

# Bewertung

March 4, 2020

## 1 Bewertung für Bachelorarbeit

Autorin: Frau Ingrid Tchilibou

Bewerter: Herr Michael Tebbe

### 1.0.1 1.1 Datensatz von Annotation laden

```
[ ]: import json
import import_ipynb
import extract_hyponyms as EH
# read file
if __name__ == '__main__':

    #data loaded
    with open('export200.json', 'r') as myfile:
        data = myfile.read()
    ideen_list = json.loads(data)
```

### 1.0.2 1.2 Ideen Wörterbuch (key = (nummerId , Id) , Value = Content)

Alle 200 Ideen werden in der ideenDict mit entsprechen nummerID und Id gespeichert

```
[ ]: ideenDict = EH.get_dictionaryIdee(ideen_list)
```

### 1.0.3 1.3 DictionaryTree: Beziehungen zwischen Hypernym

Key = ein Hypernym(Eltern Knoten)

Value = List von Hypernyme (Kinder Knoten die als Direkthypernym der Eltern Knoten haben)

i) Verfügbare Funktionen

a) EH.build\_tree(dictionaryTree) Zeigt der gesamte Baum

b) Ein Spezifisches Knoten mit Kinder als unterbaum anzeigen lassen  
EH.build\_part\_of\_tree(Tree,knoten) wobei knoten den form synset.pos.id hat  
Beispiel: EH.build\_part\_of\_tree(dictionaryTree, "entity.n.01")

```
[ ]: synsetsList      = EH.get_synsetsList(ideen_list)    #key = Synset, Value = [list
      ↳ von ID]
      dictionaryTree = EH.analyse(synsetsList)
```

#### 1.0.4 1.4 SynsetsList, hypernymList, ideen\_synset, ideen\_hypernym Wörterbücher

a) Verfügbare Funktionen:

- i) EH.plot\_balken:hypernym(hypernymList,prozent): Balkendiagramm Darstellung: Anzahl von Ideen pro Hypernym Hypernym mit Anzahl an Ideen weniger als "prozent" werden in Others gruppiert. Und "Others" wird als eine geordnete Liste von der kleinsten bis zur größten angezeigt. Beispiel: EH.plot\_balken\_hypersyn(hypernymList,20)
- ii) EH.plot\_hypersyn\_fenster(hypernymList,minimum,maximum) Hypernym mit Anzahl an Ideen Zwischen minimum and maximum Tabelle mit Anzahl an Ideen, Überschneidung und Anzahl an Hypernym Beispiel: EH.plot\_hypersyn\_fenster(hypernymList,26,40)
- iii) EH.plot\_mapping\_idee\_Synset(ideen\_synset,prozent) Balkendiagramm Darstellung: Anzahl von Synset pro Idee IdeenId mit Anzahl an Synsets weniger als "prozent" werden in Others gruppiert. Und "Others" wird als eine geordnete Liste von der kleinsten bis zur größten angezeigt.
- iv) EH.plot\_mapping\_idee\_Hypersyn(ideen\_hypersyn,prozent) Balkendiagramm Darstellung: Anzahl an Hypernym pro Idee IdeenId mit Anzahl an Hypernym weniger als "prozent" werden in Others gruppiert. Und "Others" wird als eine geordnete Liste von der kleinsten bis zur größten angezeigt.

```
[ ]: hypernymList      = EH.get_hypersynDict(synsetsList) #Key = hypersyn ,      Value
      ↳ = [list von ID]
ideen_synset          = EH.mapping_idee_to_synsets(ideen_list) #key = (nummerId,Id)
      ↳ Value = [Listsynsets]
ideen_hypersyn        = EH.mapping_idee_to_hypersyn(ideen_synset) # key =
      ↳ (nummerId,Id) Value = [Listhypersynen]
print("Anzahl Synsets", len(synsetsList.keys()))
print("Anzahl Hypersyn", len(hypernymList.keys()))
```

