LEXIA

Legal Information Analysis, Exploration, and Reasoning Platform

Abstract	**Lexia** ist ein kollaboratives Web Framework zur semantischen Erschließung von textuellen Dokumenten. **Lexia** ist Teil der interdisziplinären Lexalyze Initiative der Technischen Universität München und wird seit 2014 kontinuierlich weiterentwickelt. **Lexia** besteht aus verschiedenen lose gekoppelten Softwarekomponenten und folgt einer service-orientierten Architektur. Im Kern wurde eine leicht anzupassende Softwarearchitektur implementiert, die sich zur Analyse von großen unstrukturierten, textuellen Datenbeständen eignet. Der Fokus liegt auf dem Erkennen relevanter Textpassagen und Klassifizieren von Dokumenten bzw. Dokumentteilen. Umgesetzt wird dies durch integratives Zusammenspiel von regelbasierten Ansätzen und heuristischen Verfahren aus dem Bereich Machine Learning. In mehreren Forschungsprojekten mit Kooperationspartner wurde **Lexia** bereits erfolgreich zur Analyse von Gesetzen, Urteilen, Verträgen, Entwürfen und Nachrichten eingesetzt.
Projektleiter	**Bernhard Waltl** b.waltl@tum.de
Lehrstuhl	**Prof. Dr. Florian Matthes** Software Engineering für betriebliche Informationssysteme Boltzmannstraße 3 85748 Garching bei München Internet: wwwmatthes.in.tum.de

1 Hintergrund

Die zunehmende Digitalisierung dringt in viele Bereiche der Gesellschaft vor. Die Verfügbarkeit von Informationen ist durch den hohen Vernetzungsgrad immer stärker gegeben. Dabei spielen Informationssysteme eine wichtige Rolle. Diese stellen Infrastruktur und Ordnungsrahmen bereit innerhalb derer Information abgespeichert, abgerufen, geteilt, und wiederverwendet wird. Gerade im betrieblichen Kontext spielen diese eine wichtige Rolle, da im Zeitalter der Wissensarbeit das Wissensmanagement zur fundamentalen Herausforderung von Unternehmen wird.
Der Lehrstuhl Software Engineering für betriebliche Informationssysteme[1] (sebis) untersucht seit seiner Gründung die Ausgestaltung und Umsetzung betrieblichen Informationssysteme. In den

[1] wwwmatthes.in.tum.de

letzten Jahren hat sich aufgrund von drei wesentlichen Beobachtungen der Einsatz von Daten- und Textmining Verfahren als zentral für zukünftige Entwicklungen herausgestellt:

1. Verfügbarkeit digitaler Daten
2. Performance der Algorithmen
3. Leistungsfähigkeit der Infrastrukturen

Die Forschungsgruppe an der TUM untersucht daher die Potentiale von Informationssystemen im Zusammenspiel mit Textmining und Machine Learning. Hierbei haben sich bereits diverse Kooperationen mit Industriepartnern in Forschungs- und Entwicklungsprojekten ergeben.

2 Anwendungsfälle

Zahlreiche Anwendungsfälle konnten bereits identifiziert werden, die nachfolgend kurz vorgestellt werden. Sofern vorhanden wird auch auf öffentlich verfügbare und bereits publizierte Arbeitsergebnisse verwiesen.

2.1 Unterstützung von Redaktionsstäben

Die Strukturierung von Dokumenten auf Basis von Metadaten oder Inhalten, die aus dem Dokument extrahiert, werden können ist derzeit noch eine weitverbreitet manuelle Tätigkeit. Dabei werden Domänenexperten dazu ausgebildet wichtige Information aus einem potentiell sehr großen Dokument zu extrahieren. Üblicherweise dient dieser Vorgang dazu das unstrukturierte Dokumente in eine (semi-)strukturierte Form zu überführen.

Dieser manuelle Prozess ist sehr zeit-, und kostenintensiv und lässt sich bis zu einem hohen Grad sehr gut durch computerlinguistische Verfahren unterstützen. Regelbasierte Verfahren können in Kombination mit heuristischen Machine Learning Verfahren Vorschläge zur Klassifikation und Extraktion von Metadaten machen, die von Endbenutzern dann angenommen oder verworfen werden. Die zur Verfügung gestellte Usereingabe (Trainingsdaten) können dienen als Grundlage für weitere Verfeinerung (sog. Active learning Komponenten).

