In the mapping study, however, we retained ‘artificial intelligence’ as the index keywords because we also wanted to see if we could obtain information about other AI fields.

"The machine learning life cycle is a cyclical process followed by data science projects" [18]. It defines the steps to achieve a set goal through machine learning. The AI model life cycle is broken down into steps, which vary depending upon perspective and degree of detail. DataRobot [18], for example, divides the life cycle of machine learning into five steps(see Figure 1.1.1): define project goals; acquire and explore data; build model; interpret model; and implement, document and maintain.

Figure 1.1.2 shows the stages of AI defined by Google, which consist of source and prepare the data; code; train, evaluate and tune the model; deploy; get predictions from the model; monitor; and mode manage and version[19]. Based on the Microsoft definition, a machine learning life cycle consists of six steps(see Figure 1.1.3): train model, package model, validate model, deploy model, monitor model and retrain model [20]. [21] describe the life cycle of one of the subcategories of machine learning, deep learning: find successful, well-known reference model and data preparation; create/update model; train and test model; evaluate model. As shown in Figure 1.1.4, [22] explains the life cycle management of supervised machine learning applications, dividing the life cycle into four parts: requirement analysis (which comprises data and system requirement analysis); data-oriented works (for example, data collection, learning and labelling); model-oriented works (which comprises a small loop structure of model training, model design and construction, model evaluation, and model optimisation); and DevOps works (which comprises model deployment and monitoring).

The problem with these defined steps of the AI life cycle are detailed and elaborate. They do not reflect the problems within the specific steps, such as the reproducibility and traceability of the model and the fairness of the data. Therefore, in our systematic mapping research, we took this into consideration when classifying the life cycle of the AI model.

To ensure the mapping study would cover as much research on AI model life cycle management as possible, based on the above discussion of AI model life cycle classification, we divided the life cycle into two parts as follows (see Figure 1.1.5):

- The workflow which has three categories:
    - Code meets data
    - Model-oriented works
    - DevOps works
- The technical debt of artificial intelligence which includes **traceability**, **reproducibility**, and **verifiability**.

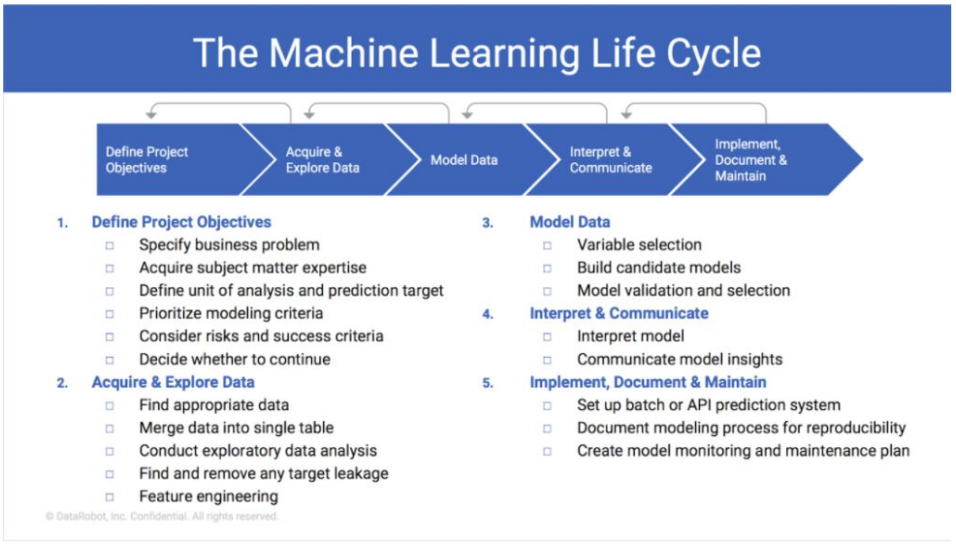


Figure 1.1.1: Steps and components of machine learning lifecycle defined by DataRobot [18]

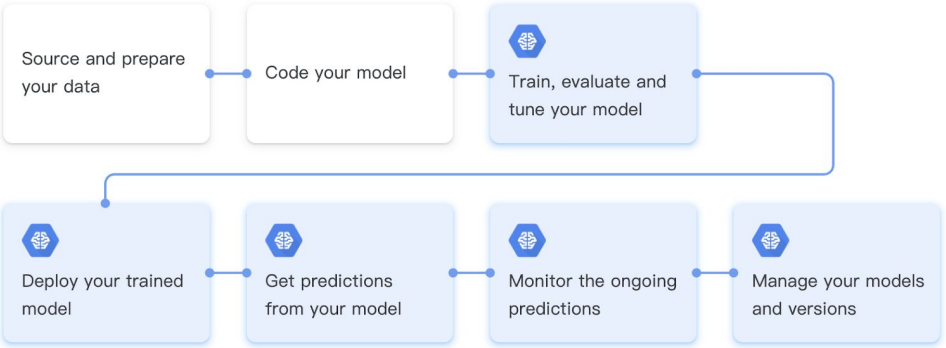


Figure 1.1.2: Steps and components of machine learning lifecycle defined by Google [19]

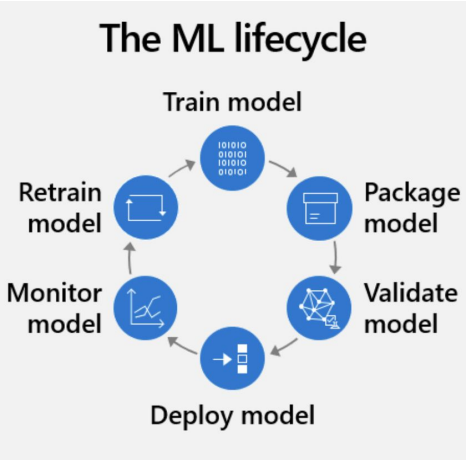


Figure 1.1.3: Steps and components of machine learning lifecycle defined by Microsoft [20]

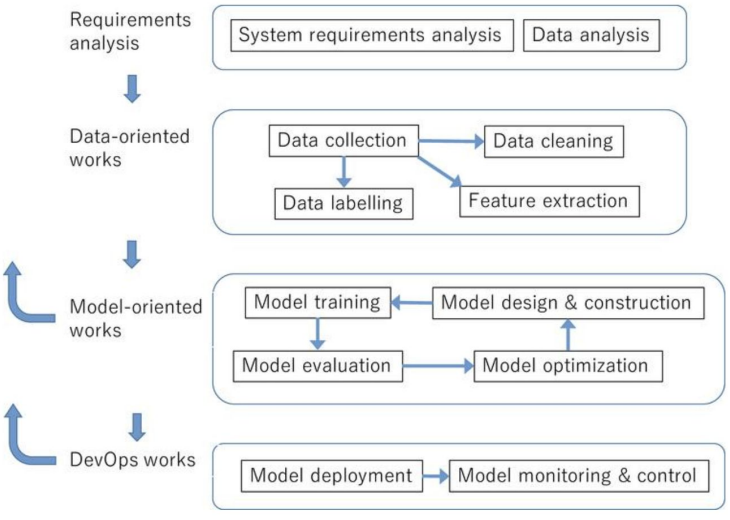


Figure 1.1.4: Steps and components of machine learning lifecycle defined by Microsoft [22]

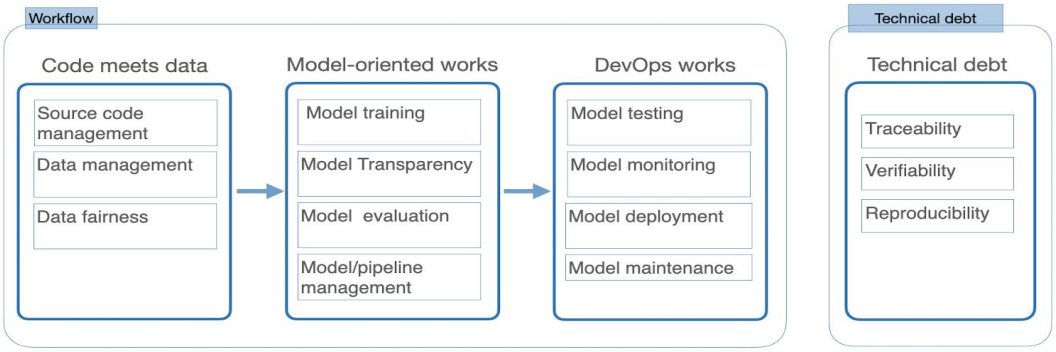


Figure 1.1.5: Keywords of AI model lifecycle

1.3 Current AI-related systematic mapping studies

Most of the systematic mapping studies in the field of AI focus on the applications of AI in various fields, or the intersection between AI and these specific fields. For example, [18] reviewed the application of machine learning in various software engineering activities, such as how to use machine requirements to automate and simplify software inspection techniques. The systematic mapping study was conducted at the intersection between machine learning and software engineering detection technology. [19] studied 131 articles from the ACM Digital Library and IEEE Xplore related to AI, network security and used the systematic mapping method to draw a systematic map, which confirmed the effectiveness of AI methods in intrusion detection systems. Another area of artificial intelligence is decision support systems, which can help decision makers make better decisions when they need to consider factors in many scientific fields and face complex problems. [25] carried out a systematic mapping study of fisheries and aquaculture and decision support systems.

None of the AI-related systematic mapping studies are about AI model life cycle management, and only fragmented work about the specific sub topic of AI lifecycle management. For instance, through systematic mapping research, [26] conceptualised and classified the ethics of AI, and found 37 keywords related to the ethics of AI in 83 papers. [27] conducted a system mapping study on the testing of machine learning systems. After researching 37 selected articles, they identified the trends in the field of machine learning testing and discovered research opportunities, such as testing machine learning programs using reinforcement learning.

In the process of researching the literature on AI-related mapping studies, we found a literature review on the AI model life cycle [28]. In it, the authors discuss the challenge of AI applications from the perspective of software engineering and provide a qualitative summary to answer two research questions: “(1) What software engineering challenges for machine learning applications have been discussed and potentially exist? (2) Which knowledge area is closely related to each of them?” The authors mapped the challenge of AI applications to software engineering topics based on the Software Engineering Body of Knowledge (SWEBOK), which is the guideline for classifying software engineering topics into knowledge areas. However, due to the nature of the study, it provides a qualitative summary and lacks a high level overview and classification. Therefore, we conducted

systematic mapping study, provided demographics and classifications to answer a wide range of research questions.

1.4 Research questions

Considering the background and research gaps described above, we conducted a systematic mapping study on artificial intelligence life cycle management, and defined our research questions as follows:

- What Research exists in the field of AI model life cycle management?

To make the research question achieve the purpose of constructing a knowledge system related to a specific topic [29,30,31], we also defined five sub-questions according to mapping study guidelines [29,30,31]. Each question has a clear goal related to the research topic and specifies which data will be extracted from the selected paper. The five sub-questions are as follows:

- RQ1: In what years, from which sources, countries, universities, and by which researchers were these research papers published?
- RQ2: What research approaches do these studies apply?
- RQ3: What most frequently applied research methods?
- RQ4: Which subtopics of AI model life cycle management have already been investigated?
- RQ5: What most investigated topics about AI model life cycle management, and how has this changed over time?

2. Methodology

This chapter describes the methodology of the mapping study. The system mapping study is divided into five parts based on the guideline described in [29]:

Defining the research questions: The main research questions and sub-questions are described in section 1.3.

Conducting the search: This step consisted of two parts—identifying keywords and formulating search queries. Section 2.1.1 explains how we identified keywords related to the research question based on the PICO model to ensure the search covered as many papers as possible related to AI model lifecycle management. Section 2.1.2 discusses why we choose DBLP and Scopus to conduct the search, as well as the syntax of each database. Section 2.1.3 describes how we defined the queries based on syntax to remove papers that were out of the scope of this research. After completing this step, we identified 3884 papers.

Screening of papers: The screening of papers was completed by two researchers to ensure the objective and fairness of the screening process. We conducted three rounds of paper screening; this consisted of checking titles, abstracts, and assessing the content of the papers. In each step, we labelled the papers as “include”, “exclude” or “uncertain”. Based on the criteria, we filtered out the papers that were out of scope. A total of 264 of the 3884 papers were retained for the next step.

Keywording: In this step, we thoroughly assessed the content of all 264 papers several times. For each paper, we selected keywords from the keywords sets. In the end, we classified all the publications into six major categories with 31 sub-categories.

Data extraction: we listed the information that needed to be extracted to answer the research questions.

2.1 Conducting the Search

We used the PICO model to define keywords related to the research question. We formulated the search queries for each database based on the keywords and the syntax of the selected database.

2.1.1 Defining the keywords based on PICO

As described in section 1.4, our main research question was “What are the lifecycle management challenges/topics of AI?”. In this study, as shown in Table 1.1.1, we used the PICO (population, intervention, comparison and outcomes) model suggested by Kitchenham and Charters [29] to explain the research question:

Population	In our context, the population is a specific part and category of AI, for instance, machine learning, deep learning etc.
Intervention (Phenomenon of Interest)	In our context, interventions are the specific aspects of lifecycle management. As described in section1.2 and shown in Figure 1.1.5, in our mapping study we roughly classified the topics related to AI model lifecycle management into four categories: code meets data, model-oriented work, DevOps-related work, and technical debts. The subtopics in each category are the phenomenon of interest.
Comparison	Comparison is what the intervention is being compared with [29]. No empirical comparison was made.
Outcomes	No measurable outcomes were considered.

Table 1.2.1: Main component extracted based on PICO model

Based on the PICO model, keywords were defined from two major aspects: artificial intelligence-related and lifecycle-related as shown in Table 1.1.2:

- AI-related: **Artificial Intelligence, AI, Machine Learning, ML, Deep Learning, DL, Neural Network**
- AI model lifecycle-related (see Figure 1.1.5):

  I. From the perspective of the process of AI life cycle management,we divided artificial intelligence life cycle into three stages to extract keywords.
     A. Code meets data: Including **data management, data fairness** and **source code management**.
     B. Model-oriented works include **model training and evaluation, model transparency** and **model/pipeline management.**
     C. DevOps works consist of **model deployment, model monitoring, model maintenance** and **model testing.**

  II. From the perspective of the main technical debt of artificial intelligence: **traceability, reproducibility and verifiability.**

  III. Other keywords related to standards: **standard, guideline, best practice, lifecycle and platform.**

Population, AI-related	Artificial Intelligence, AI, Machine Learning, ML, Deep Learning, DL, Neural Network
Phenomenon of Interest, AI model lifecycle-related	data management, data fairness, source code management, model training and evaluation, model transparency, model/pipeline management, model deployment, model monitoring, model maintenance, model testing, traceability, reproducibility and verifiability, standard, guideline, best practice, lifecycle, platform
Comparison	No comparison was considered.
Outcomes	No measurable outcomes were considered.

Table 1.2.2: Keywords extracted based on PICO model

The relation among the keywords in one group should be Boolean “or”, and the search query should contain at least one keyword from each group, which means the relationship between two groups should be Boolean “and”.

2.1.2 Database

2.1.2.1 Why Scopus and DBLP

To ensure high-quality search results and wide coverage and to manage the workload of the mapping study, we chose two of many potential databases: Scopus and DBLP. The reasons for selecting these databases are as follows:

DBLP offers bibliographic information, especially in the area of computer science. It has strict criteria regarding the venue, editors and authors, standards, and long-term availability to better index a journal article or conference to ensure high-quality references [32]. DBLP covers a significant portion of ACM, IEEE, Springer and some smaller digital libraries [33]. “Scopus is the largest abstract and citation database in peer-reviewed literature.[71]” It provides an overview of comprehensive research results in different fields, including computer science [34]. [35] Compared four databases including DBLP and Scopus; the results indicated that DBLP and Scopus had a greater number of unique articles indexed. The precision score of DBLP was 0.79 with a range from 0 to 1, while all of the other database scores were less than 0.1. Compared with other databases that are not specific for computer science, Scopus had a better coverage than others for the field of computer science.