#### Frage 1: Welche Themenbereiche gibt es in dem Datensatz ?

```
[ ]: EH.plot_hypersyn_fenster(hypernymList,5,7)
```

59 ideas: application, technology, profession -> technology can be used to do a job rest maybe: tech can be used at home, not on the job (e.g. sports)

```
[ ]:
```

```
[ ]: #Antwort:
      #print("Synsets", synsetsList.keys())
```

```
#print("Hypernyms", hypernymList.keys())
EH.plot_balken_hypernym(hypernymList,10)
```

EH.plot\_balken\_hypernym(hypernymList,5) Topics: bad person invesigation building container mammal.... 11 - 16 occurences

EH.plot\_balken\_hypernym(hypernymList,10) Topics: residence home ... - technology

## Frage 2: Finde unerwartete Ideen bzw. unerwartete Themenbereiche oder Outlier (5 Stück)

```
[ ]: #Antwort:
      Livestock, Student
```

### 1.0.5 1.5 Daten Reduktion: Pro Ideen genau 1 Synset

Synset die am Wenigstens in Datensatz vorkommt bzw. genau ein mal!

a) verfügbare Funktionen:

- i) EH.plot\_occurence\_synset(synsets\_occurence,prozent)  
Balkendiagramm Darstellung: Pro Synset wird die Anzahl an wiederholung in der Datensatz angezeigt  
Synsets mit Anzahl an Occurence weniger als "prozent" werden in Others gruppiert.  
Und "Others" wird als eine geordnete Liste von der kleinsten bis zur größten angezeigt.

ii) Alle in 1.3 genannten Tree Funktionen

```
[ ]: synsets_occurence      = EH.occurence_of_synset(ideen_synset) #key = Synset, Value = Anzahl von vorkommen
      ideen_synsetList_clearly = EH.ideen_synsetDict_clearly(ideen_synset, synsets_occurence) #key = (nummerId,Id), Value = [ein Synset]
      synsetList_clearly      = EH.synsetDict_clearly(ideen_synsetList_clearly, synsetsList) #key = Synset, Value = [list von ID]
      dictionaryTree_clearly  = EH.analyse(synsetList_clearly)
```

## Frage 3: Mit dem neuen Daten was sind Die Themenbereiche?

```
[ ]: #Antwort

      hypernymList_clearly      = EH.get_hypernymDict(synsetList_clearly)
      EH.plot_balken_hypernym(hypernymList_clearly,1)

      #filter through hypernyms and choose the ones that might be topics
```

### 1.0.6 1.6 Abstraktionsgrad eine Idee

Verfügbare Funktionen:

- i) EH.Top\_Abstract\_and\_concret(Abstraktionsgrad,ideenDict,top)  
gibt ein Tupel mit top Abstrakte und Konkrete Ideen

```
beispiel: listTop2_Abstrakt,listTop2_KonKret = EH.Top_Abstrakt_and_concret(Abstractionsgrad,id
```

```
[ ]: Abstractionsgrad = EH.Abstactionsgradfunc(ideen_synset)  #key = (numId,Id)
    ↳value = distance
```

**Frage 4: Was sind die abstraktesten Ideen in dem Datensatz? Was sind die konkretesten Ideen in dem Datensatz? (5 Stück)**

```
[ ]: #Antwort
listTop5_Abstrakt,listTop5_KonKret = EH.
    ↳Top_Abstrakt_and_concret(Abstactionsgrad,ideenDict,5)

print('abstrakt:', listTop5_Abstrakt)
print()
print('konkret: ', listTop5_KonKret)
```

**Frage 5: Welche wichtigen Informationen über die Daten können wir mit Hilfe der bereitgestellten Wörterbücher noch erhalten?**

## 2 Antwort

find rare ideas by looking at overlaps between synsets. Hypothesis: rare ideas produce 'strange' overlaps

synsets per idea k tree of one idea

compare two trees

use corpus-based methods to filter out uninformative topics