Veröffentlichung

Waltl, B.; Landthaler, J.; Scepankova, E.; Matthes, F.; Geiger, T.; Stocker, C.; Schneider, C.: Automated extraction of semantic information from german legal documents, IRIS: Internationales Rechtsinformatik Symposium, Salzburg, Austria, 2017

Landthaler, J.; Waltl, B.; Matthes, F.: Unveiling References in Legal Texts - Implicit versus Explicit Network Structures, IRIS: Internationales Rechtsinformatik Symposium, Salzburg, Austria, 2016

2.2 Analyse von Gesetzen, Urteilen und Verträgen

Die Analyse rechtlich relevante Texte, insbesondere Gesetze, Urteile und Verträge ist für Anwälte, Rechtswissenschaftler und Personen der Rechtspflege von fundamentaler Bedeutung. Textanalyse Verfahren sind hierbei insbesondere vorteilhaft, wenn auf Basis von linguistischer Muster oder Ähnlichkeitsanalyse schnell Aussagen über den Inhalt eines Dokuments gemacht werden müssen. Der Vorteil von Algorithmen besteht in der Geschwindigkeit und Präzision mit den großen Dokumentkorpora analysiert werden.

Algorithmen sind beispielsweise in der Lage Normen hinsichtlich der Funktion (Legaldefinition, Verbot, Erlaubnis, Freistellung, etc.) und des Inhalts (Haftung, Zahlung, IP, etc.) zu bewerten. Diese Klassifizierung erfolgt in Lexia durch ein hybrides und mehrstufiges Verfahren, in dem zunächst regelbasierte Ansätze (Pattern Definitions, Dictionaries und Thesauri) verwendet werden. Nach und nach werden heuristische Verfahren (Naive Bayes, SVM[2], log. Regression) dazu verwendet um linguistische Grenzfälle aufzulösen und ggf. einer Klasse zuzuordnen.

Veröffentlichung
Waltl, B.; Matthes, F.; Waltl, T.; Grass, T.: LEXIA - A Data Science Environment for Semantic Analysis of German Legal Texts, IRIS: Internationales Rechtsinformatik Symposium, Salzburg, Austria, 2016

Waltl, B.; Zec, M.; Matthes, F.: LEXIA: A Data Science Environment for Legal Texts, Jurix: International Conference on Legal Knowledge and Information Systems, Braga, Portugal, 2015

2.3 Verbessern von Such- und Explorationsprozesse

Nach der semantischen Erschließung eines rechtlich-relevanten Dokuments, die durch Textanalyse Verfahren durchgeführt werden kann, bleibt die Frage nach der Einbettung dieser Information in den Suchprozess noch ungelöst. Im Rahmen der Forschungen von Lexia werden intelligente und intuitive Benutzeroberflächen erforscht, die es erlauben komplexe Informationen in das Dokument einzubetten, ohne den Benutzer dabei zu überfordern. Dies kann beispielsweise durch farbliches Hervorheben relevanter Information oder durch Einbettung der Information in sogenannte Such-Facetten erfolgen. Letztere stellen eine deutliche Verbesserung der Volltextsuche, die mittlerweile zum Standard von Suchdatenbanken zählt, dar.

Des Weiteren ist das System in der Lage Suchanfragen zu verarbeiten um auch verwandte oder ähnliche Suchbegriffe in die Suche ein- oder auszuschließen. Auch die Ähnlichkeitssuche über

[2] Support Vector Machine

Dokumente oder Dokumentteile wurde in mehreren Forschungsarbeiten behandelt und signifikante Ergebnisse hinsichtlich der Brauchbarkeit erzielt.

Veröffentlichung
Landthaler, J.; Waltl, B.; Holl, P.; Matthes, F.: Extending Full Text Search for Legal Document Collections using Word Embeddings, Jurix: International Conference on Legal Knowledge and Information Systems, Sofia Antipolis, France, 2016

Waltl, B.; Altamirano, Laura S.; Matthes, F.: Applying Lexical Knowledge to Support Search and Navigation in Legal Databases, IRIS: Internationales Rechtsinformatik Symposium, Salzburg, Austria, 2016

2.4 Formalisieren von Entscheidungsstrukturen

Neben der Analyse von textueller Information erlaubt es Lexia Entscheidungsstrukturen zu formalisieren und dieses anschließend automatisiert auswerten zu lassen. Dabei muss die Entscheidungsstruktur zunächst abgebildet werden. Dies erfolgt in einer FEEL[3] die neben logischen auch arithmetischer Operationen unterstützt. In Lexia ist die Erstellung der Formalisierung strikt von der Ausführung getrennt. Das heißt die Daten (Fakten) auf Basis derer automatisiert Entscheidungen getroffen werden sind getrennt von der Entscheidungslogik.

