Lowcomote

RS & Monitoring for LCDP

林嘉恩

清华大学

2022 年 11 月 2 日

,

Lissette Almonte Garcia

- 2020 年发了第一篇关于自动生成推荐系统集成到 MDE 中的文章，相对来说只是理论为主。
    - *Towards automating the construction of recommender systems for low-code development platforms*
- 2021 年把之前的工作集成到了 eclipse 以外的平台进行测试
    - *Automating the Synthesis of Recommender Systems for Modelling Languages*
- 2022 年在 RS 配置中加入数据预处理
    - *Building recommender systems for modelling languages with DROID*
- 2022 年发表了一篇综述，虽然对这项工作没有什么特别的改进，但是可以帮助我们系统地认知这个领域
    - *Recommender Systems in Model-Driven Engineering*

Research Questions

1. In **which ways** can recommender systems assist in the different tasks within MDE processes?
   - complete and repair artefacts, and work over models
2. **Which recommendation** techniques are most commonly used to support MDE tasks, and **how** are recommenders for MDE **evaluated**?
   - knowledge-based followed by content-based
   - offline experiments
3. What are the **main opportunities** in recommender systems for MDE solutions?
   - model transformations or code generators
   - creating, reusing or finding artefacts
   - effective repositories of MDE artefacts that mitigate the current lack of data
   - adapting RSs to the user's needs
   - mechanisms to exploit the crowd knowledge via collaborative filtering
   - the userbased evaluation of RSs within MDE

Dimensions

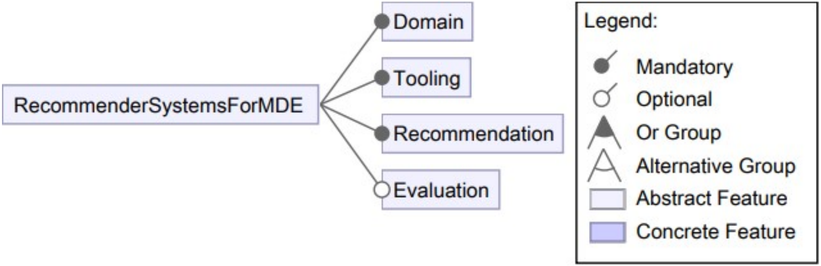

图 1: Dimensions for analysing the use of RSs in MDE.

文章分了四个维度来叙述模型驱动工程中的推荐系统，分别是推荐系统在什么领域解决什么问题、提出的工具有什么特点、使用到了什么样的推荐系统以及怎么评价这个推荐系统。

RS in MDE: domain

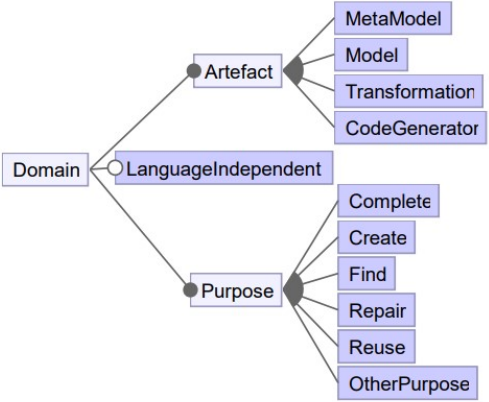

图 2: Domain dimensions for RSs in MDE.

重点关注：Complete

RS in MDE: tooling

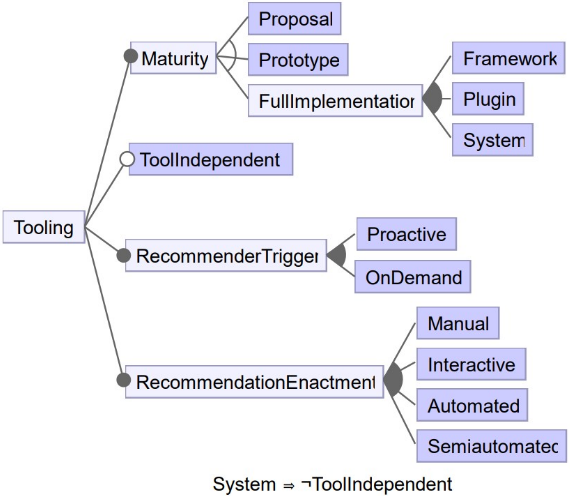

图 3: Tooling dimensions for RSs in MDE.

