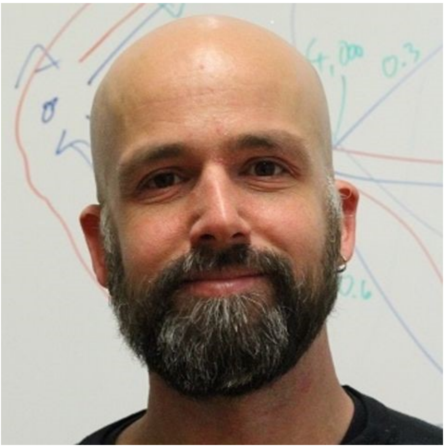
Matching with Transformers in MELT

Sven Hertling, Jan Portisch, Heiko Paulheim
University of Mannheim

Joint Work

Sven Hertling
Data and Web Science Group,
University of Mannheim
sven@informatik.uni-mannheim.de

Jan Portisch
Data and Web Science Group,
University of Mannheim / SAP SE
jan@informatik.uni-mannheim.de



Prof. Dr. Heiko Paulheim
Data and Web Science Group,
University of Mannheim
heiko@informatik.uni-mannheim.de

Agenda

UNIVERSITY OF MANNHEIM
School of Business Informatics and Mathematics

- Motivation
- What is MELT
- Transformers in MELT
- Experiments
- Conclusion & Future Work

Motivation

Motivation

UNIVERSITY OF MANNHEIM
School of Business Informatics and Mathematics

- the transformer architecture achieved breakthrough results in various NLP domains
- this poses the question in how far transformers can be beneficial for the ontology matching domain

UNIVERSITY OF MANNHEIM
School of Business Informatics and Mathematics

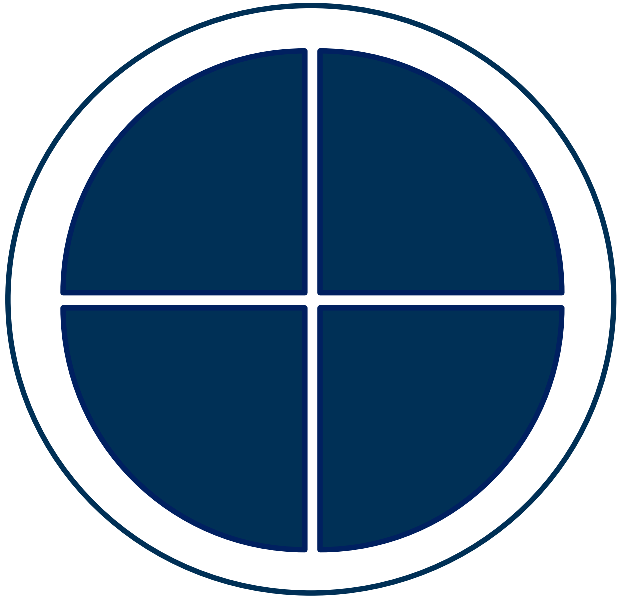
What is MELT?

What is MELT?

UNIVERSITY OF MANNHEIM
School of Business Informatics and Mathematics

- **Easy** matcher development
- Re-Usable Matcher Componetns
- **Non-Java** matcher development
- **Maven** support

- Allow **for parameter optimization**



- Facilitate **matcher packaging** (SEALS, HOBBIT, Docker Web)
- Facilitate **matcher submission**

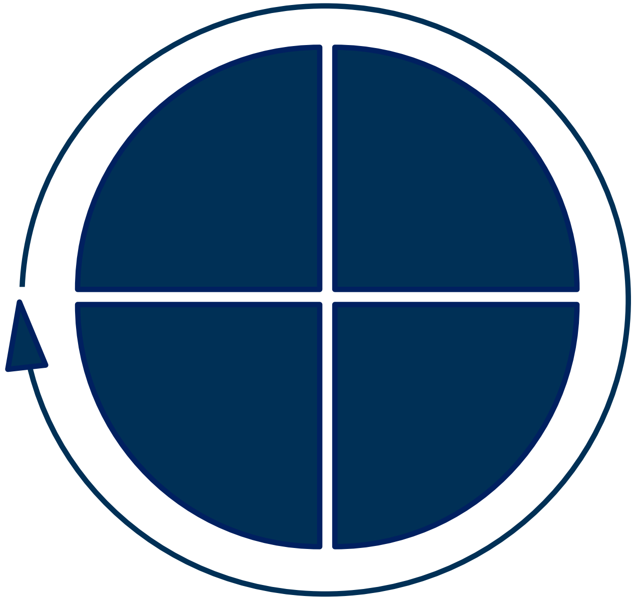
- **Advanced evaluation** capabilities
- **Evaluation before packaging**
- Allow for **interactive visualization**