2.1.2.2 Syntax

Each database has its own syntax of query strings. Here, we only list those used in our mapping study.
For Scopus, the grammar used to formulate the query is as follows:

1. Boolean and: separate the words with “AND" to find documents that contain all of the terms.
2. Boolean or: separate the words with “OR” to find documents that contain any of the terms.
3. Boolean and not: separate the words with “AND NOT"; this excludes documents that include the specified term.
4. Asterisk (*): Replace multiple characters in a word to search keywords from the same root.
5. Searches words from the title, abstract, and keywords: TITLE-ABS-KEY()
6. Searches words from the title: TITLE ()
7. Preceding (Pre/n): For example, “machine Pre/2 learning” means that “machine” must be no more than two words apart from “learning”.
8. Within (W/n): It does not matter which word comes first. For example, “model W/1 test” means that phrases such as “test AI model” and “model test” will be found.

For dblp, the grammar used to formulate the query is as follows:

1. Boolean and: separate words by space. This finds documents that contain all of the terms.
2. Boolean or: connect words by pipe symbol (|). This finds documents that contain any of the terms.

3. Prefix search is the default in dblp. This searches keywords from the same root. For example, the keyword "sig" will identify "SIGIR" and "signal".

As described in section 2.1.1, we defined our keywords as two groups: AI-related keywords and AI lifecycle-related keywords. The relation among the keywords in one group should be Boolean “or”, and the search query should contain at least one keyword from each group. This means the relation between the two groups should be Boolean “and”.

2.1.3 Defining the query strings

2.1.3.1 DBLP

For dblp, the phrase search operator (.) was disabled due to technical problems, and dblp does not support brackets to indicate the order of precedence rules. For example, to express:

“Artificial Intelligence” or AI

we had to split the phrase “Artificial Intelligence”. This led to the search query:

Artificial| AI Intelligence | AI

Note that “|” means Boolean “or”, and space “ ” means Boolean “and”. This query identifies papers that contain “Artificial Intelligence” or “AI”, as well as some noise papers that contain “Artificial AI” or “Intelligence AI”. If the wrong query is used, such as “Artificial Intelligence | AI”, only the noise papers that contain “Artificial AI” or “Intelligence AI” will be identified instead of the desired papers containing “Artificial Intelligence” or “AI”.

Taking the above into account, we defined our dblp search queries as shown below. To make it easy to read, here we used separate lines to represent a space “ ” (which means Boolean “and”).

AI | ML | DL | Artificial | Machine | Deep | Neural
AI | ML | DL | Artificial | Machine | Deep | Network
AI | ML | DL | Artificial | Machine | Learning | Neural
AI | ML | DL | Artificial | Machine | Learning | Network
AI | ML | DL | Artificial | Learning | Deep | Neural
AI | ML | DL | Artificial | Learning | Deep | Network
AI | ML | DL | Artificial | Learning | Neural
AI | ML | DL | Artificial | Learning | Network
AI | ML | DL | Intelligence | Machine | Deep | Neural
AI | ML | DL | Intelligence | Machine | Deep | Network
AI | ML | DL | Intelligence | Machine | Learning | Neural
AI | ML | DL | Intelligence | Machine | Learning | Network
AI | ML | DL | Intelligence | Learning | Deep | Neural
AI | ML | DL | Intelligence | Learning | Deep | Network
AI | ML | DL | Intelligence | Learning | Neural
AI | ML | DL | Intelligence | Learning | Network
traceab | reproduc | verif | guideline | standard | lifecycle| pipeline | deploy | maintain | fairness | transparency | platform

2.1.3.2 Scopus

As mentioned above, Scopus accepts the Boolean “and not”; therefore, we also defined keywords to filter out papers that were out of our research scope to reduce the workload in the next step. We named these as exclusive keywords. In this work, we only focused on AI model lifecycle management and not on the applications of AI, using AI to predict or forecast something, or using AI to detect defects, etc. The exclusive keywords are as follows:

predict, forecast, diagnosis, defect, fault, fpga, sensor application, AI method for, using PRE/2 AI, AI approach for, "Application of" PRE/2 "AI", by PRE/2 AI, "based on" PRE/2 AI.

where the keyword “AI” could be replaced by other keywords from the AI-related group in table 1.2.2.

The Scopus queries were formulated as follows:

TITLE-ABS-KEY ("ai model" OR "ml model" OR "artificial intelligence model" OR "machine learning model" OR "neural network model" OR "deep learning model" OR "dl model")

AND TITLE ("deep learning" OR "dl" OR "ai" OR "ml" OR "artificial intelligence" OR "machine learning" OR "neural network")

AND TITLE-ABS-KEY ((model W/1 test) OR traceability OR verifiability OR reproducibility OR guideline OR standard OR "best practice" OR lifecycle OR "data management" OR "source code" OR pipeline OR deploy* OR maintain* OR fairness OR transparen* OR platform)*

AND NOT TITLE (("using" PRE/2 "deep learning") OR ("using" PRE/2 "dl") OR ("using" PRE/2 "ai") OR ("using" PRE/2 "ml") OR ("using" PRE/2 "artificial intelligence") OR ("using" PRE/2 "machine learning") OR ("using" PRE/2 "neural network"))

AND NOT TITLE (("Application of" PRE/2 "dl") OR ("Application of" PRE/2 "ai") OR ("Application of" PRE/2 "ml") OR ("Application of" PRE/2 "artificial intelligence") OR ("Application of" PRE/2 "machine learning") OR ("Application of" PRE/2 "neural network"))

AND NOT TITLE (("Application of" PRE/2 "deep learning") OR ("based on" PRE/2 "dl") OR ("based on" PRE/2 "deep learning"))

AND NOT TITLE (("by" PRE/2 "deep learning") OR ("by" PRE/2 "dl") OR ("by" PRE/2 "ai") OR ("by" PRE/2 "ml") OR ("by" PRE/2 "artificial intelligence") OR ("by" PRE/2 "machine learning") OR ("by" PRE/2 "neural network"))

AND NOT TITLE (("based on" PRE/2 "ai") OR ("based on" PRE/2 "ml") OR ("based on" PRE/2 "artificial intelligence") OR ("based on" PRE/2 "machine learning") OR ("based on" PRE/2 "neural network"))

AND NOT TITLE ("deep learning approach for" OR "dl approach for" OR "ai approach for" OR "ml approach for" OR "artificial intelligence approach for" OR "machine learning approach for" OR "neural network approach for")

AND NOT TITLE ("deep learning method for" OR "dl method for" OR "ai method for" OR "ml method for" OR "artificial intelligence method for" OR "machine learning method for" OR "neural network method for")

AND NOT TITLE (predict OR forecast*)*

AND NOT TITLE (sensor)

AND NOT TITLE (fpga)*

AND NOT TITLE (("diagno" OR "defect*" OR "fault*"))*

2.1.4 Search results

We only considered papers published in English from 2005 to 2020. After this step, we found 3884 papers in total; among them, 2127 publications were from DBLP and 1757 publications were from Scopus.

2.2 Screening Papers

In this step, we conducted three rounds of screening based on defined criteria to exclude papers that were out of scope. The details and results of each round can be checked in Figure 1.2.1

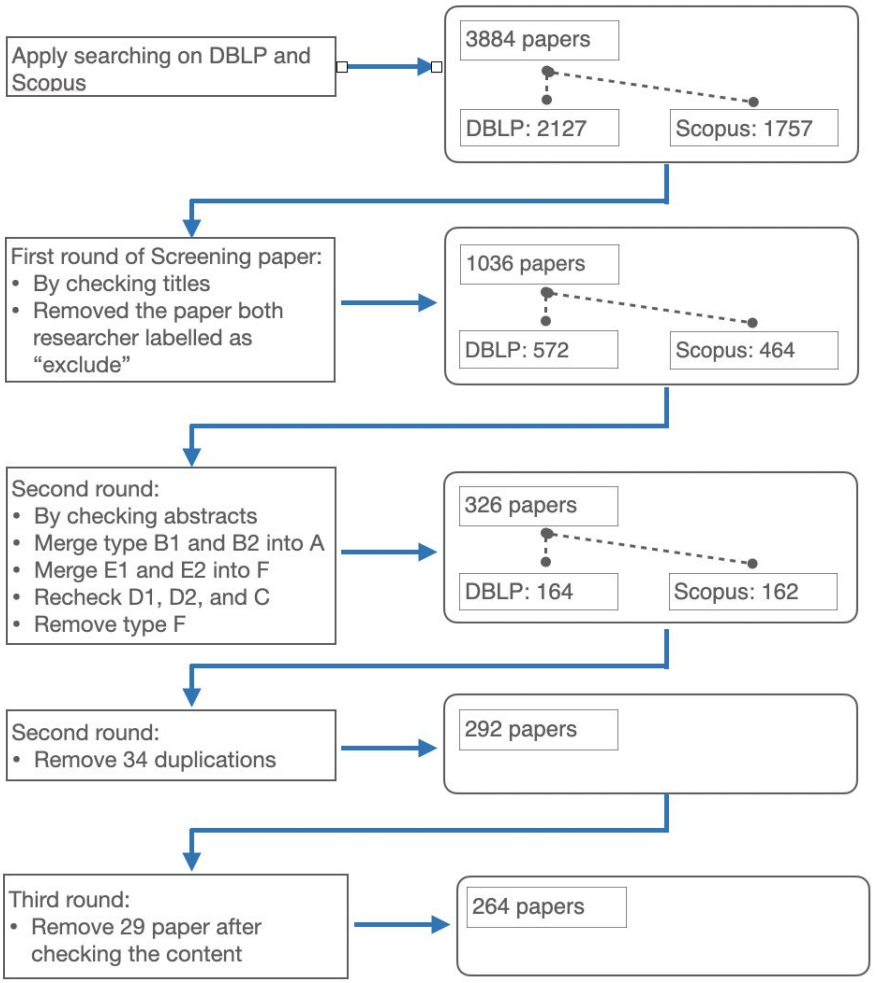


Figure 1.2.1: Three rounds of paper screening

2.2.1 Criteria

We applied the following criteria to remove papers that were outside the scope of our study:

- Studies not presented in English
- Duplicates
- Studies not accessible in full-text
- Books and grey literature
- Studies that discussed applications of AI or the design of new AI algorithms unrelated to AI model lifecycle management

In the first two rounds, two researchers checked all the publications identified in the previous steps and marked them as "include", "exclude" or "uncertain". Type F papers (see Figure 1.2.3), i.e., those that both researchers marked as "exclude", were not carried forward.

Label	Researcher 1			
Researcher 2		Include	Uncertain	Exclude
	Include	A	B1	D1
	Uncertain	B2	C	E1
	Exclude	D2	E2	F

Table 1.2.3: Labels of the papers

2.2.2 First round

Two researchers checked the title of all 3884 papers identified in the "conduct search" step and labelled the papers as either "include", "exclude" or "uncertain". Papers that both researchers labelled as "exclude" were removed from further consideration. In the end, 1036 papers remained, among which 464 were from Scopus and 572 were from DBLP. Table 1.2.4 and 1.2.5 show the results after the first round.

Scopus	Research 1			
Research 2		Include	Uncertain	Exclude
	Include	A:100	B1:22	D1:22
	Uncertain	B2:63	C:44	E1:89
	Exclude	D2:35	D2:89	F:1293

Table 1.2.4: Numbers of each labels of Scopus papers after checking the titles

DBLP	Researcher 1			
Researcher 2		Include	Uncertain	Exclude
	Include	A: 95	B1:45	D1:49
	Uncertain	B2:66	C:41	E2:123
	Exclude	D2:84	D2:69	F:1555

Table 1.2.5: Numbers of each labels of DBLP papers after checking the titles

2.2.3 Second round

Two researchers checked the abstracts of all 1036 papers identified in the first round and again labelled them as "include", "exclude" or "uncertain". Table 1.2.6 and Table 1.2.7 show the results after this step.

Scopus	Researcher 1			
Researcher 2		Include	Uncertain	Exclude
	Include	A:127	B1:3	D1:2
	Uncertain	B2:16	C:18	E1:1
	Exclude	D2:31	E2:7	F:257

Table 1.2.6: Numbers of each labels of Scopus papers after checking the abstracts

DBLP	Researcher 1			
Researcher 2		Include	Uncertain	Exclude
	Include	A:126	B1:3	D1:36
	Uncertain	B2:15	C:3	E1:5
	Exclude	D2:26	E2:5	F:353

Table 1.2.7: Numbers of each labels of Scopus papers after checking the abstracts

Papers that one of the researchers labelled as "uncertain" but the other labelled as "include" or "exclude" were relabelled: Type B1 and B2 were merged as type A, and E1 and E2 were merged as type F.

Next, we rechecked all the papers that were labelled as type C. In the end, 27 of these papers were relabelled as A and the rest were classified as F

All type F papers were removed. There were 326 papers left among which 162 were from Scopus and 164 were from DBLP. After merging these papers from the two databases, we found 34 duplicate articles. All duplicates were removed, and the remaining 292 papers proceeded to the third round of screening.

2.2.4 Third round

In this step, one researcher went through the contents of the 292 papers from the second round to further remove articles that were out of scope. This researcher also defined keywords for the next step, known as "keywording". A total of 29 papers were removed in the third round. Therefore, after the "Screening papers" step, 264 papers remained.

2.3 Keywording

We performed the first classification assessment of papers during the third round of screening by going through the contents of all the papers. In that step, we listed as many keywords as possible for

each paper to form a collection of keywords. Because new keywords become apparent when reading different papers, the papers read in the beginning are not well classified. Therefore, in the keywording step, we went through all 264 articles repeatedly. When reading papers, we selected suitable keywords from the keywords that were stated in the paper, and each paper was labelled with multiple keyword tags. As shown in Figure to1.2.2, we classified all of the papers into six major categories:

- Trustworthy
- Lifecycle management (from an overall perspective)
- Data management
- Model management
- Production
- Computing System/Architecture

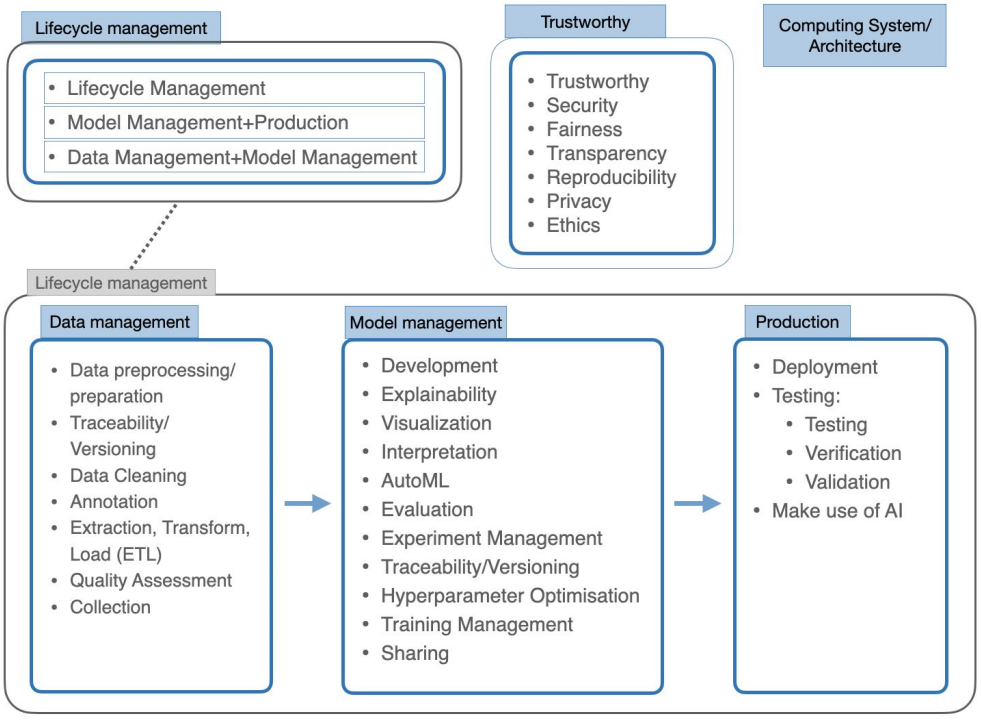


Figure 1.2.2: Topics of the AI lifecycle management

For each category, there are several subtopics. Computing system/architecture is a separate group because those papers mainly discussed the operating system or involved hardware. Such papers could also be defined as out of scope. However, these concepts are tangentially related to AI model lifecycle management, so we retained this category but listed it separately.