RS in MDE: recommendation

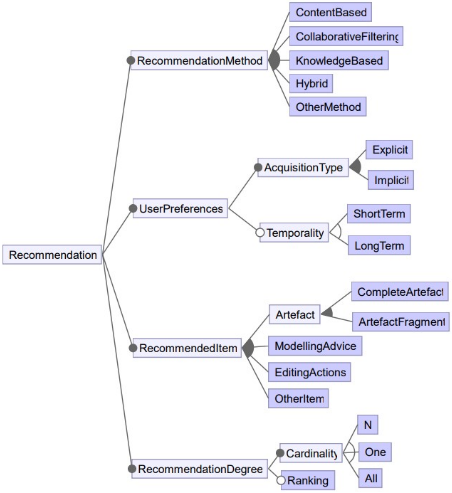

图 4: Recommendation dimensions for RSs in MDE.

RS in MDE: Evalution

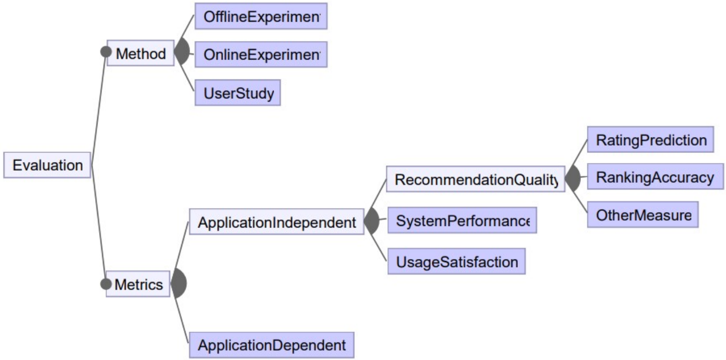

图 5: Evaluation dimensions for RSs in MDE.

RS in MDE: Evalution

- Offline Experiment (4 种数据来源)
  - 生成数据
  - 从现成的库中获取数据 (✓)
  - 从现有文章中获取实例
  - 从公司获取现实世界的数据
- Online Experiment 在线实验 (和 A/B 测试的差异?)
- User Study 用户研究 (三种类型)
  - 使用 RS 来完成任务
  - 用户在 A/B 测试中使用 RS
  - 将推荐的项目和专家推荐的对比

RS in MDE: Evalution

- precision: percentage of the recommended items relevant
- recall: percentage of relevant items included in the recommendation list
- F1: a harmonic mean of previous two
- USC: percentage of users that the RS can recommend
- ISC: popularity of what is recommended
- nDCG: whether the most relevant items are top

Motivating example

- class modeling

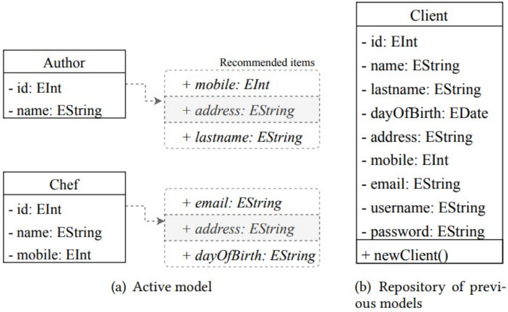

图 6: Motivating example.

- other LCDPs may use alternative modelling notations, and the recommendation task may be different as well

Background: Recommender System

- Matrices

- **user-item matrix**:rating given by the user u to the item i
- **item-feature matrix**: each cell is set to 1 if the item has the feature, and to 0 otherwise

	I_0	I_1	I_2	I_3
U_0	2	-	-	1
U_1	1	5	5	2
U_2	5	-	3	2
U_3	4	5	4	-

(a) User-item matrix

	f_0	f_1	f_2	f_3
I_0	1	1	0	0
I_1	1	0	1	0
I_2	1	0	0	0
I_3	1	0	0	1

(b) Item-feature matrix

图 7: Examples of matrices used in RSs.

