

Conceptualization and Development of ML-based Recommender Systems for Software Engineering



Candidate: Claudio Di Sipio

Advisor: Prof. Davide Di Ruscio

Co-Advisors:

Dr. Juri Di Rocco

Dr. Phuong Thanh Nguyen

Università degli studi dell'Aquila / Italy

XXXV Doctoral cycle

a.a. 2021/2022

About me

- 2016-2018: Master degree at University of l'Aquila
- 2018-2019: Research fellowship at Univaq
 - API recommendation by using code cloning techniques
- 2019-2023: Ph.D. in Computer Science



Hobbies: running, play tennis, writing novels 😊

- Recommender system for software engineering
 - Categorization of OSS repositories
 - API function call recommendations
- Model-Driven engineering
 - Modeling assistants
 - MDE approach to design recommender systems
- Quality aspects in recommender systems
 - Adversarial attacks
 - Bias and fairness



Davide Di Ruscio
Associate professor



Juri Di Rocco
Assistant professor



Phuong T. Nguyen
Assistant professor



Riccardo Rubei
Post-doc researcher

Collaborations



Massimiliano Di Penta
Full professor
University of Sannio



Cezar Sas
Ph.D. student
University of Gronigen

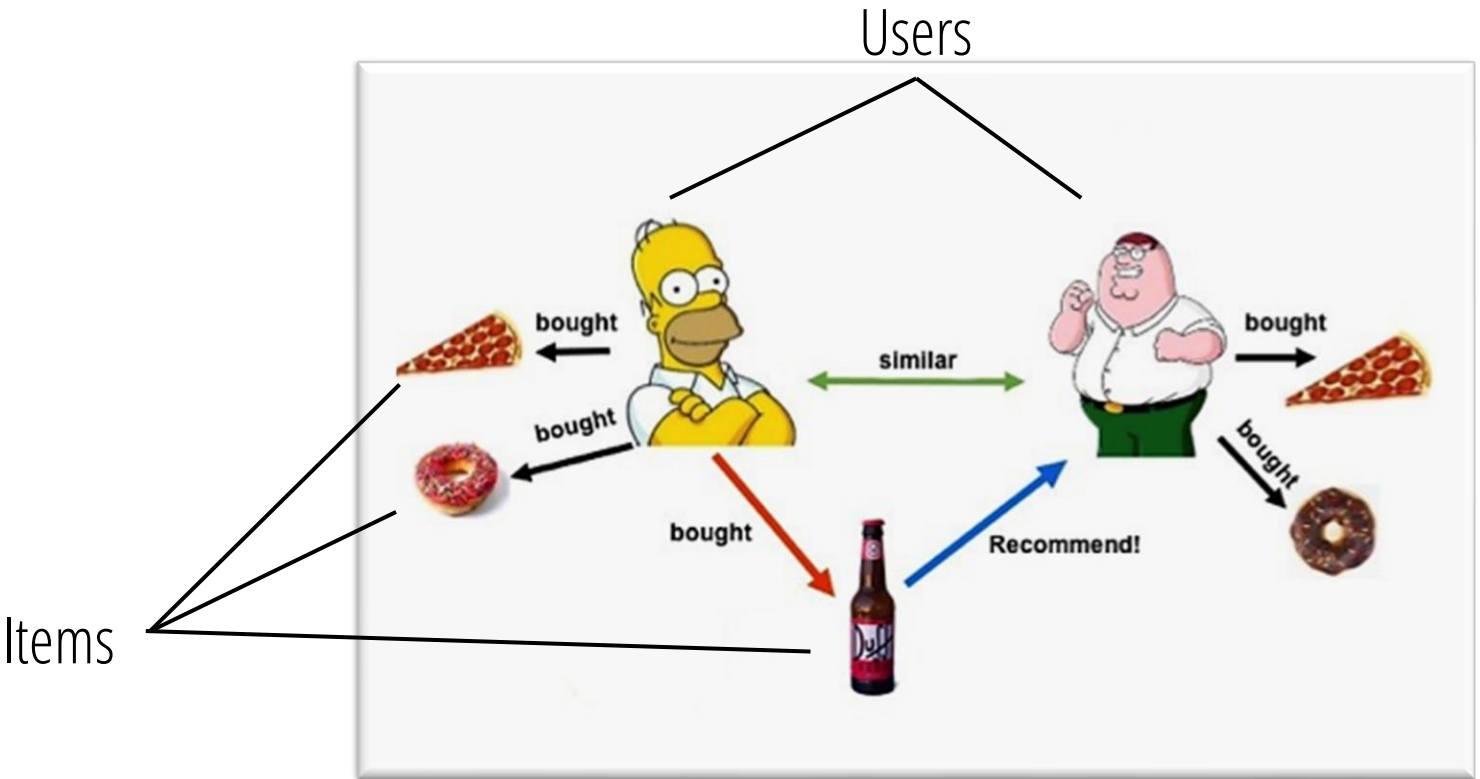


Andrea Capiluppi
Associate professor
University of Gronigen



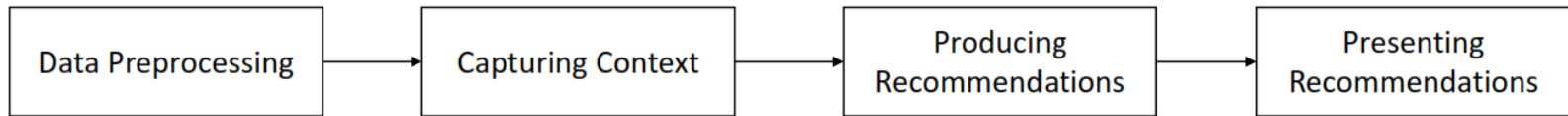
Leonardo Mariani
Full professor
University of Milano-Bicocca

Recommender system in a nutshell



Recommender system for software engineering

- Recommender systems* (RS) are complex software system that comes in handy to alleviate the burden of choice in different domains
- Recommender systems in software engineering** (RSSEs) has been defined

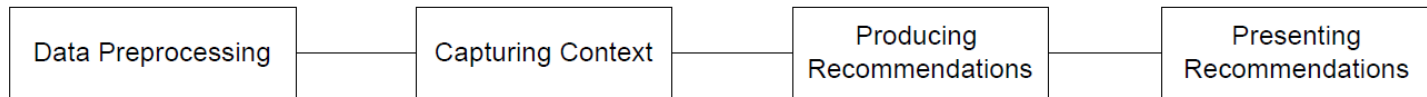


*RSSE phases as described by Robillard et. al***

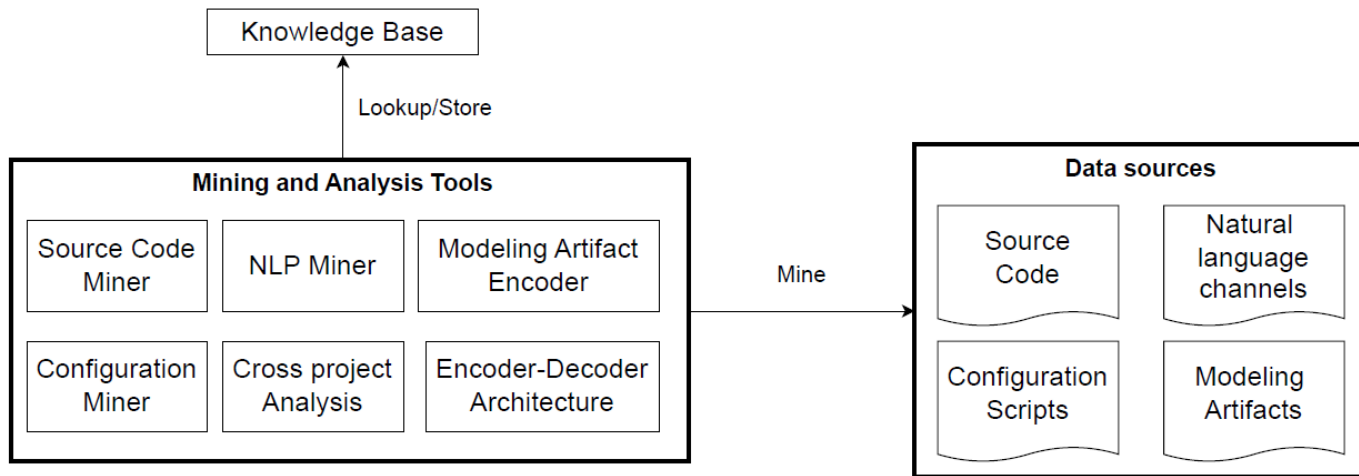
* Francesco Ricci, Lior Rokach, and Bracha Shapira. *Introduction to Recommender Systems Handbook*, pages 1–35. Springer US, Boston, MA, 2011. ISBN 978-0387-85820-3.

