Job title classification

Rahkakavee Baskaran

Department of Economics

December 16, 2021

Universität Konstanz

Research Task

Examine the classification of German job titles into the classes of the KLDB 2010 taxonomy. The analysis is based on three perspectives:

1) vectorization techniques
2) classifiers
3) including of additional knowledge

Table of Contents

Related Work – Challenges

Domain-specific
some work for the English-speaking job market

traditional (Zhu et al 2017) vs. Deep Learning Methods (Neculoiu et al 2016)

framing as classification task (Wang et al. 2019) vs. framing as a string representation approach of similar job titles (Decorte et al. 2021)

Multiclass
decomposition into multiple binary problems vs. naturally handling

no clear answer so far

different approaches like SVM(Guo and Wang 2015) or Convolutional Neural Networks (Farooq etc. al 2017)

Short texts

Related Work – Challenges

Domain-specific

some work for the English-speaking job market

traditional (Zhu et al 2017) vs. Deep Learning Methods (Neculoiu et al 2016)

framing as classification task (Wang et al. 2019) vs. framing as a string representation approach of similar job titles (Decorte et al. 2021)

Multiclass

decomposition into multiple binary problems vs. naturally handling

no clear answer so far

different approaches like SVM(Guo and Wang 2015) or Convolutional Neural Networks (Farooq etc. al 2017)

short text

Related Work – Short Text Classification

- job titles have often not more than 50 characters

→sparseness

→few word co-occurrence

→missing shared context

→noisiness

→ambiguity

Related Work – Short Text Classification

- Approaches
  - criticism of “Bag of words” context
  - representation of documents as an extraction of semantic relationships
  - **dense** (Wang et al. 2017) vs. **sparse vectors** (Chen et al. 2019)
    - sparse: tf-idf and count vectorizer
    - dense: word2vec, doc2vec etc.
  - no consensus for classifiers

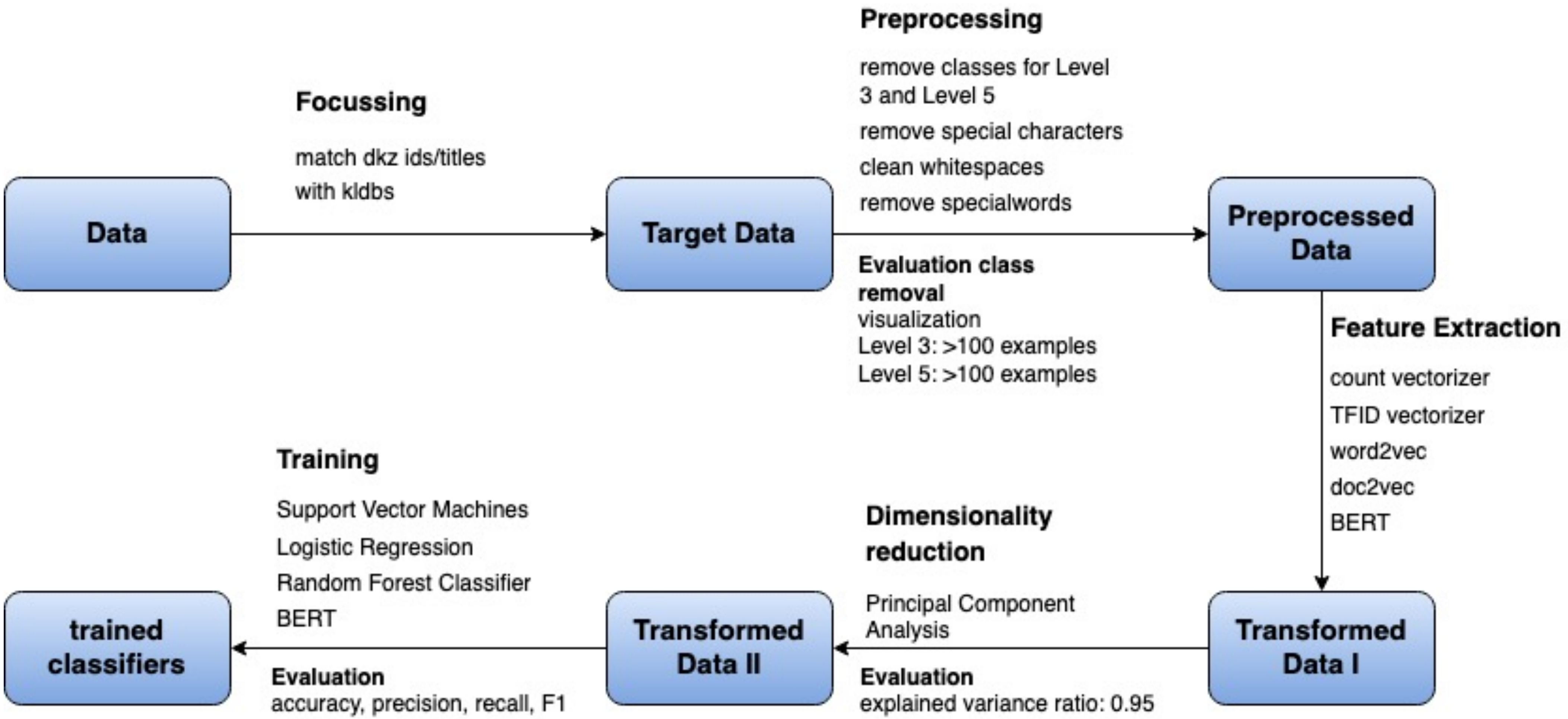
Research Gap

- no classification attempts for the German job market
- but would facilitate several downstream tasks:

