Protocol for Systematic Literature Review on Model-driven Engineering for Software Systems with ML Components

Hira Naveed

March 10, 2023

1 Rationale

As a rapidly advancing field several branches of Artificial Intelligence (AI) have emerged, these include Machine Learning (ML), Natural Language Processing (NLP), Expert Systems and Robotics to name a few. ML is one of the subsets of AI which enables machines to perform specific tasks by learning from data and making informed decisions. Over the last decade, the availability of large volumes of data, advanced processing algorithms and high computational power have all contributed to the extensive popularity of ML. The widespread use of ML can be seen through several applications around us from face recognition on mobile phones to self-driving cars and more.

This systematic review will be the first of its kind to review literature on Model-driven Engineering approaches for software systems with ML components. Conducting an SLR on this topic will be beneficial for me in the following ways:

1. I would be able to systematically analyze the literature based on guidelines in [1]
2. I would get a thorough understanding of the existing work in the domain
3. I can identify future challenges and limitations in existing studies and address one of them in my PhD
4. I can identify the top venues to publish in this area
5. Get a publication

2 Research Questions

This section describes the PICOC and research questions defined for this SLR.

Population	Software systems with ML components
Intervention	Model-driven engineering approaches for software systems with ML components
Comparison	Not applicable
Outcomes	The consequence of using MDE for software systems with ML
Context	*Include:* MDE approaches for software systems with MLcomponents *Exclude:* AI approaches automating, recommending, enhancing the MDE process, MDE approaches for systems with AI components other than ML, Model-based testing approaches for ML systems

Table 1: PICOC for this SLR

RQ1. **What are the motivations and goals behind applying Model-driven engineering to software systems with ML components?**

(a) What is the motivation/goals behind the presented approach?

(b) What are the benefits of the approach presented in the study?

(c) What type of ML systems does the study target?
These can be software systems with ML components related to Machine Learning, Deep Learning, Computer Vision, Reinforcement learning or any other subset of ML. [2]

(d) What software platform(s) does the study target?
The target platforms can be mobile, desktop, Web, IoT, CPS or cross-platform.

(e) What is the domain of this study?
Some possible target domains can be health, education, finance, and retail.

(f) Who are the target users of the approach in the study?
The target users of the approach presented in the study can be MDE experts, ML experts, Software engineers, Domain experts or end-users.

(g) What are the final outcomes of the study?
The outcomes of a study can include one or more of the following: framework, guidelines, techniques, tools, models, modeling language(s), evaluations result of an MDE approach.

RQ2. **What are the Model-driven engineering approaches for ML components-based systems presented in the literature?**

(a) What information has been modeled?

(b) Are the models expressed visually or textually?

(c) What are the domain models used for modeling?

(d) What modeling language is used?

(e) How many and what types of models are created (requirements model/data model/process model etc)?

(f) What is the automation level supported by the transformations in the study?

- If the study does M2T transformation, is the code usable? Is it a runnable application or does it need manual effort to work?

(g) What are the MDE phases considered in this study?
These phases can be Model to Model Transformation (M2M) or Model to Code Transformation (M2C) or both.

- If the study does M2T transformation, how are models converted to code? What generator is used?
- If the study does M2T transformation, what is the completeness of the generated solution?

(h) What are the artifacts generated by the MDE approach? These can be models, code, scripts, configurations and more.

(i) Does the approach support forward engineering or reverse engineering?

(j) What phases of ML software system development are considered in this study? These phases can be Problem Scoping, Data Acquisition, Data Exploration, Modelling, Training, Evaluation or any combination of them. GUI for ML?

(k) Does the approach consider training data for the ML components?

(l) What are the ML libraries used in the study? E.g. tensorflow, Keras, Pandas etc

(m) How does the approach support the engineering of software systems with ML components? This can be a framework, guidelines, model transformer, modelling language, code generator etc.

(n) Does the MDE approach in the study consider responsible AI principles?

If yes, which principles does the study consider? This can be fairness, trustworthiness, privacy, explainability and more.