Except for the computing system/architecture and trustworthy categories, the remaining four categories all have the including and included implication. The entire field of artificial intelligence development can be divided into three stages/parts, namely, data management, model management and production. Hence, we defined these three keywords as topics. Publications discussing lifecycle management from a holistic perspective or involving two or more stages were classified in the lifecycle management category. The trustworthy category contained papers that discussed the indicated trustworthy-related topics. Importantly, each category also includes itself as a subset. For

example, articles that discussed multiple topics related to trustworthy were classified as trustworthy. To classify articles more clearly, each article only appears once in the major category. Articles involving multiple main categories were classified according to the main topic of the article. To describe the classification in more detail, the article can appear in multiple different subcategories. For a related discussion, please refer to section 3.3.

2.4 Data Extraction

- RQ1: In which year and from what sources, countries, universities, and researchers were these studies published?

Data to be extracted: The countries, universities, researchers, publishers, conferences, and journals.

- RQ2: What research approaches did these studies apply?

Data to be extracted: Research type used

- RQ3: What were the most frequently applied research methods?

Data to be extracted: The number of articles of each research type

- RQ4: What subtopics of AI model lifecycle management have already been investigated?

Data to be extracted: Classification of all the publications

- RQ5: What are the most investigated topics about AI model lifecycle management, and how have these changed over time?

Data to be extracted: The number of articles in each category; publications of each category vs. year

3 Result and Discussion

3.1 RQ1: In what years, from which sources, countries, universities, and by which researchers were these research papers published?

The first article on the life cycle of AI models was published in 2009 (Section 3.4). This article is about model management. From 2017, publications on this topic began to grow rapidly. The research is conducted in many countries. Among them, the United States produces the most publications, accounting for about 35–44% of the total, (the count depends on whether only the first author is considered). Publications from the IEEE and ACM accounted for 58% of the total. 69% of the research on artificial intelligence lifecycle management is in the form of conference papers. The largest number of these were presented at the International Conference on Management of Data, with 10 papers in total. Among universities and companies, IBM produced the most publications (22), half of which discuss trustworthiness in AI model lifecycle management. Most authors have written only one publication; the papers by authors with two to three publications tend to be interrelated, such as updates of previous research.

3.1.1 Country

The country is determined according to the institution where each author is located. There are two points to note: First, since most papers are multiply authored, we have made two different classifications by country. The first of these assigns authorship on the basis of the first author only; the second classification accounts for all of the authors. This latter classification is essential when all authors have contributed equally so, to be fair, all countries that appear are counted. However, countries that appear multiple times in one paper are counted only once.

As shown in Figure 1.3.1 and Figure 1.3.2, despite the changes to the country's accounting method, the top five countries remain unchanged, namely the United States, Germany, China, the United Kingdom, and Canada. The United States in particular accounts for a large proportion of all publications, about 40%, while countries other than these 'big five' each accounted for less than 3%.

Country	Number of publications	Percentage
United States	116	44.11%
Germany	21	7.98%
China	18	6.84%
United Kingdom	13	4.94%
Canada	11	4.18%
India	7	2.66%
Australia	7	2.66%
Switzerland	6	2.28%
Japan	6	2.28%
Singapore	5	1.90%
Austria	5	1.90%
Spain	5	1.90%
Norway	4	1.52%
Belgium	4	1.52%
South Korea	4	1.52%
Brazil	3	1.14%
Czech Republic	3	1.14%
Italy	3	1.14%
Others	22	8.37%

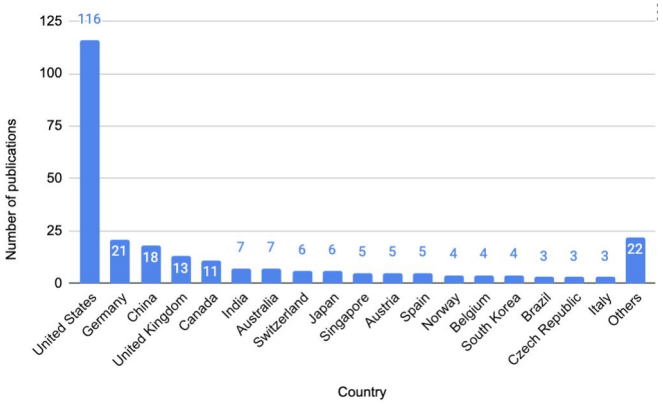


Figure 1.3.1: Statistics about the country, considering only the first author

Country	Number of publications	Percentage
United States	128	36.26%
Germany	27	7.65%
United Kingdom	22	6.23%
China	22	6.23%
Canada	17	4.82%
India	13	3.68%
Spain	10	2.83%
Switzerland	9	2.55%
South Korea	9	2.55%
Singapore	9	2.55%
Japan	8	2.27%
Australia	8	2.27%
Netherlands	6	1.70%
Austria	6	1.70%
Italy	5	1.42%
Brazil	5	1.42%
Belgium	5	1.42%
Norway	4	1.13%
France	3	0.85%
Czech Republic	3	0.85%
Others	26	7.37%

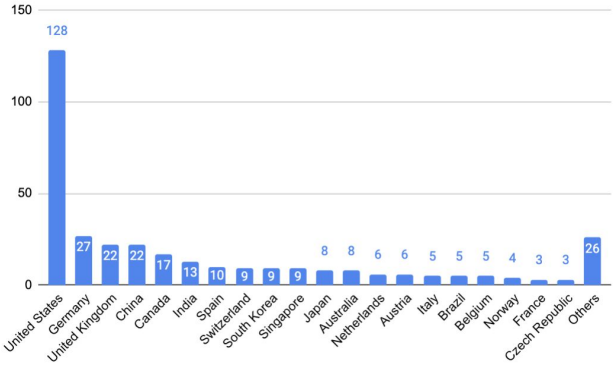


Figure 1.3.2: Statistics about the country, considering all authors

3.1.2 Citation

Since the number of citations recorded by different platforms/databases is different, the citation counts used here come from Google scholar.

Table 1.3.1 lists the 20 most-cited publications, as well as their publication types, publishers, and authors. Each of the top five publications has been cited more than 200 times.

Nine publications discuss trustworthiness topics, five discuss fairness, three focus on security, and one discusses ethics.

Among the top 20 publications in terms of citation, eleven articles discussed a topic in model management and/or model production. No publication focused on data management (such as data preparation, data cleaning, etc.). The ranking of citations shows that fairness is more popular than other topics.

Another point to note is that most of the heavily-cited publications appeared before 2019, and only one paper on AI ethics was published in 2019. Older publications have had more time to be cited, so we also calculated the average number of citations per year since publication. For example, for the most-cited paper, "DeepXplore: Automated Whitebox Testing of Deep Learning Systems", the average number of citations per year is 467/(2020-2017)=156. We take the difference of three years (2020–2017) instead of four years because the data was collected in the middle of 2020, and the article's publication month is unknown.

Publication	Citations (total)	Citations/year	Conference/Journal Name	Type	Publisher	Lifecycle Topic	Year
DeepXplore: Automated Whitebox Testing of Deep Learning Systems	467	156	Symposium on Operating Systems Principles	Conference Paper	ACM	Testing	2017
Deep learning for smart manufacturing: Methods and applications	425	213	Journal of Manufacturing Systems	Journal Articles	Elsevier	Guidelines for using AI	2018
Petuum: A New Platform for Distributed Machine Learning on Big Data	379	76	IEEE TRANSACTIONS ON BIG DATA	Journal Articles+Conference and Workshop Papers	IEEE	Deployment	2015
Decision-based adversarial attacks: Reliable attacks against black-box machine learning models	363	182	International Conference on Learning Representations	Conference Paper	International Conference on Learning Representations, ICLR	Security	2018
The Measure and Mismeasure of Fairness: A Critical Review of Fair Machine Learning	217	109	Economics and Computation (EC 2018) and International Conference on Machine Learning	Conference Paper	ACM	Fairness	2018
Visualizing Dataflow Graphs of Deep Learning Models in TensorFlow	191	96	IEEE Transactions on Visualization and Computer Graphics	Journal Articles	IEEE	Visualization	2018
TFX: A TensorFlow-Based Production-Scale Machine Learning Platform	190	63	ACM SIGKDD International Conference on Knowledge Discovery and Data Mining	Conference and Workshop Papers	ACM	Model Management+Production	2017
Artificial Intelligence: the global landscape of ethics guidelines	182	182	Nature Machine Intelligence ER	Journal Articals	Springer	Ethics	2019
ActiVis: Visual Exploration of Industry-Scale Deep Neural Network Models	180	90	IEEE Transactions on Visualization and Computer Graphics	Journal Articles	IEEE	Visualization	2018
Opportunities and challenges in developing deep learning models using electronic health records data: A systematic review	161	81	Journal of the American Medical Informatics Association	Journal Articles+Conference and Workshop Papers	Oxford University Press	Development	2018
Poison frogs! Targeted clean-label poisoning attacks on neural networks	158	79	Conference on Neural Information Processing Systems	Conference Paper	Neural information processing systems foundation	Security	2018
AI Fairness 360: An Extensible Toolkit for Detecting, Understanding, and Mitigating Unwanted Algorithmic Bias	131	66	IBM Journal of Research and Development	Journal Articles	IBM press	Fairness	2018
Improving Fairness in Machine Learning Systems: What Do Industry Practitioners Need?	112	112	Conference on Human Factors in Computing Systems	Conference and Workshop Papers	ACM	Fairness	2019
DeepMutation: Mutation Testing of Deep Learning Systems	109	55	IEEE International Symposium on Software Reliability Engineering	Conference Paper	IEEE	Testing	2018
ModelDB: A system for machine learning model management	101	25	International Conference on Management of Data	Conference Paper	ACM	Traceability/Versioning	2016
Ensuring fairness in machine learning to advance health equity	95	48	Annals of internal medicine	Journal Articles	American College of Physicians	Fairness	2018
Fairer machine learning in the real world: Mitigating discrimination without collecting sensitive data	84	28	Big Data & Society (BD&S)	Journal Articles	SAGE Publications Ltd	Fairness	2017
Interpretable machine learning in healthcare	79	40	Bioinformatics, Computational Biology and Biomedicine	Conference Paper	IEEE	Security	2018
The LRP toolbox for artificial neural networks	79	40	Journal of Machine Learning Research	Journal Articles	Microtome Publishing	Interpretation	2018
The LRP toolbox for artificial neural networks	79	20	Journal of Machine Learning Research	Journal Articals	Microtome Publishing	Explainability	2016

Table 1.3.1: 20 most-cited publications, publication types, publishers, and authors

3.1.3 Publisher

Most of the research was published by IEEE and ACM (77 and 61 articles respectively), together accounting for about 58% of the total. Springer published 28 articles accounting for 11.8%. Each of the remaining publishers' publications accounted for less than 5% of the total.

Publisher	Number of publications	Percentage
IEEE	77	32.35%
ACM	61	25.63%
Others	37	15.55%
Springer	28	11.76%
Elsevier	11	4.62%
AAAI	6	2.52%
CEUR-WS	5	2.10%
USENIX	4	1.68%
SPIE	3	1.26%
ijcai.org	3	1.26%
IBM	3	1.26%

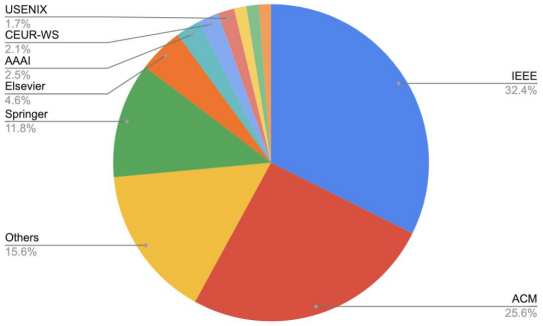


Figure 1.3.3: Statistics about the publisher

3.1.4 Conference

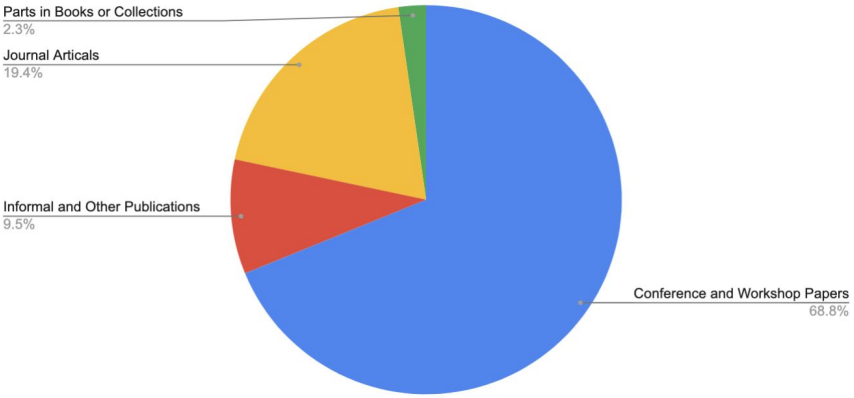
Conference papers are presented at many types of conferences. Table 1.3.2 lists eight conferences that featured more than two articles. Among them, the International Conference on Management of Data featured a total of 10 related articles. There was no paper about trustworthiness, though other topics were covered. Three papers discussed model management-related issues. Eight papers were presented at the AAAI/ACM Conference on Artificial Intelligence, Ethics, and Society; 6 of these are classified as addressing 'trustworthiness' in this mapping.

Conference	Number of publications
International Conference on Management of Data	10
AAAI/ACM Conference on Artificial Intelligence, Ethics, and Society	8
IEEE International Conference on Data Engineering	5
SIGKDD Conference on Knowledge Discovery and Data Mining	4
USENIX Conference on Operational Machine Learning	3
International Joint Conferences on Artificial Intelligence	3
IEEE International Conference on Big Data	3
The ACM Conference on Human Factors in Computing Systems	3

Table 1.3.2: All of the conference which published more than two papers

3.1.5 Publication type

68.8% of all publications are conference papers, 19.4% are journal articles, 9.5% are informative and other publications, and the remaining six publications are parts in books or collections.



Document type	Number of publications	Percentage
Conference and Workshop Papers	181	68.82%
Informal and Other Publications	25	9.51%
Journal Articles	51	19.39%
Parts in Books or Collections	6	2.28%

Figure 1.3.4: Statistics about the type of the publications

3.1.6 Company, University, or Organization

197 different companies, universities or organizations have published research on the life cycle management of artificial intelligence models. Four of the universities/companies produced more than five publications. As shown in Table 1.3.3, IBM has produced 23 articles, Google produced 8 articles, Carnegie Mellon University and the University of Maryland each produced 5 related publications.

Company/ University	Number of Publications
IBM	23
Google	8
Carnegie Mellon University	5
University of Maryland	5

Table 1.3.3: All of the universities/companies who published more than five papers

Among IBM's 23 related papers, 11 are on trustworthiness topics including security, model transparency, and fairness.

3.2 Research Type (RQ2 and RQ3)

RQ2: What research approaches do these studies apply? RQ3: What most frequently applied research methods?

A: 38.4% of the articles proposed a solution for a topic related to life-cycle management of AI models. Solution paper 25.5% of the articles are evaluation papers. Philosophical and Validation papers each account for about 16%. The most frequently applied research methods are solution proposals. However, in general, the proportion of the above four categories is relatively even.

3.2.1 Research type

There are six research types. They are solution paper, validation paper, evaluation paper, philosophical paper, opinion paper, and experience paper. The definition of research refers to the following documents:
It should be noted articles may range widely, and hence be classified into multiple types. For clarity, we sort publications according to the following rules.