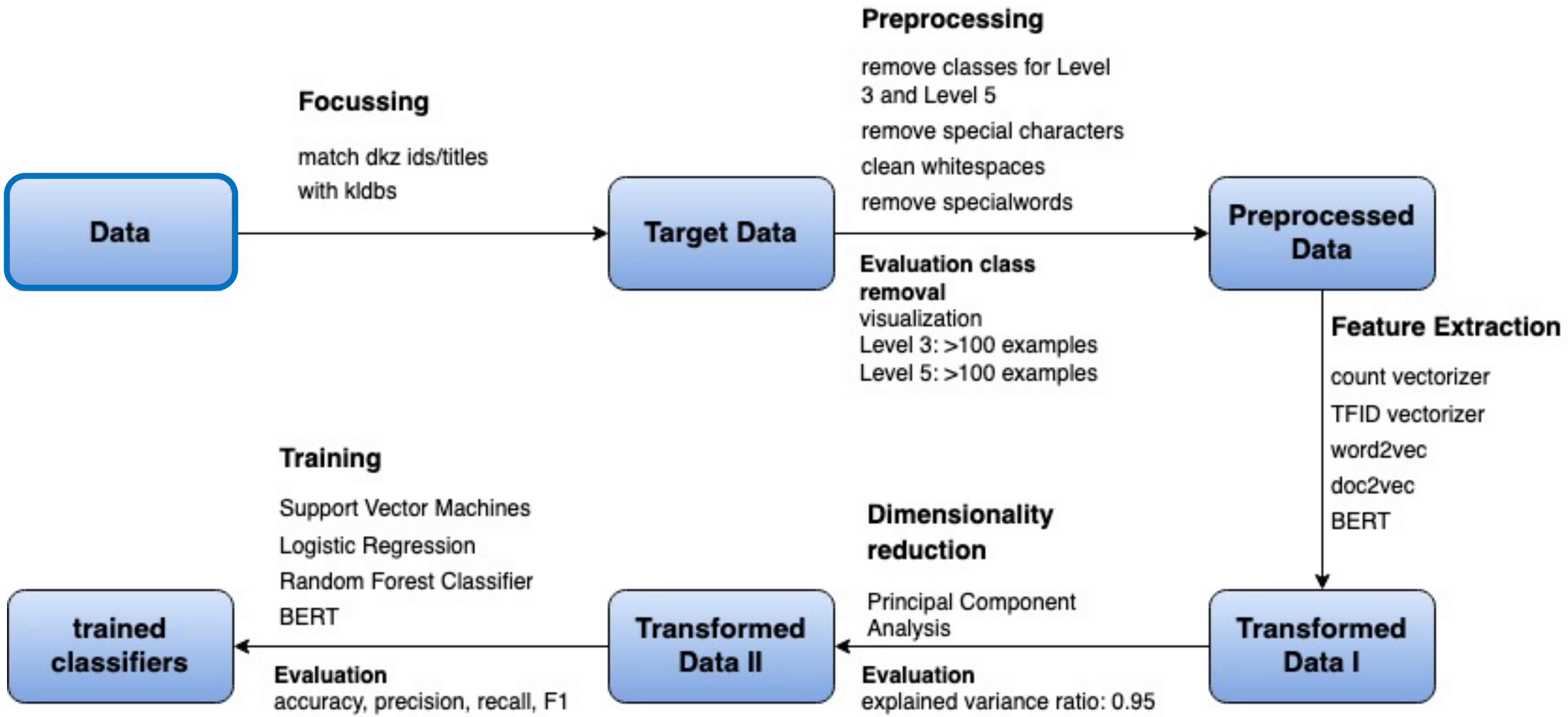
job market analyses | **improvement of job search engines** | **improvement of job recommendation systems**

- solid database

Method


Data
Focussing
match dkz ids/titles
with kldbs
Target Data
Preprocessing
remove classes for Level
3 and Level 5
remove special characters
clean whitespaces
remove specialwords
Evaluation class
removal
visualization
Level 3: >100 examples
Level 5: >100 examples
Preprocessed
Data
Feature Extraction
count vectorizer
TFID vectorizer
word2vec
doc2vec
BERT
Transformed
Data I
Dimensionality
reduction
Principal Component
Analysis
Evaluation
explained variance ratio: 0.95
Transformed
Data II
Training
Support Vector Machines
Logistic Regression
Random Forest Classifier
BERT
Evaluation
accuracy, precision, recall, F1
trained
classifiers

Method


Focussing
match dkz ids/titles
with kldbs
Data
Target Data
Preprocessing
remove classes for Level
3 and Level 5
remove special characters
clean whitespaces
remove specialwords
Evaluation class
removal
visualization
Level 3: >100 examples
Level 5: >100 examples
Preprocessed
Data
Feature Extraction
count vectorizer
TFID vectorizer
word2vec
doc2vec
BERT
Transformed
Data I
Dimensionality
reduction
Principal Component
Analysis
Evaluation
explained variance ratio: 0.95
Transformed
Data II
Training
Support Vector Machines
Logistic Regression
Random Forest Classifier
BERT
Evaluation
accuracy, precision, recall, F1
trained
classifiers