- **Streamlined** development process
- **Integration** with existing tooling
- **OAEI support**
- **Extensibility**

What is MELT?

Matcher **Development** with **Transformers**

Matcher **Fine-Tuning**



Matcher **Submission**

Matcher **Evaluation**

There is MUCH more to MELT

UNIVERSITY OF MANNHEIM
School of Business Informatics and Mathematics

Multi-Threaded Matcher Execution

Ontology **Caching** Services

> 50 matchers

> 25 filters

Execution of SEALS, HOBBIT, WEB packages from within MELT

TRY IT!

MELT @GitHub

Alignment **Extensions**

OAEI-Track Organizer Tools

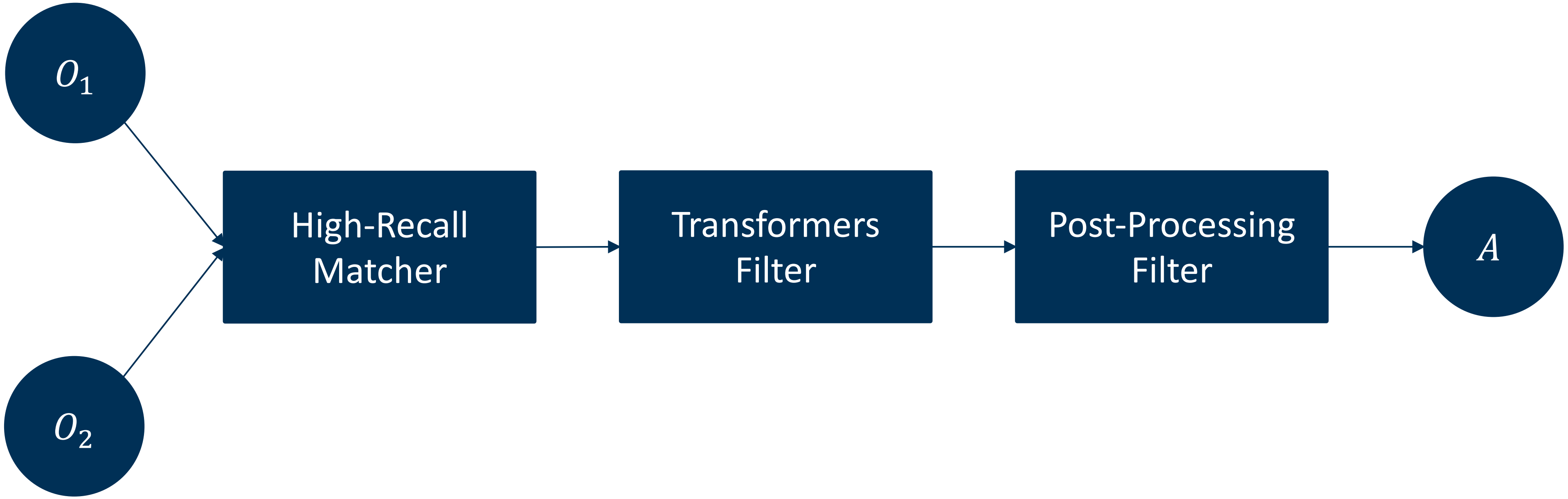
ExecutionResult **Indexing**

One-Time **Auto-Download** of OAEI Tracks

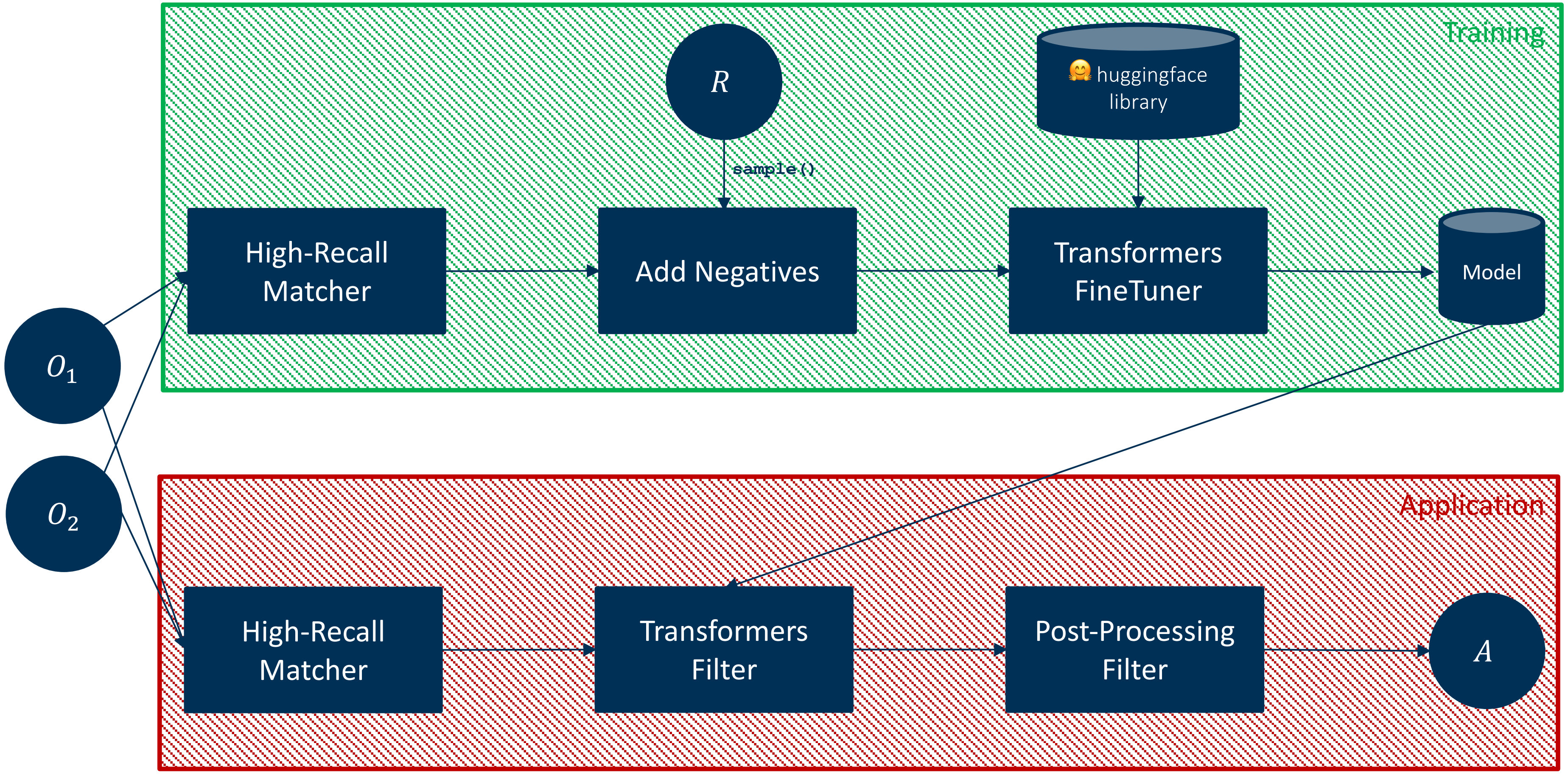
Matcher **Pipelining**

Transformers in MELT

The Transformer Pipeline