The main difference between solution papers and evaluation papers is: if the publication lacks a clear and complete description of the implementation and its results, or is not a real-world case study, the paper is deemed a solution paper.

The difference between solution papers and validation papers is: the validation paper has a clear experimental setup, discussion about the results, and plots. If the publication contains only the proposed solution without clear and sufficient validation, it is deemed a solution paper.

As for evaluation and validation, many publications do not fall cleanly one way or the other. But, if the solution is implemented in the real world, tested, and confirmed, then the publication is deemed an evaluation paper. Validation papers, in contrast, discuss validation/testing of projects only in the laboratory or in the non-real world.

Philosophical papers are those that suggest classification of fields, or review of some topics. If a publication discusses and structures the current challenges for certain topics, but also proposes each subtopic a solution, then it is deemed a solution paper rather than a philosophical paper.

It can be seen from Table 1.3.4 that solution papers are the most numerous, constituting 38.4% of the dataset. Next are the evaluation papers, accounting for 25.5%. Both philosophical papers and validation papers account for about 16%. There are ten Opinion papers, constituting 3.8%. There are only two experience papers.

Research Type	Count	Percentage
Solution	101	38.40%
Evaluation	67	25.48%
Philosophical	42	15.97%
Validation	41	15.59%
Opinion	10	3.80%
Experience	2	0.76%

Table 1.3.4: Statistics for research type

3.3 Topic (RQ4 and RQ5)

RQ4: Which subtopics of AI model life cycle management have already been investigated? RQ5: What most investigated topics about AI model life cycle management?

A: According to the description in section 2.3, all of the papers are classified into 6 categories:

- Trustworthiness
- Lifecycle management (from an overall perspective)

- Data management
- Model management
- Production
- Computing System/Architecture

and 31 sub-categories.

Among them, the total number of papers related to “trustworthiness” and “model management” is relatively large, with 83 articles on trustworthiness and 66 articles on model management. For all sub-topics, the proportion of papers about deployment, lifecycle management (from an overall perspective), security, and fairness is slightly higher than other topics, all at around 10%.

3.3.1 Main topics

As mentioned in section 2.3, we classified all the 264 papers into six primary categories/topics, and each category has subcategories/topics. For primary classification, all articles are counted only once, and cross-topic articles are classified according to the main topic. For example, if the article proposes a model management (model visualization, model versioning/traceability) method, which also involves trustworthiness (reproducibility), that method helps to improve trustworthiness. In this case, this article will be classified as model management at the first level. For the secondary classification, that is, the subtopics of the primary classification. Articles that discuss multiple subtopics will be counted multiple times. The article in the above article is hence counted in the ‘model versioning/traceability’ classification, and also in ‘model visualization’.

We can see from table 1.3.5 that up to 63.5% of the publications are about lifecycle management (both as a whole or discussing a single part of it). Among these, data management-related articles are the scarcest, with only 13 articles, accounting for 5%. The most discussed topic is model management. This topic accounts for 25% of all publications. There are 66 of these articles, not counting publications on model management in “lifecycle management as a whole”. Trustworthiness articles account for one-third of all papers. There are only 15 publications related to Computing System/Architecture. This is because such articles are not very relevant to the topic of this paper, so the criteria are not clear. Therefore, papers related to this topic are most likely to be excluded.

Lifecycle Topic	Number of Publications	Percentage
Trustworthy	83	31.56%
Model Management	66	25.10%
Production	45	17.11%
Lifecycle Management	43	16.35%
Data Management	13	4.94%
Computing System/Architecture	13	4.94%

Table 1.3.5: Statistics for six main topics

3.3.2 Data management

There are 15 publications about data management. Two of these include two subtopics, leading to 17 counts in total. Most of the publications under this category are mainly on data preprocessing/preparation. The rest of the topics all only have one or two publications.

Data management topic	Number of publications	Percentage
Data preprocessing/preparation	6	40.00%
Traceability/Versioning	2	13.33%
Data Cleaning	2	13.33%
Annotation	2	13.33%
Extraction, Transform, Load (ETL)	1	6.67%
Quality Assessment	1	6.67%
Collection	1	6.67%

Table 1.3.6: Statistics for the subtopics under the category: data management

Table 1.3.6 classifies according to secondary topics. Articles with multiple topics are counted repeatedly. We can see that, in all secondary categories, deployment-related articles are the most numerous: 26 articles discuss lifecycle management from an overall perspective, and there are articles addressing security and fairness. The fifth topic is explanation/interpretation, with 23 articles.

3.3.3 Model management

The "model management" category has a total of ten subcategories, namely Development, Explainability/Interpretation, Visualization, Interpretation, AutoML, Evaluation, Experiment Management, Traceability/Versioning, Hyperparameter Optimization, Training Management, Sharing. A total of 66 articles discuss model management. Eight of these discuss two subtopics, so there are 74 counts in total. More than 10 publications discuss model development, model explanation/Interpretation, and visualization. In contrast, there are relatively few discussions about, for example, training management and hyperparameter optimization. These two subtopics may be more involved in algorithm/model development than in life cycle management. In addition, and more remarkably, most of the publications in the visualization category discuss deep learning/neural network visualization.

More publications discussed about model management can be found in the "lifecycle management" category.

Model Management topic	Number of publications	Percentage
Development	15	20.27%
Explainability	25	18.92%
Visualization	10	13.51%
Interpretation	9	12.16%
AutoML	8	10.81%
Evaluation	5	6.76%
Experiment Management	4	5.41%
Traceability/Versioning	3	4.05%
Hyperparameter Optimization	2	2.70%
Training Management	2	2.70%
Sharing	2	2.70%

Table 1.3.7: Statistics for the subtopics under the category: model management

There are usually two kinds of papers about explanation: using mathematical models to explain ML models, and explaining model features. In the process of reading the literature, we found that many documents confuse explainability and interpretability. Explainability refers to the explanation of internal mechanics, while interpretability focuses more on human understanding and causality[36],such as what causes the model to choose to make a specific prediction.

Among the visualization-related papers , [82] [81] [83] [84] proposed visualization solutions for model components (such as neural networks and neuron layers), and others for example [85], provided visualization of data flow. The solutions proposed mainly focus on the simple interpretation of ML algorithms through some interactive tools, and/or the visual abstraction of complex processes/components. However, in the latter case, visualizations that are too abstract are usually counterproductive.

3.3.4 Production

A total of 45 articles discuss production, and no articles are counted more than once. Most articles on production discuss deployment, though six articles discuss guidelines for AI applications. Among the 30 articles on deployment, 11 articles are about distributed deployment, and 13 articles address cloud-related topics such as cloud storage, cloud computing, etc. From this we can see that distributed systems and cloud computing are closely related to the deployment of artificial intelligence. For the topic "testing", two publications discuss verification, and three are about validation.

More publications about production can be found in the "lifecycle management" category.

Production management topic	Number of publications	Percentage
Deployment	30	66.67%
Testing	9	20.00%
Make use of AI	6	13.33%

Table 1.3.8: Statistics for the subtopics under the category: production

3.3.5 Lifecycle Management

There are 43 articles classified in this category. No articles in this category are counted multiple times.

Lifecycle Management has four sub-categories. If it includes data management, model management, production two or more are in this category. Lifecycle management is also a subcategory of its own, containing all articles discussing artificial intelligence management from a holistic perspective.

Lifecycle management topic	Number of publications	Percentage
Lifecycle Management	26	60.47%
Model Management+Production	10	23.26%
Data Management+Model Management	5	11.63%
Pipeline Debugging	2	4.65%

Table 1.3.9: Statistics for the subtopics under the category: Lifecycle management

3.3.6 Trustworthy

Trustworthiness has seven subtopics, namely Security Fairness Transparency Reproducibility Privacy Ethics Trustworthiness. Trustworthiness is also listed as its own subtopic because there are articles discussing multiple related topics. There are a total of 83 trustworthiness articles, all of which are counted once. Among them, the most publications discuss security and fairness, each with 26 articles, each accounting for 31%. Transparency has 12 articles accounting for 15%. The remaining topics are reproducibility, privacy and ethics, each of which accounts for less than 10%.

Trustworthy Topic	Number of publications	Percentage
Security	26	31.33%
Fairness	26	31.33%
Transparency	12	14.46%
Reproducibility	7	8.43%
Privacy	6	7.23%
Ethics	5	6.02%
Trustworthy	1	1.20%

Table 1.3.10: Statistics for the subtopics under the category: Trustworthy

In machine learning, the purpose of fairness usually refers to design algorithms to make fair predictions across various demographic groups [42]. According to different classification methods, 26 fairness related publications can be divided into: fairness in results, and fairness in process (fairness in process can be further divided into fairness in data and other resources, fairness in algorithms); static fairness, and dynamic fairness. The fairness of the algorithm is often related to interpretability. For example, improving the interpretability of the black box can improve the fairness of the algorithm.

We also found that although topic transparency has not received as much attention as the topics security and commonality, many articles related to other topics are related to transparency, for example explanation/interpretability, ethics, visualization and fairness. For details on related paper, please check section 1.2 in Part 2

3.3.7 Subtopics

We split the main category and count articles related to subtopics. For subtopics with only one article, we classify it as others. As shown in Table 1.3.11, deployment, lifecycle management, security, and fairness are the most discussed topics.

Lifecycle Topic	Number of publications	Percentage
Deployment	30	10.99%
Lifecycle Management	26	9.52%
Security	26	9.52%
Fairness	26	9.52%
Development	15	5.49%
Explainability/Interpretation	23	5.13%
Computing System/Architecture	13	4.76%
Transparency	12	4.40%
Model Management+Production	10	3.66%
Visualization	10	3.66%
Interpretation	9	3.30%
Testing	9	3.30%
AutoML	8	2.93%
Reproducibility	7	2.56%
Data preprocessing/preparation	6	2.20%
Serving	6	2.20%
Privacy	6	2.20%
Traceability/Versioning	5	1.83%
Data Management+Model Management	5	1.83%
Evaluation	5	1.83%
Ethics	5	1.83%
Experiment Management	4	1.47%
Data Cleaning	2	0.73%
Annotation	2	0.73%
Pipeline Debugging	2	0.73%
Training Management	2	0.73%
Sharing	2	0.73%
Hyperparameter Optimization	2	0.73%
Others	4	1.47%

Table 1.3.11: Statistics for all subtopics

3.4 Year & Trend (RQ5)

RQ5: How have the topics about AI model lifecycle management changed over time?

A: The number of studies on the life cycle management of AI models has increased over time. Especially from 2016 to 2019, the number of publications each year is double that of the previous year. The data for 2020 is not complete. The specific main topic and subtopic fluctuate slightly depending on the year, but the growth trend can still be seen.

The publication information mentioned in this mapping study was collected in the first half of 2020, so the 2020 publication information is not complete. But, from the data covering 2005 to 2019, we can see that the research on artificial intelligence life cycle management has been growing. From 2005 to 2008, there were no publications on this topic. In 2009, there was one article on “model management”. Beginning in 2017, the number of papers began to grow rapidly, reaching 138 in 2019. However, the total number of articles on artificial intelligence life cycle management is still relatively small.

From Figure 1.3.6 and Figure 1.3.7, we can see that the total number of articles related to all topics is increasing over time.

Year	Number of publications	Percentage
2020	19	7.20%
2019	138	52.50%
2018	57	21.70%
2017	29	11.00%
2016	12	4.60%
2015	3	1.10%
2014	2	0.80%
2013	1	0.40%
2012	1	0.40%
2011	0	0.00%
2010	0	0.00%
2009	1	0.40%
2008	0	0.00%
2007	0	0.00%
2006	0	0.00%
2005	0	0.00%

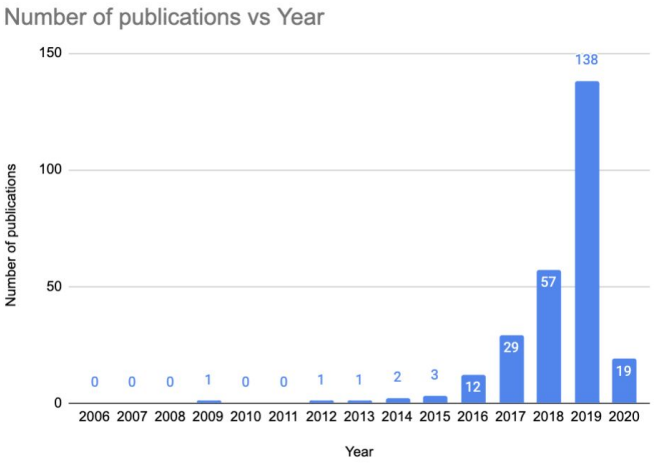


Figure 1.3.5: Number of publications each year from 2005 to 2020

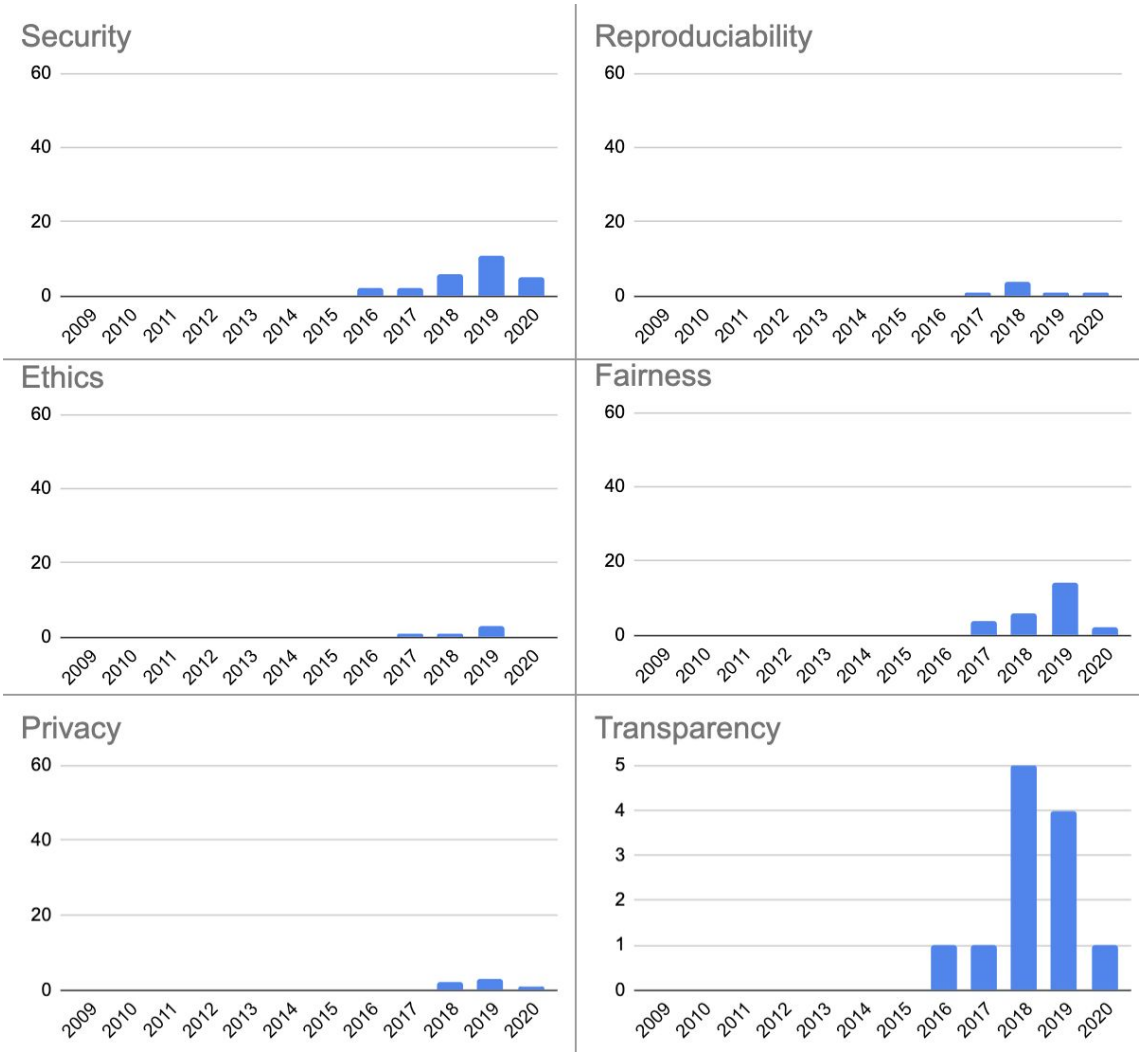


Figure 1.3.6: Number of publications of sub topics under "trustworthy" each year from 2005 to 2020

