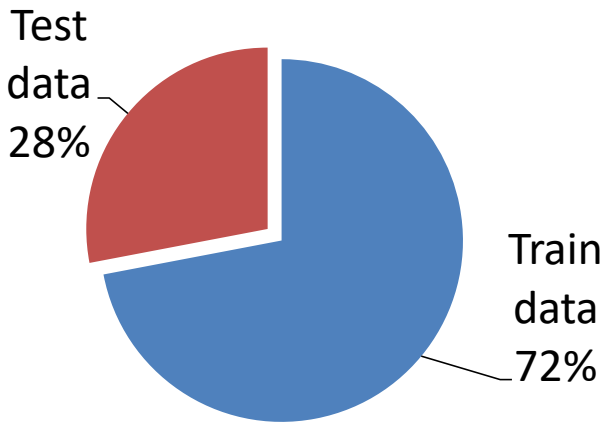
# Sprint Review
# 12.08.19-04.09.19

erstellt von Oleg Nekhayenko

https://github.com/nekhayen/Assignment

# Aufgabenbereich 1a- Supervised Learning

- **Input Data**: Reuters-21578 Evaluationskorpus für Dokumentenklassifikation (Mod-Apte split)
- 10788 Newsartikel in 90 Newskategorien




- Jedem Newsartikel von 1 bis 15 Kategorien (label) zugewiesen -> **Multilabel Classification**
- **Aufgabe:** Newskategorien aus dem Test data vorherzusagen

# Aufgabenbereich 1a,b- Supervised Learning

**Klassifikatoren:**

- **K-Nearest-Neighbor (KNN)** und **Support Vector Classifier (SVC)**

- automatisierte Suche nach den best passenden Parametern und Cross Validation (CVGridSearch)

**Ergebnisse**:

| Klassifikator | Subset accuracy über alle Kategorien |
|---|---|
| KNN | 0.4442 |
| SVC | 0.8182 |

# Aufgabenbereich 1a, b- Supervised Learning

Macro average quality numbers*

| Klassifikator | Recall | Precision | F1 |
|---|---|---|---|
| KNN | 0.2881 | 0.5801 | 0.3552 |
| SVC | 0.4282 | 0.6714 | 0.4980 |

SVC F1 14% ⬆

Micro average quality numbers**

| Klassifikator | Recall | Precision | F1 |
|---|---|---|---|
| KNN | 0.4367 | 0.4962 | 0.4646 |
| SVC | 0.8128 | 0.9395 | 0.8715 |

SVC F1 41% ⬆

# Aufgabenbereich 1a, b- Supervised Learning

- CVGridsearch optimale Parameter(CV=3):

| KNN CVGridsearch | Beste Parameter |
|---|---|
| neighbors': [1, 3, 5] | 1 |
| metric':['euclidean', 'manhattan'] | euclidean |
| weights':['uniform','distance']} | uniform |
| **SVC CVGridsearch** | |
| C': [1, 10, 100, 1000] | 10 |
| gamma': [10, 1, 0.1, 0.01, 0.001, 0.0001] | 0.1 |
| kernel': ['rbf'] | rbf |

# Aufgabenbereich 1a, b- Supervised Learning

Möglichkeiten zur Verbesserung der Text-Klassifizierung:

1. N-gram statt Wort beim Tfidf Vectorizer -> Accuracy steigen
2. CVRandomized search nach Parametern
3. Lemmatization statt Stemming
4. Vollständigere Stoppwortliste
5. Andere Klassifikatoren einsetzen
6. Nutzung von default Tfidf Vectorizer preprocessing statt eigene Funktion

# Aufgabenbereich 2- Unsupervised Learning

- **Input Data**: Reuters-21578 Evaluationskorpus für Dokumentenklassifikation (Mod-Apte split)
- **Aufgabe:** optimale Anzahl an Cluster-Themenbereichen im Korpus bestimmen
- **Annahme:** von 4 bis 20 Themenbereiche im Reuters Korpus
- Überprüfung der Annahme durch eigene Implementierung von Clustering-Algorithmus
- Keine genaue Festlegung von Anzahl der Themenbereiche aufgrund von „noisy text data“