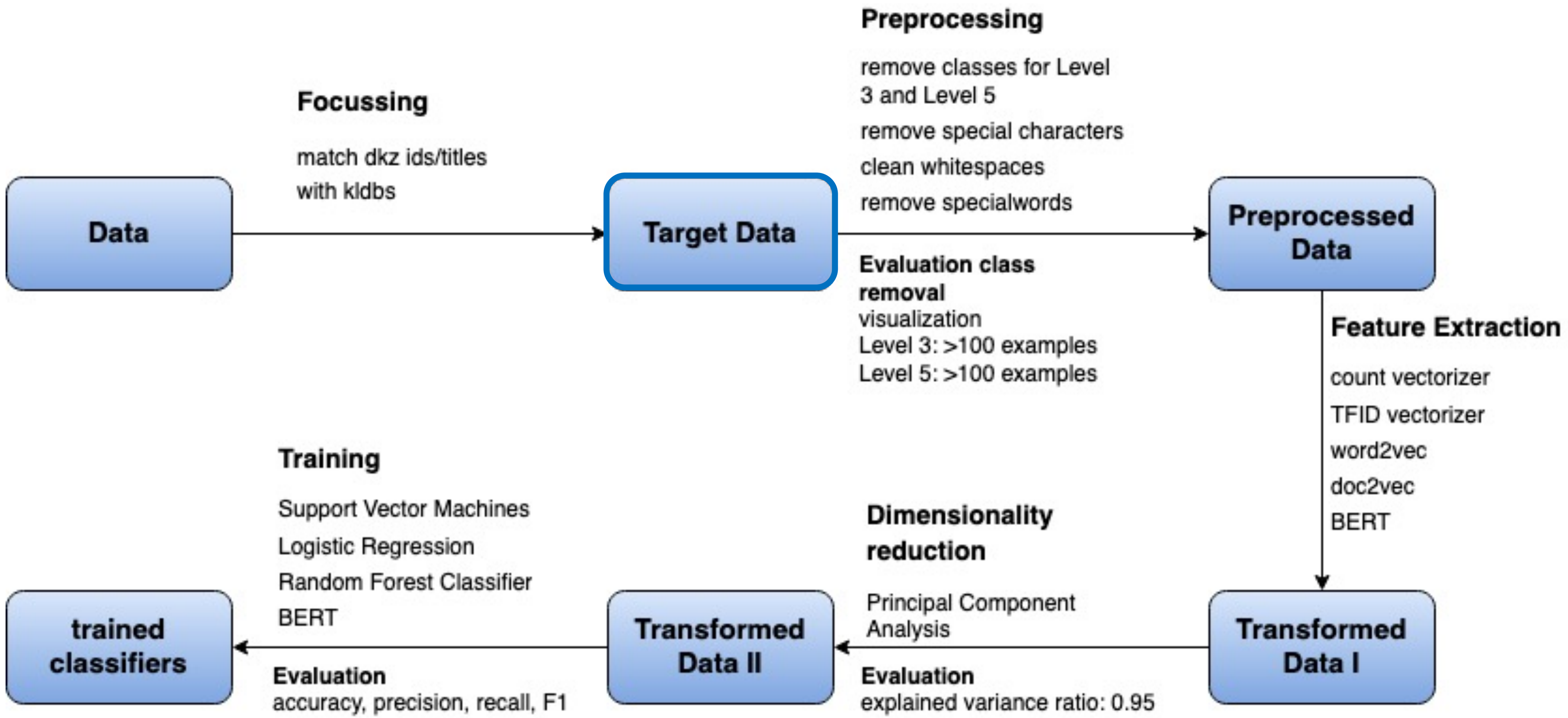
Data

- data from the job portal of the ”Bundesagentur für Arbeit” (BA):
  - job title
  - “Dokumentationskennziffer” (Dkz)
- KldB Taxonomy 2010

Name	Level	Number of classes	Example
Berufsbereiche	1	10	4: Naturwissenschaft, Geografie und Informatik
Berufshauptgruppen	2	37	43: Informatik-, Informations- und Kommunikationstechnologieberufe
Berufsgruppen	3	144	434: Sofwareentwicklung
Berufsuntergruppen	4	700	4341: Berufe in der Sofwareentwicklung
Berufsgattungen	5	1286	43412: Berufe in der Sofwareenetwicklung - fachlich ausgerichtete Tätigkeiten

Overview of KldB (edited after (Bundesagentur für Arbeit, 2011)

Method

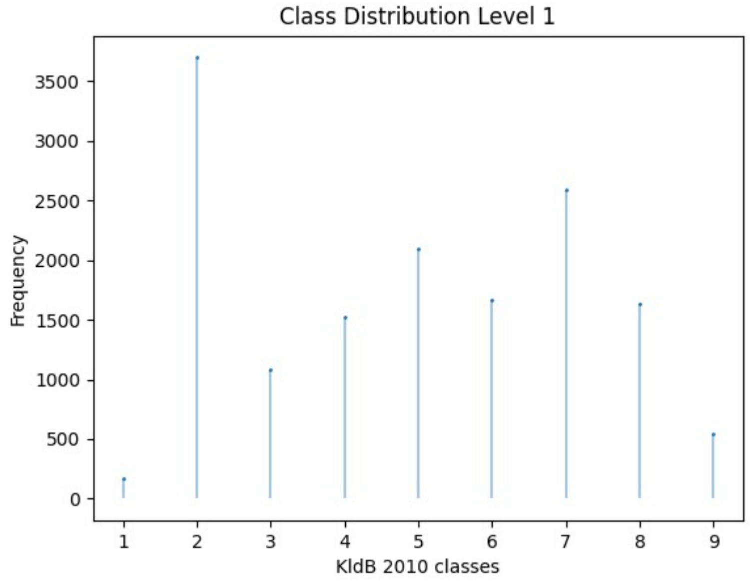

Data
Focussing
match dkz ids/titles
with kldbs
Target Data
Preprocessing
remove classes for Level
3 and Level 5
remove special characters
clean whitespaces
remove specialwords
Evaluation class
removal
visualization
Level 3: >100 examples
Level 5: >100 examples
Preprocessed
Data
Feature Extraction
count vectorizer
TFID vectorizer
word2vec
doc2vec
BERT
Transformed
Data I
Dimensionality
reduction
Principal Component
Analysis
Evaluation
explained variance ratio: 0.95
Transformed
Data II
Training
Support Vector Machines
Logistic Regression
Random Forest Classifier
BERT
Evaluation
accuracy, precision, recall, F1
trained
classifiers

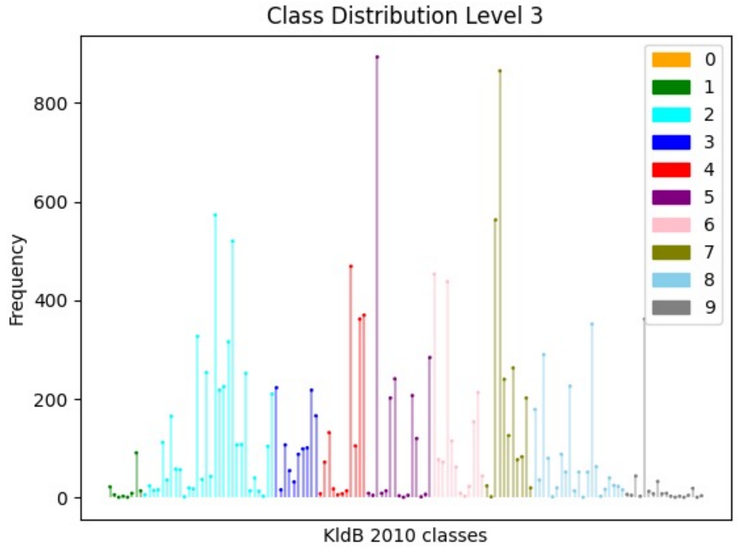
Target Data

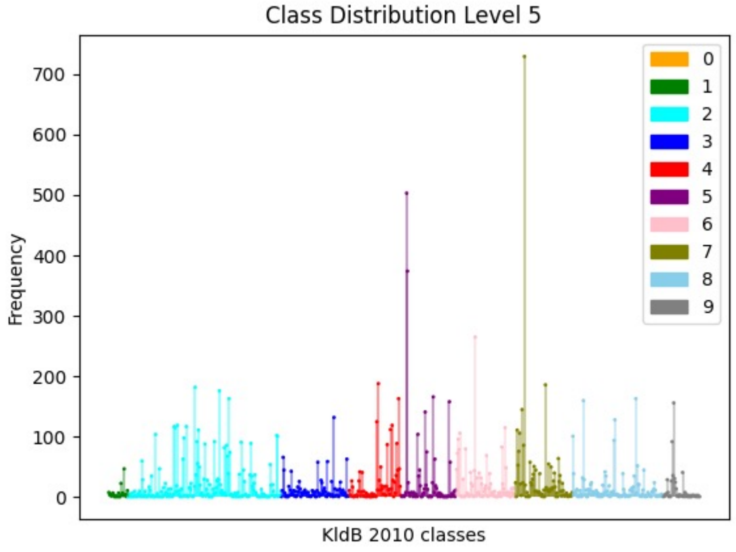
- the ”Dkz” id from the “BA” can be matched with the KldB class id
- trainings data example (Level 5)

```
[{'id': '82312', 'title': 'Friseur oder Friseurmeister (m/w/d)'},
 {'id': '81713', 'title': 'Köln - Kinder-Physiotherapeut (m/w/d) TZ'},
 {'id': '73203',
  'title': 'Sachbearbeiterin / Sachbearbeiter (m/w/d) Qualitätssicherung'}]
```

