

## Allgemeines

- Bei den ersten drei Aufgaben wären „zwei von drei“ zu verteilen.
- Da die Daten in vielen Aufgaben neben Ausgabe auch zu speichern sind und die Struktur aber schwer beschreibbar ist möglichst neo4j (Graphendatenbank) einsetzen.
- Wenn Thesauri eingesetzt werden möglichst frei verfügbare Daten verwenden.
- Generell sind die Aufgaben verhandelbar, auch ganz andere Aufgaben sind in Absprache möglich.
- Verwendung von Tools wie spaCy, NLTK, pattern, RASA-NLU bitte im Vorfeld klären.
- Generell möglichst viel Spezialitäten aus der deutschen Sprache nutzen (STTS-Tagset)
- Fortschrittsberichte als kleinste Seminarvorträge an folgenden Terminen
  - 22.05.2019
  - 05.06.2019
  - 19.05.2019
  - 26.05.2016
- Abgabe spätestens in Woche ab 09.09.2019
- Abschluss-Kolloq ! (Termin tbd)
- Die Lösungen müssen als lauffähige Installation in der Infrastruktur des Labors 0.25 hinterlassen werden.
- Einfache Projekt-Doku (Foliensatz)

## Aufgabe 1: ICD-Klassifikation (1)

Entwickeln Sie einen Klassifikator, der anhand von Real-Daten sowie den offiziellen Diagnosen aus dem Internet eine automatische ICD-10-Klassifikation durchführt.

## Aufgabe 2: ICD-Klassifikation (2)

Entwickeln Sie einen ATNP zur Klassifikation der ICD-Diagnosen.

## Aufgabe 3: ICD-Klassifikation (3)

Entwickeln Sie eine Grammatik für die ICD-Diagnosen und implementieren Sie diese in NLTK.

## Aufgabe 4: Chatbot mit RASA-NLU

Entwickeln Sie einen Chatbot, der für die Homepage der HS Aalen „Gespräche“ v.a. für Studieninteressierte bietet. Der Bot soll möglichst portabel sein (Bildungsdaten KBW)

## Aufgabe 5: Wissensextraktor

Entwickeln Sie einen Bot, der eine Homepage verschlagwortet und das extrahierte Wissen in Form von „Sätzen“ einer Graphendatenbank (SPO – Subjekt Prädikat Objekt) ausgibt.

## Aufgabe 6: Suchmaschine

Entwickeln Sie eine Suchmaschine, die zusätzlich zur Volltextverschlagnwortung eine Verschlagnwortung gem. Zipf / Luhn bietet. Bei der Abfrage sollen Synonym- und Homonymprobleme berücksichtigt werden, beim Ranking TF und IDF.

## Aufgabe 7: Klassifikation von Texten

Klassifizieren Sie die die Texte der Homepage der HS Aalen nach Kriterien wie „W / MINT“ oder gar nach Fakultäten. (andere Textkorpora sind ebenfalls zulässig)

## Aufgabe 8: NLP-Suche-Backend

Entwickeln Sie ein Backend, welches zu Suchanfragen eine Grammatikstruktur bzgl. Satz und Wörtern ausgibt.  
(Aufgabe vielleicht etwas zu „klein“?)

## Aufgabe 9: Prüfungs-Autokorrektur

Entwickeln Sie ein Programm, das Freitext-Prüfungen – auf Basis eines Scans – versucht einer Autokorrektur (auf Basis von Schlagwörtern, Regeln o.ä.) zu unterziehen.

## Aufgabe 10: Klassifikation von EVASYS-Freitexten

Versuchen Sie, EVASYS-Kommentare zu klassifizieren (Freitext, Stichwörter, Positiv-Negativ, ....)

## Aufgabe 11: Check von Modulhandbüchern

Klassifizieren Sie Modulhandbuch-Texte, ob und ggf. wie weit sie formale Vorgaben - insbesondere den didaktischen Schlüsselklauseln wie „Kompetenz erwerben“) genügen.

## Aufgabe 12: Text-Check

Entwickeln Sie einen Text-Checker, der auf Probleme in Texten (Schreibfehler, Grammatikfehler, ggf. Komma-Fehler etc.) hinweist.