# Aufgabenbereich 2- Unsupervised Learning

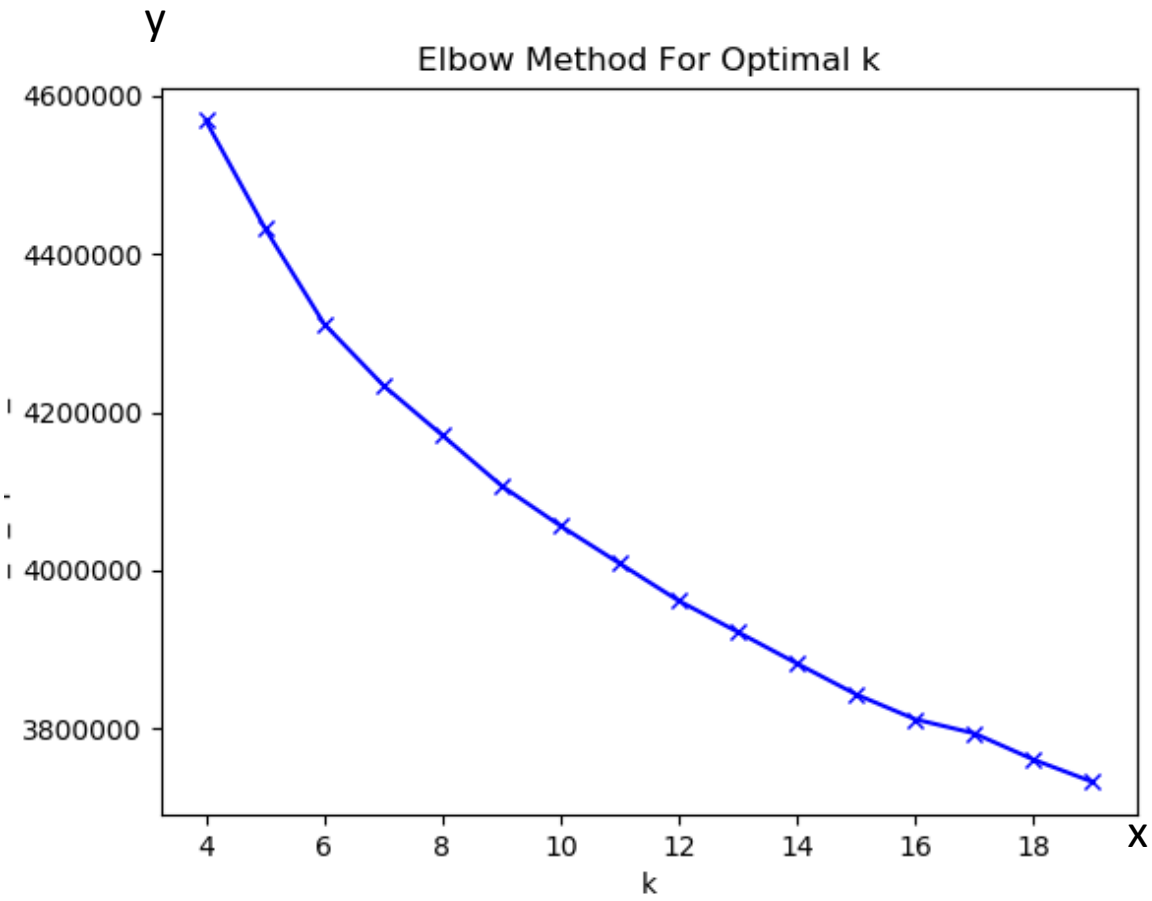
Implementierung von Clustering-Algorithmus:

- Einlesen von Reuters raw text Dataset
- Data preprocessing
  - Erstellen von List of Strings. Satz=String
  - Entfernen von leeren Strings und \n
- Initialisierung von „BERT-as-service“ client
  - Auswahl von pretrained Bert Model uncased_L-12_H-768_A-12 und Parameter
  - Übergabe von preprocessed text an Bert as service

Initialisierung von K-Means Clustering mit mit n_clusters in range von (4,20)

Festlegung vom optimalen K nach dem Elbow Method

# Aufgabenbereich 2- Unsupervised Learning




X- Anzahl der Cluster
Y-Summe der quadrierten Abstände der Samples zu ihrem nächstgelegenen Clusterzentrum (Inertia)

# Aufgabenbereich 2- Unsupervised Learning

Möglichkeiten zur Verbesserung der Implementierung:

1. Data Preprocessing
2. Fine tuning der Parameter von Bert Client
3. Andere Methoden zur Festlegung von optimalen K
4. Anderer Cluster Algorithmus