Dies hat den Vorteil, dass Personen nicht notwendigerweise über die Entscheidungslogik informiert sein müssen und nur ihren Fall bearbeiten können. Die Fakten werden in ein Formular eingetragen und das System kann logische Schlüsse und Ableitungen vollautomatisch durchführen. Die Formalisierung erlaubt es Entscheidungen zu optimieren bzw. Inkonsistenzen aufzuzeigen.

Veröffentlichung
Waltl, B.; Oppmann, D.; Reschenhofer, T.; Matthes, F.: Modeling, Execution and Analysis of Formalized Legal Norms in Model Based Decision Structures, ***Working paper to appear in 2017***

[3] Friendly Enough Expression Language

3 Technologie

Die Abbildung 1 zeigt das zu implementierende Framework in einem kompakten, leicht abgewandelten UML Komponentendiagramm. Diese zeigt die einzelnen Komponenten, deren Zusammensetzung und den Austausch zwischen diesen anhand von Daten und Informationsströmen.

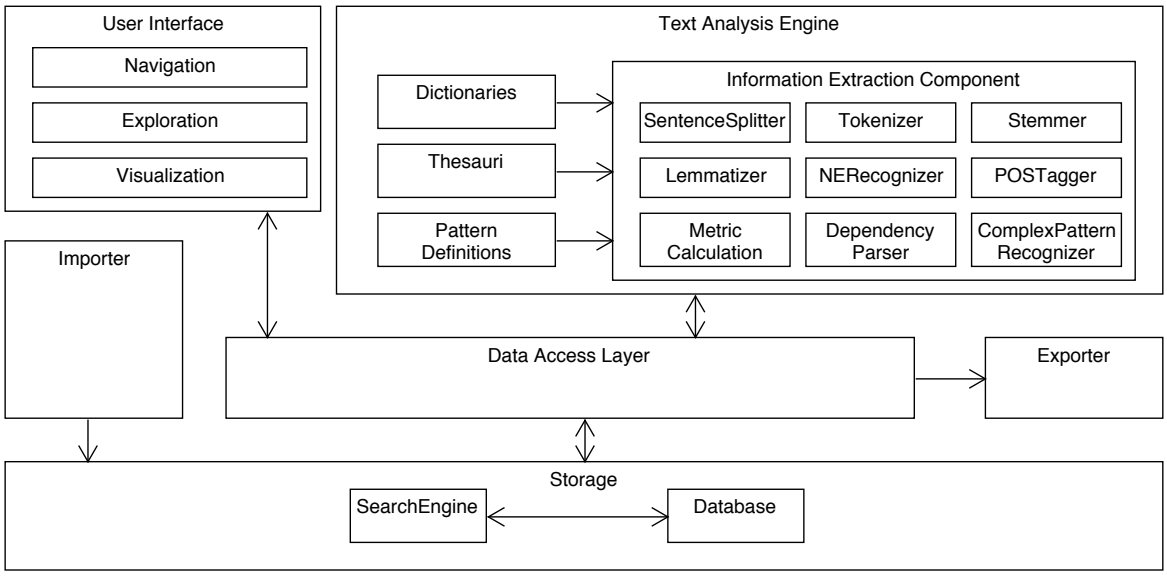

Abbildung 1. Übersicht über die modulare Architektur der Software Komponenten des Frameworks

Dabei ist Lexia in sechs wesentliche Bestandteile gegliedert:

- **Importer**
  Das Hinzufügen neuer Dokumente (Gesetze, Urteile, Kommentare, Artikel, Verträge, etc.) erfolgt über eine eigene Komponente die den jeweiligen Datentyp erfasst und in das interne Datenmodell überführt.
- **Exporter**
  Der Exporter ermöglich den Zugriff auf den internen Datenbestand über eine wohldefinierte API. Derzeit ist eine zugeschnitten REST API, sowie ein Export von Daten im CSV Format implementiert. Dies ermöglicht die Weiterverwendung des Systems in einem komplexen Ökosystem und im Zusammenspiel mit anderen Systemen (zB SAP, Exchange, etc.)
- **Storage**
  Daten werden interne in einem effizienten Suchcluster abgelegt. Dieser kann hervorragend mit textuellen Daten umgehen und liefert sehr gute Funktionalität hinsichtlich

textueller Attribute und Aggregationen. Hierfür wurde der dokumentenbasierte Index ElasticSearch[4] gewählt.