RQ3. **How are existing studies on Model-driven engineering for ML components-based systems evaluated and what are the tools available?**

(a) How is the approach evaluated?

(b) What examples/case studies is the approach applied to? This can be academic or industrial case studies.

(c) Are the tools available?

(d) Is the data for the tool available?

(e) What are the benchmark datasets used for evaluation?

(f) What are the meta-tools and frameworks, the tool is built on? E.g. EMF, Eclipse IDE

(g) Is the work presented in the study practical? Is there a demonstration? (Practically usable?)

(h) What are the metrics used for the evaluation of ML systems?

(i) What are the metrics used for the evaluation of the MDE approach?

RQ4. **What is the quality of existing research studies on Model-driven engineering for software systems with ML components and what are the gaps?**

(a) What are the strengths of this study?

(b) Is the MDE approach in the study scalable?

(c) Are the MDE models in the study reusable?

(d) What are the limitations of the study discussed in the paper?

(e) What are the limitations/gaps of the study not discussed in the paper?

(f) What is the recommended future work in the study?

3 Search Strategy

This section explains the digital libraries I will be using to find relevant studies along with the initial papers I found. The section will also include the search string and searching strategy for this SLR.

The search string:

("machine learning" OR "supervised learning" OR "unsupervised learning" OR "semi-supervised learning" OR "reinforcement learning" OR "deep learning" OR "ensemble learning" OR "neural network" OR "naive bayes" OR "decision tree" OR "deep boltzmann machine" OR "deep belief network" OR "convolutional neural network" OR "recurrent neural network" OR "generative adversarial network" OR "auto encoder" OR "gradient boosted regression trees" OR "adaboost" OR "gradient boosting machine" OR "random forest" OR "perceptron" OR "back propagation" OR "hopfield network" OR "radial basis function network" OR "linear regression" OR "stepwise regression" OR "logistic regression" OR "least square regression" OR "adaptive regression" OR "locally estimated scatterplot" OR "k means clustering" OR "k medians clustering"

OR "hierarchical clustering" OR "mean shift clustering" OR "expectation maximization" OR "gaussian naïve bayes" OR "multinomial naïve bayes" OR "bayesian network" OR "bayesian belief network" OR "k nearest neighbour" OR "learning vector quantization" OR "self organizing map" OR "locally weighted learning" OR "transfer learning" OR "support vector machines" OR "classification and regression tree" OR "CHAID" OR "conditional decision tree" OR "decision stump" OR "long short term memory network" OR "gaussian mixture" OR "hidden markov model" OR "Q learning" OR "temporal difference learning" OR "dimensionality reduction") AND ("model driven development" OR "model driven engineering" OR "model based software engineering" OR "model driven software engineering" OR "model transformation" OR "model driven architecture" OR "low code solution" OR "low code application" OR "low code applications" OR "low code paradigm" OR "low code approach" OR "low code platform" OR "low code development" OR "low code/no code" OR "no code/low code" OR "no code development platform")

Six databases have been selected for this study: IEEE Xplore, ACM Digital Library, Wiley, Web of Science and Springer and Scopus. The search criteria are restricted to research articles (conference papers, workshop papers and journal articles) without any time limit. Considering the differences in each database, I have refined the search string to find the most relevant studies for this SLR. Search strings for each database are shown in in the table below.