** M. Robillard, R. Walker and T. Zimmermann, "Recommendation Systems for Software Engineering," in IEEE Software, vol. 27, no. 4, pp. 80-86, July-Aug. 2010, doi: 10.1109/MS.2009.161.

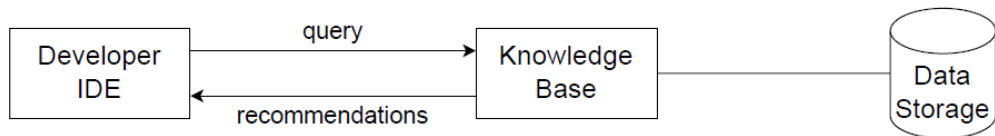
RSSEs: Conceptual framework



Data Preprocessing



Capturing Context, Producing Recommendations, Presenting Recommendations Phases

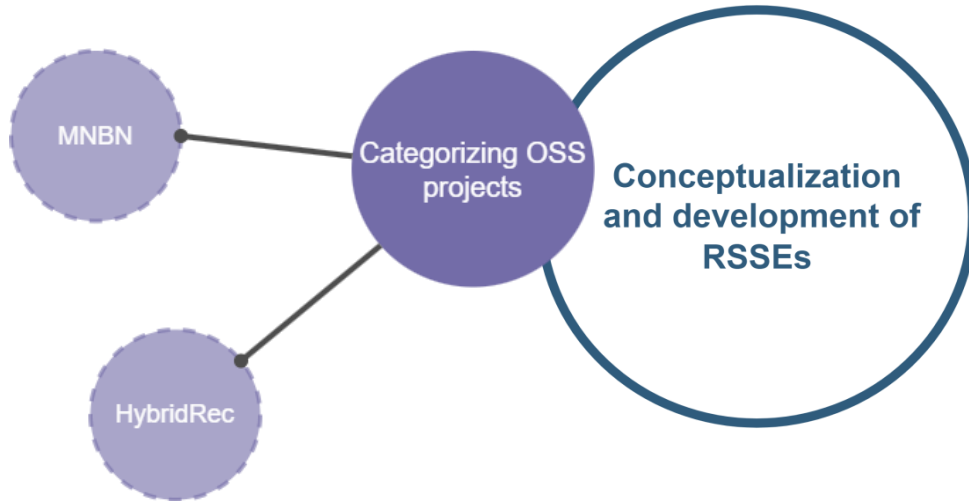


Categorizing GitHub projects

UNIVERSITÀ
DEGLI STUDI
DELL'AQUILA



DISIM
Dipartimento di Ingegneria
e Scienze dell'Informazione
e Matematica



Setting the context

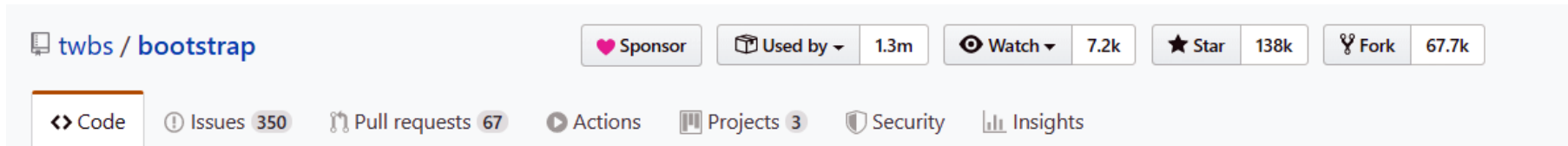
- In 2017, GitHub introduced *topics* to help developers increase the reachability of their repositories
- Assigning wrong labels can compromise the discoverability*
- As first attempt, we develop an automatic approach based on Multinomial Naive Bayesian network (MNBN)**

[*] Hudson Borges, André C. Hora, and Marco Tulio Valente. *Understanding the Factors That Impact the Popularity of GitHub Repositories*. In 2016 IEEE International Conference on Software Maintenance and Evolution, ICSME 2016, Raleigh, NC, USA, October 2-7, 2016, pages 334–344. IEEE Computer Society

[**] Claudio Di Sipio, Riccardo Rubei, Davide Di Ruscio, and Phuong T. Nguyen. *A multinomial naïve bayesian (mnb) network to automatically recommend topics for github repositories*. In Proceedings of the Evaluation and Assessment in Software Engineering, EASE '20

Motivating example

12



The screenshot shows the GitHub repository page for twbs/bootstrap. The repository name is 'twbs / bootstrap'. The page features several interactive buttons: 'Sponsor' (with a heart icon), 'Used by' (with a box icon and a dropdown arrow, showing 1.3m), 'Watch' (with an eye icon and a dropdown arrow, showing 7.2k), 'Star' (with a star icon, showing 138k), and 'Fork' (with a fork icon, showing 67.7k). Below these are navigation tabs: 'Code' (selected), 'Issues' (350), 'Pull requests' (67), 'Actions', 'Projects' (3), 'Security', and 'Insights'.

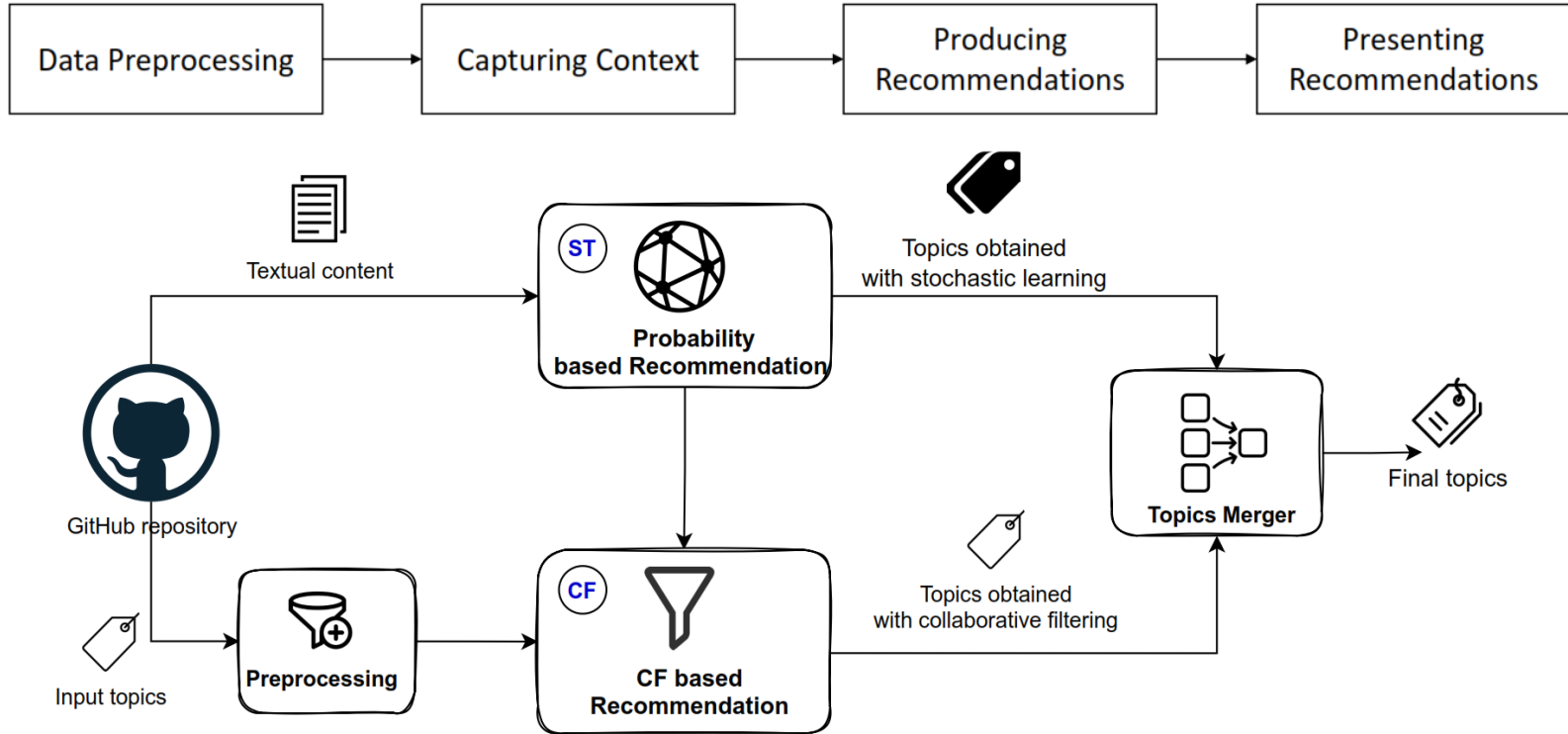
The most popular HTML, CSS, and JavaScript framework for developing responsive, mobile first projects on the web.