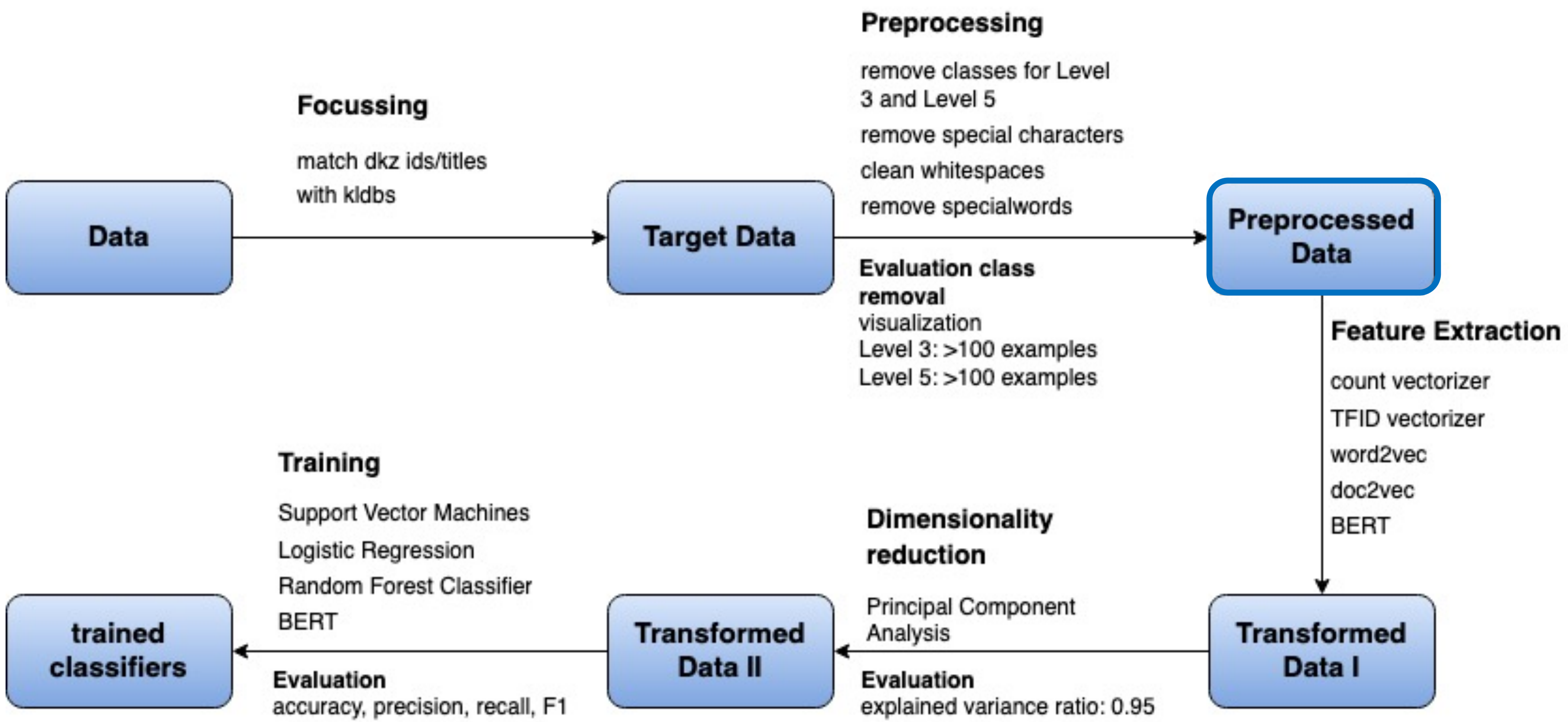
Target Data – Class Distribution







Method

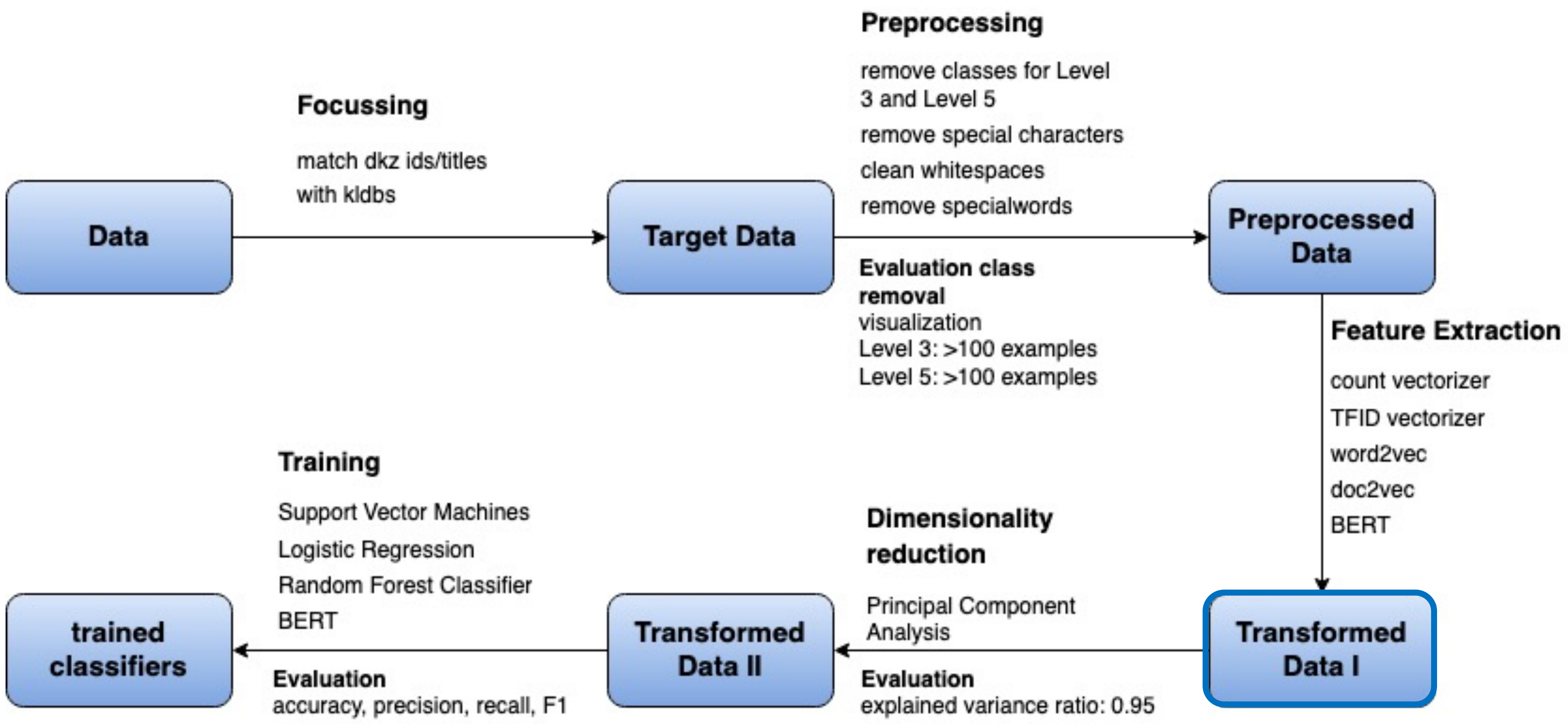

Data
Focussing
match dkz ids/titles
with kldbs
Target Data
Preprocessing
remove classes for Level
3 and Level 5
remove special characters
clean whitespaces
remove specialwords
Evaluation class
removal
visualization
Level 3: >100 examples
Level 5: >100 examples
Preprocessed
Data
Feature Extraction
count vectorizer
TFID vectorizer
word2vec
doc2vec
BERT
Transformed
Data I
Dimensionality
reduction
Principal Component
Analysis
Evaluation
explained variance ratio: 0.95
Transformed
Data II
Training
Support Vector Machines
Logistic Regression
Random Forest Classifier
BERT
Evaluation
accuracy, precision, recall, F1
trained
classifiers

Preprocessed Data

- remove stopwords, special characters, lowercase
- Zipf’s law: most frequent words → get a list of special words → remove from title
- remove classes for Level 3 and Level 5 (Threshold 100)
- Input data example

```
[{'id': '82312', 'title': 'friseur friseurmeister'},
 {'id': '81713', 'title': 'köln kinder physiotherapeut tz'},
 {'id': '73203',