Database	Database Specific Search String
IEEE Xplore	("machine learning" OR "supervised learning" OR "unsupervised learning" OR "semi-supervised learning" OR "reinforcement learning" OR "deep learning" OR "ensemble learning" OR "neural network" OR "naive bayes" OR "decision tree" OR "deep belief network" OR "convolutional neural network" OR "recurrent neural network" OR "generative adversarial network" OR "auto encoder" OR "gradient boosted regression trees" OR "adaboost" OR "gradient boosting machine" OR "random forest" OR "perceptron" OR "back propagation" OR "radial basis function network" OR "linear regression" OR "stepwise regression" OR "logistic regression" OR "least square regression" OR "adaptive regression" OR "k means clustering" OR "k medians clustering" OR "hierarchical clustering" OR "mean shift clustering" OR "expectation maximization" OR "gaussian naïve bayes" OR "multinomial naïve bayes" OR "bayesian network" OR "bayesian belief network" OR "k nearest neighbour" OR "learning vector quantization" OR "locally weighted learning" OR "transfer learning" OR "support vector machines" OR "classification and regression tree" OR "conditional decision tree" OR "decision stump" OR "long short term memory network" OR "gaussian mixture" OR "hidden markov model" OR "Q learning" OR "temporal difference learning" OR "dimensionality reduction") AND ("model driven development" OR "model driven engineering" OR "model based software engineering" OR "model driven software engineering" OR "model transformation" OR "model driven architecture" OR "low code solution" OR "low code application" OR "low code applications" OR "low code paradigm" OR "low code approach" OR "low code platform" OR "low code development" OR "low code/no code" OR "no code/low code" OR "no code development platform")
ACM Digital Library	[[All: "machine learning"] OR [All: "supervised learning"] OR [All: "unsupervised learning"] OR [All: "semi-supervised learning"] OR [All: "reinforcement learning"] OR [All: "deep learning"] OR [All: "ensemble learning"] OR [All: "neural network"] OR [All: "naive bayes"] OR [All: "decision tree"] OR [All: "deep boltzmann machine"] OR [All: "deep belief network"] OR [All: "convolutional neural network"] OR [All: "recurrent neural network"] OR [All: "generative adversarial network"] OR [All: "auto encoder"] OR [All: "gradient boosted regression trees"] OR [All: "adaboost"] OR [All: "gradient boosting machine"] OR [All: "random forest"] OR [All: "perceptron"] OR [All: "back propagation"] OR [All: "hopfield network"] OR [All: "radial basis function network"] OR [All: "linear regression"] OR [All: "stepwise regression"] OR [All: "logistic regression"] OR [All: "least square regression"] OR [All: "adaptive regression"] OR [All: "locally estimated scatterplot"] OR [All: "k means clustering"] OR [All: "k medians clustering"] OR [All: "hierarchical clustering"] OR [All: "mean shift clustering"] OR [All: "expectation maximization"] OR [All: "gaussian naïve bayes"] OR [All: "multinomial naïve bayes"] OR [All: "bayesian network"] OR [All: "bayesian belief network"] OR [All: "k nearest neighbour"] OR [All: "learning vector quantization"] OR [All: "self organizing map"] OR [All: "locally weighted learning"] OR [All: "transfer learning"] OR [All: "support vector machines"] OR [All: "classification and regression tree"] OR [All: "chaid"] OR [All: "conditional decision tree"] OR [All: "decision stump"] OR [All: "long short term memory network"] OR [All: "gaussian mixture"] OR [All: "hidden markov model"] OR [All: "q learning"] OR [All: "temporal difference learning"] OR [All: "dimensionality reduction"]] AND [[All: "model driven development"] OR [All: "model driven engineering"] OR [All: "model based software engineering"] OR [All: "model driven software engineering"] OR [All: "model transformation"] OR [All: "model driven architecture"] OR [All: "low code solution"] OR [All: "low code application"] OR [All: "low code applications"] OR [All: "low code paradigm"] OR [All: "low code approach"] OR [All: "low code platform"] OR [All: "low code development"] OR [All: "low code/no code"] OR [All: "no code/low code"] OR [All: "no code development platform"]]
Springer	'("machine learning" OR "supervised learning" OR "unsupervised learning" OR "semi-supervised learning" OR "reinforcement learning" OR "deep learning" OR "ensemble learning" OR "neural network" OR "naive bayes" OR "decision tree" OR "deep boltzmann machine" OR "deep belief network" OR "convolutional neural network" OR "recurrent neural network" OR "generative adversarial network" OR "auto encoder" OR "gradient boosted regression trees" OR "adaboost" OR "gradient boosting machine" OR "random forest" OR "perceptron" OR "back propagation" OR "hopfield network" OR "radial basis function network" OR "linear regression" OR "stepwise regression" OR "logistic regression" OR "least square regression" OR "adaptive regression" OR "locally estimated scatterplot" OR "k means clustering" OR "k medians clustering" OR "hierarchical clustering" OR "mean shift clustering" OR "expectation maximization" OR "gaussian naïve bayes" OR "multinomial naïve bayes" OR "bayesian network" OR "bayesian belief network" OR "k nearest neighbour" OR "learning vector quantization" OR "self organizing map" OR "locally weighted learning" OR "transfer learning" OR "support vector machines" OR "classification and regression tree" OR "CHAID" OR "conditional decision tree" OR "decision stump" OR "long short term memory network" OR "gaussian mixture" OR "hidden markov model" OR "Q learning" OR "temporal difference learning" OR "dimensionality reduction") AND ("model driven development" OR "model driven engineering" OR "model based software engineering" OR "model driven software engineering" OR "model transformation" OR "model driven architecture" OR "low code solution" OR "low code application" OR "low code applications" OR "low code paradigm" OR "low code approach" OR "low code platform" OR "low code development" OR "low code/no code" OR "no code/low code" OR "no code development platform")'
Wiley	"(("machine learning" OR "supervised learning" OR "unsupervised learning" OR "semi-supervised learning" OR "reinforcement learning" OR "deep learning" OR "ensemble learning" OR "neural network" OR "naive bayes" OR "decision tree" OR "deep belief network" OR "convolutional neural network" OR "recurrent neural network" OR "generative adversarial network" OR "auto encoder" OR "gradient boosted regression trees" OR "adaboost" OR "gradient boosting machine" OR "random forest" OR "perceptron" OR "back propagation" OR "linear regression" OR "stepwise regression" OR "logistic regression" OR "least square regression" OR "adaptive regression" OR "k means clustering" OR "k medians clustering" OR "hierarchical clustering" OR "mean shift clustering" OR "expectation maximization" OR "gaussian naïve bayes" OR "multinomial naïve bayes" OR "bayesian network" OR "bayesian belief network" OR "k nearest neighbour" OR "learning vector quantization" OR "locally weighted learning" OR "transfer learning" OR "support vector machines" OR "classification and regression tree" OR "conditional decision tree" OR "decision stump" OR "long short term memory network" OR "gaussian mixture" OR "hidden markov model" OR "Q learning" OR "temporal difference learning" OR "dimensionality reduction") AND ("model driven development" OR "model driven engineering" OR "model based software engineering" OR "model driven software engineering" OR "model transformation" OR "model driven architecture" OR "low code solution" OR "low code application" OR "low code applications" OR "low code paradigm" OR "low code approach" OR "low code platform" OR "low code development" OR "low code/no code" OR "no code/low code" OR "no code development platform"))" anywhere