<https://getbootstrap.com>

css bootstrap javascript html scss css-framework sass

Bootstrap GitHub projects and its topics

Proposed solution: HybridRec*

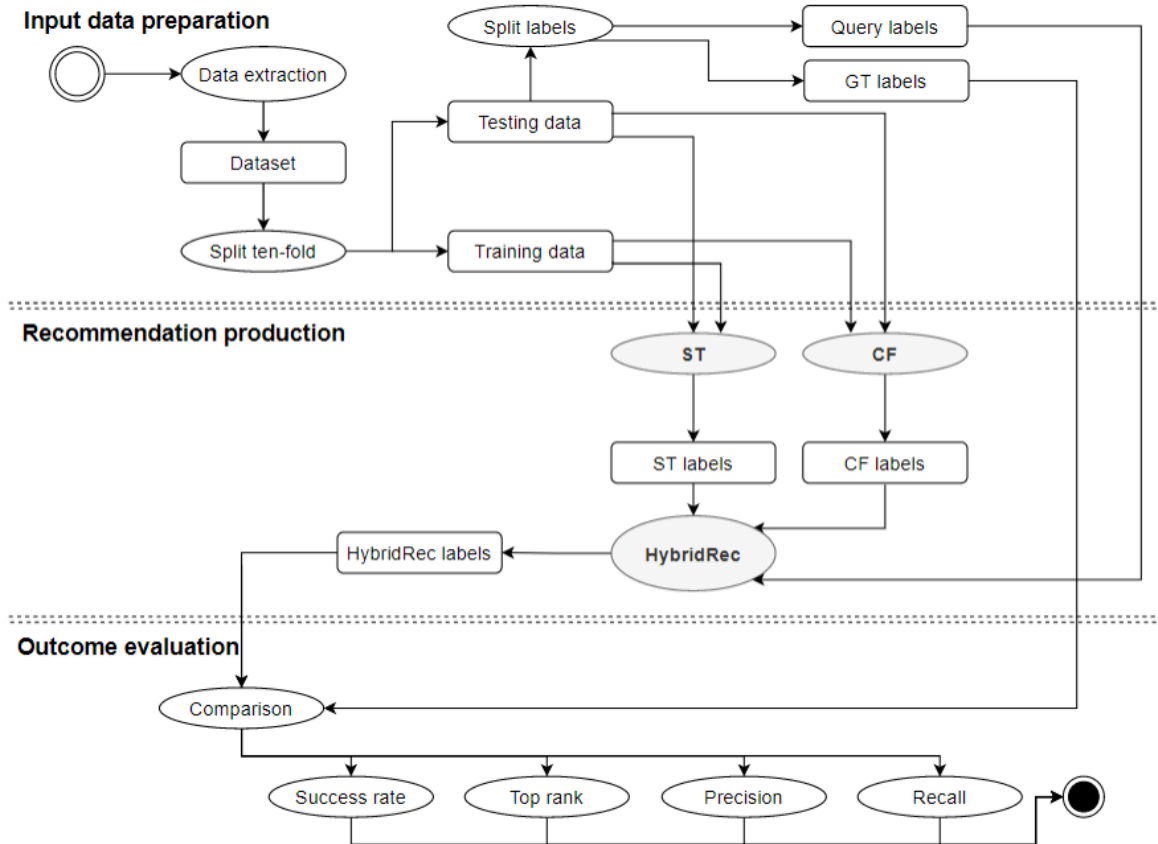


$$\text{success rate}@N = \frac{\text{count}_{p \in P} (|GT(p) \cap (\cup_{r=1}^N REC_r(p))| > 0)}{|P|}$$

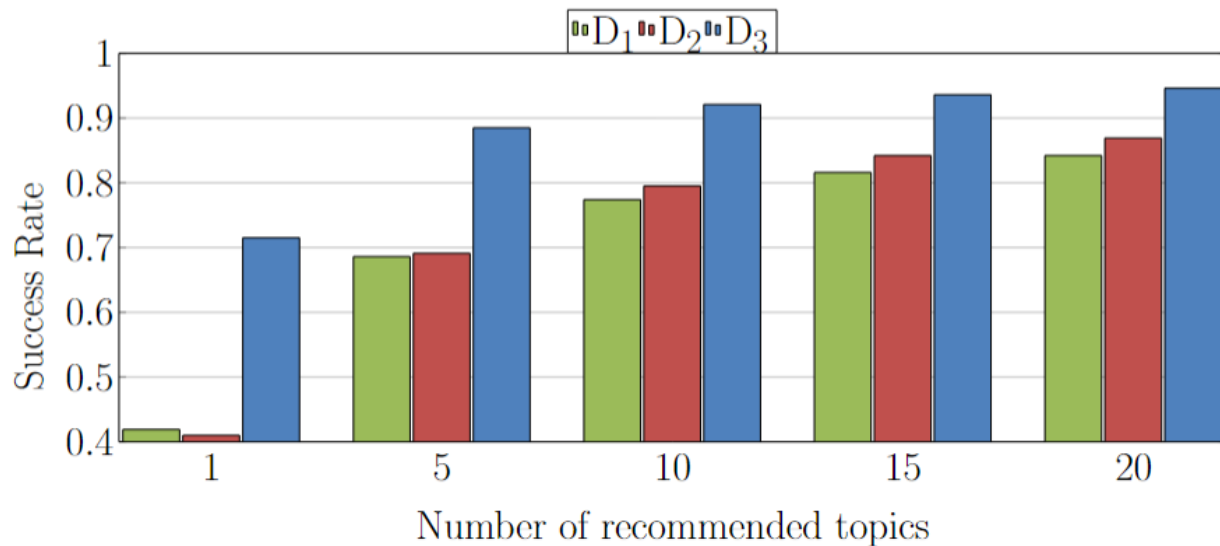
$$\text{precision}@N(p) = \frac{\sum_{r=1}^N |GT(p) \cap REC_r(p)|}{N} \quad \text{recall}@N(p) = \frac{\sum_{r=1}^N |GT(p) \cap REC_r(p)|}{|GT(p)|}$$

$$\text{Top rank} = \frac{\text{TpRank}(r)}{|R|} \times 100\% \quad \text{coverage}@N = \frac{|\cup_{p \in P} \cup_{r=1}^N REC_r(p)|}{|I|}$$

Evaluation process



	D ₁	D ₂	D ₃
# of artifacts	11,694	6,253	5,620
# of topics	19,337	455	6,442
Avg. number of topics	8.24	6.70	8.60
Avg. frequency of topics	16.13	42.10	29.90



HybridRec: Contributions and limitations

Contributions

- 👍 It works also for MVN projects
- 👍 Combining different approaches increases the performances
- 👍 It can cover all the types of GitHub topics

Limitations

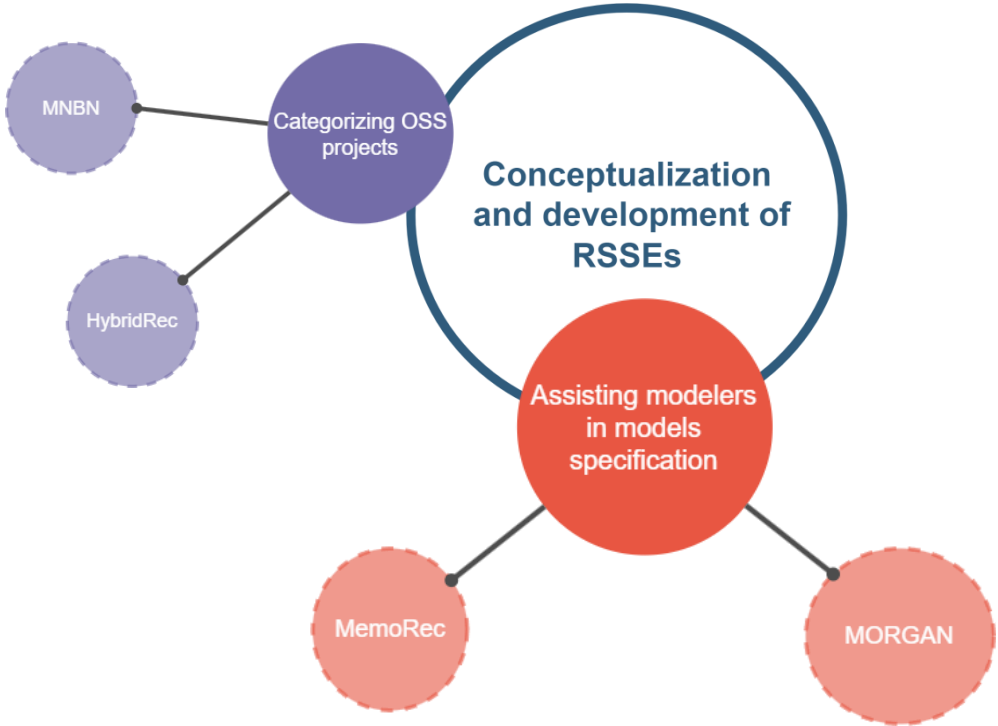
- 👎 No comparison with a baseline
- 👎 Not able to analyze the source code

