DUUI: A Toolbox for the Construction of a new Kind of Natural Language Processing

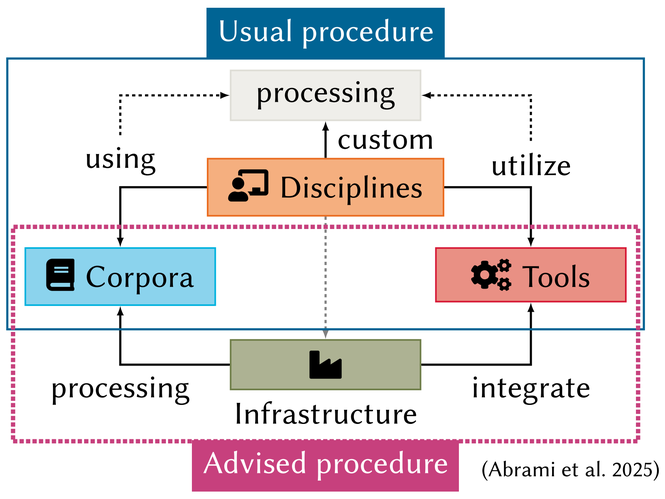
Giuseppe Abrami Daniel Baumartz Alexander Mehler

Goethe University Frankfurt | Text Technology Lab

DHd 2025 3-7 March 2025
Bielefeld, Germany

Motivation: Natural Language Processing

- Today's natural language processing (NLP) is characterized by an ever-increasing amount of (heterogeneous) tools, models and accessible corpora.
- This leads to non-trivial challenges for various disciplines, which result in a considerable time investment in order to perform the analyses in a moderate amount of time with moderate use of resources.
- Furthermore, non-standardized data formats for input and output complicate the interchangeability of tools and the ability to analyze them, which leads to situations such as these:



(Abrami et al. 2025)

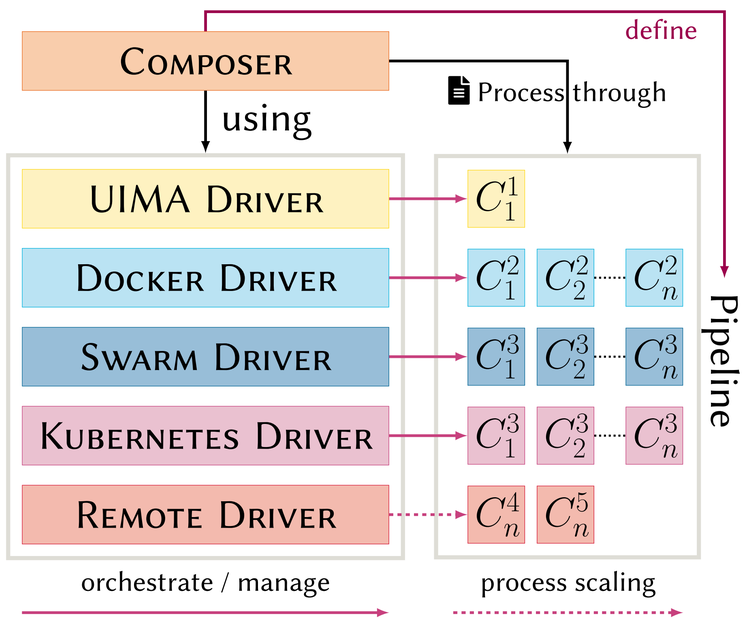
- In order to address this challenge and enable a novel approach to corpus processing in a distributed and effective way, Docker Unified UIMA Interface (DUUI – Leonhardt et al. 2023) was developed, utilizing the UIMA (Ferrucci et al. 2009) annotation standard by encapsulating different analysis methods in microservice-oriented container solutions (Component).

DUUI: Features

- Utilizes Docker-based microservice architectures to:
  - Capturing heterogeneous annotation landscapes
  - Capturing heterogeneous implementation landscapes
  - Implementation in different runtime environments by Drivers.
- Enabling horizontal (cluster-based) and vertical (parallel processing) scaling with Docker Swarm and Kubernetes (Abrami et al. 2025)
- Easy extensibility (Abrami and Mehler 2024) and usability through strict container encapsulation of components using Lua (Ierusalimschy, Figueiredo, and Celes 2007).