Scopus	(TITLE-ABS-KEY("machine learning" OR "supervised learning" OR "unsupervised learning" OR "semi-supervised learning" OR "reinforcement learning" OR "deep learning" OR "ensemble learning" OR "neural network" OR "naive bayes" OR "decision tree" OR "deep belief network" OR "convolutional neural network" OR "recurrent neural network" OR "generative adversarial network" OR "auto encoder" OR "gradient boosted regression trees" OR "adaboost" OR "gradient boosting machine" OR "random forest" OR "perceptron" OR "back propagation" OR "radial basis function network" OR "linear regression" OR "stepwise regression" OR "logistic regression" OR "least square regression" OR "adaptive regression" OR "k means clustering" OR "k medians clustering" OR "hierarchical clustering" OR "mean shift clustering" OR "expectation maximization" OR "gaussian naïve bayes" OR "multinomial naïve bayes" OR "bayesian network" OR "bayesian belief network" OR "k nearest neighbour" OR "learning vector quantization" OR "locally weighted learning" OR "transfer learning" OR "support vector machines" OR "classification and regression tree" OR "conditional decision tree" OR "decision stump" OR "long short term memory network" OR "gaussian mixture" OR "hidden markov model" OR "Q learning" OR "temporal difference learning" OR "dimensionality reduction") AND TITLE-ABS-KEY("model driven development" OR "model driven engineering" OR "model based software engineering" OR "model driven software engineering" OR "model transformation" OR "model driven architecture" OR "low code solution" OR "low code application" OR "low code applications" OR "low code paradigm" OR "low code approach" OR "low code platform" OR "low code development" OR "low code/no code" OR "no code/low code" OR "no code development platform"))
Web of Science (WoS)	("machine learning" OR "supervised learning" OR "unsupervised learning" OR "semi-supervised learning" OR "reinforcement learning" OR "deep learning" OR "ensemble learning" OR "neural network" OR "naive bayes" OR "decision tree" OR "deep belief network" OR "convolutional neural network" OR "recurrent neural network" OR "generative adversarial network" OR "auto encoder" OR "gradient boosted regression trees" OR "adaboost" OR "gradient boosting machine" OR "random forest" OR "perceptron" OR "back propagation" OR "radial basis function network" OR "linear regression" OR "stepwise regression" OR "logistic regression" OR "least square regression" OR "adaptive regression" OR "k means clustering" OR "k medians clustering" OR "hierarchical clustering" OR "mean shift clustering" OR "expectation maximization" OR "gaussian naïve bayes" OR "multinomial naïve bayes" OR "bayesian network" OR "bayesian belief network" OR "k nearest neighbour" OR "learning vector quantization" OR "locally weighted learning" OR "transfer learning" OR "support vector machines" OR "classification and regression tree" OR "conditional decision tree" OR "decision stump" OR "long short term memory network" OR "gaussian mixture" OR "hidden markov model" OR "Q learning" OR "temporal difference learning" OR "dimensionality reduction") (All Fields) AND ("model driven development" OR "model driven engineering" OR "model based software engineering" OR "model driven software engineering" OR "model transformation" OR "model driven architecture" OR "low code solution" OR "low code application" OR "low code applications" OR "low code paradigm" OR "low code approach" OR "low code platform" OR "low code development" OR "low code/no code" OR "no code/low code" OR "no code development platform") (All Fields)
Science Direct	("machine learning" OR "supervised learning" OR "unsupervised learning" OR "neural network" OR "reinforcement learning") AND ("model driven engineering" OR "model driven architecture" OR "low code" OR "low code/no code")