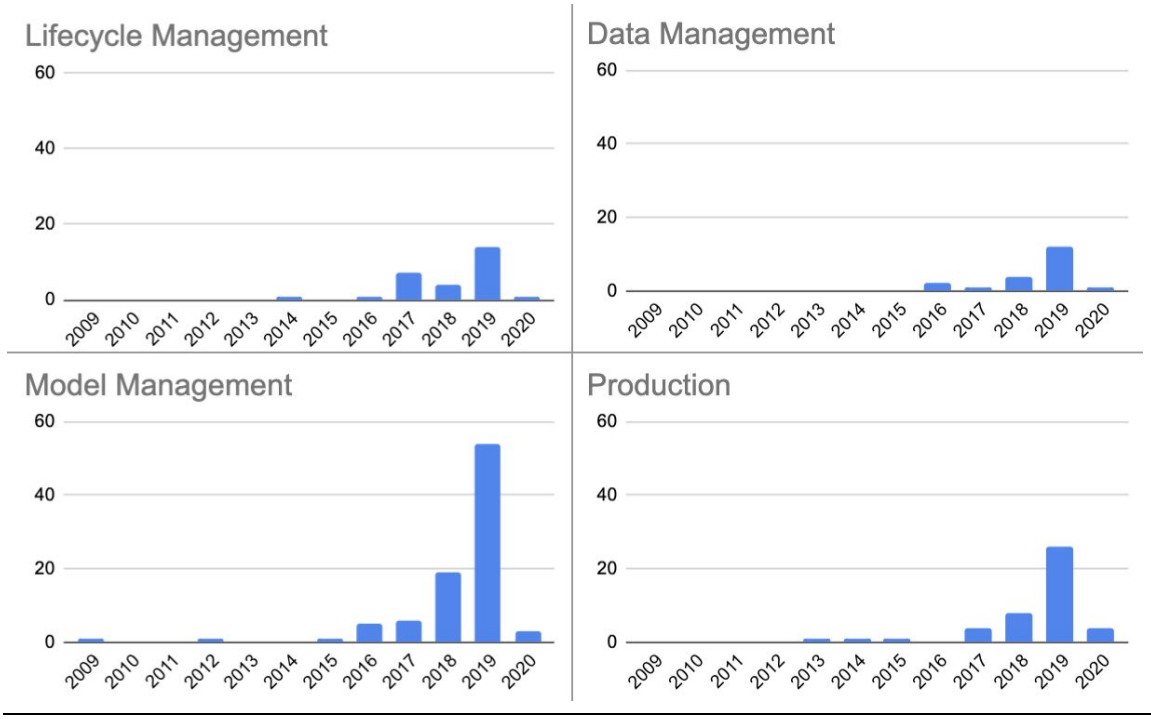


Figure 1.3.7: Number of publications of sub topics under "lifecycle" each year from 2005 to 2020

3.5 Research Type vs Topic

Figure 1.3.8 shows that, in the cross-correlation of topic with research type, the trustworthiness and model management solution papers appear the most, with 33 and 31 papers respectively. Trustworthiness is the most frequent topic among opinion papers, philosophical papers, solution papers and verification papers. No matter what kind of article, there are very few discussions on data management, only 5 or fewer. The only two experience papers discuss the life cycle management of AI models from a holistic perspective. Except for data management and computer system/architecture, the number of occurrences of other topics in the evaluation paper is basically the same.

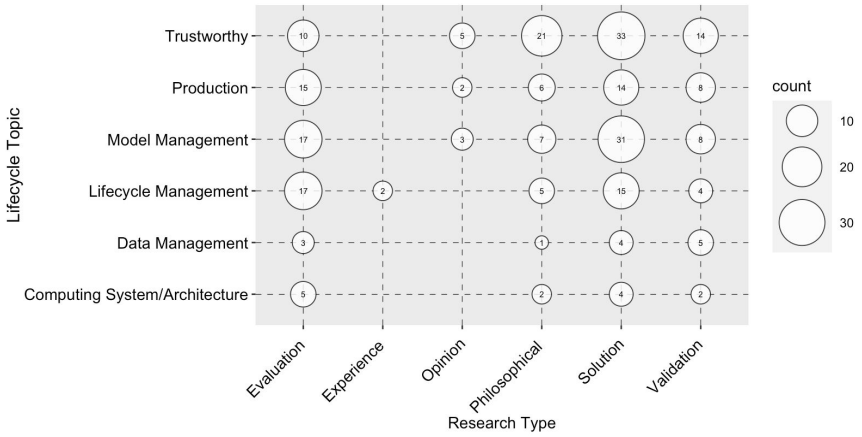


Figure 1.3.8: Bubble plot of research type vs Topic

4. Conclusion

Through this systematic mapping study, we found that, although the total number of publications is still small, research on artificial intelligence lifecycle management has shown a rapid upward trend since 2017. Various universities, companies, and other organizations from all over the world have participated in the research related to this topic, and related papers have been published in many different journals/conferences. Companies such as IBM have produced more publications than other organizations. Publications from the United States account for about half of the total. Many researchers have paid attention to this topic, but most of them only have one publication related to this topic.

About 40% of publications propose a solution related to AI model lifecycle management, and a quarter of the articles are evaluation papers.The proportion of conference papers in publishing is relatively high, though most of the conferences listed featured only one or two papers.

We have categorized all articles into 6 main categories. Among these, the topics that have received the most attention are trustworthiness and model management, while data management has received less attention. Among the 31 sub-categories, model deployment, AI lifecycle management (overall perspective), security and fairness received the most attention. In addition, among all current publications, the three most-cited are all about production. To a certain extent, this also reflects the attention of production. For all sub-categories, except for deployment, the number of articles on other topics is less than 30, while there are only 10 topics with at least 10 articles. Almost every topic has plenty of room to continue research.

We also noticed that when referring to specific topics, many articles are confused about topics with similar meanings. This may be due to the lack of a clear and authoritative definition of those topics, because the current research on the life cycle of artificial intelligence is still in its infancy.

For the classification of articles, we aimed to cover every step and problem of the artificial intelligence life cycle as much as possible; the overall level of understanding is retained. In order to achieve this goal, we need to make the classification as fine as possible, but also to avoid double counting as much as possible. Therefore, for example, the main category, AI lifecycle management, contains three other main categories: data management, model management, and production. This category was devised to minimize the double counting of articles discussing multiple topics. Moreover, we also say that each main category recurs as its own subcategory. In addition, we only repeat the count when discussing the subcategories under the specific main category.

III. Solution for AI Democratization and Transparency

1. Introduction

In this chapter, we first introduce the motivation of the research. Because we only classify articles based on the main topics discussed, some potential topics are not included in the classification of systematic maps, such as AI democratisation. In addition, we learned from the results of the mapping study that transparency, visualisation, and documentation can democratise artificial intelligence. Transparency is an abstract and broad concept, and the other two themes can be considered subcategories of transparency. In addition, there is significant research on visualisation, but few articles are related to AI documents. Combined with the current trend of democratisation of AI, we decided to conduct research on this potential topic and contribute to it practically. In other words, we decided to design a tool to automatically generate model documents to improve the transparency of the model. The artificial intelligence documentation tool aims to help non-artificial intelligence experts quickly understand and use artificial intelligence models, thereby promoting its democratisation. In section 1.2, we discuss the three main themes related to model democratisation, namely transparency, visualisation, and AI documentation. In section 1.6, we clarify the AI stakeholder. Finally, we formulate the research questions.

1.1 Motivation

Due to the fast, efficient, and accurate problem-solving ability of artificial intelligence, AI models are widely used in different domains. Many domain users who are not AI experts use machine learning. However, for most AI development, usage, and deployment, there is still a need for experienced experts like machine learning engineers and data scientists. The democratisation of artificial intelligence, that is, the process of enabling other stakeholders, including domain users, to understand and use artificial intelligence models, has become an important issue in life cycle management. As we mentioned in the systematic mapping study on AI model lifecycle management, there is less attention paid to the life cycle management of artificial intelligence. Moreover, in the limited research, there are fewer direct discussions about the democratisation of artificial intelligence. However, in our previous research on the artificial intelligence life cycle, we structured and categorised AI model lifecycle management into 6 categories with 31 subtopics. We found that some papers mentioned AI democratisation, but they mainly discussed the topics as follows:

- Transparency: the interpretation of the model itself, the interpretation of its results/decision, and the transparency of the design process.
- Visualisation: A common example is the visualisation of the machine learning process [43].
- AI documentation is a summary of useful AI model information.
- Other topics related to AI democratisation, such as increasing the participation of domain users in the design process by having domain users and ML experts collaborate closely in applying ML to practical problems, for example [43].

1.2 Transparency

From the results of the mapping study, we found that in the trustworthy category, there are few papers that directly discuss transparency, but papers in many other subcategories address the issue

of transparency. Furthermore, their proposed solutions, such as model interpretability, model visualisation, and ethics guidelines, can help improve transparency.

By reading the transparency-related papers obtained during the mapping study, we also found that the current research attaches great importance to domain users and non-AI experts in AI transparency. We obtained insight into the different factors of model transparency and their classifications. We also learned the basic method of solving the problem of transparency, that is, through the model explanation/interpretation and model description (AI documentation). A description of the related publications follows.

Transparency is the interpretation of the model, results/decision, and design process. We found 12 papers from the mapping study: [44] provided a comparative analysis of current transparency solutions, finding that the current transparency solutions lack "user cognitive response communication" and "domain knowledge". They proposed a solution to incorporate domain experts into the development process to increase model transparency. The article emphasises the importance of domain users in the development and use of AI. [45] clarified the definition of transparency, which can be used as more than a deontological framework. The paper also reiterated the importance of transparency. It clarifies the five components of transparency evaluation: a complete description of the purpose, the scope of application, data source, human interpretability, and monitoring of adverse events and emergency plans. [47] is a short article about artificial intelligence (AI) systems and autonomous systems (AS). It offers an overview of transparency, trust, and liability issues. [46] proposed a framework containing two documents (AI Validation Document, Deployment Disclosure Document) to structure the scope and requirements of the decision-making system to ensure its transparency. [48] focused on cognitive systems engineering. In the paper, the issue of transparency in automation equipped with machine learning is discussed. [49] and [50] improved transparency through documentation. In [51], the authors enhanced the transparency of deep learning by explaining its features. [50] and [52] proposed or discussed the interaction method to explain and interpret machine learning models to improve transparency. In [80], the author proposed a solution to explain the 'black box' by collecting real-time internal status.

1.3 Visualisation

Generally, the visualisation of AI models mainly focuses on the simple interpretation of ML algorithms through some interactive tools and the visual abstraction of complex processes. In the latter case, visualisations that are too abstract are usually counterproductive. Among the papers obtained from the mapping study, [37], [38], [39], and [40] proposed visualisation solutions for model components (such as neural networks and neuron layers) and others, for example [41], provided visualisation of data flow.

As a method of describing the model, visualisation can help model users and developers understand the model, thereby contributing to the transparency and democratisation of the model.

1.4 AI Documentation

Documentation is not a new topic for software engineering. There has been a lot of research on traditional software documentation. Developers regard documentation as an essential part of completing software engineering well [54], [55]. At the same time, there are also explorations for the

automation of traditional software documentation [56]. However, we found that there is little research on the documentation of artificial intelligence models.

[57] describes standardised documentation for the data in AI projects that includes the information contained within the data, data collection, intended use, and other concerns, for example, fairness and privacy.

Fact sheets [56, 58] propose a guideline to generate the documents to help increase trust in AI services. Their target group is model consumers. Fact sheets are created by AI providers/developers for examination by AI consumers. The content of the fact sheets focuses on topics like fairness, privacy, and safety.

[57] divided documents into internal and external categories. The external document mainly helps to win trust for the model, reduce abuse, and help users refer to the ethics level. It is a reference when allocating resources for the model. Compared to an internal document that contains a lot of details, an external document is more like a summary. External documents are more helpful to domain users and end users. The improvement of the documentation norms and processes will help improve AI transparency [36], [57], thereby solving the problem of democratisation of artificial intelligence.

A previous publication of AFR [59] mentioned an article on AI documentation that was published by Google in 2019 and defines the concept of the model card [60], which is a document that contains a brief description of the machine learning model/project. That article also describes the most basic component of the model card: model details, expected use cases, data, metrics, performance results, etc. The model card is a solution to the transparency of machine learning. Developers can use it to emphasise the advantages of the model and inform the end user of disadvantages to avoid inappropriate usage. For ordinary users, the model card provides simple explanations of complex technologies. It can help them quickly understand the model.

1.5 AI Stakeholder

Stakeholders can be roughly divided into three categories [61]:

- AI experts such as machine engineers and computer vision engineers
- Domain experts. They work in various fields and need to use artificial intelligence models to solve work or scientific research problems in this field.
- End users. Their work is related to AI. There is no need to use or train the model in their work. It is enough to have a basic understanding of the model.

Compared with AI experts, end users and domain experts may have more demand for AI documentation tools. This was confirmed in the following user study.

1.6 Research Question

Among the above three themes, transparency is relatively abstract and broad. The other two themes, visualisation and AI documents, can also be regarded as subcategories of transparency. In the research related to the life cycle management of AI models, model visualisation has attracted more attention, while AI documents, which are traditional SE topics and common tools in the development of traditional software, have received little attention in the AI field. In addition, in ING, attention has been paid to related research. Therefore, the goal of our research is to improve the democratisation

of AI by automatically generating AI documentation. Taking all the above into consideration, we defined our research questions:

How can the democratisation of AI be increased by automatically generating AI documents?
The sub-questions are as follows:

- RQ1: What content should be included in the AI document so that model users have the most basic understanding of the model?
- RQ2: Which information can be used to generate AI documents automatically? Which resource does this information come from?
- RQ3: How to design the tool to generate AI documents?
- RQ4: How to design a user learning assessment tool?

2. Component of the AI documents (RQ1)

By analysing and comparing the existing AI document solutions and guidelines in section 3.2, we chose to include project information, developer information, model information, data information, artifacts information, and version information in the automatic documents. The reasons are as follows:

In the paper [58], the author proposed a guide for establishing AI documents, namely the fact sheet. In interviews with AI developers, they summarised and screened 10 basic questions(see Table 2.2.1) that need to be answered when establishing a fact sheet. Summarising these ten questions, we found that their corresponding topics are models (uses, factors), data, artifacts (evaluation indicators and hyperparameters), and the interpretability of the model. The concept of model card proposed in the literature [60] also includes the content of model, data, and artifacts. The concept of the model card is more biased towards ordinary users, so unlike the fact sheet, it hides the interpretability of the model. The model card also describes the detailed information that needs to be included in each category:

- Model: Including developer information, model generation time, model version, model type, parameters, fairness conditions, reference information, licenses, etc.
- Intended use: Including main intended use, main intended user, and out-of-scope use cases
- Factors: Including relevant factors and evaluation factors
- Metrics: Including model performance evaluation, decision thresholds, and mutation methods
- Data: Including data set, motivation, preprocessing, and training data description

In addition to these topics, the model card also includes quantitative analysis and ethical considerations.

Due to the diversity of artificial intelligence projects, we expanded the content of the fact sheet and reduced the content of model reports to limit the content of manual input. In addition, considering the actual problems of the ING project stored in the GitLab private library, we also included the project information in the AI document.

Machine learning is different from traditional software development. The entire development process requires a lot of training. After repeated hyperparameter adjustments, a model with better performance will be gradually obtained, but how to compare with the previous model. How to record the pre-trained models. Take this into consideration, we also included model version information.