  'title': 'sachbearbeiterin sachbearbeiter qualitätssicherung'}]
```

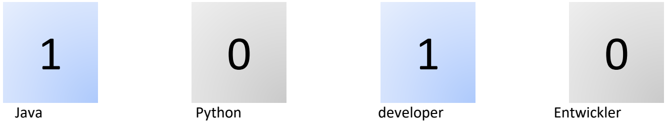
Method

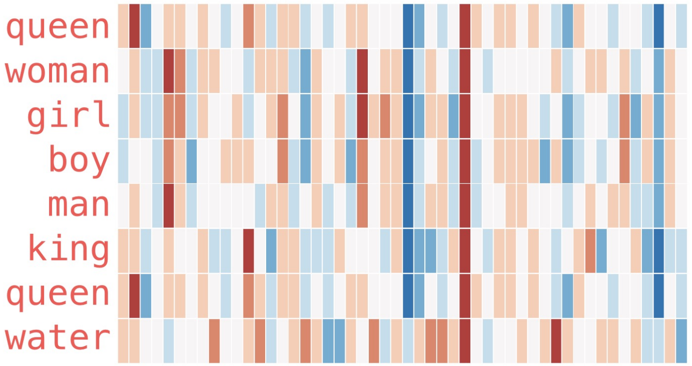

Focussing
match dkz ids/titles
with kldbs
Preprocessing
remove classes for Level
3 and Level 5
remove special characters
clean whitespaces
remove specialwords
Data
Target Data
Preprocessed
Data
Evaluation class
removal
visualization
Level 3: >100 examples
Level 5: >100 examples
Feature Extraction
count vectorizer
TFID vectorizer
word2vec
doc2vec
BERT
Training
Support Vector Machines
Logistic Regression
Random Forest Classifier
BERT
Dimensionality
reduction
Principal Component
Analysis
trained
classifiers
Transformed
Data II
Transformed
Data I
Evaluation
accuracy, precision, recall, F1
Evaluation
explained variance ratio: 0.95

Transformed Data I

- sparse methods: count vectorizer and tf-idf vectorizer.
- dense methods
  - word2vec:
    - pretrained model from google with fine tuning
    - with and without additional information
  - doc2vec
    - with and without additional information (epochs=10, vector size=300, window=5)
  - BERT pretrained model
  - BERT with fine tuning and one classification layer

