A Data Science Environment for Legal Texts

Bernhard WALTL [a,1] , Marin ZEC [a] and Florian MATTHES [a]
[a] *Software Engineering for Business Information Systems, Technische Universität München, Germany*

Introduction, Motivation and Contribution

Since the beginning of legal informatics, data analysis has been very attractive to the scientific community in different domains and areas. Modern algorithms and mining technologies are able to unveil structural, such as network-like features and semantic properties that might be contained explicitly or implicitly in texts. This becomes evident by screening the topics of recent scientific conferences and workshops (e.g., ICAIL 2015 [1]). However, only few efforts were spent on the development of a generic environment taking into account the characteristics of legislative systems, which would foster the re-usage of results, such as models, components or frameworks.

This paper proposes a reference architecture, which can easily be extended to specific research questions and use cases. It is designed and implemented as a generic environment allowing state-of-the-art text analysis. We performed a case study on the German tenancy law. Thereby, we analyzed the evolution the law in the last 25 years. Due to space limitations the results can be obtained on request or on wwwmatthes.in.tum.de.

Reference Architecture for Legal Data Science

In order to set up a data science environment that can easily be extended and adapted the architectural design has incorporate basic software engineering design principles such as low coupling, reuse of components and easy extensibility [2].

Figure 1 shows the comprehensive reference architecture, covering an importer, a data storage and access layer, a text mining engine, an exporter, and a user interface. Based on the reference architecture we developed a collaborative web application with a Java back-end using the Play Framework.

Data Model, Storage and Access The data model used in the environment is able to represent laws, judgments and every kind of unstructured but nested textual information. Thereby, we follow a recent knowledge engineering approach (i.e., Wiki), whereby relevant attributes are stored in Key-Value-Maps, such that the usage of attributes is not constrained beforehand.

[1] Corresponding Author: Bernhard Waltl, Software Engineering for Business Information Systems, Boltzmannstr. 3, 85748 Garching bei München, Germany; E-mail: b.waltl@tum.de.

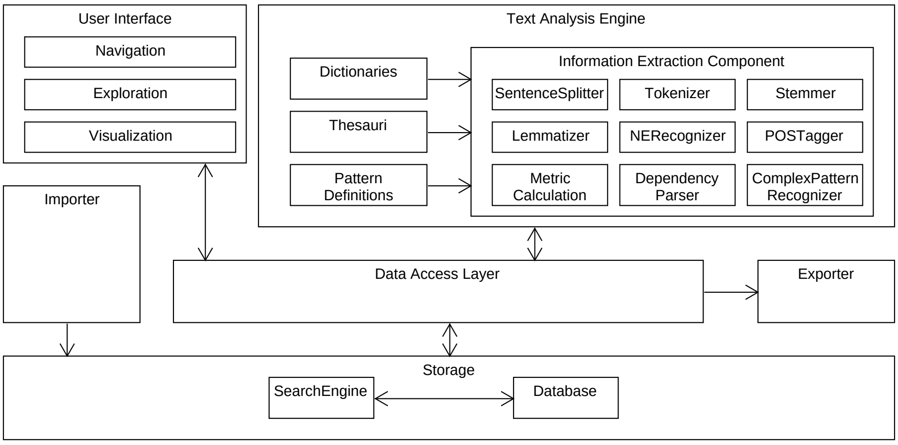

Figure 1. Legal data science reference architecture for German legal data.

Text Mining Engine The text mining engine consists of a variety of reusable components and can easily be extended by new components. As baseline architecture Apache UIMA was used. Beside state of the art analysis components, such as metric calculation, POS tagging, Named-Entity-Recognition (NER), Deep Parsing, we are using Apache UIMA Ruta (Rule-based text annotation). Apache Ruta allows us to formulate more complex rules than plain regular expression by considering linguistic properties and semantic features.

Importer The importing structure is required to transform the input data, which can be of any data type (pdf, xml, JSON, etc.), into the data model of our system. Thereby the structure reflects modular design principles such that it can easily be extended to new data sources and channels (e.g., OData, RESTful APIs, etc.).

Exporter The exporter component provides interfaces for other applications (e.g., REST APIs) to use and reuse the information stored. Based on use cases the exporter component can easily be enhanced and adapted to support more functionality.

User Interface The user interface is encapsulated from the data model and logic of the application. Depending on the concrete use cases various specific implementations are possible. Those can be tailored to navigation, exploration or visualization.

Acknowledgment

This research was sponsored in part by the German Federal Ministry of Education and Research (BMBF) (project “Software Campus (TU München)”, grant no. 01IS12057).

References

[1] *Proceedings of the 15th International Conference on Artificial Intelligence and Law*, 2015. [Online]. Available: http://dl.acm.org/citation.cfm?id=2746090

[2] R. S. Pressman, *Software engineering: a practitioner’s approach.* Palgrave Macmillan, 2005.