So, to sum up, we can already answer the research question： “What content should be included in the AI document so that model users have the most basic understanding of the model?”

A: Our AI documents contain the following information:

- Intended use: Main intended use, intended users, and out-of-scope use cases
- Factors and subgroups: Instrumentation, environment, group, and attributes
- Metrics: Metrics and their values

- Data: A short description and source, its categories and basic statistics information
- Developer: Name, contact email, and number of commits
- Project: Project name, ID, URL, and where to find the original document provided by the developer
- Versioning information: Model and its corresponding generation time, metrics, hyperparameters, and developers

Questions	Docmentation Section
1. What is this model for?	Model information(Intended use)
2. What domain was it designed for?	Model information(Intended use)
3. Information about the training data (if appropriate)?	Data information
4. Information about the model (if appropriate)?	Model information
5. What are the model's inputs and outputs?	Data information, and Model information
6. What are the model's performance metrics? (Accuracy, Bias, Robustness, Domain Shift, Other metrics that you think are appropriate for this model)	Artifacts (Metrics)
7. Information about the test set?	Data information
8. Can a user get an explanation of how your model makes it decisions?	Model explanation
9. In what circumstances does the model do particularly well	Model information (Metrics)
10. Based on your experience in what circumstances does the model perform poorly?	Model information (Metrics)

Table 2.2.1: Summarized and screened questions that need to be answered when establishing FactSheet[58], and their corresponding topics

3. Information and Sources (QR2)

All components of the AI document can be divided into two categories, which are automatically generated content:

- Metrics: Metrics and their value
- Data: A short description, source, its categories and basic statistics information
- Developer information: Name, contact email, number of commits
- Project: Project Name, ID, URL, where to find the original document provided by the developer
- Versioning information: Model, and its corresponding generation time, metrics, hyperparameters, developers

And what needs to be entered manually:

- Intended use: Main intended use, intended users, out-of-scope use cases
- Factors and subgroups: Instrumentation, environment, group, attributes

This section describes the sources for automatic generation, and why they were selected.

3.1 Versioning Information

Version control records a certain file at a specific time so that after the system is changed, the previous specific version can be called. Ad hoc data processing and pretraining are the fastest ways to get some ideas during machine learning projects, and the process of developing and using the model requires a lot of training. However, these pre-trained models, model modification, pipelines, data, and metrics will become complicated over time. Furthermore, when a problem arises, the machine learning engineer may need to go back and check the earlier version of the model, data, etc. Therefore, version control of models, data, and metrics is an important part of the AI model development. Currently, most developers manually record documents for version control. The disadvantage of this method is that it may cause confusion and retraining if it cannot be updated in time, even if there is a record. Therefore, we incorporate version control into our solution.

There are different types of version control, including source code version, data version, and model version. In the previous systematic mapping study of AI model lifecycle management, we found five studies on the topic of traceability/version control. Among them, the tool proposed by Vartak et al., ModelDB, is one of the earliest tools to solve the problem of model version control. It helps users to record the hyperparameters, metrics, and dataset information during training and can automatically track and index the model through SQL and visualisation methods. However, ModelDB is a management tool specifically for models built in scikit-learn and spark.ml [61]. Different from ModelDB, DisDAT [62] is a tool for data version control. It abstracts data into bundles, which are versioned, immutable collections of data, and then manages the bundles to achieve data version control. The entire artificial intelligence model development life cycle is about converting the raw data into a well-trained model. This complex process contains multiple steps including data processing, storing, and transformation. It may involve stand-alone servers, clouds or HPC clusters. [63] proposed a solution of tracking data during the entire life cycle while keeping the execution overhead low. [64] mainly discussed the reproducibility and traceability of deep neural network (DNN)–based multimedia analysis. In [65], the authors discussed the challenges of the trainability, providing a brief introduction of tracking the artifact, which includes source code, test results, and development plans in general. However, it only offers rough ideas rather than solutions or structured philosophical discussion.

Our AI model documentation solution aims to provide users and especially non-AI experts with an initial understanding of the basic AI model or project rather than focus on the internal structure of the model, source code, and other details of the model. Therefore, we choose to provide model versioning information in the document, as well as model-related hyperparameters and metrics information instead of source code or data versioning information. In the above literature, the solution provided by ModelDB is more suitable as a reference for our research. In this section, we compared existing model versioning tools similar to ModelDB. Finally, MLflow was selected as an auxiliary tool to generate versioning information in AI model documents.

The existing version control tools include DVC, Pachyderm, Sacred, Neptune, MLflow, Amazon SageMaker, Polyaxon, SigOpt, Cortex, Wandb, Come, etc.

DVC is a data pipeline building tool instead of only data versioning like its name. It is closely related to Git. Many developers currently use Git+GNU Make instead of DVC because DVC is more complicated than Make, and the advantage of DVC is not apparent. This may be an obstacle to the spread of DVC. Also, DVC sets each experiment as a branch, which may cause trouble for a large number of experiments. For our documentation tool, DVC can provide useful information, such as hyperparameter tracking in its 0.93 version [65], but this is not the main function of DVC. The primary understanding of AI projects or models does not require most of the information provided by DVC, such as the information of reproducible pipelines. This tool can ensure that sources like data, configuration, and code are in sync among the team members. It is more useful for the development team's work than as an auxiliary tool to our documentation tools.

MLflow [67] is a management tool for machine learning lifecycle management. Its function includes recording training artifacts, such as parameters, tags, metrics, etc.; recording conda or docker environments; and storing models in a unified format. All information will be stored in the current training document and can also be stored in databases such as MySQL, SQLite, etc. The use of MLflow is simple: Select a paragraph in the source, set the range with mlflow.start_run() and mlflow.end_run(), and use, for example, mlflow.log_param() or mlflow in the circled range.log_metric() and so on to record artifacts. The user can choose whether to set the argument of start_run(): experiment_id and run_name, and record the artifacts under the specified experiment. mlflow.log_model() can be used to store the model. Unlike ModelDB, which integrates tightly with SparkML and scikit-learn, MLflow is lightweight and compatible with many additional tools, platforms, and frameworks. Compared with other versioning tools, it provided limited information, but that information is exactly what we need in the AI documentation. Therefore, MLflow is a good choice as an auxiliary tool for our solution.

Sacred [68] is also a tool for configuring and logging ML experiments. It is relatively difficult to integrate. Compared with MLflow, it is not that lightweight. It offers source code versioning information, but it is unnecessary for our current AI documentation solution. Furthermore, to choose a tool that is not entirely popularised as the source of our solution, MLflow is more suitable since it has higher popularity and activity than Sacred [69].

Pachyderm, Neptune, Come, and Sagemaker are not open source, and Polyaxon is an enterprise-grade platform for large-scale deep learning applications. The popularity of the remaining tools is much lower than that of MLflow.

Since this project is being carried out at ING, we also consider the opinions of ING employees. Combining the above comparison and analysis, we decided to use MLflow as an auxiliary tool for our solution as AI documents to provide information about the model version.

3.2 Project Information

Git is a popular distributed version control system (VCS) that helps developers track source code by recording source code changes. Developers can easily call the code of the previous version. It can also ensure that code is in sync among team members. GitHub is a version control platform/repository with Git as the core that is used to store and review source code as well as manage and share projects. Similarly, GitLab is also a web repository based on Git. The main difference between the two is that GitHub has private repositories and shared repositories with private repositories of more than three people being charged; GitLab focuses on building private warehouses for free and can be deployed on their own servers.

’ ’The initial version of our tool is suitable for GitLab projects because this project was conducted through ING Bank, and ING’s projects are stored and shared through GitLab. Information about the project in the AI document, such as project ID, creation date, URL, contributors, etc., is obtained through the GitLab API.

3.3 Model Information

Model details include model developer, date, model version, citation/reference information, parameters, license, where to send questions or comments about the model, and more.

As described in section 2.2, GitLab projects are our target projects. Information such as date, contributor/developer information, contact email address, etc., are obtained through GitLab API. The citation/reference information, as well as the license information, is obtained by searching keywords on the files in the Gitlab repository. While the model information and artifacts, as described in section 2.1, are obtained through the version control tool MLflow.

3.4 Data Information

The data used for training is stored locally or online. Usually, fixed functions are called in the source code to access the data. You can use the Python standard library, pandas or NumPy. Therefore, we can automatically obtain data information by analysing the source code, and after obtaining the data, generate statistical information of the data through different functions.

In summary, we use the source code, markdown file, MLflow log information, Git information, and the content manually entered by the developer to create the model document

3.5 Intended Use and Factors

In addition to the above, according to the recommendations of [70], the model documentation also includes the intended use and factors. The content of these two parts is manually entered by the author. Through the description of "intended use", "intended user" and "out of scope" provided by the developer, the model document enables readers to quickly understand the purpose of the model and the application scenarios and methods of the non-model. "Factors" refer to all factors that may affect the performance of the model. The information in this section is the fairness factor that needs to be considered during model evaluation. This section includes four aspects: “groups, “inducements”, “environment”, and “attributes”. Among them, “groups” refers to data with the same characteristics, including but not limited to demographics and phenotype category, etc. For example, data annotation is done by those people. Regarding the “instrument”, for example, which camera was used to collect the image data, and what is its hardware indicator. An example of "environment" in computer vision is the light conditions in the surrounding environment during. “Factors” also include other technical attributes that lead to different model performance.

4. Implementation details (RQ3)

Figure 2.4.1 shows the main functions other than the GUI. "model_info()" first finds whether the specified model file exists in the git tree. If so, it will find the creation date and last modification date, author, and other information and store them. Then find the license document in the git tree, call the internal function "__check_if_file_contains_certain_strings()" to search for the document and check whether it contains "GPL", "BSD", "MIT", "Mozilla", "Apache" " , "LGPL" . model_info() returns the basic information about the model and the type of license. data_and_algorithm_info() searches the git tree to find if there is a source code file containing how to access the dataset. Through the analysis of the source code, check whether it contains "csv.reader", "loadtxt(", ".read_csv(""http://"".csv" and other patterns to know whether the data is stored online or locally. And use pandas to generate statistical information. citation_info() searches markdown documents in the repository and then checks whether there is a fixed regular expression to extract citation information. read_model() loads the checkpoint and reads the variables in it. The result will not be reflected in the model document, but saved in a separate document. extract_info_from_mlflow() searches and organizes Mlflow log files. Search for metrics, parameters, and models in the Git tree, extract their information, including generation date, quantity, value, etc. The information about models and related artifacts from one experiment will be saved together (see Figure 2.4.4). list_contributors() and list_projectInfo() extracted "contributors_username", "contributors_email", "number of submissions", "project", "project id", "created in", "web_url", "readme_url", and other information via Git API().

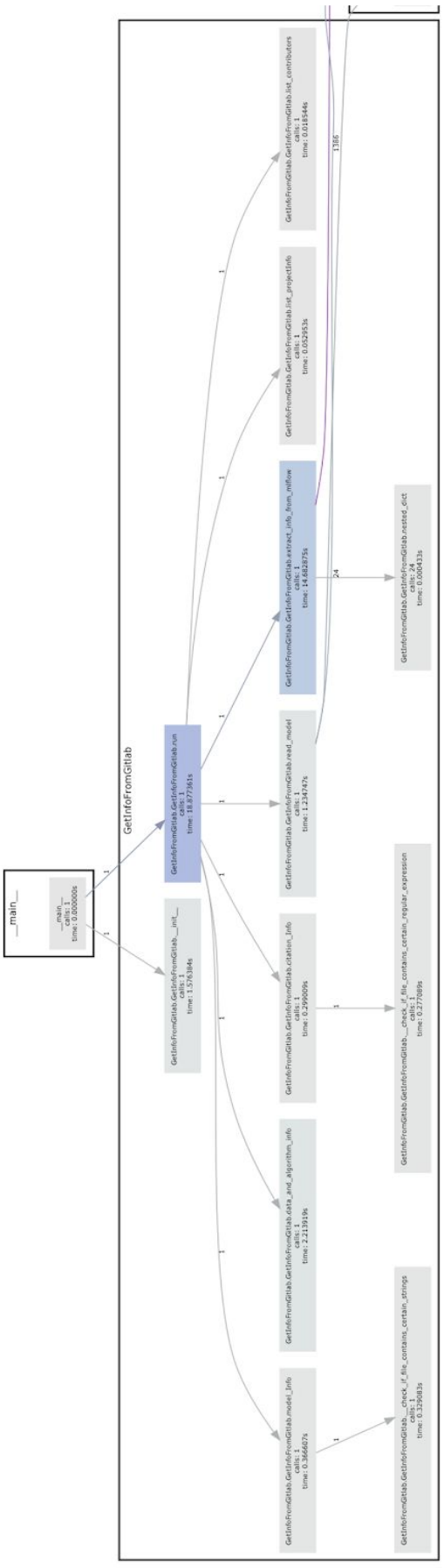


Figure 2.4.1: Workflow

The tool is packaged in a Python library that can be accessed at [70]. To use this tool, users need to install the library and run the script “example.py” under the repository where the model documentation is to be generated (please check the GitLab link above). In the first GUI, as a model developer, the user is asked to fill in relevant information about the intended use, factors, and subgroups. At the same time, users need to enter GitLab access information, such as project name, URL, and token, to access the private library on GitLab. If they have a specific checkpoint to check, the user can also provide the model name. After clicking the “Save” button, the tool will save the manually filled parts. After clicking the confirmation button in the “Access Information” section, the tool will analyse resources such as MLflow log information, Git information, source code, and called libraries, and will generate the following in the GUI:

- Intended use: Main intended use, intended users, and out-of-scope use cases
- Factors and subgroups: Instrumentation, environment, group, and attributes
- Metrics: Metrics and their values
- Data: A short description and source, its categories and basic statistics information
- Developer: Name, contact email, and number of commits
- Project: Project name, ID, URL, where to find the original document provided by the developer
- Versioning information: Model and its corresponding generation time, metrics, hyperparameters, and developers