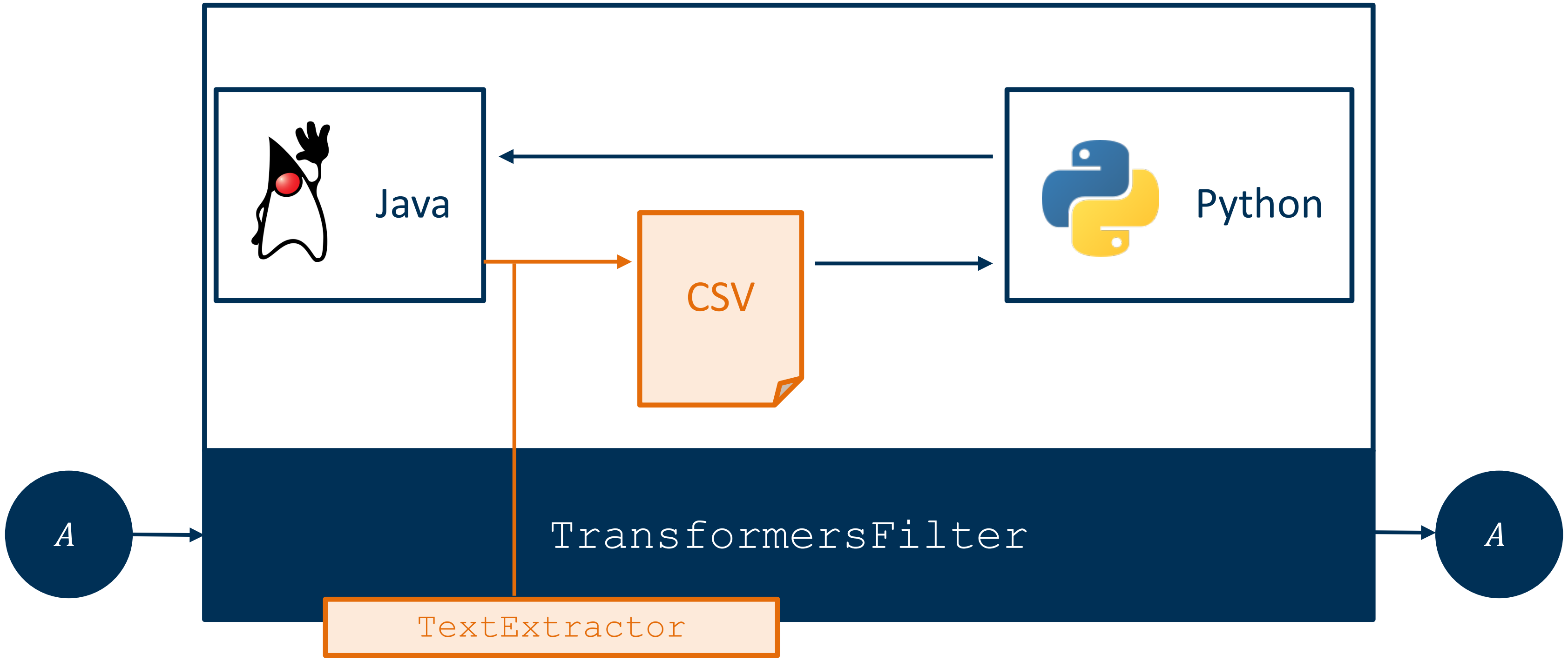
The Transformer Pipeline



Generating Negatives

- positives:
  - (1) reference
  - (2) high-precision sytem
- negatives
  - generate randomly (`AddNegativesRandomlyAbsolute`, `AddNegativesRandomlyShare`)
  - generate through one-to-one assumption, e.g. incomplete or unknown reference `AddNegativesRandomlyOneOneAssumption`
- all new strategies implement interface `AddNegatives`