Database	Papers Found	Duplicates Removed
IEEE Xplore	142	132
ACM Digital Library	547	529
Springer	1948	1934
Wiley	270	261
Scopus	363	196
Web of Science (WoS)	230	87
Science Direct	434	431

Table 2: Initial Papers found w.r.t. Databases (March 2023)

The filtering process is conducted in a series of steps, I will begin by removing duplicates from the pool of potential papers. After that I will screen papers based on title and abstract and remove irrelevant ones. Then, screen papers again skimming through the entire text. Finally, I would be doing forward and backward snowballing to find any missed out papers.

Results from first screening:

WoS	4 green, 2 yellow	/87
Wiley	0 green, 2 yellow	/261
Scopus	12 green, 2 yellow	/196
IEEE	16 green, 2 yellow	/132
ACM	22 green, 9 yellow	/529
Springer	12 green, 11 yellow	/1934
ScienceDirect	6 green, 2 yellow	/431
Total	72 green, 30 yellow	

Results from second screening:

WoS	2 green, 0 yellow	/87

Wiley	0 green, 0 yellow	/261
Scopus	10 green, 0 yellow	/196
IEEE	12 green, 0 yellow	/132
ACM	16 green, 1 yellow	/529
Springer	9 green, 0 yellow	/1934
ScienceDirect	6 green, 1 yellow	/431
Total	55 green, 2 yellow	

Endnote has been used for duplicate removal and Mendley will be used as the reference management tool.