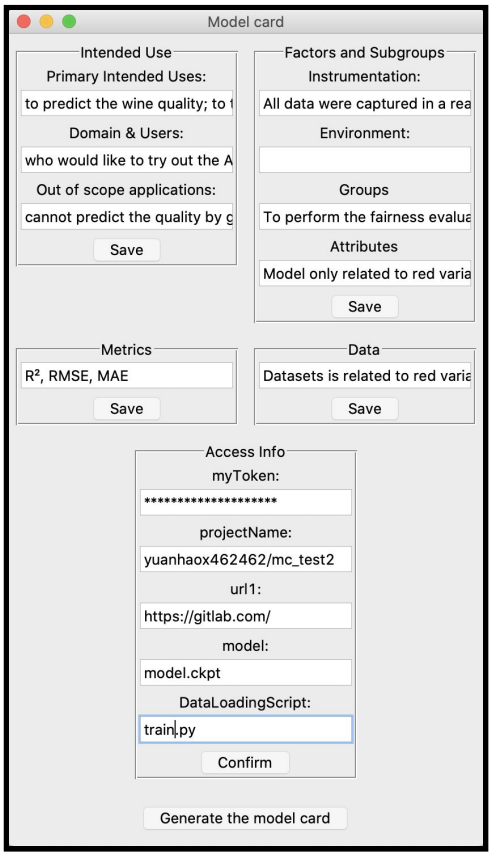


Figure 2.4.2: The GUI for manual input

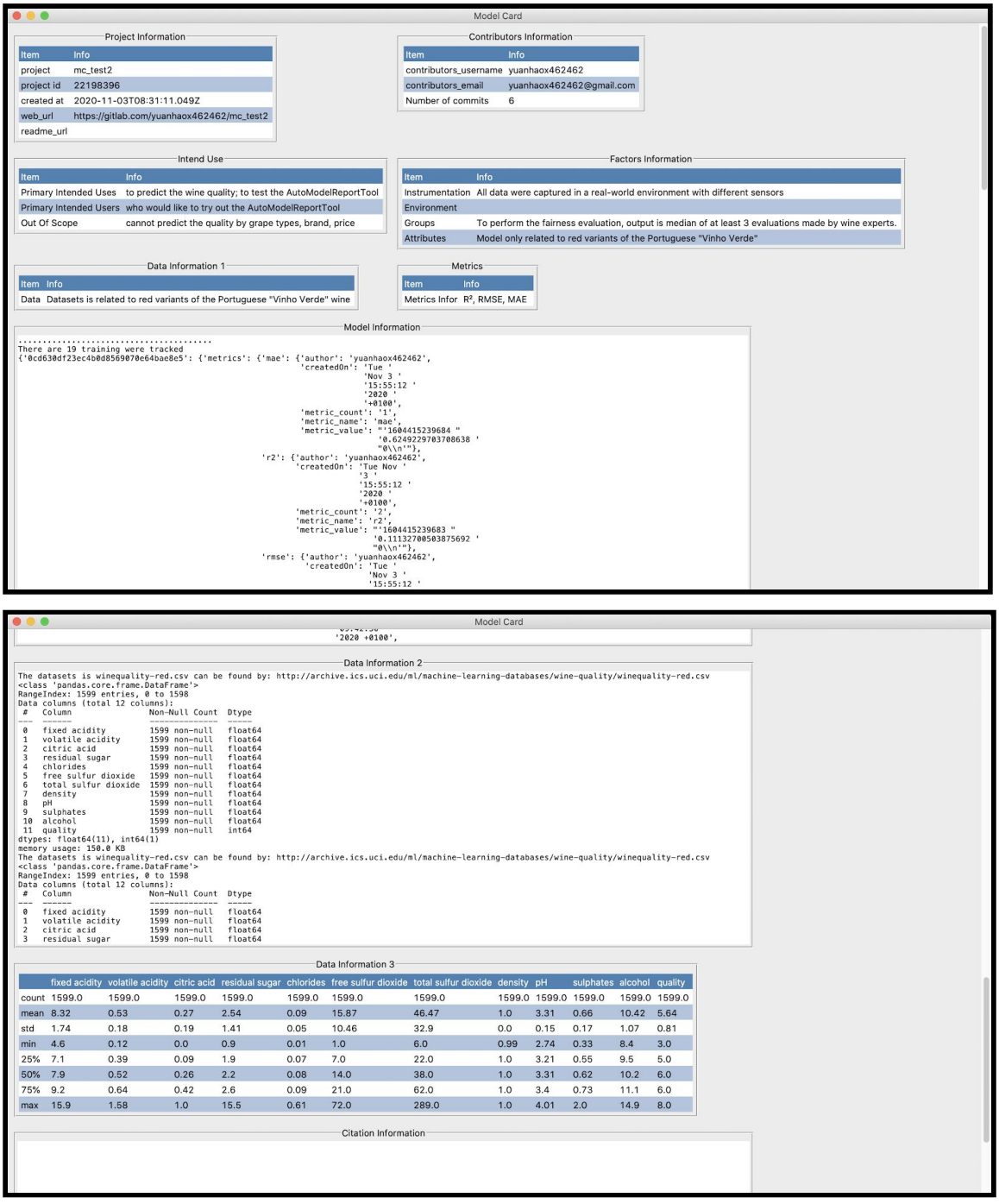


Figure 2.4.3: Generated document

```
There are 19 training were tracked
{'0cd630df23ec4b0d8569070e64bae8e5': {'metrics': {'mae': {'author': 'yuanhaox462462',
                                                         'createdOn': 'Tue '
                                                                      'Nov 3 '
                                                                      '15:55:12 '
                                                                      '2020 '
                                                                      '+0100',
                                                         'metric_count': '1',
                                                         'metric_name': 'mae',
                                                         'metric_value': "'1604415239684 "
                                                                         '0.6249229703708638 '
                                                                         "0\\n'"},
                                                  'r2': {'author': 'yuanhaox462462',
                                                         'createdOn': 'Tue Nov '
                                                                      '3 '
                                                                      '15:55:12 '
                                                                      '2020 '
                                                                      '+0100',
                                                         'metric_count': '2',
                                                         'metric_name': 'r2',
                                                         'metric_value': "'1604415239683 "
                                                                         '0.11132700503875692 '
                                                                         "0\\n'"},
                                                  'rmse': {'author': 'yuanhaox462462',
                                                           'createdOn': 'Tue '
                                                                        'Nov 3 '
                                                                        '15:55:12 '
                                                                        '2020 '
                                                                        '+0100',
                                                           'metric_count': '3',
                                                           'metric_name': 'rmse',
                                                           'metric_value': "'1604415239682 "
                                                                           '0.7919616045104533 '
                                                                           "0\\n'"}},
                                       'model': {1: {'author': 'yuanhaox462462',
                                                     'content': 'ElasticNet(l1_ratio=0.2, '
                                                                'random_state=42)',
                                                     'count': 1,
                                                     'createdOn': 'Tue Nov 3 '
                                                                  '15:55:12 '
                                                                  '2020 +0100',
                                                     'name': 'model.pkl'}},
                                       'params': {'alpha': {'author': 'yuanhaox462462',
                                                            'createdOn': 'Tue '
                                                                         'Nov 3 '
                                                                         '15:55:12 '
                                                                         '2020 '
                                                                         '+0100',
                                                            'params_count': '1',
                                                            'params_value': "'1.0'"},
                                                  'l1_ratio': {'author': 'yuanhaox462462',
                                                               'createdOn': 'Tue '
                                                                            'Nov '
                                                                            '3 '
                                                                            '15:55:12 '
                                                                            '2020 '
                                                                            '+0100',
                                                               'params_count': '2',
```

Figure 2.4.4: Versioning Information

5. User Study (RQ4)

5.1 Interviewee

We conducted user research to evaluate our prototype. Participants come from different industries, work in different positions, and have different AI knowledge levels. Since the interview involves the background investigation of the participants, to protect the privacy of the participants, we only list the overall statistical information of the company/school and industry where the interviewee is located. We follow the 'interviewee's wishes and hide the names of certain companies. Of the 14 interviewees, 6 were graduate students, PhDs, and research assistants from Delft University of Technology, Reno University of Nevada, ETH Zurich, and York University. The other six are from NXP, ING, Heineken, a semiconductor company, and a medical device company. Respondents' industries include electrical engineering, semiconductor, medicine, embedded system, materials science, banking, FMCG, and astronomy. Their positions include graduate student, doctor, research assistant, software engineer, data engineer, IC designer, algorithm engineer, and market analyst.

5.2 Interview and Questionnaire

5.2.1 Design

The whole interview is divided into four parts: pre-questions, Introduction and tool presentation, post-questions. Questions include single choice, multiple choice, quantitative scoring, description/short answer questions.

The first part is not an introduction to our tools and research, but directly asks interviewees about personal information, and AI-related experience, background and opinions. This is to prevent unfair guidance due to introduction of the background information. After completing the questionnaire survey, we introduced the motivation and the purpose of this project , that is, we completed the mapping study on artificial intelligence life cycle management, and carried out classification and theoretical research on the entire topic. Now, we would like to contribute from a practical perspective. In the mapping study, we classified the papers according to the main topics discussed, therefore, some potential topics are not included in the classification map, such as the democratization of AI. In addition, we learned from the results of cartographic research that visualization, transparency, and documentation can promote the democratization of artificial intelligence. Among them, transparency is an abstract and broad concept, and the other two themes can be used as subcategories of transparency. In addition, there are many researches on visualization, but few related to AI documents, so we decided to take AI documents as the direction of this research. We proposed a solution to automatically generate model documentation to improve the clarity of the model. The artificial intelligence documentation tool aims to help non-specialist quickly understand and use AI models, thereby promoting democratization. In the second part, in addition to the background introduction, we also introduced the content and usage of the model document. Finally, we present and test our tool in the same case project. After the demonstration is complete, in the third part, we will ask some questions that echo the previous questions and get feedback about the tool.

5.2.2 Pre-questionaire

1. How do you understand and use AI?
    a. I studied in school and used it at work.

b. I studied at school, but I have not used it at work.
c. Did not study in school, but used in work.
d. Never studied in school, not used for work, personal hobbies.

2. .What kind of data do you use AI to process?
    a. Numbers
    b. Image
    c. Both

3. How does your work relate to AI?
    a. Currently engaged in development work/research related to AI
    b. Sometimes used in current work/study
    c. No AI-related technology is currently used, but have plan to use
    d. No AI-related technology is currently used, the current work has nothing to do with AI, and there are no plans to use this AI

4. How long has the AI model/algorithm been used recently?
    a. Months ago
    b. Over a year ago
    c. Is using
5. Please describe in as much detail as possible the model you have used recently, or the most impressive model.

6. For the above models, do you think that the introduction documents (like markdown files) provided by the developers are sufficient for you to understand the purpose of the model, factors that affect the results, model evaluation indicators, data set information, etc. ? Or you still need to gather information from other sources (such as the source code) ?
    a. Yes, the documentation provides enough information
    b. No, I still need to find the information by myself
    c. No, since no documentation is provided

7. The clarity of the above model documents: (The range is from 1 to 5, where 1 is completely unclear and 5 is completely clear)

8. What are your pain points in AI development or usage?
    a. I want to adjust the algorithm, but I don't understand the algorithm
    b. Problems on training models and tuning parameters
    c. Version control: Lost model information, and its related artifact information
    d. Cannot find a suitable model to solve my problem
    e. They stated that the developer had provided insufficient information on considerations such as metadata, model use, and factors affecting performance
    f. Other pain poits.

9. 9. Do you think that a clearly described model document can solve the following problems?
    a. I want to adjust the algorithm, but I don't understand the algorithm
    b. Problems on training models and tuning parameters
    c. Version control: Lost model information, its related artifact information
    d. Cannot find a suitable model to solve my problem
    e. The developer did not provide sufficient information, such as data information, model usage
    f. The introduction document is not helpful

10. When sharing your well-trained/developed AI model with others, will you provide an introduction document?

11. Does your document contain the following information?

    a. Model details
    b. Intended use
    c. Metrics
    d. Factors
    e. Data
    f. Ethics related tips
    g. Quantitative analysis: some test results

12. As a developer, if there is a tool that can automatically generate documents for AI models, would you like to use it?
    a. Yes, Since it can help me record the model and share it with others to help them quickly understand my model
    b. Not sure, according to ease of use
    c. I don't want to use, since I have my own habits and recording methods
    d. No, I don't keep document for AI projects

13. For the item in a given link, can you quickly understand its model information, data set, metrics, etc.?

5.2.3 Post-questions

14. Through the model documentation provided by this tool, can you now quickly understand the relevant information of the above model? (The range is from 1 to 5, where 1 is completely unclear and 5 is completely clear)
15. Does the tool provide guidance on generating model documents? (The range is from 1 to 5, where 1 is completely not agree and 5 is completely agree)
16. Do you think this tool can solve the pain points you mentioned before?
17. Is the tool easy to use?(The range is from 1 to 5, where 1 is completely easy to use and 5 is very difficult to use)
18. Are there too many parts of the tool that need to be added manually?
19. As a developer, do you use this tool for your own use?
20. When sharing a trained/developed AI model with others, do you use this tool to generate a model card?
21. As a model user, do you want developers to provide such model cards?

6. Results and Discussion (checking done)

6.1 Pre-questions

All participants have used AI technology in their studies or work. 12 out of 14 participants' current work or study are related to AI. 85.7 percent of participants learned AI technology at school, the rest learned it during work. 6 out of 10 workers use AI as domain users and AI developers.

28.6 percent of participants only use AI to process images, 35.7 percent use it to digitize data, and the rest 35.7 percent use it for both kinds of data.

6 out of 14 participants are currently using artificial intelligence models in their work or study, while 5 others have used artificial intelligence models a few months ago. As for the remaining 3 participants, the last time they used the artificial intelligence model was at least a year ago.

Their descriptions of the model were very brief, including only the intended use of the model. Only one participant offered a brief description of the data.

For the model they described, only 14.3 percent of developers provided sufficient documentation; the remaining 85.7 percent of the developers either provided incomplete documentation, from which an intuitive understanding of the model was impossible, or provided no documentation. In evaluating the clarity of the limited information provided by the document, however, six participants awarded four points or more on a scale of one to five, these two are exactly. Therefore, the evidence we have obtained is that most documents provide incomplete information, but for limited information, its clarity is relatively high.

Participants' choice of pain points for model development and use covered all five options: (1) want to adjust the algorithm, but I don't understand the algorithm; (2) problems on training models and tuning parameters; (3) version control: Lost model information, and its related artifact information; (4) they could find no suitable model to solve their own problems; and (5) they stated that the developer had provided insufficient information on considerations such as metadata, model use, and factors affecting performance. One participant chose "other pain points" in answer to this question, listing resource issues such as training speed and computing power.

When asked whether a clearly described document could help solve the above pain points, the participants selected all the same options. However, not all participants felt that documentation could solve the pain points of developers who provided insufficient information, while some felt that documentation could also help them understand algorithms. In the post-question discussion, we found that participants had different understandings of the content covered by the documentation. Half said that the document should include more information like training guidelines to help them train their own data.

21.4 percent of the models provided to others included no documentation, The documents the other 78.6 percent of the participants provided included the seven following types of information: model details, intended usage, influencing factors, metadata, evaluation metrics, some training results, and ethics-related information. No participant chose ethics-related information. Only one participant said

that the document he provided would include all types of information except ethics, while the rest stated that they would provide two to five types of information. Information on the model itself and received more attention than information on the data set.

78.6 percent of participants said they would be willing to use semi-automatic document generation tools. The remaining 21.4 percent said they were unsure, and that it depended on the ease of use of the tool.

Regarding the GitHub project that was shown to them without documentation, three participants said that the information about the project was basically clear from browsing the source code. Three participants said that they were not completely sure but had some understanding. The remaining 8 participants either thought they could not understand the project at all, or could get very little understanding or were simply guessing from browsing other information.

6.2 Post-questions

After we presented our tool and had the participants try it out, all participants reported a better understanding of the model than before. 11 participants gave an evaluation of "mostly understood" or "completely understood." Two stated that they were not completely sure but had some understanding. One participant said it is still largely incomprehensible. In the post-question discussion, we asked that participant's opinion in detail, which will be explained later.

All participants agreed that the tool supplied guidance on generating AI model/project introduction documents. Seven participants awarded three points ("mostly clear guidance") or four points ("totally clear guidance").

Regarding the AI-development or use-process pain points that the tool solved, 11 of the 14 participants felt that this tool could solve the problem of insufficient information, 10 of the 14 participants participants felt that it could solve the problem of version control, and 6 of the 14 participants felt that the tool could help them find a model to suit their own problems.

Half of the participants thought the tool was easy to use, while the others held the opposite opinion. Three participants believed that the reason for the difficulty of use was too much manual input, and they don't really understand the item like instrumentation and groups. From the perspective of the model developer, some complaints are about version control, which requires calling the mlflow function in the source code. Although mlflow is easy to use, changing the source code is still something that model developers want to avoid.

11 of 14 participants expressed their willingness to use the tool when they need to share the model with others. Participants who were unwilling to use the tool reported that there might be environmental compatibility issues, that they did not want to spend time on documentation, or that they are more accustomed to recording documents in their own way.

There are also 11 participants who want model developers to provide such model documents when sharing models.

Suggestions for improving the tool itself included the following: (1) add the function of regular automatic saving and highlight the changed information; (2) add information such as GPU settings and training time to the original content; (3) add instructions for the tool itself; (4) improve the limitations; (5) add the visualization; and (6) add citation information.

7. Conclusion

In short, participants believe that model documentation tools can provide clear guidance for generating model documentation, and the generated model documentation can greatly improve their understanding of the model. This tool can solve the problem of insufficient model information, lack of or missing version information, and cannot quickly determine whether the model is suitable for their problems. Half of users think the tool is easy to use, while the rest think it is difficult to use. The reason is due to the design of the GUI, the manual input section is too detailed, and the participants have no relevant tools (such as Gitlab, Mlflow) experience.