Assisting modelers in specifying models and metamodels

UNIVERSITÀ
DEGLI STUDI
DELL'AQUILA



DISIM
Dipartimento di Ingegneria
e Scienze dell'Informazione
e Matematica

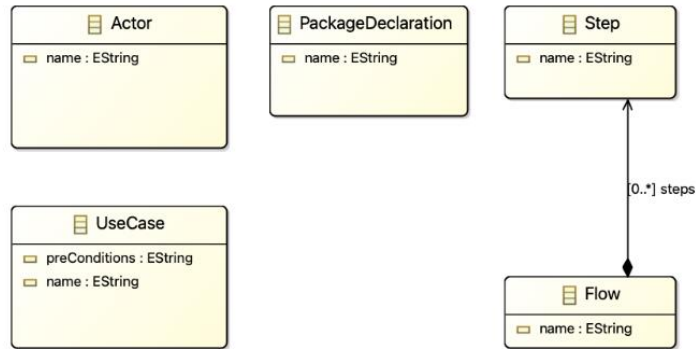


- MDE employs models as first-class artifacts to facilitate the overall development lifecycle*
- Recently, intelligent modeling assistants (IMAs) have been proposed to automatize their specification**
- Selecting a proper encoding scheme and suitable data is still open challenges

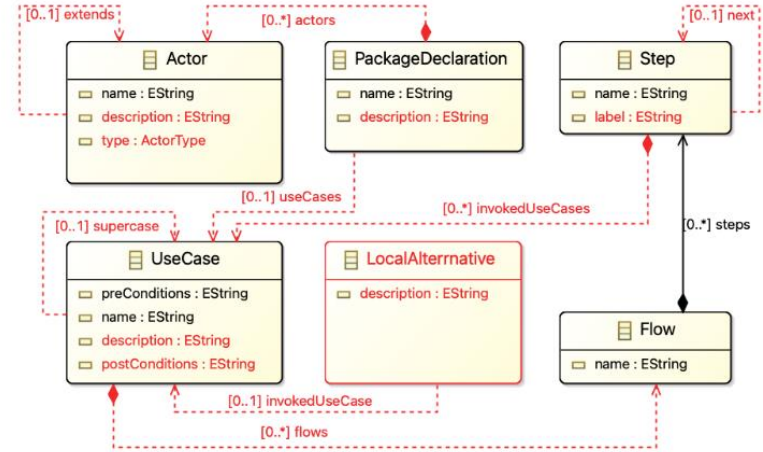
[*] Marco Brambilla, Jordi Cabot, and Manuel Wimmer. *Model-driven software engineering in practice*. Synthesis lectures on software engineering, 3(1):1–207, 2017.

[**] Gunter Mussbacher, Benoit Combemale, Jörg Kienzle, Silvia Abrahão, Hyacinth Ali, Nelly Bencomo, Márton Búr, Loli Burgueño, Gregor Engels, Pierre Jeanjean, Jean- Marc Jézéquel, Thomas Kühn, Sébastien Mosser, Houari Sahraoui, Eugene Syriani, Dániel Varró, and Martin Weysow. *Opportunities in intelligent modeling assistance*. Software and Systems Modeling, 19(5):1045–1053, September 2020.