4 Study Selection

This section details the inclusion and exclusion criteria for this SLR.

4.1 Inclusion Criteria

1. Papers about MDE for software systems with ML components
2. Full text of the article is available
3. Studies that have been used in academia with references from literature

4.2 Exclusion Criteria

1. Papers about Machine Learning (or its subsets) but not using an MDE approach
2. Papers about MDE for any subset of AI other than ML.
3. Papers about model-based testing of systems with ML components.
4. Studies leveraging AI to improve, enhance, or automate MDE
5. Short papers that are less than four pages
6. Studies not in English language
7. Conference or workshop papers if an extended journal version of the same paper exists
8. Papers with inadequate information to extract (Irrelevant Papers)
9. Non-primary studies (Secondary or Tertiary Studies)
10. Gray literature (unpublished work), books, book chapters, posters, discussions, opinions and keynotes, magazine articles, opinion papers, experience papers, comparison papers (only compare two existing approaches).
11. Vision papers with no concrete implementation

5 Data Extraction

This section describes the qualitative data to be extracted from the studies.

1. Publication details — authors, title, demographics, year, venue, citation count, publisher.
2. What is the motivation/goal of this study?
3. What is the benefit of this study?
4. What type of ML systems are considered in the study?
5. What software platform(s) does the study target?
6. What is the domain of the study?
7. Who are the target end users of the approach presented in the paper?
8. What are the final outcomes of the study?
9. What information has been modeled in the study?
10. Are the models expressed textually or visually?
11. What are the domain models used in the study?
12. What modeling language is used in the study?
13. How many models are created and what is the type e.g. data models, requirements models, design models, business process models etc.?
14. What is the automation level of the model transformations in the study?
15. What is the usability of the generated code? Does it need manual effort to work?
16. What are the MDE phases considered in this study?
17. How are models converted to code?
18. What is the completeness of the generated solution?
19. What are the artifacts generated by the approach?
20. Does the approach support forward engineering or reverse engineering?
21. What phase(s) of ML development does the study support?
22. Does the approach consider training data for ML components?
23. How does the MDE approach in the study support development of ML systems?

24. Does the study consider responsible AI characteristics? If yes, which ones?
25. What are the human-centric aspects or human values considered in the study?
26. What methods have been used to evaluate the study?
27. What type of examples and/or case studies are used in the study, are they academic or industrial?
28. Are the tools mentioned in the study available?
29. Is data for the tools mentioned in the study available?
30. What are the benchmark datasets used for evaluation?
31. What are the meta-tools and frameworks the tool is built on? E.g. EMF, Eclipse IDE
32. Is the work presented in the study practically usable?
33. What are the metrics used for the evaluation of ML systems?
34. What are the metrics used for the evaluation of the MDE approach?

35. What are the strengths of the study?
36. Is the MDE approach scalable to large systems?
37. Are the MDE models in the study reusable?
38. What are the limitations of the study discussed in the paper?
39. What are the limitations of the study not mentioned in the paper?
40. What are the research gaps and future challenges/opportunities reported in the study?

6 Data Synthesis

Quantitative data synthesis will be conducted to identify patterns and trends in the data, for example, x% of studies develop their own DSML whereas y% use UML. Identifying the number of studies per year, the types of intelligent systems generated from models, the types of modeling languages and more according to the information extracted from the studies.

Information that is deemed important to share with the research community will be included as qualitative data. This will include interesting papers from each category, in the discussion section, for example, interesting relationships, major findings and gaps.

7 References

[1] Kitchenham, Barbara, O. Pearl Brereton, David Budgen, Mark Turner, John Bailey, and Stephen Linkman., "Systematic literature reviews in software engineering–a systematic literature review," *Information and Software Technology,* vol. 51, 2009.

[2] Azimi, Mohsen, Armin Dadras Eslamlou, and Gokhan Pekcan., "Data-Driven Structural Health Monitoring and Damage Detection through Deep Learning: State-of-the-Art Review," *sensors,* 2020.