Proposed Approach: Overview of Architecture

1. designer provides the meta-model(class diagram)
2. repository of models(data)
3. designer uses a textual DSL to define the meta-model elements
4. framework will generate a tailored RS for the LCDP
5. developers will be offered the recommendations within the LCDP environment
    - tips over the diagram elements
    - example fragments
    - query-answer chatbots addressed in natural language

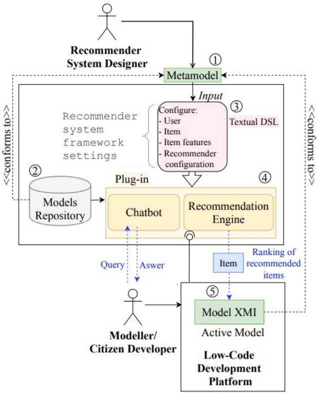

图 8: Overview of the proposed approach.

Proposed Approach: DSL(step 1-2)

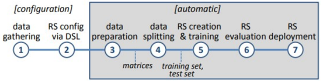

图 9: Overview of process.

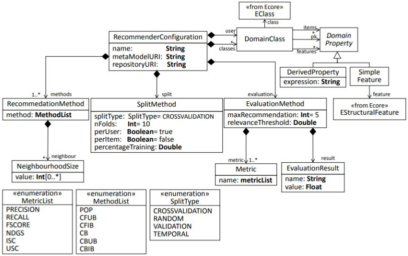

图 10: Meta-model of the DSL.

```
Metamodel: "/SimpleOOPL.ecore"
Repository: "/Instances/"

//Definition of user and items
Users: ClassDeclaration {
  Items: attributes, methods, superclasses; }

//Definition of primary keys (pks) and features
ClassDeclaration {
  pk: name; }

AttributeDeclaration {
  pk: attrName;
  features: attrName, attrType; }

MethodDeclaration {
  pk: name;
  features: name, returnType; }

//Recommender preferences
Recommendations {
  //split configuration
  Split {
    splitType: CrossValidation;
    nFolds: 10;
    perUser: true;
    percentageTraining: 0.8; }

  //methods configuration
  Methods {
    collaborativeFiltering: pop, cfub(2,3,5,10), cfib;
    contentBased: cb;
    hybrid: cbub(2,3,5,10), cbib(2,3,5,10); }

  //evaluation configuration
  Evaluation {
    metrics: precision, recall, f1, ndgs, isc, usc;
    maxRecommendations: 5;
    relevanceThreholds: 0.5; }}
```

图 11: Example of recommender system configuration.

Proposed Approach: Data preparation(step 3)

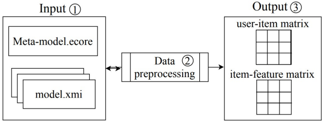

图 12: Data preparation steps.

1. retrieves the collection of models
2. extracts the model objects
3. generates a user-item matrix and an item-feature matrix

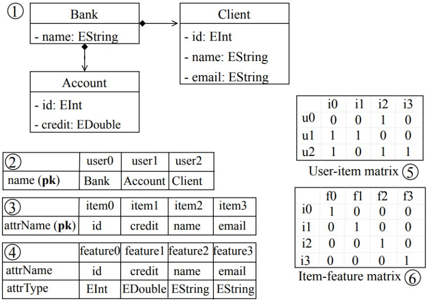

图 13: Example of data preparation.

Proposed Approach: Recommendation engine(step4 -7)

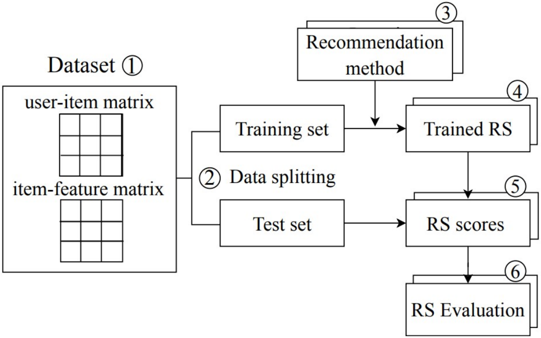

图 14: Steps to build the recommendation engine.

each candidate RS is evaluated according to the specified metrics, and the results are made available for the designer inspection