Motivating example

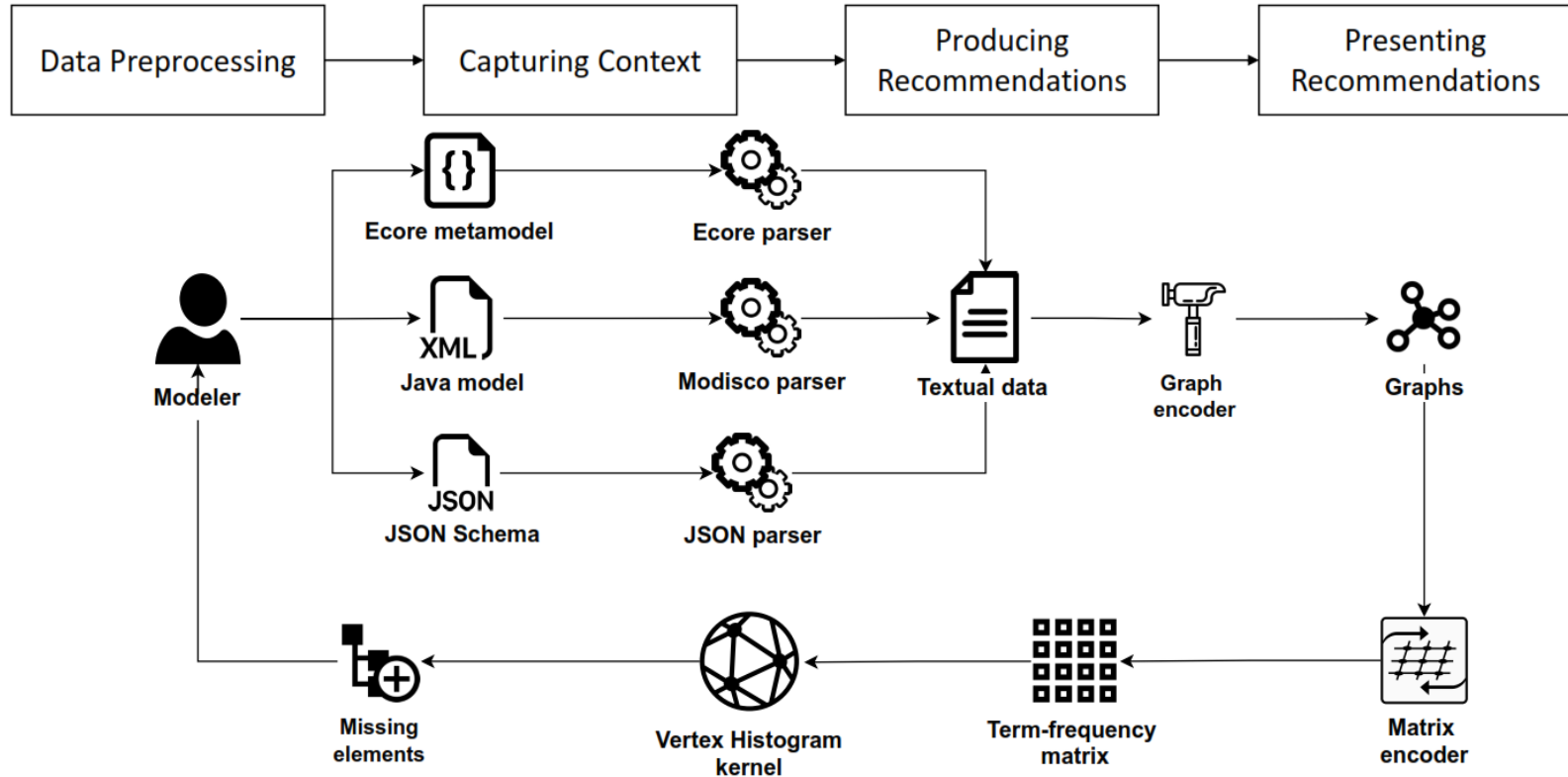


Partial metamodel

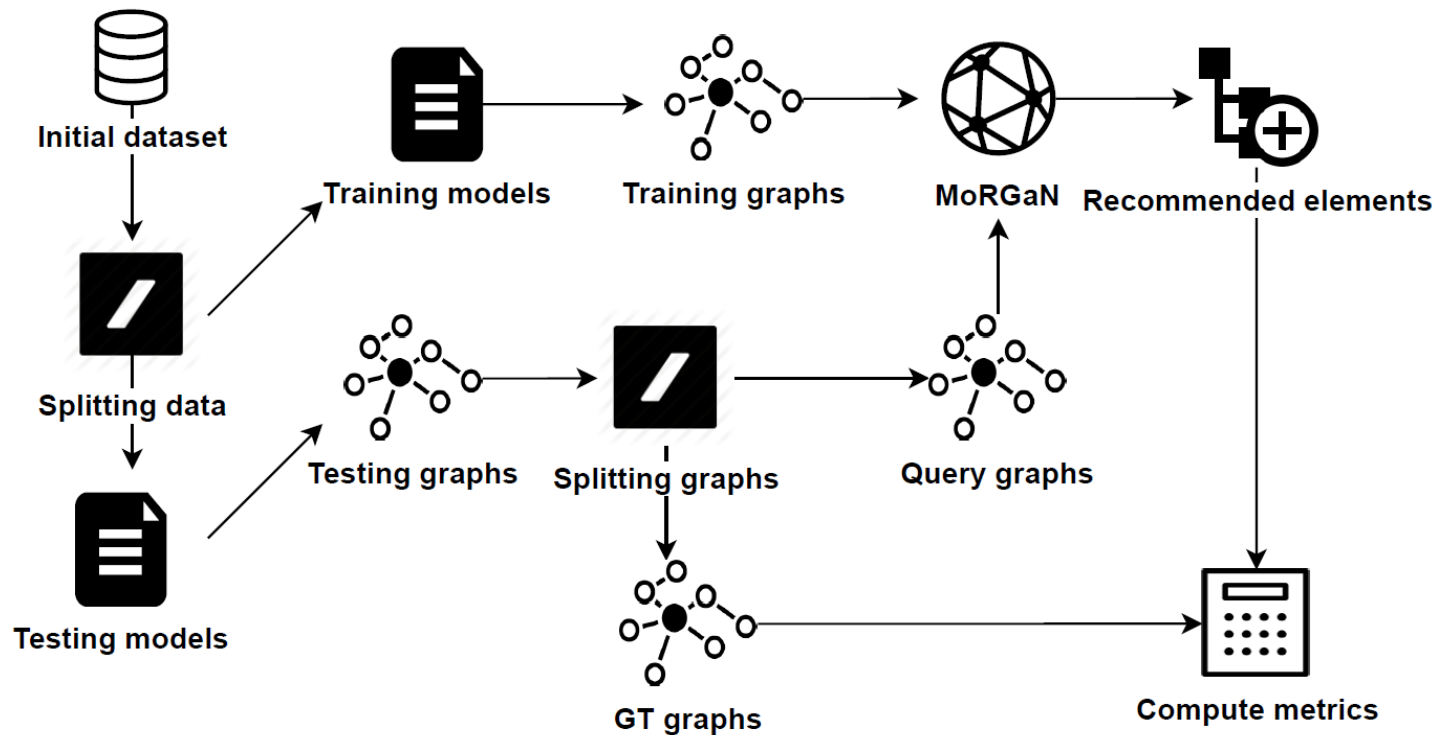


Complete metamodel

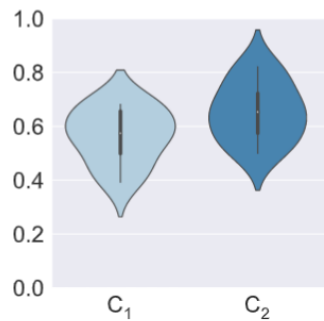
Proposed solution: MORGAN*



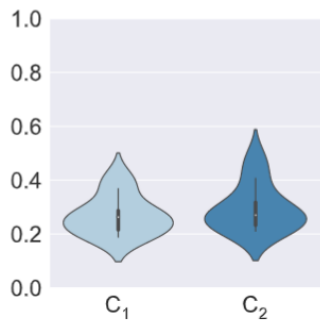
Evaluation process



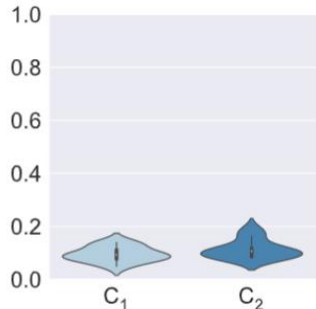
Models results



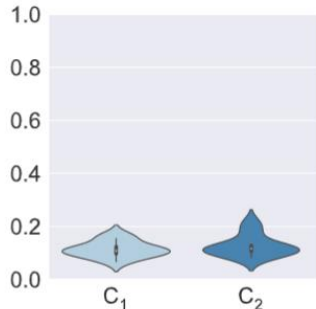
(a) Success rate



(b) Precision

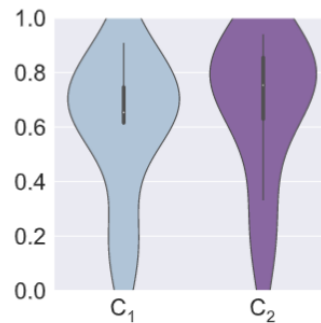


(c) Recall

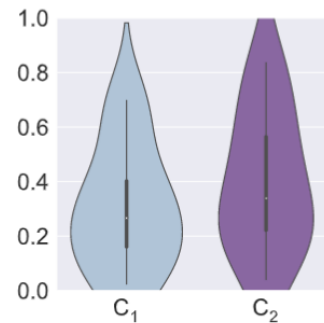


(d) F-Measure

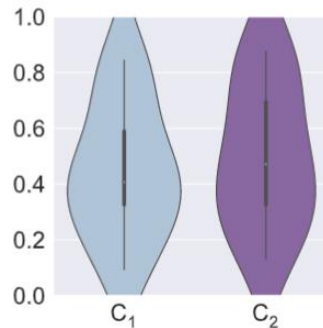
Metamodels results



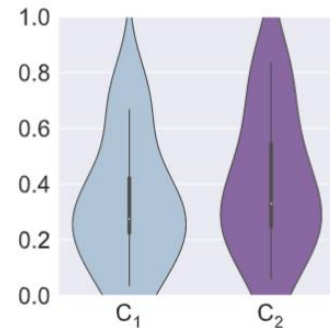
(a) Success rate



(b) Precision



(c) Recall



(d) F-Measure

Contributions

- 👍 It was the first approach using graph kernel similarity to support modeling activities
- 👍 It can support both metamodel and model completion

Limitations

- 👎 Scalability issue when large datasets are considered
- 👎 No user evaluation

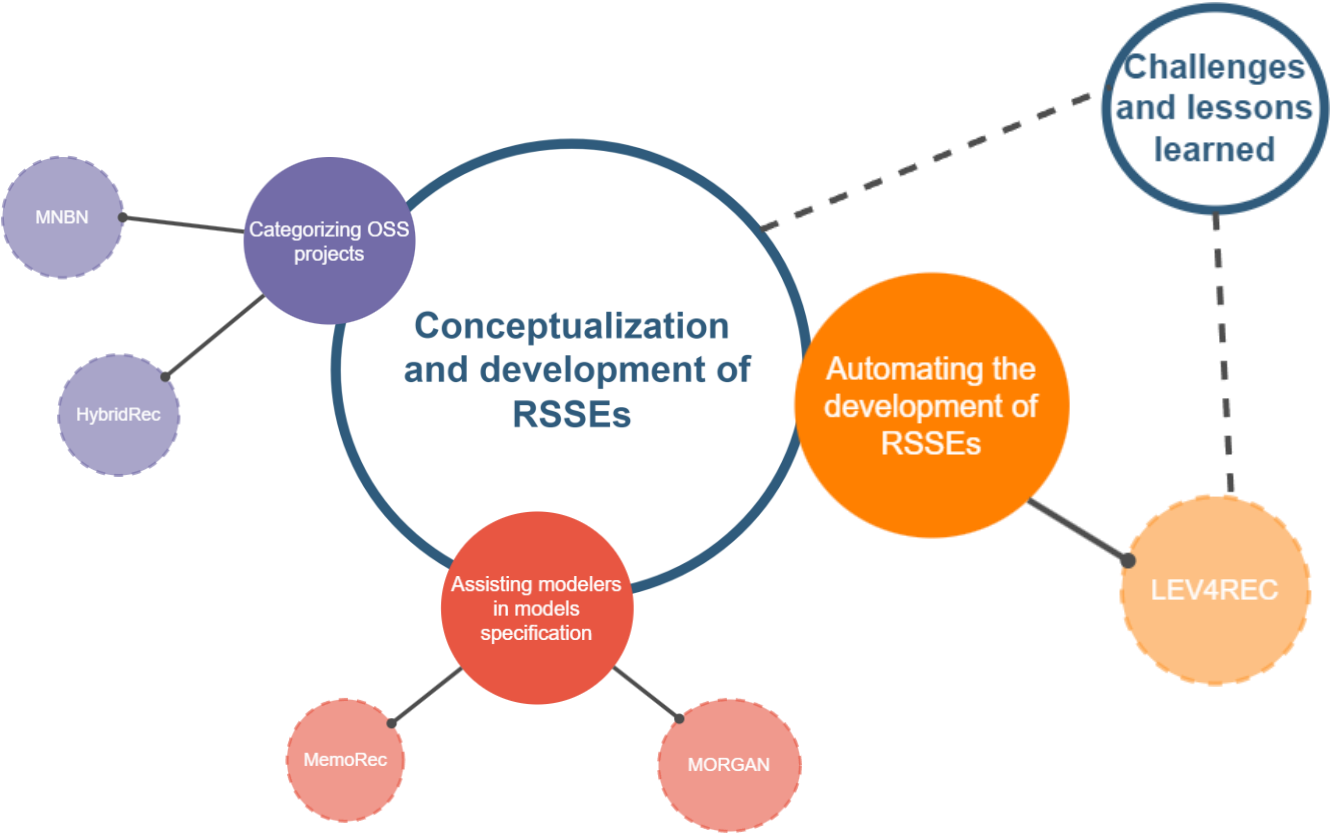
Automating the design and the deployment of RSSEs