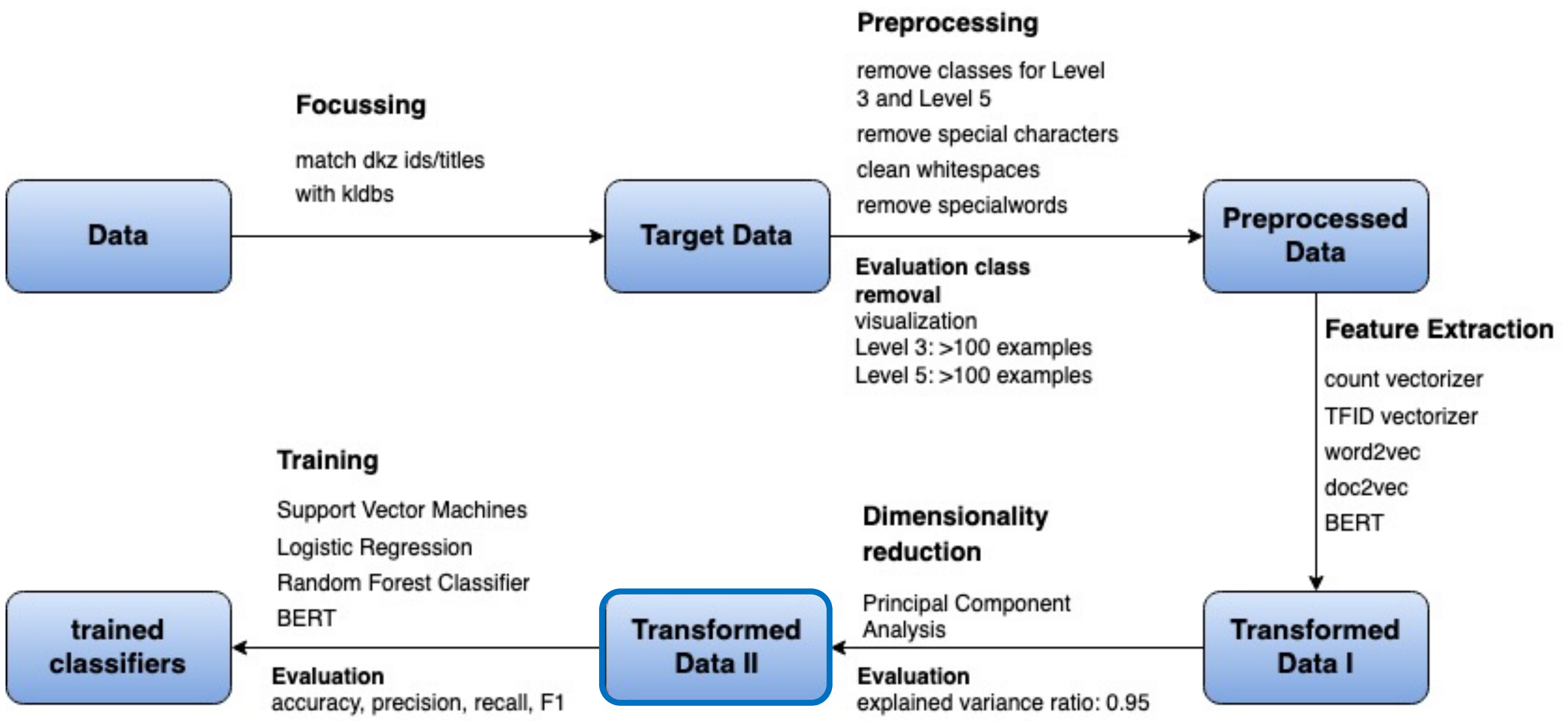
Example 1: „Java developer“





Source: Alammar, J. The illustrated Word2vec. 2019 <https://jalammar.github.io/illustrated-word2vec/> accessed: 09.12.2021

Method



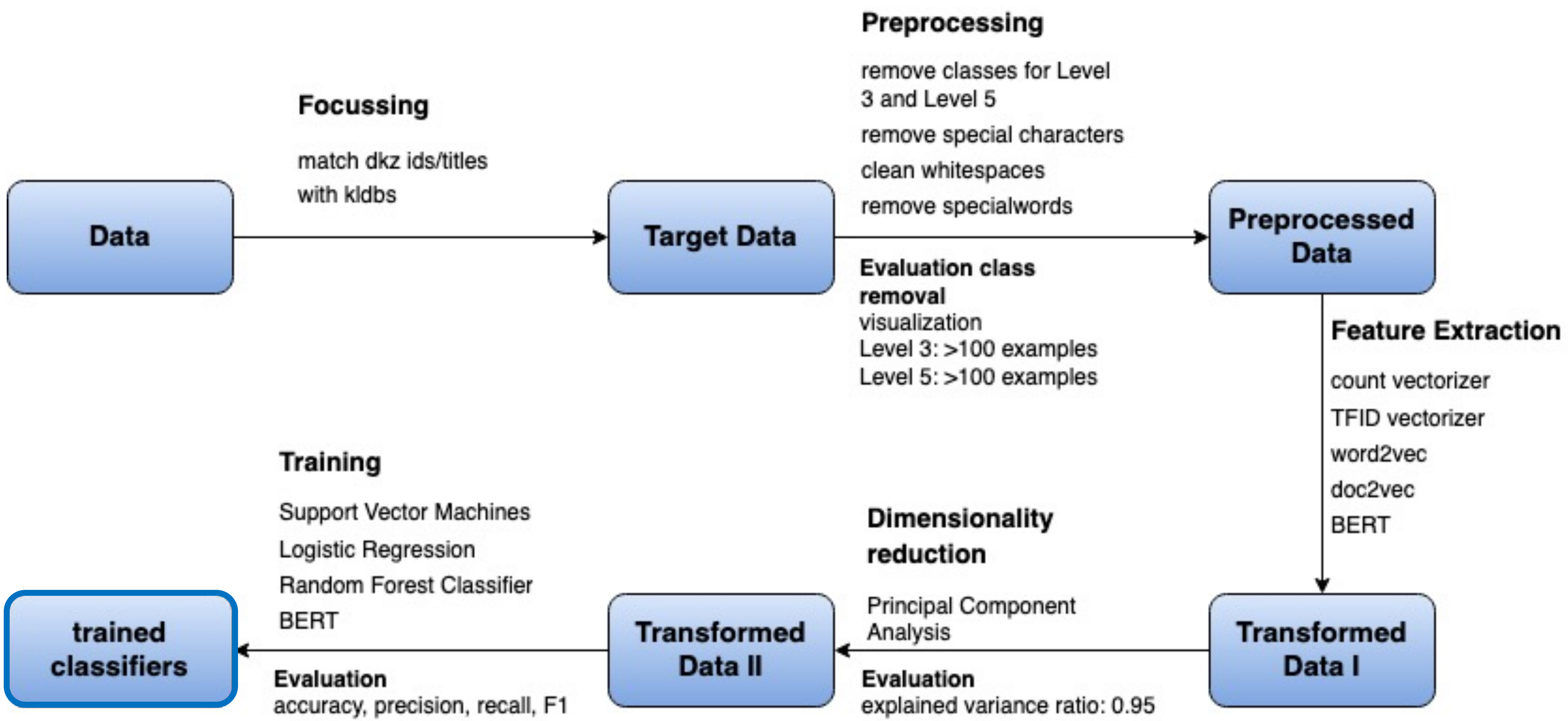
Data
Focussing
match dkz ids/titles
with kldbs
Target Data
Preprocessing
remove classes for Level
3 and Level 5
remove special characters
clean whitespaces
remove specialwords
Evaluation class
removal
visualization
Level 3: >100 examples
Level 5: >100 examples
Preprocessed
Data
Feature Extraction
count vectorizer
TFID vectorizer
word2vec
doc2vec
BERT
Transformed
Data I
Dimensionality
reduction
Principal Component
Analysis
Evaluation
explained variance ratio: 0.95
Transformed
Data II
Training
Support Vector Machines
Logistic Regression
Random Forest Classifier
BERT
Evaluation
accuracy, precision, recall, F1
trained
classifiers

Transformed Data II

Principal component Analysis (PCA)

- reduces features to n_components to speed up the classifiers
- choose n_components such that the explained variance ratio is approx. 0.95

Results


Preprocessing
remove classes for Level 3 and Level 5
remove special characters
clean whitespaces
remove specialwords
Focussing
match dkz ids/titles with kldbs
Data
Target Data
Preprocessed Data
Evaluation class removal
visualization
Level 3: >100 examples
Level 5: >100 examples
Feature Extraction
count vectorizer
TFID vectorizer
word2vec
doc2vec
BERT
Training
Support Vector Machines
Logistic Regression
Random Forest Classifier
BERT
Dimensionality reduction
Principal Component Analysis
trained classifiers
Transformed Data II
Transformed Data I
Evaluation
accuracy, precision, recall, F1
Evaluation
explained variance ratio: 0.95

Evaluation metrics

- Evaluation measurements

Accuracy: How well does the classifier classifies across all classes?

