**Ähnlichkeit von Job-Posts**

1. Finden von ähnlichen Job-Posts

Stellen Sie sich vor, ein/e Bewerber/in hat eine für ihn / sie interessante Stelle gefunden. Nun  
würde er/sie gerne automatisch ähnliche Stellen vorgeschlagen bekommen, die auch  
interessant sein könnten. Wie könnte eine entsprechende Funktionalität implementiert  
werden?  
  
**a.** Definieren Sie zunächst, wann eine Stelle in diesem Szenario als ähnlich gesehen  
werden sollte.  
**b.** Überlegen Sie dann, mit welchen Ansätzen diese Art von Ähnlichkeit erfasst werden kann

**a)** Wann sind Job-Posts ähnlich? Was macht Job-Posts aus (Eigenschaften)?

Eigenschaften:

* Job Beschreibung
* Job Titel
* Kategorie/Thema
* Ort (Stadt, Land, Unternehmen)
* Plattform
* Berufsart

Arten der Ähnlichkeit:

* Ansprüche
* Fachgebiet
  + Beruf
* Gehalt
* Firma
* Plattform
* Ort
* Berufsart
* Textlänge
* Grammatik

Ähnlichkeiten - wichtig für uns: (Sortieren erst nach 1, dann 2,…)

1. Fachgebiet
2. Spezifische Inhalte
3. Berufsart
4. Ansprüche (Skills)
5. Ort
6. Gehalt
7. (Firma)

**b)**

**Ansatz 1 Word-Similarities**

Finden von Job-Posts mit ähnlichen Themen durch Berechnung von TF-IDF-Vektoren und die Cosinus-Distanz bestimmt die Ähnlichkeit.

**Referenzen:**

* <https://www.kaggle.com/general/202112>

**Ziele:**

* TF-IDF-Vektoren
* Cosinus-Distanz bestimmen
* Dadurch Ranking erstellen

Falls Zeit noch übrig:

**Topicmodelling Erweiterung**

**Idee:**

Verwendung eines Topicmodells

Wie kann man Ähnlichkeit berechnen?

* Spezifische Inhalte
  + Beschreibung -> ähnliche Begriffe = Topicmodelling
  + Ähnlichkeit = Umso mehr Keywords enthalten desto besser! Umso mehr andere Keywords, desto schlechter!

Reverenzen:

* <https://www.kaggle.com/code/thebrownviking20/topic-modelling-with-spacy-and-scikit-learn/notebook>

**Ansatz 2 Word2Vec-Similarity** + Eigene Features / Point-Gain-Proceedings

**Idee:**

Ähnlichkeit durch Word2Vec Ansatz.

Falls noch Zeit vorhanden:

Verwendung eines Punktesystems:

Es werden Punkte für Ähnlichkeit vergeben, dabei existieren verschiedene Kategorien. So gibt es für das selbe Thema am meisten Punkte.

Für die ausgewählten Bereiche, werden die Punkte berechnet und die Ähnlichkeit hochgestuft.

Wie kann man Ähnlichkeit berechnen?

* Spezifische Inhalte
  + Word2Vec (Vortrainiert)
    - <https://www.geeksforgeeks.org/python-word-similarity-using-spacy/>
    - <https://ashutoshtripathi.com/2020/09/04/word2vec-and-semantic-similarity-using-spacy-nlp-spacy-series-part-7/>
* Ort
  + Reale Distanz
    - <https://www.tutorialspoint.com/how-to-get-the-longitude-and-latitude-of-a-city-using-python>
    - <https://www.geeksforgeeks.org/python-calculate-distance-between-two-places-using-geopy/>
* Gehalt
  + Differenz der Gehälter
* Kategorie
  + Falls in selber Kategorie = Boost/extra Punkte
* Job-Title
  + Falls selber Job-Title oder teils etwas darin enthalten = Boost/extra Punkte
* Berufsart
  + Vollzeit-Vollzeit = 0
  + Vollzeit-Halbtags = 1
  + Halbtags-Vollzeit = 2
  + Halbtags-Halbtags = 0
  + Undefined-… = keine Punkte
* Contract = Halbzeits?
* Anzahl an Verben/Wörtern,… (?)

Ziel:

* Word2Vec laden
* Ähnlichkeit durch Gewichte bestimmen
* Wissensbasierte-Empfehlungssystem
  + Andere Features mit einbinden lassen
    - Ähnlichkeit Subjektiv!!!