UNIVERSITÀ
DEGLI STUDI
DELL'AQUILA



DISIM
Dipartimento di Ingegneria
e Scienze dell'Informazione
e Matematica

Conceptual map



Lessons learned from developed RSSE

Requirements

#R1: Importance of a clear requirement definition process

#R2: Users skepticism

Evaluation

#E1: Candidate baselines might not be reusable

#E2: User studies are cumbersome, and they can take a long time to be conducted and completed

#E3: In certain contexts, the k-fold cross-validation technique is a good alternative to user studies

Development

#D1: Selecting the right representation can be of paramount importance

#D2: Do not pretend to immediately find the optimal solution (move on iteratively)

#D3: Start first with the techniques you already know and move on from there

Setting the context

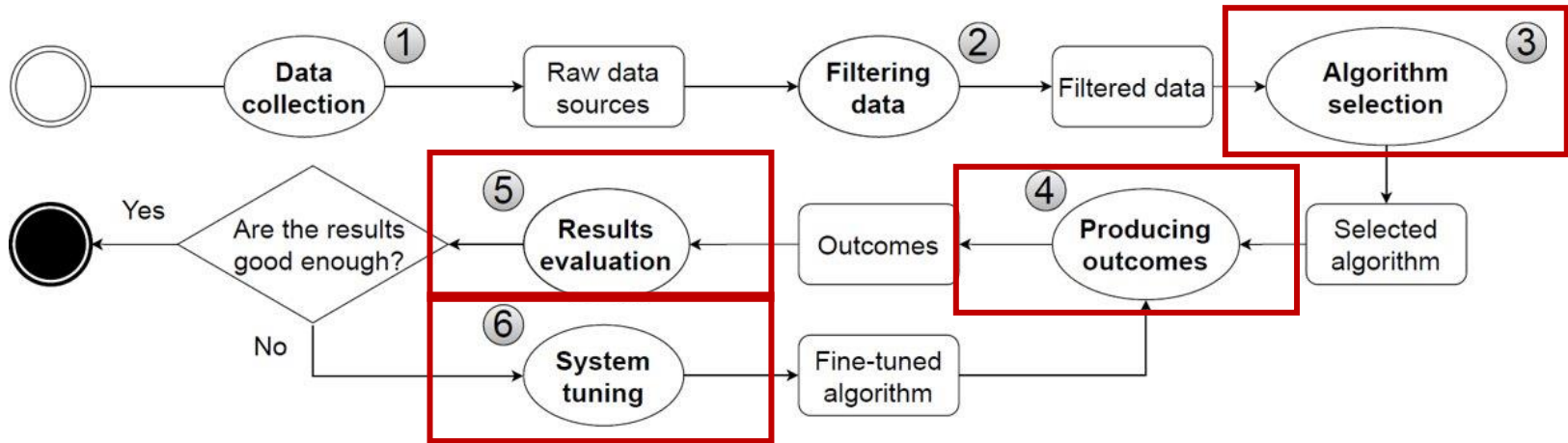
- Several frameworks have been proposed to automatize the development of RS^{*},^{**}

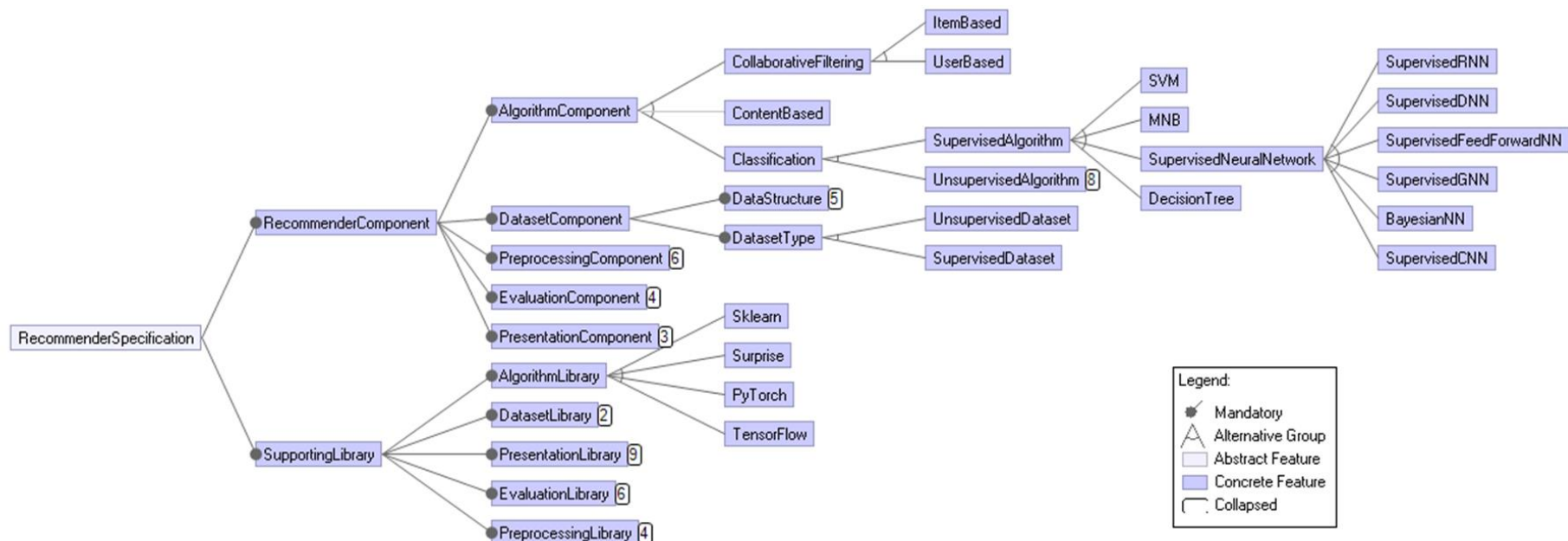
- None of the existing approaches consider constraints at the design time

*Almonte, L., Perez-Soler, S., Guerra, E., Cantador, I., de Lara, J.: Automating the Synthesis of Recommender Systems for Modelling Languages p. 14 (2021)