TextExtractors & Technical Details



TextExtractors & Technical Details

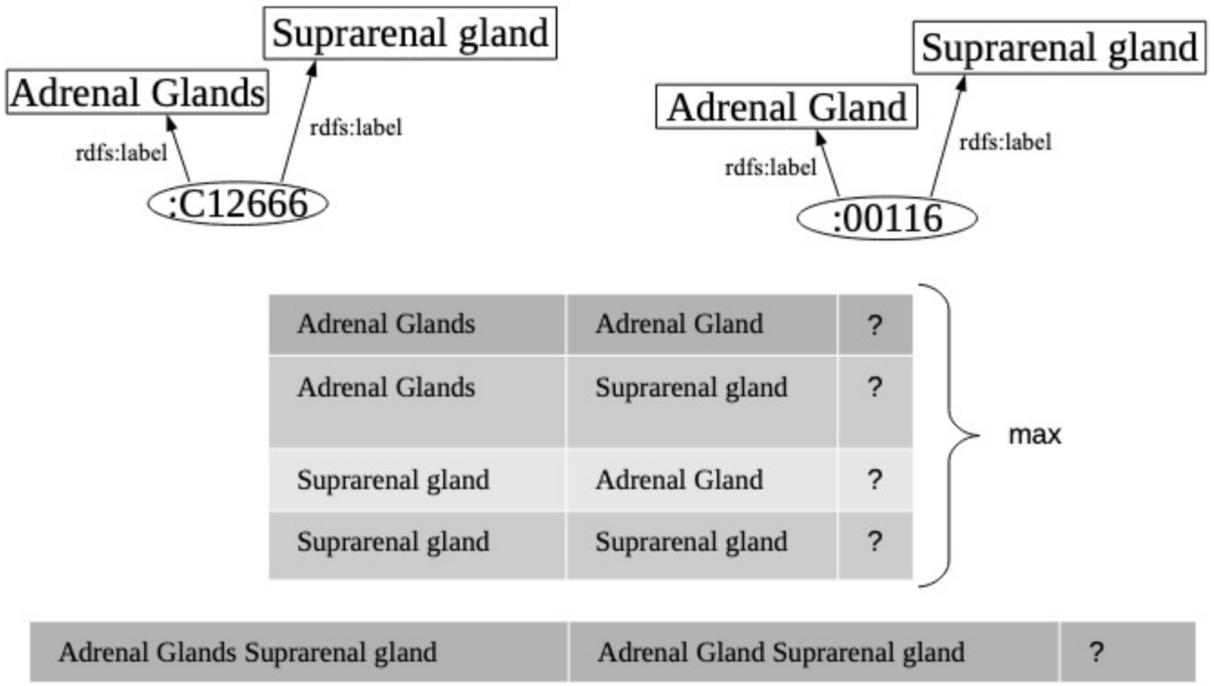


Fig. 2. Optional multi-text mechanisms implemented in class `TransformersFilter`.

