

## Anwendungen der KI (SS 2018)

### Aufgabenblatt 1

---

#### Generelle Anmerkungen für's Praktikum:

- Die Aufgaben werden jeweils im Rahmen des nächsten Praktikums besprochen. Ich komme hierfür bei den einzelnen Teams vorbei.
- Bereiten Sie sich auf eine kurze Besprechung zu folgenden Punkten vor: (1) Welche Ergebnisse haben Sie erzielt? Bereiten Sie insbesondere Beispielergebnisse vor, die sie zeigen können. (2) Gibt es Bugs und Unklarheiten im Code / in der Fragestellung? (3) Erscheinen Ihnen Ergebnisse unplausibel?

#### Aufgabe 1.1 (Vorbereitung)

Finden Sie sich zu **Zweier-Teams** zusammen. Wählen Sie einen Team-Namen. Legen Sie sich unter <http://gitlab.cs.hs-rm.de/> ein **Git-Projekt** mit Ihrem Team-Namen an. Machen Sie Prof. Ulges zum Projekt-Mitglied.

Hosten Sie Ihren Code während des gesamten Semesters im Git-Projekt und checken Sie Zwischenergebnisse zeitnah ein.

#### Aufgabe 1.2 (Python-Crashkurs))

Bearbeiten Sie den **Python-Crashkurs** (siehe Homepage). Dieser stellt Python auf ca. 80 Slides vor (und ist an Folien des Ex-Kollegen Prof. Barth aus der MI-Veranstaltung "Programmieren 3" angelehnt). Sie finden 5 Aufgaben (die Slides mit Überschrift "Do it Yourself"). Lösen Sie die Aufgaben.

#### Aufgabe 1.3 (ElasticSearch einrichten)

Arbeiten Sie sich in ElasticSearch (ES) für Textsuche ein. Grundlagen hierfür finden Sie hier:

- **Die ElasticSearch-Referenz:**  
<https://www.elastic.co/guide/en/elasticsearch/reference/current/index.html>
- Wie man ES aus **Python** benutzt:  
<https://elasticsearch-py.readthedocs.io>

#### Richten Sie nun Ihre eigene ElasticSearch-Instanz ein:

- a) Laden Sie elasticsearch herunter:  
<https://www.elastic.co/de/downloads/elasticsearch>

- b) Erstellen Sie sich eine Kopie des config-Files `my_config/elasticsearch.yml`. Tragen Sie die Ordner ein, in denen ES Ihre Daten und Logs speichern soll, z.B.:

```
path.data: /home/ulges/my_data/foo/  
path.logs: /home/ulges/my_logs/bar/
```

- c) Rufen Sie `elasticsearch` auf (Sie sollten einige Ausgaben auf der Konsole sehen):

```
ES_PATH_CONF=my_config/ bin/elasticsearch
```

- d) Testen Sie ob ES verfügbar ist: Der Aufruf `curl localhost:9200` sollte Ihnen eine Antwort mit verschiedenen Daten (u.a. der Tagline “You know, for Search”) liefern.

#### Aufgabe 1.4 (ElasticSearch mit Python)

Stellen Sie aus Python eine Verbindung zu Ihrer ElasticSearch-Instanz her:

- a) Schreiben Sie ein kleines Python-Skript, das (1) fünf Dummy-Dokumente (mit jeweils einem einfachen Titel und einem Satz als Inhalt) indexiert, und (2) mit dem Sie per Text-Query in dem Index suchen können.  
*Hinweise: Mit `indices.create()` erstellen Sie einen neuen Index. Mit `index()` fügen Sie ein Dokument hinzu, mit `helpers.bulk()` mehrere Dokumente auf einmal (deutlich effizienter). Mit `search()` suchen Sie nach einem Query. Verwenden Sie einen simplen `MatchQuery`.*
- b) Prüfen Sie ob Sie die gewünschten Dokumente auch unter Flexion finden. Falls nicht, verwenden Sie einen Analyzer, der ein Stemming der Terme durchführt (so dass Sie z.B. bei der Suche nach `patients` Dokumente mit dem Term `patient` finden). Verifizieren Sie anhand eines kurzen Tests, dass Ihre Änderung erfolgreich war.
- c) (**schwierig**): Wenn Sie beim Aufruf von `search()` den Parameter `explain` verwenden, erhalten Sie eine detaillierte Analyse, wie die Scores zustandekommen. Welches Scoring-Verfahren verwendet ElasticSearch per Default?