Experiment: Setup

- Dataset
  - Synthetic: classes from the Internet
  - SyntheticExtended: extend first one by synonyms
  - AtlanEcore: 300 Ecore meta-models
- configuration: first two are **??**, last one is similar to it
- data splitting: 10-fold cross-validation, 8:2
- recommendation methods: top 5
  - **collaborative filtering**: cfubk, cfibk, pop(baseline)
  - **content-based**: cb **hybrid**: cbubk
- evalution metrics(ranking-based)
  - precision、recall、F1、USC、ISC、nDCG

Experiment: Result

	Synthetic						SyntheticExtended						AtlanEcore					
Method	prec.	recall	F1	nDCG	ISC	USC	prec.	recall	F1	nDCG	ISC	USC	prec.	recall	F1	nDCG	ISC	USC
pop	0.048	**0.221**	0.079	**0.177**	0.015	**1.000**	0.046	0.207	0.076	0.170	0.015	**1.000**	0.018	0.083	0.029	0.055	0.002	**1.000**
cfub2	0.060	0.185	0.091	0.144	0.035	0.719	**0.093**	0.242	**0.135**	0.179	0.043	0.698	**0.241**	0.362	**0.289**	**0.323**	0.048	0.332
cfub3	0.054	0.190	0.084	0.145	**0.036**	0.802	0.088	0.256	**0.132**	0.188	0.046	0.802	**0.211**	**0.367**	**0.268**	**0.322**	0.055	0.372
cfub5	0.054	0.206	0.085	**0.165**	**0.036**	0.898	0.083	**0.276**	0.128	**0.202**	0.048	0.837	0.179	**0.368**	0.241	0.321	0.061	0.415
cfub10	**0.066**	0.193	**0.099**	0.146	0.032	0.600	0.074	**0.289**	0.118	**0.219**	**0.049**	0.919	0.140	0.347	0.200	0.297	**0.067**	0.482
cfib	0.053	**0.207**	0.085	0.147	**0.038**	0.901	0.064	0.239	0.101	0.172	**0.049**	0.921	0.092	0.273	0.138	0.225	0.063	0.627
cb	0.018	0.086	0.030	0.086	0.008	**1.000**	0.018	0.086	0.030	0.085	0.006	**1.000**	0.005	0.022	0.008	0.010	0.001	**1.000**
cbub2	0.016	0.015	0.016	0.016	0.002	0.968	**0.096**	0.176	0.125	0.124	0.033	0.633	0.200	0.246	0.221	0.220	0.035	0.311
cbub3	0.016	0.032	0.022	0.026	0.005	0.968	0.065	0.196	0.098	0.142	0.040	0.745	0.155	0.259	0.194	0.230	0.048	0.410
cbub5	0.016	0.057	0.025	0.039	0.010	0.968	0.057	0.203	0.089	0.147	0.043	0.896	0.113	0.276	0.160	0.238	0.058	0.483
cbub10	0.016	0.070	0.026	0.044	0.012	0.968	0.052	0.211	0.083	0.158	0.043	0.993	0.079	0.269	0.122	0.228	**0.065**	0.558
cbib2	**0.095**	0.199	**0.129**	0.131	0.026	0.539	0.014	0.010	0.012	0.011	0.002	0.973	0.001	0.001	0.001	0.001	0.000	0.697
cbib3	0.049	0.166	0.075	0.120	0.030	0.634	0.013	0.020	0.016	0.018	0.004	0.973	0.001	0.002	0.002	0.002	0.000	0.697
cbib5	0.036	0.131	0.056	0.098	0.033	0.864	0.013	0.039	0.019	0.027	0.008	0.973	0.001	0.005	0.002	0.003	0.001	0.697
cbib10	0.032	0.138	0.051	0.112	0.034	**1.000**	0.013	0.049	0.020	0.031	0.010	0.973	0.001	0.006	0.002	0.004	0.001	0.697

图 15: Results of the experiment.

- The AtlanEcore dataset has the best overall performance.