→ Problem: favours majority classes

in addition

Precision: ability of model to return only relevant cases

Recall: ability of model identify all relevant instances

F-Score: harmonic mean between precision and recall

First Results – Level 1

Accuracy

	LR	SVM	RF
CountVectorizer	0.71	0.68	0.64
TFIDF	0.71	0.69	0.64
Word2Vec	0.55	0.53	0.62
Doc2Vec	0.48	0.46	0.56
BERT	0.65	0.65	0.58

Precision (p), Recall (r), F1 - Macro

	LR	SVM	RF
CountVectorizer	p: 0.74, r: 0.60, F1: 0.64	p: 0.73, r: 0.56, F1: 0.60	p: 0.67, r: 0.54, F1: 0.57
TFIDF	p: 0.75, r: 0.60, F1: 0.63	p: 0.74, r: 0.57, F1: 0.62	p: 0.65, r: 0.53, F1: 0.55
Word2Vec	p: 0.52, r: 0.40, F1: 0.42	p: 0.46, r: 0.41, F1: 0.41	p: 0.62, r: 0.54, F1: 0.56
Doc2Vec	p: 0.43, r: 0.34, F1: 0.35	p: 0.39, r: 0.33, F1: 0.33	p: 0.60, r: 0.41, F1: 0.43
BERT	p: 0.67, r: 0.57, F1: 0.60	p: 0.63, r: 0.56, F1: 0.58	p: 0.70, r: 0.46, F1: 0.50

	Accuracy	Precision	Recall	F1
BERT CLF Level 1	0.76	0.73	0.70	0.71
BERT CLF Level 3	0.53	0.56	0.46	0.46
BERT CLF Level 5	0.60	0.59	0.54	0.53

Notes: Short training data set with 15000 examples (for Level 3 and 5 results see Appendix A)

1. Vectorization Techniques

good performance

- differences depends on classifier
- for example, besides Doc2vec, all vectorizations performs quite similar for RF

- Word2vec and Doc2vec dense techniques have poor performance for LR and SVM poor

2. Classifiers

- LR and SVM perform better than RF
- there is no huge difference between LR and SVM

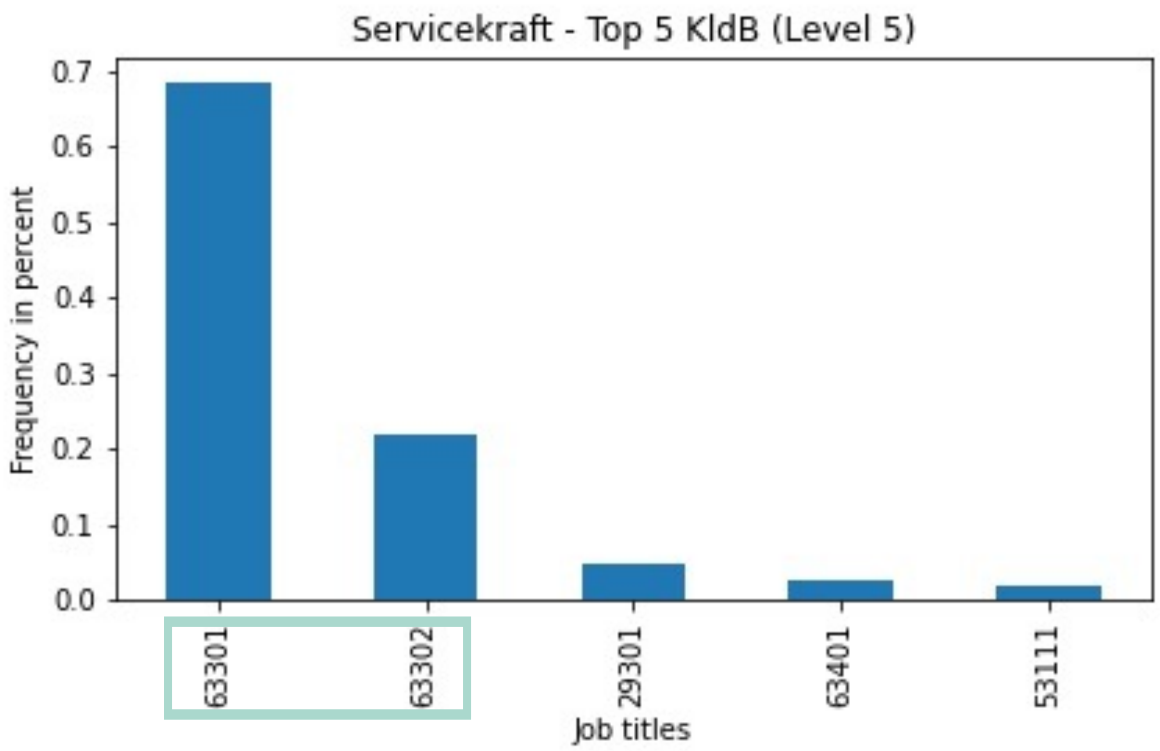
3. additional information

Discussion/Limitation

Occupation: „Servicekraft“

<u>Ambiguity between level of requirement</u>

- 6330**1**: „Berufe im Gastronomieservice (ohne Spezialisierung) – **Helfer/Anlerntätigkeiten“**
- 6330**2**: „Berufe im Gastronomieservice (ohne Spezialisierung) - **fachlich ausgerichtete Tätigkeiten“**