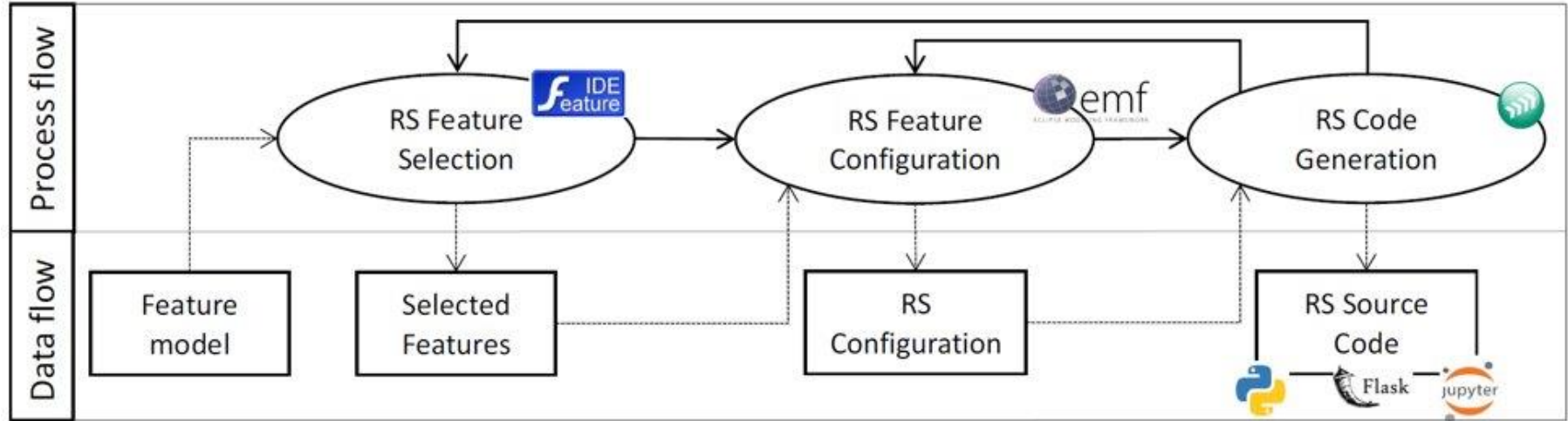
**Anelli, V.W., Bellogin, A., Ferrara, A., Malitesta, D., Merra, F.A., Pomo, C., Donini, F.M., Di Noia, T.: Elliot: A comprehensive and rigorous framework for reproducible recommender systems evaluation. In: Proceedings of the 44th International ACM SIGIR Conference on Research and Development in Information Retrieval, SIGIR'21, p. 2405–2414. Association for Computing Machinery, New York, NY, USA (2021).

Motivating example



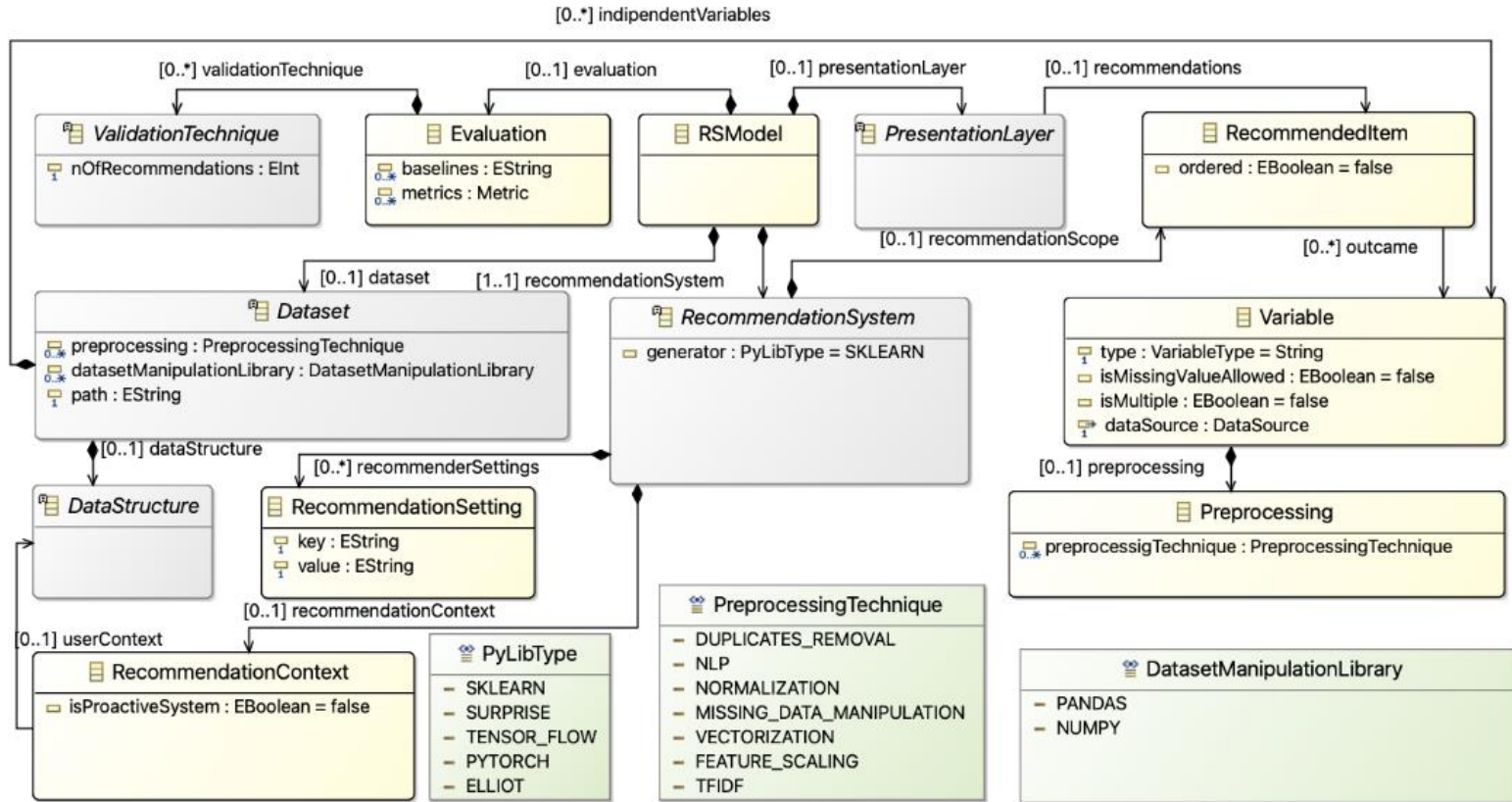


Proposed solution: LEV4REC*,**



*Di Sipio, C., Di Ruscio, D., and Nguyen, P.T. *Democratizing the development of recommender systems by means of low-code platforms*, In Proceedings of the 23rd ACM/IEEE International Conference on Model Driven Engineering Languages and Systems: Companion Proceedings, MODELS '20, New York, NY, USA, 2020. Association for Computing Machinery. ISBN 9781450381352. DOI: <https://doi.org/10.1145/3417990.3420202>

**Di Sipio, C., Di Rocco, J., Di Ruscio, D. and, Nguyen, P.T., 2021. *A Low-Code Tool Supporting the Development of Recommender Systems*. In Fifteenth ACM Conference on Recommender Systems (RecSys '21). Association for Computing Machinery, New York, NY, USA, 741–744. DOI: <https://doi.org/10.1145/3460231.3478885>



- ▼ RecommenderModel
 - ▼ RecommenderConfiguration
 - ▼ AlgorithmConfiguration
 - ▼ CollaborativeFiltering
 - ItemBased
 - UserBased
 - > Classification
- ▼ Configuration statistics
 - Number of selected features: 31
 - Number of manually selected features: 8
 - Number of unselected features: 47

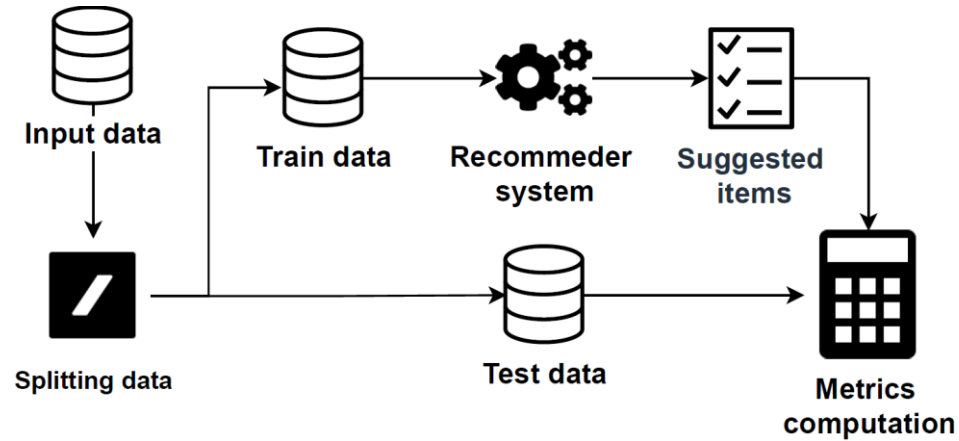
Feature specification

- ▼ RS Model CrossRec
 - ▼ Supervised Dataset crossRecData.csv
 - Variable GithubProject
 - Variable LibraryDependence
 - > Graph Graph
 - Web Application
 - ▼ Evaluation
 - > Cross Validation Ten Fold
 - > Filtering RS CollaborativeFiltering