- **Data Access Layer**
  Der Zugriff auf die Daten erfolgt einheitlich über den Data Access Layer. Hierbei werden Aggregationen durchgeführt und einheitliche Abfragen an den Storage gestellt. Dies verhindert die doppelte Implementierung von Software und ermöglicht einfachere Optimierung von potentiell sehr komplexen Abfragen.
- **User Interface**
  Das User Interface ermöglicht die Interaktion mit dem System. Dabei können Datenbestände (Textdokumente) gesichtet und durchsucht werden. Hier können auch diverse Verarbeitungsprozesse angestoßen und parametrisiert werden.
  Außerdem wurden alternative Repräsentationsformen auf den zugrundliegenden Datenbestand implementiert (Dashboard und Netzwerkview).
- **Text Analysis Engine (TME)**
  Die Text Mining Engine stellt das eigentliche Herzstück des Frameworks dar. Hierin sind die verschiedenen Softwarekomponenten implementiert und deren Abhängigkeiten umgesetzt. State-of-the-Art Komponenten wurden hierbei wiederverwendet bzw. auf die Domäne des Rechts angepasst. Anhand verschiedener zT sehr komplexer linguistischer Vorgänge wurde die Leistungsfähigkeit der computer-linguistischer Komponenten untersucht deren Potentiale Grenzen so präzise als möglich aufgezeigt.

Das Kernstück der TME ist eine sog. „Pipes&Filters“ Architektur (siehe Abbildung unten), die es erlaubt beliebig komplexe Datenverarbeitungen zu konfigurieren und durchzuführen. Dabei werden Softwarekomponenten aneinandergereiht und über wohldefinierte Schnittstellen verbunden. Die einzelnen Komponenten erfüllen dabei genau eine einzige Aufgabe und können hochspezialisiert entwickelt werden, zB Erkennen von Verweisungen in Texten. Innerhalb einer Pipeline bauen diese Komponenten dann aufeinander auf und können sich auf die Ergebnisse der vorgehenden Komponenten berufen. Eine nachfolgende Komponente kann also die Verweisungen auflösen die zuvor erkannt wurden. Das hat außerdem den Vorteil, dass diese Komponenten ohne Probleme verbessert werden können hinsichtlich Präzision, Recall und Performance und keine anderen Komponenten von diesen Änderungen betroffen sind. Das Gesamtergebnis der Verarbeitung hingegen steigt. Die Komponenten können auch einfach an andere Forschungsgruppen weitergegeben werden und

[4] https://www.elastic.co/

von diesen angepasst werden. Idealerweise führt dies zu einem gemeinsamen Pool von Text Mining Komponenten von deren Weiterentwicklung jeder profitieren kann.

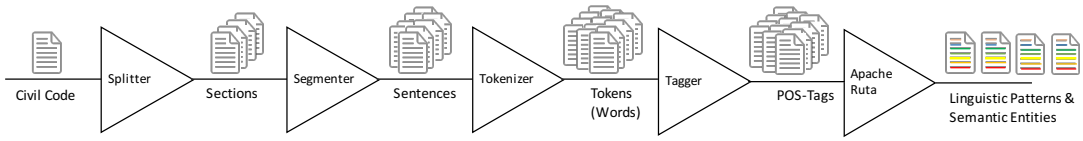

Abbildung 2. Illustration der Pipes&Filter Architektur der Text Mining Engine

Die Pipes&Filters Architektur hat auch noch einen weiteren softwaretechnischen Vorteil: die Ausführung lässt sich gut parallelisieren und die Implementierung eignet sich für die verteilte und webbasierte Infrastrukturen (Multi-Threading), zB Cloud-Services. Damit entspricht sie dem Stand der Wissenschaften und Technik.