Architecture

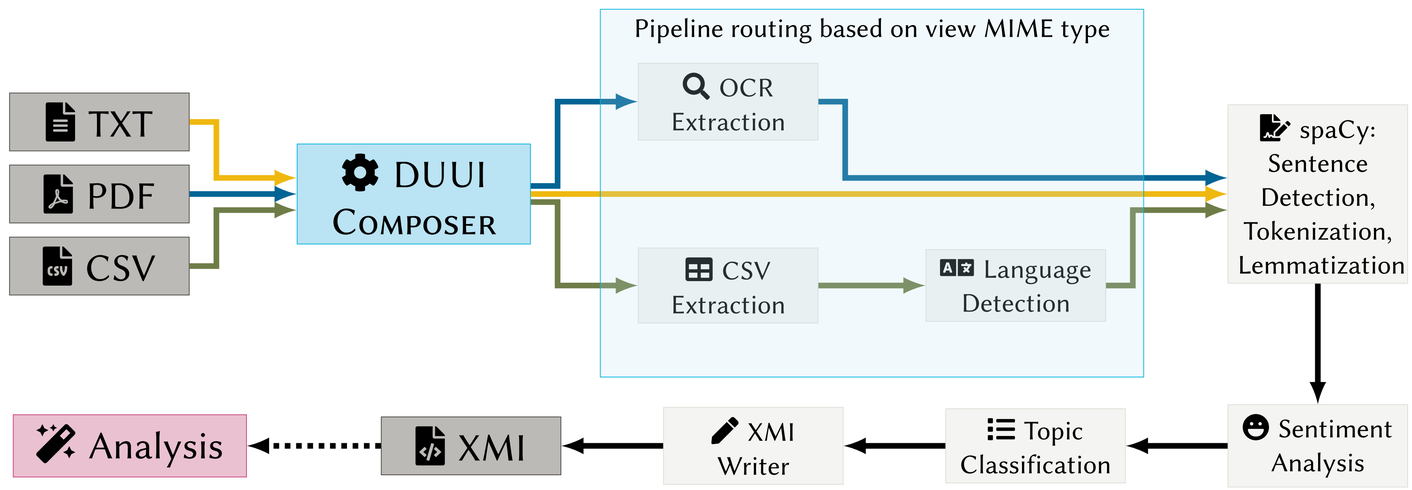
(Abrami et al. 2025, slightly modified)



Construction of a type of NLP

```
DUUIComposer composer = new DUUIComposer()
   .withWorkers(3)
   .withSkipVerification(true)
   .withLuaContext(new DUUILuaContext().withJsonLibrary());

composer.addDriver(new DUUIDockerDriver(), new DUUIUIMADriver(), new KubernetesDriver());

DUUIAsynchronousProcessor reader = new DUUIAsynchronousProcessor(new DUUIFileReader(
      inDir.toString(), ".xmi.gz",
      1, 0, false, "", false, "de", 0,
      outDir.toString(), ".xmi.xmi.gz"
));

composer.add(new DUUIDockerDriver.Component("docker.texttechnologylab.org/duui-spacy")
        .withImageFetching()
        .withScale(3));

composer.add(
   new DUUIDockerDriver.Component("docker.texttechnologylab.org/duui-ddc-fasttext:2.3.2")
      .withParameter("ddc_variant", "ddc2")
      .withParameter("selection", "text"));

composer.add(
   new KubernetesDriver.Component("docker.texttechnologylab.org/duui-transformers-sentiment")
      .withParameter("model_name", "oliverguhr/german-sentiment-bert")
      .withParameter("selection",
       "de.tudarmstadt.ukp.dkpro.core.api.segmentation.type.Sentence"));

JSONObject llmArgsJson = new JSONObject();
llmArgsJson.put("model", "deepseek-r1:70b");
llmArgsJson.put("temperature", 0.8);
composer.add(
   new DUUIDockerDriver.Component("docker.texttechnologylab.org/duui-core-llm-rating:0.0.2")
      .withParameter("llm_args", llmArgsJson.toString())
      .build().withTimeout(1000L));

composer.add(new DUUIUIMADriver.Component(createEngineDescription(XmiWriter.class,
   XmiWriter.PARAM_TARGET_LOCATION, outDir.toString()
)).build());

composer.run(reader, "duui_pipeline");
composer.shutdown();
```

DUUI Use-Case



Utilizing heterogeneous tools, such as spaCy, sentiment analysis and topic classification, to process diverse data sources, including PDF documents, CSV tables, and text files, to generate a standardized output suitable for systematic analysis.

Future Work

- Implementation of further Drivers and Components
- Development of a dynamic, web- and API-based DUUI interface
- Breaking down the barriers toward multicodal processing of UIMA documents

Resources

DockerUnifiedUIMAInterface @ TTLab

References

Abrami, Giuseppe, Markos Genios, Filip Fitzermann, Daniel Baumartz, and Alexander Mehler (2025). "Docker Unified UIMA Interface: New perspectives for NLP on big data." In: *SoftwareX* 29, p. 102033.

Abrami, Giuseppe and Alexander Mehler (Aug. 2024). "Efficient, uniform and scalable parallel NLP pre-processing with DUUI: Perspectives and Best Practice for the Digital Humanities." In: *Digital Humanities Conference 2024 - Book of Abstracts (DH 2024)*. Ed. by Jajwalya Karajgikar, Andrew Janco, and Jessica Otis. DH. Washington, DC, USA: Zenodo, pp. 15–18.

Ferrucci, David, Adam Lally, Karin Verspoor, and Eric Nyberg (2009). *Unstructured Information Management Architecture (UIMA) Version 1.0*. OASIS Standard.

Ierusalimschy, Roberto, Luiz Henrique de Figueiredo, and Waldemar Celes (2007). *The Evolution of Lua*.

Leonhardt, Alexander, Giuseppe Abrami, Daniel Baumartz, and Alexander Mehler (2023). "Unlocking the Heterogeneous Landscape of Big Data NLP with DUUI." In: *Findings of the Association for Computational Linguistics: EMNLP 2023*. Ed. by Houda Bouamor, Juan Pino, and Kalika Bali. Singapore: Association for Computational Linguistics, pp. 385–399.

Gefördert durch DFG Deutsche Forschungsgemeinschaft 326061700
Specialised Information Service Biodiversity Research 5404