Property	Value
Library	<input checked="" type="checkbox"/> SURPRISE
Name	<input checked="" type="checkbox"/> Ten Fold
NOF Recommendations	<input checked="" type="checkbox"/> 10
Number Of Fold	<input checked="" type="checkbox"/> 10

Model fine-tuning

Results: Quantitative analysis



	Collaborative filtering		Classification	
Avg. metrics	CROSSREC	LEV4REC	AURORA	LEV4REC
Precision	0.623	0.634	0.945	0.950
Recall	0.622	0.552	0.938	0.949
F1	0.623	0.590	0.942	0.949

Qualitative evaluation: Focus group*

Participant	Recommender Systems	Low-code Engineering	Software Product Lines	Model-driven Engineering
P1	Expert	Outsider	Outsider	Outsider
P2	Good Knowledge	Good Knowledge	Good Knowledge	Good Knowledge
P3	Expert	Expert	Expert	Expert
P4	Familiar	Good Knowledge	Familiar	Expert
P5	Expert	Expert	Expert	Expert

Pro

- 👍 Decent degree of automation
- 👍 Generalizability

Cons

- 👎 Usability, i.e., improving the user interface
- 👎 Extensibility, i.e., providing dedicated extension point

*J. Kontio, L. Lehtola, and J. Bragge. **Using the focus group method in software engineering: obtaining practitioner and user experiences**. In Proceedings. 2004 International Symposium on Empirical Software Engineering, 2004. ISESE '04., pages 271–280, 2004. doi: 10.1109/ISESE.2004.1334914.

Posta - Claudio Di Sipio - Outlo... x | Riccardo Rubei - Google Scholar x | Overleaf, Online LaTeX Editor x | LEV4REC - Chose your preferen... x | +

localhost:8891/lev4rec/

Gmail YouTube Maps shamsa-abid/TACE... dblp: Search for "" Google Scholar rjust/defects4j: A D... ACM Digital Library How to Code BERT... cleanlab - PyPI flasgger/flasgger: E... Tutti i preferiti

LEV4REC Feature model

Quickly and easily set your recommender system preferences.

Algorithm Component

- Collaborative filtering
- Content based
- Classification
- MiningAlgorithm

Dataset component

Data structure

- TextualData
- GraphData
- Matrix
- ARFF
- Tree

Dataset type

- UnsupervisedDataset
- SupervisedDataset

Preprocessing Data Component

- NLP
- FeatureScaling
- Vectorization

11°C Preval. nuvol.

Cerca

08:57 19/10/2023

Contributions

- 👍 An MDE-based environment to specify recommender systems
- 👍 It can cover additional languages apart from Python, e.g., Java, C++
- 👍 Each component can be extended by modifying the two models

Limitations

- 👎 User interface needs improvement
- 👎 Provide dedicated extension point
- 👎 Only a subset of features has been implemented

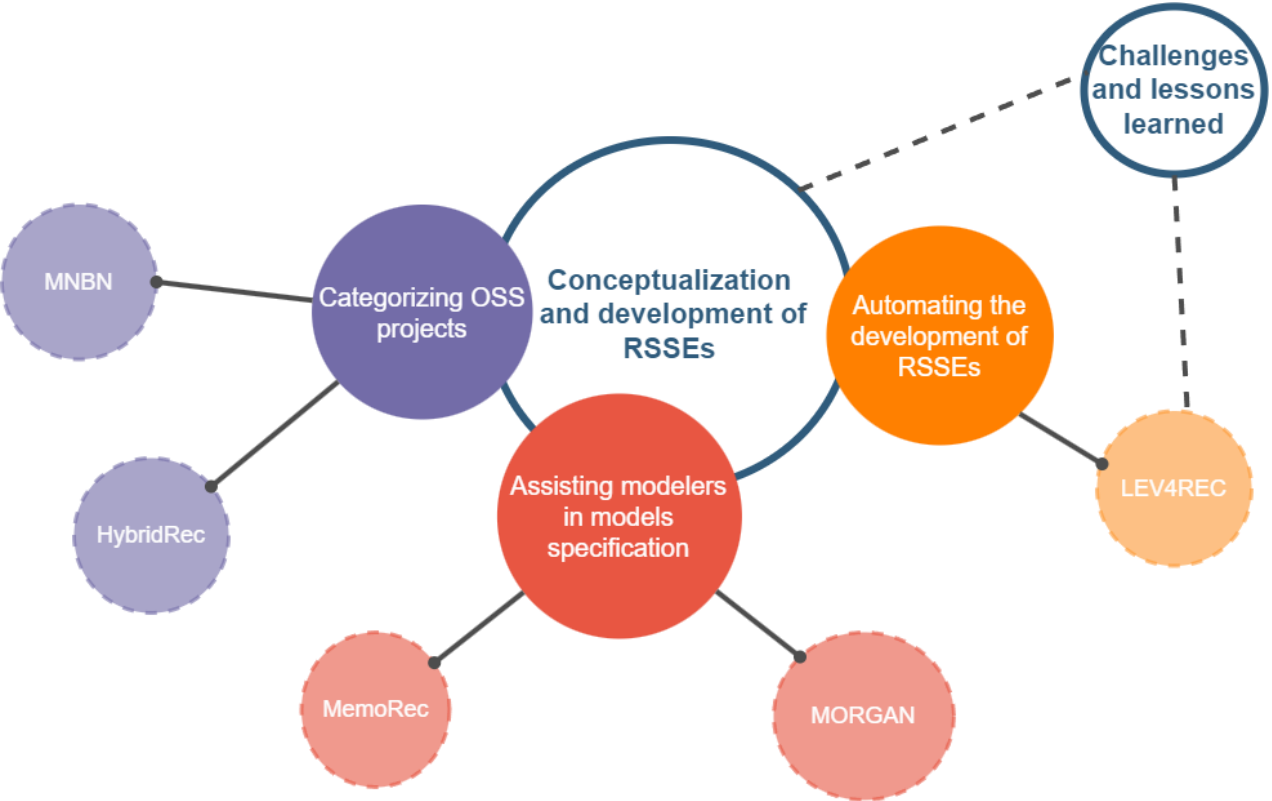
Conclusion

UNIVERSITÀ
DEGLI STUDI
DELL'AQUILA



DISIM
Dipartimento di Ingegneria
e Scienze dell'Informazione
e Matematica

Conceptual map



Categorizing Github

MNBN

HybridRec

Repository	Real topics	Featured topics	Recommended topics
alstudmuffin/SpaceshipGenerator	python,blender-scripts, spaceship, procedural-generation, game-development, 3d	python, 3d	shell, terminal, 3d, opengl, python
0xAX/go-algorithms	golang, algorithm, data-structures, go, sort, tree-structure	algorithm, data-structures, go	data-structures, algorithm, twitter, library, go

Modeling assistance

MemoRec

MORGAN

Context	Recommended item
Step	<i>attribute</i> finalState:EStrng <i>reference</i> continuation:Step <i>reference</i> initialState:initState <i>reference</i> finalStates:FinalState
Flow	<i>attribute</i> finalizeFlow:EBoolean <i>attribute</i> eventPatternId:EStrng <i>reference</i> initialState:initState <i>reference</i> finalStates:FinalState
Step	<i>metaclass</i> StepAlternative:EClass <i>metaclass</i> Automaton <i>metaclass</i> FSM <i>metaclass</i> mFSM

- Embody user feedback in an agnostic way*
- RSSEs for IoT application domain**
- Investigate qualitative aspects in RSSEs, e.g., fairness***, trustworthiness, explainability

*Rubei, R., **Di Sipio, C.**, Di Rocco, J., Di Ruscio, D., and Nguyen, P.T., **Endowing third-party libraries recommender systems with explicit user feedback mechanisms**, 2022 IEEE International Conference on Software Analysis, Evolution and Reengineering (SANER), 2022, pp. 817-821, DOI: 10.1109/SANER53432.2022.00099

Di Rocco J. and **Di Sipio C., *ResyDuo: Combining data models and CF-based recommender systems to develop Arduino projects*, 5th International Workshop on Multi-Paradigm Modeling for Cyber-Physical Systems (MPM4CPS'23)

***Nguyen P.T, Rubei R., Di Rocco J, **Di Sipio C.**, Di Ruscio D., Di Penta M, *Dealing with Popularity Bias in Recommender Systems for Third-party Libraries: How far Are We?*, The 20th Mining Software Repositories (MSR 2023).