Hyperparameter Optimization

UNIVERSITY OF MANNHEIM
School of Business Informatics and Mathematics

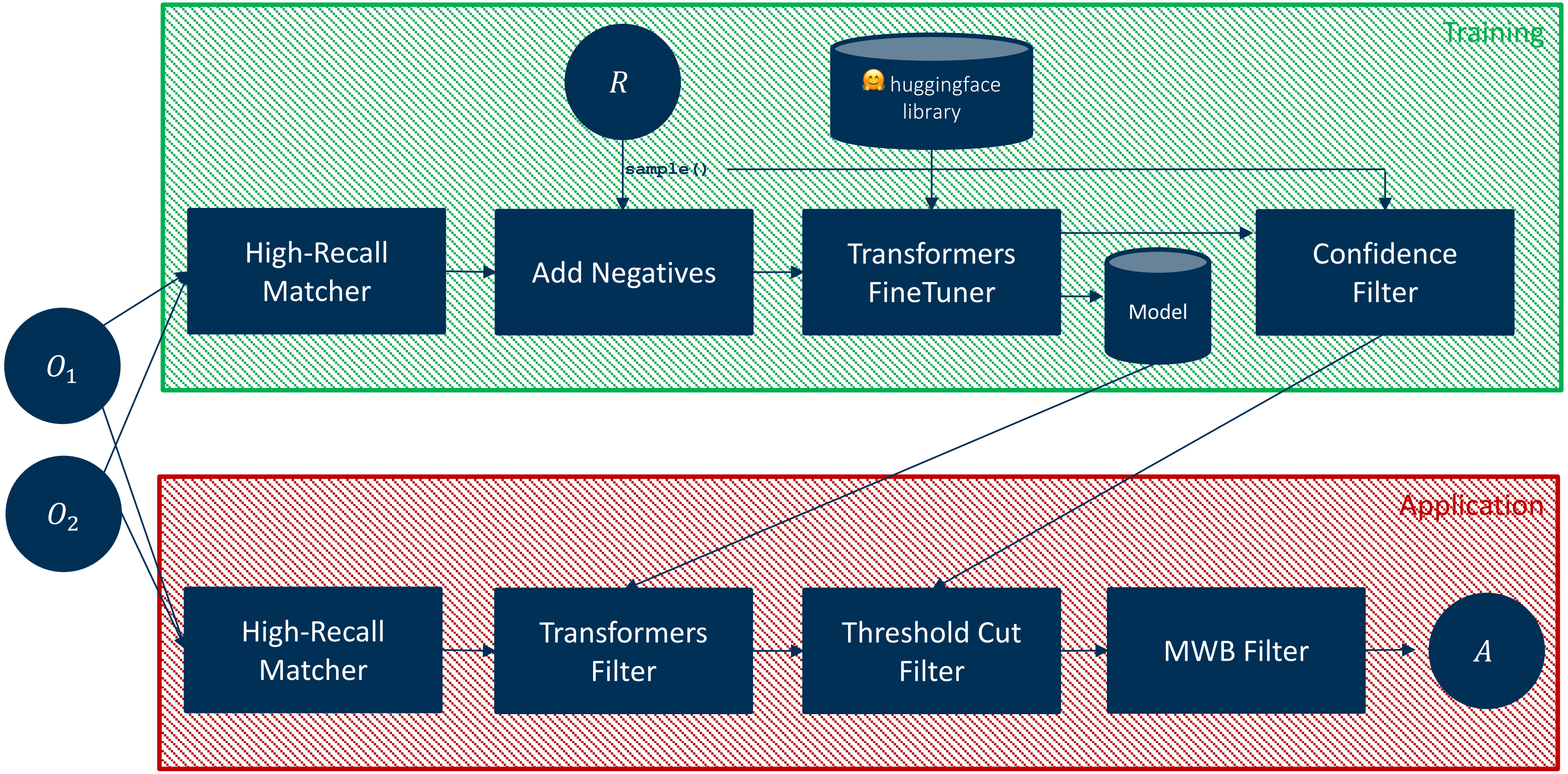
- via Ray Tune
- initial hyperparameter search space
  - learning rate: loguniform distribution between 10^{-6} and 10^{-4}
  - epochs: random choice between 1 and 5
  - seed: uniform distribution between 1 and 40
  - batch size: random choice of 4, 8, 16, 32, 64
    (max size autoamtically determined)
- optimizable metrics: loss, accuracy, F1, recall, precision, AUC
- class `TransformersFineTunerHpSearch`

It’s not complicated!

```
public class TransformerApplyExample extends MatcherPipelineYAAAJena {

    @Override
    protected List<MatcherYAAAJena> initializeMatchers() {
        List<MatcherYAAAJena> list = new ArrayList<>();

        // some recall matcher
        list.add(new RecallMatcher());

        // transformer filter
        list.add(new TransformersFilter(
                new TextExtractorAllLiterals(),
                "albert-base-v2"));

        // some post processing steps
        list.add(new ConfidenceFilter(0.75));
        list.add(new MaxWeightBipartiteExtractor());

        return list;
    }
}
```

Experiments

Experiments | Pipeline



Experiments

- tracks: *Anatomy*, *Confernce*, *Knowledge Graph*
- with and without fine-tuning
- sampling rate: `20%`
- models
  - `bert-base-cased`
  - `roberta-base`
  - `albert-base-v2`

Results

UNIVERSITY OF MANNHEIM
School of Business Informatics and Mathematics

		Conference			Anatomy			Knowledge Graph		
		P	R	F1	P	R	F1	P	R	F1
Baseline	**SimpleString**	0.710	0.498	0.586	0.964	0.708	0.816	0.909	0.727	0.808
	High Recall	0.450	0.561	0.179	0.037	0.942	0.071	0.167	0.915	0.283
Zero-Shot	**bert-base-cased (mrpc-tuned)**	0.650	0.548	**0.594**	0.531	0.817	0.644	0.739	0.714	0.726
Fine-Tuned (per Track)	**bert-base-cased**	0.748	0.361	0.487	0.726	0.689	0.707	0.941	0.789	**0.859**
	roberta-base	0.667	0.498	0.570	0.715	0.749	0.732	0.400	0.388	0.393
	albert-base-v2	0.812	0.397	0.533	0.854	0.825	**0.839**	0.687	0.665	0.676