Participants engaged in materials research said that the automatic generation of model cards is of little help to their work. Because for this industry, their main job is to simulate various materials at the atomic level to observe the stress response. As model users, the model they use is the most basic model and they make changes on this basis. There is no special Sota model specifically for their profession, and they do not use Sota's AI model during their work. In addition, as a developer, since the core technology of the industry is simulation modeling of various materials, data processing and source code are rarely shared. Therefore, if it is only for own use, the advantages of this tool are not great. Participants as domain users mentioned that the model is not easy to use, not because of the lack of introduction documents, but as domain users, setting up the environment is a big problem, and training itself is very difficult for them. Issues such as the lack of guidelines and guidance for the environment setup and parameters tuning have priority than the requirements of the model documentation.

According to the opinions of the participants, the future improvement directions of the model document itself include increasing GPU settings and training time, adding regular automatic update functions, emphasizing the version control information of the document itself, and so on. The model documentation tool is a prototype to improve democratization and transparency. Due to time constraints, it only implements some main functions, and has only been tested in multiple case projects, but not in real word projects. These need to be improved in future work.

References

1. McCarthy, J., Minsky, M.L., Rochester, N. and Shannon, C.E., 2006. A proposal for the dartmouth summer research project on artificial intelligence, august 31, 1955. AI magazine, 27(4), pp.12-12.

2. Holyoak, K.J., 1987. Parallel distributed processing: explorations in the microstructure of cognition. Science, 236, pp.992-997.

3. Dreyfus, H., Dreyfus, S.E. and Athanasiou, T., 2000. Mind over machine. Simon and Schuster.

4. Yanan, Z., 2019. Analysis of Artificial Intelligence Hypothesis from the Perspective of Philosophy.

5. Wang, J., 2017. Symbolism vs. Connectionism: A Closing Gap in Artificial Intelligence.

6. Welsh, R., 2019. Defining Artificial Intelligence. SMPTE Motion Imaging Journal, 128(1), pp.26-32.

7. Blenk, A., Kalmbach, P., Kellerer, W. and Schmid, S., 2017, August. O'zapft is: Tap your network algorithm's big data!. In Proceedings of the Workshop on Big Data Analytics and Machine Learning for Data Communication Networks (pp. 19-24).

8. Kocheturov, A., Pardalos, P.M. and Karakitsiou, A., 2019. Massive datasets and machine learning for computational biomedicine: trends and challenges. Annals of Operations Research, 276(1-2), pp.5-34.

9. AI For Fintech Research. 2020. icai. [ONLINE] Available at: https://se.ewi.tudelft.nl/ai4fintech/. [Accessed 11 November 2020].

10. Chaurasia, V. and Pal, S., 2014. Data mining approach to detect heart diseases. International Journal of Advanced Computer Science and Information Technology (IJACSIT) Vol, 2, pp.56-66.

11. Rashid, T.A., Abdullah, S.M. and Abdullah, R.M., 2016. An intelligent approach for diabetes classification, prediction and description. In Innovations in Bio-Inspired Computing and Applications (pp. 323-335). Springer, Cham.

12. Gao, Q. and Lin, M., 2013. Linguistic features and peer-to-peer loan quality: A machine learning approach. Available at SSRN.

13. Dixon, M., Klabjan, D. and Bang, J.H., 2017. Classification-based financial markets prediction using deep neural networks. Algorithmic Finance, 6(3-4), pp.67-77.

14. Li, J., Sun, L., Yan, Q., Li, Z., Srisa-An, W. and Ye, H., 2018. Significant permission identification for machine-learning-based android malware detection. IEEE Transactions on Industrial Informatics, 14(7), pp.3216-3225.

15. Cheng, R., Song, Y., Chen, D. and Ma, X., 2018. Intelligent positioning approach for high speed trains based on ant colony optimization and machine learning algorithms. IEEE Transactions on Intelligent Transportation Systems, 20(10), pp.3737-3746. and Machine Learning Algorithms

16. Khomh, F., Adams, B., Cheng, J., Fokaefs, M. and Antoniol, G., 2018. Software engineering for machine-learning applications: The road ahead. IEEE Software, 35(5), pp.81-84.

17. Arpteg, A., Brinne, B., Crnkovic-Friis, L. and Bosch, J., 2018, August. Software engineering challenges of deep learning. In 2018 44th Euromicro Conference on Software Engineering and Advanced Applications (SEAA) (pp. 50-59). IEEE.

18. Durelli, V.H., Durelli, R.S., Borges, S.S., Endo, A.T., Eler, M.M., Dias, D.R. and Guimaraes, M.P., 2019. Machine learning applied to software testing: A systematic mapping study. IEEE Transactions on Reliability, 68(3), pp.1189-1212.

19. Wiafe, I., Koranteng, F.N., Obeng, E.N., Assyne, N., Wiafe, A. and Gulliver, S.R., 2020. Artificial intelligence for cybersecurity: a systematic mapping of literature. IEEE Access, 8, pp.146598-146612.

20. Docs.microsoft.com. 2020. Introduction To Mlops And ML Lifecycle - Learn. [online] Available at: <https://docs.microsoft.com/en-us/learn/modules/start-ml-lifecycle-mlops/2-mlops-introduction> [Accessed 17 November 2020].

21. Miao, H., Li, A., Davis, L.S. and Deshpande, A., 2017, April. Modelhub: Deep learning lifecycle management. In 2017 IEEE 33rd International Conference on Data Engineering (ICDE) (pp. 1393-1394). IEEE.

22. Kumeno, F., 2019. Sofware engneering challenges for machine learning applications: A literature review. Intelligent Decision Technologies, 13(4), pp.463-476.

23. Machine Learning Life Cycle | DataRobot Artificial Intelligence Wiki. 2020. DataRobot. [ONLINE] Available at: https://www.datarobot.com/wiki/machine-learning-life-cycle/. [Accessed 11 November 2020].

24. Google Cloud. 2020. Machine learning workflow | AI Platform | Google Cloud. [ONLINE] Available at: https://www.datarobot.com/wiki/machine-learning-life-cycle/. [Accessed 11 November 2020].

25. Mathisen, B.M., Haro, P., Hanssen, B., Björk, S. and Walderhaug, S., 2016. Decision support systems in fisheries and aquaculture: A systematic review. arXiv preprint arXiv:1611.08374.

26. Vakkuri, V. and Abrahamsson, P., 2018, June. The key concepts of ethics of artificial intelligence. In 2018 IEEE International Conference on Engineering, Technology and Innovation (ICE/ITMC) (pp. 1-6). IEEE.

27. Sherin, S. and Iqbal, M.Z., 2019. A systematic mapping study on testing of machine learning programs. arXiv preprint arXiv:1907.09427.

28. Kumeno, F., 2019. Software engineering challenges for machine learning applications: A literature review. Intelligent Decision Technologies, 13(4), pp.463-476.

29. Keele, S., 2007. Guidelines for performing systematic literature reviews in software engineering (Vol. 5). Technical report, Ver. 2.3 EBSE Technical Report. EBSE.

30. Petersen, K., Feldt, R., Mujtaba, S. and Mattsson, M., 2008, June. Systematic mapping studies in software engineering. In 12th International Conference on Evaluation and Assessment in Software Engineering (EASE) 12 (pp. 1-10).

31. Petersen, K., Vakkalanka, S. and Kuzniarz, L., 2015. Guidelines for conducting systematic mapping studies in software engineering: An update. Information and Software Technology, 64, pp.1-18.

32. Dblp.org. 2020. Dblp: What Are The Criteria For Dblp To Index A Journal Or Conference?. [online] Available at: <https://dblp.org/faq/5210119.html> [Accessed 17 November 2020].

33. Reitz, F. and Hoffmann, O., 2010, September. An analysis of the evolving coverage of computer science sub-fields in the DBLP digital library. In International Conference on Theory and Practice of Digital Libraries (pp. 216-227). Springer, Berlin, Heidelberg.

34. Blog.scopus.com. 2020. About | Elsevier Scopus Blog. [online] Available at: <https://blog.scopus.com/about> [Accessed 17 November 2020].

35. Cavacini, A., 2015. What is the best database for computer science journal articles?. Scientometrics, 102(3), pp.2059-2071.

36. Experiences with Improving the Transparency of AI Models and Services

37. Murugesan, S., Malik, S., Du, F., Koh, E. and Lai, T.M., 2019. DeepCompare: Visual and interactive comparison of deep learning model performance. IEEE computer graphics and applications, 39(5), pp.47-59.

38. Kahng, M., Andrews, P.Y., Kalro, A. and Chau, D.H.P., 2017. A cti v is: Visual exploration of industry-scale deep neural network models. IEEE transactions on visualization and computer graphics, 24(1), pp.88-97.

39. Hu, Q., Ma, L. and Zhao, J., 2018, December. Deepgraph: A pycharm tool for visualizing and understanding deep learning models. In 2018 25th Asia-Pacific Software Engineering Conference (APSEC) (pp. 628-632). IEEE.

40. Xie, C., Qi, H., Ma, L. and Zhao, J., 2019, May. DeepVisual: a visual programming tool for deep learning systems. In 2019 IEEE/ACM 27th International Conference on Program Comprehension (ICPC) (pp. 130-134). IEEE.

41. Wongsuphasawat, K., Smilkov, D., Wexler, J., Wilson, J., Mane, D., Fritz, D., Krishnan, D., Viégas, F.B. and Wattenberg, M., 2017. Visualizing dataflow graphs of deep learning models in tensorflow. IEEE transactions on visualization and computer graphics, 24(1), pp.1-12.

42. Rashid, T.A., Abdullah, S.M. and Abdullah, R.M., 2016. An intelligent approach for diabetes classification, prediction and description. In Innovations in Bio-Inspired Computing and Applications (pp. 323-335). Springer, Cham.

43. Zhou, J. and Chen, F., 2018. 2D Transparency Space—Bring Domain Users and Machine Learning Experts Together. In Human and Machine Learning (pp. 3-19). Springer, Cham.

44. 2D Transparency Space - Bring Domain Users and Machine Learning Experts Together

45. Benrimoh, D., Israel, S., Perlman, K., Fratila, R. and Krause, M., 2018, June. Meticulous Transparency—An Evaluation Process for an Agile AI Regulatory Scheme. In International Conference on Industrial, Engineering and Other Applications of Applied Intelligent Systems (pp. 869-880). Springer, Cham.

46.Berscheid, J. and Roewer-Despres, F., 2019. Beyond transparency: a proposed framework for accountability in decision-making AI systems. AI Matters, 5(2), pp.13-22.

47. Thelisson, E., 2017, August. Towards Trust, Transparency and Liability in AI/AS systems. In IJCAI (pp. 5215-5216).

48. Fallon, C.K. and Blaha, L.M., 2018, July. Improving automation transparency: Addressing some of machine learning's unique challenges. In International Conference on Augmented Cognition (pp. 245-254). Springer, Cham.

49. Hind, M., Houde, S., Martino, J., Mojsilovic, A., Piorkowski, D., Richards, J. and Varshney, K.R., 2020, April. Experiences with Improving the Transparency of AI Models and Services. In Extended Abstracts of the 2020 CHI Conference on Human Factors in Computing Systems (pp. 1-8).

50. Raji, I.D. and Yang, J., 2019. ABOUT ML: Annotation and Benchmarking on Understanding and Transparency of Machine Learning Lifecycles. arXiv preprint arXiv:1912.06166.

51. Kuwajima, H., Tanaka, M. and Okutomi, M., 2019. Improving transparency of deep neural inference process. Progress in Artificial Intelligence, 8(2), pp.273-285.

52. Stoffel, F., 2018. Transparency in Interactive Feature-based Machine Learning: Challenges and Solutions (Doctoral dissertation).

53. Zhou, J., Khawaja, M.A., Li, Z., Sun, J., Wang, Y. and Chen, F., 2016. Making machine learning useable by revealing internal states update-a transparent approach. International Journal of Computational Science and Engineering, 13(4), pp.378-389.

54. Robillard, M.P., Marcus, A., Treude, C., Bavota, G., Chaparro, O., Ernst, N., Gerosa, M.A., Godfrey, M., Lanza, M., Linares-Vásquez, M. and Murphy, G.C., 2017, September. On-demand developer documentation. In 2017 IEEE International Conference on Software Maintenance and Evolution (ICSME) (pp. 479-483). IEEE.

55. Hind, M., Houde, S., Martino, J., Mojsilovic, A., Piorkowski, D., Richards, J. and Varshney, K.R., 2020, April. Experiences with Improving the Transparency of AI Models and Services. In Extended Abstracts of the 2020 CHI Conference on Human Factors in Computing Systems (pp. 1-8).

56. McBurney, P.W. and McMillan, C., 2014, June. Automatic documentation generation via source code summarization of method context. In Proceedings of the 22nd International Conference on Program Comprehension (pp. 279-290).

57. Gebru, T., Morgenstern, J., Vecchione, B., Vaughan, J.W., Wallach, H., Daumé III, H. and Crawford, K., 2018. Datasheets for datasets. arXiv preprint arXiv:1803.09010.

58. Arnold, M., Bellamy, R.K., Hind, M., Houde, S., Mehta, S., Mojsilović, A., Nair, R., Ramamurthy, K.N., Olteanu, A., Piorkowski, D. and Reimer, D., 2019. FactSheets: Increasing trust in AI services through supplier's declarations of conformity. IBM Journal of Research and Development, 63(4/5), pp.6-1.

59. Haakman, M., Cruz, L., Huijgens, H. and van Deursen, A., 2020. AI Lifecycle Models Need To Be Revised. An Exploratory Study in Fintech. arXiv preprint arXiv:2010.02716.

60. Mitchell, M., Wu, S., Zaldivar, A., Barnes, P., Vasserman, L., Hutchinson, B., Spitzer, E., Raji, I.D. and Gebru, T., 2019, January. Model cards for model reporting. In Proceedings of the conference on fairness, accountability, and transparency (pp. 220-229).

61. Gharibi, G., Walunj, V., Alanazi, R., Rella, S. and Lee, Y., 2019, June. Automated management of deep learning experiments. In Proceedings of the 3rd International Workshop on Data Management for End-to-End Machine Learning (pp. 1-4).

62. Yocum, K., Rowan, S., Lunt, J. and Wong, T.M., 2019. Disdat: Bundle Data Management for Machine Learning Pipelines. In 2019 {USENIX} Conference on Operational Machine Learning (OpML 19) (pp. 35-37).

63. Souza, R., Azevedo, L., Lourenço, V., Soares, E., Thiago, R., Brandão, R., Civitarese, D., Brazil, E., Moreno, M., Valduriez, P. and Mattoso, M., 2019, November. Provenance data in the machine learning lifecycle in computational science and engineering. In 2019 IEEE/ACM Workflows in Support of Large-Scale Science (WORKS) (pp. 1-10). IEEE.

64. Bailer, W., 2018, February. On the traceability of results from deep learning-based cloud services. In International Conference on Multimedia Modeling (pp. 620-631). Springer, Cham.

66. Dvc.org. 2020. [online] Available at: <https://dvc.org/blog/dvc-1-0-release> [Accessed 17 November 2020].

67. MLflow. 2020. Mlflow - A Platform For The Machine Learning Lifecycle. [online] Available at: <https://mlflow.org/> [Accessed 17 November 2020].

68. Sacred.readthedocs.io. 2020. Collected Information — Sacred 0.8.0 Documentation. [online] Available at: <https://sacred.readthedocs.io/en/stable/collected_information.html> [Accessed 17 November 2020].

69. Python.libhunt.com. 2020. Sacred Vs Mlflow | Libhunt. [online] Available at: <https://python.libhunt.com/compare-sacred-vs-mlflow> [Accessed 17 November 2020].

70. PyPI. 2020. Modelcard. [online] Available at: <https://test.pypi.org/project/modelcard/3.7/> [Accessed 17 November 2020].

71. Morley, J., Floridi, L., Kinsey, L. and Elhalal, A., 2020. From what to how: an initial review of publicly available AI ethics tools, methods and research to translate principles into practices. Science and engineering ethics, 26(4), pp.2141-2168.