Experiment: Research Object

1. Can a recommender system help in class modelling tasks?
   - the highest F1 value was 0.289
   - paper told us it can build an RS that helps in class modelling
2. Which recommendation method of relevant attributes, methods and superclasses has the best performance?
   - SyntheticExtended and AtlanEcore: cfub2, pop has high USC but low ISC
   - Synthetic: cbib2
3. Can hybrid approaches be beneficial for the recommendation of attributes, methods and superclasses?
   - in Synthetic, hybrid method has the better result
4. Which method performs better when considering user and item coverage in the recommendation of relevant attributes, methods and superclasses?
   - the methods having low precision and recall report high USC and low ISC

Personal Thinking

- how to build
  - experiment shows that RS perform well in modeling work, but it's just a framework
  - writting this driver software is a easy job using RankSys
- how to use in DWF
  - while customizing form or application, we hope DWF can offer us some advise
- the diffculty in building RS for DWF
  - **Data**: each user may have a little model
  - **User Similarity**: difference between users is large
  - **Result**: we don't know how it perform in DWF area

Panagiotis Kourouklidis

- 2019 年微软发的一篇关于 ML 中的软工的文章中提到了机器学习工作流的九个阶段，其中包括了 Model Monitoring
  - *Software Engineering for Machine Learning: A Case Study*
- 2020 年 Panagiotis Kourouklidis 发表了第一篇关于自动生成监控系统的文章，主要是说了一下目前 ML 模型部署后面临的问题以及他的研究计划
  - *Towards a low-code solution for monitoring machine learning model performance*
- 2021 年 Panagiotis Kourouklidis 发表了第二篇相关文章，提出了一个原始的元模型，并基于 Kubernete 给出了实现。
  - *A Model-Driven Engineering Approach for Monitoring Machine Learning Models*

ML workflow

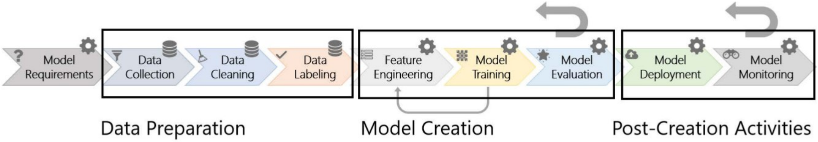

图 16: ML workflow.

Draft: data draft & concept draft

- 有监督学习中的 X，Y 通常被视为服从联合概率分布 P(X, Y) 的随机变量。由于真实世界情况复杂且不稳定，概率分布会发生变化，即概念漂移、采样漂移、先验概率漂移或更一般的数据集漂移。此处主要讨论数据漂移和概念漂移两种。
- data draft
  - 数据漂移可能是由采样机制，环境中的未知因素随时间变化引起。
  - 数据漂移不一定导致输出错误，没有数据漂移也不一定能保证输出正确。
  - 目前可以通过检查数据输入来检测数据漂移。
- concept draft
  - 概念漂移指输入-输出映射发生变化。概念漂移可能是真实的，也可能是虚拟的。
  - 真实: 由于某些未知因素发生改变，导致映射发生改变；
  - 虚拟: 映射实际并没有发生改变，只是由于其它原因观测到这个现象（比如有偏采样）。

Research Agenda

- Data Capture
  - 为了检测概念漂移，需要存储模型输出并与反馈进行对比
- Algorithm Execution
  - 希望能使用 DSL 来描述漂移检测算法，然后生成适用于不同框架的算法
- Responding to Detected Drift
  - 方法是多样的，可以包含以下两种
    - data drift：检测到漂移时简单地发送 email
    - concept drift：自动触发重新训练模型的流程

Survey

- Review: work flow of ML
- Problem
  - key concepts: data drift, concept drift
- Approach: furture research agenda
  - Data capture
  - Algorithm execution
  - Responding to detected drift

Approach

- Problem: Drift
- Approach: a model-driven approach for ML monitoring

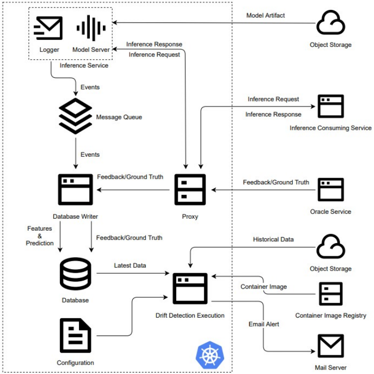

图 17: Generated Artefact