Table 1. Results of non-fine-tuned and fine-tuned transformer models (multi-text) with 20% sampling from the reference alignment. As per OAEI customs, we report micro average scores for the conference and macro average scores for the KG track.

- fine-tuning increases performance
- `albert-base-v2` and `bert-base-cased` achieved best results
- minor improvements through hyperparameter tuning

Results

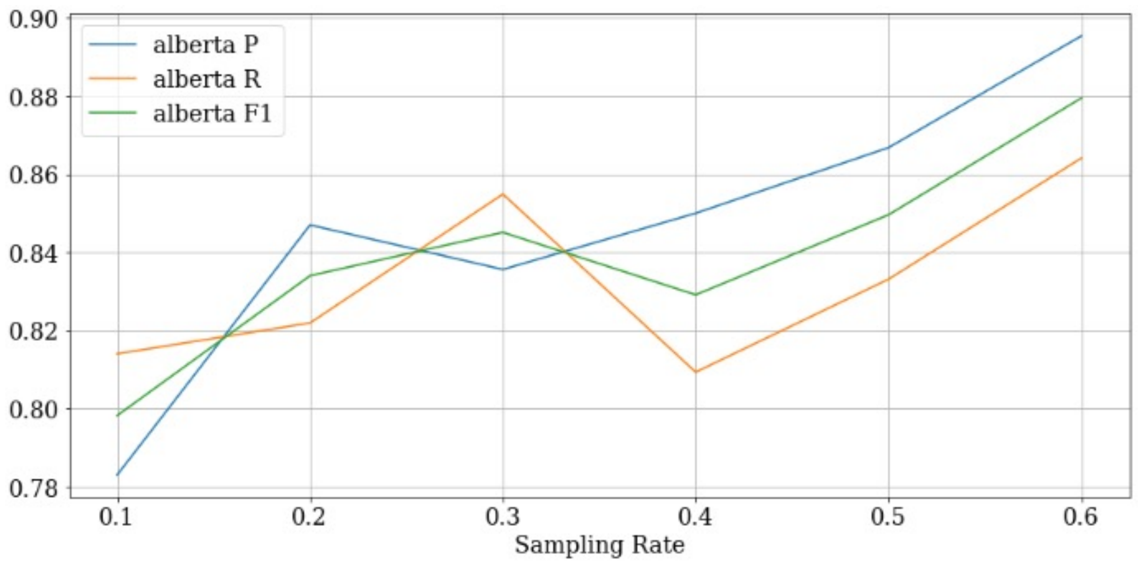


Fig. 4. `albert-base-v2` performance on the anatomy track using different reference sampling rates.

- performance increases with increasing sample rate
- low sampling (10-20%) is sufficient for good results

Conclusion and Future Work

Conclusion & Future Work

UNIVERSITY OF MANNHEIM
School of Business Informatics and Mathematics

- Transformers for MELT allow the broad usage of transformers without deep technical skills (out-of-the-box components)
- Tranformers are promising for the matching domain as shown in experiments
- in the future, we plan to
  - add further matching components (alignment repair)
  - include sentence transformers (transformers as matching component, not as filter)

UNIVERSITY OF MANNHEIM
School of Business Informatics and Mathematics

Thank you.

Sven Hertling
University of Mannheim
https://www.uni-mannheim.de/dws/people/researchers/phd-students/sven-hertling/

sven@informatik.uni-mannheim.de

Jan Portisch
University of Mannheim / SAP SE
https://www.jan-portisch.eu

jan@informatik.uni-mannheim.de / jan.portisch@sap.com

Heiko Paulheim
University of Mannheim
http://www.heikopaulheim.de/

heiko@informatik.uni-mannheim.de