Notes: Further example Appendix B

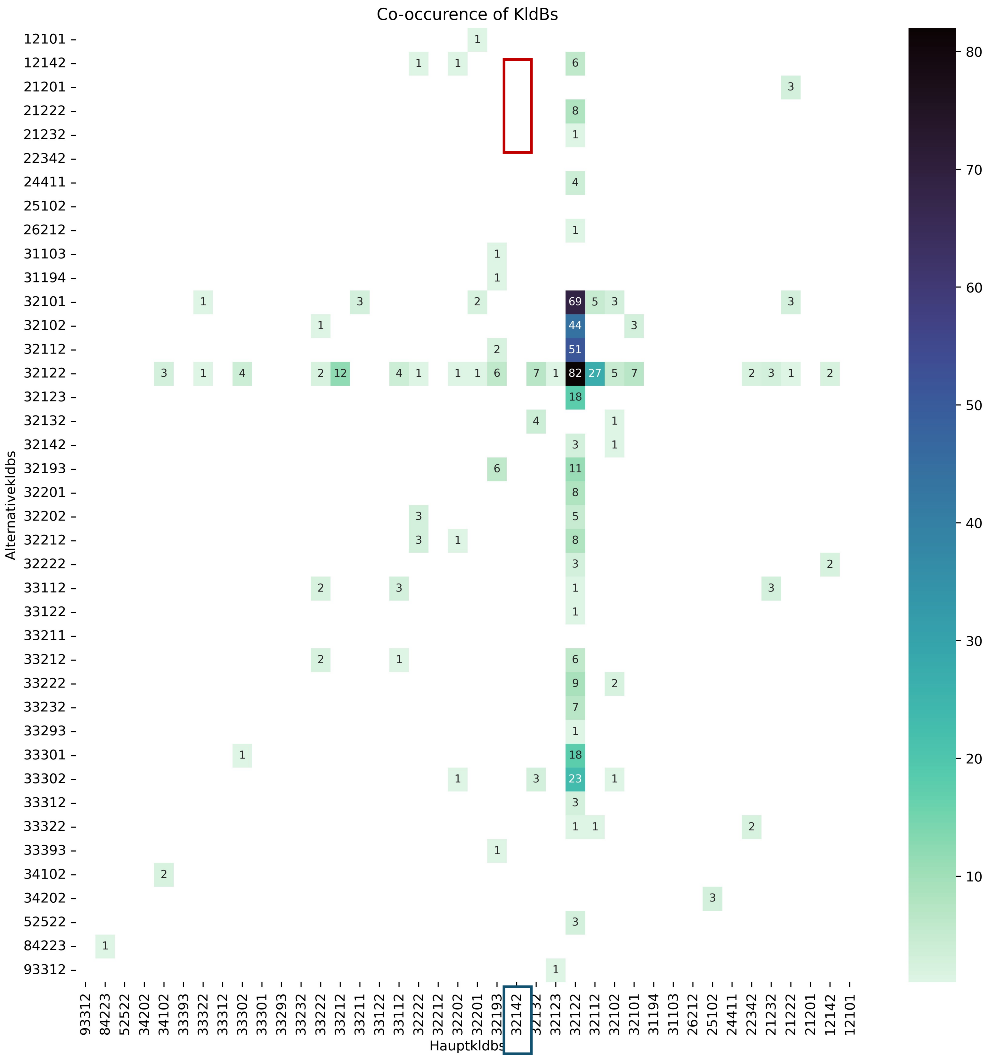
Discussion/Limitation

Occupation: „Maurer“

32122 has many alternatives
→ Differentiation is difficult

32122 often as an alternative

„Berufsuntergruppen“ (4th digit) difficult to differentiate (321**2**2, 321**1**2, 321**0**2, 321**0**1)



Discussion/Limitation

I. Pre-processing

How reasonable is it to exclude classes of level 3 and 5?

- Advantage
  - efficient in terms of time complexity
  - evaluation metrics more meaningful
- Disadvantage:
  - level 3: 46 classes remain
  - level 5: 38 classes remain

II. Evaluation

- There is also the possibility to evaluate the the classes with k-fold cross validation.
  - parameter k: number of groups that a given data sample is to be split
  - currently: one split into training and test data
  - k-fold cross validation:
    - split into k groups, one is test data the remaining groups training
    - fit model and evaluate
    - repeat with each group as test data
    - take the mean of all them and report also the confidence interval (possible with all the evaluation metrics)

Literature

Decorte, J. J., Van Hautte, J., Demeester, T., & Develder, C. (2021). JobBERT: Understanding Job Titles through Skills. arXiv preprint arXiv:2109.09605.

Farooq, A., Anwar, S., Awais, M., & Rehman, S. (2017, October). A deep CNN based multi-class classification of Alzheimer's disease using MRI. In 2017 IEEE International Conference on Imaging systems and techniques (IST) (pp. 1-6). IEEE.

Guo, H., & Wang, W. (2015). An active learning-based SVM multi-class classification model. Pattern recognition, 48(5), 1577-1597.

Neculoiu, P., Versteegh, M., & Rotaru, M. (2016, August). Learning text similarity with siamese recurrent networks. In Proceedings of the 1st Workshop on Representation Learning for NLP (pp. 148-157).

Matthes, B., & Paulus, W. (2013). Klassifikation der Berufe. Struktur, Codierung und Umsteigeschlüssel. FDZ-Methodenreport, 8, 1-35.

Wang, J., Abdelfatah, K., Korayem, M., & Balaji, J. (2019, December). DeepCarotene-Job Title Classification with Multi-stream Convolutional Neural Network. In 2019 IEEE International Conference on Big Data (Big Data) (pp. 1953-1961). IEEE.

Wang, J., Wang, Z., Zhang, D., & Yan, J. (2017, August). Combining Knowledge with Deep Convolutional Neural Networks for Short Text Classification. In IJCAI (Vol. 350).

Zhu, Y., Javed, F., & Ozturk, O. (2017, May). Document embedding strategies for job title classification. In The Thirtieth International Flairs Conference.

Appendix

A. Results

A. First Results – Level 3 – classes not removed

	LR	SVM	RF
CountVectorizer	0.48	0.50	0.43
TFIDF	0.47	0.51	0.43
Word2Vec	0.28	0.14	0.35
Doc2Vec	0.19	0.20	0.31

	LR	SVM	RF
CountVectorizer	p: 0.42, r: 0.27, F1: 0.3	p: 0.41, r: 0.35, F1: 0.36	p: 0.37, r: 0.25, F1: 0.28
TFIDF	p: 0.38, r: 0.23, F1: 0.26	p: 0.41, r: 0.35, F1: 0.36	p: 0.36, r: 0.25, F1: 0.27
Word2Vec	p: 0.15, r: 0.11, F1: 0.12	p: 0.07, r: 0.05, F1: 0.04	p: 0.23, r: 0.18, F1: 0.19
Doc2Vec	p: 0.1, r: 0.05, F1: 0.05	p: 0.13, r: 0.08, F1: 0.09	p: 0.23, r: 0.13, F1: 0.14

First Results – Level 3 – classes removed

Accuracy

	LR	SVM	RF
CountVectorizer	0.54	0.52	0.47
TFIDF	0.53	0.54	0.49
Word2Vec	0.34	0.28	0.39
Doc2Vec	0.24	0.28	0.36
BERT	0.51	0.49	0.45

Precision (p), Recall (r), F1 - Macro

	LR	SVM	RF
CountVectorizer	p: 0.63, r: 0.49, F1: 0.53	p: 0.57, r: 0.48, F1: 0.51	p: 0.55, r: 0.42, F1: 0.46
TFIDF	p: 0.66, r: 0.47, F1: 0.52	p: 0.59, r: 0.49, F1: 0.51	p: 0.58, r: 0.44, F1: 0.48
Word2Vec	p: 0.38, r: 0.26, F1: 0.28	p: 0.31, r: 0.23, F1: 0.24	p: 0.39, r: 0.34, F1: 0.35
Doc2Vec	p: 0.36, r: 0.14, F1: 0.15	p: 0.33, r: 0.23, F1: 0.25	p: 0.42, r: 0.29, F1: 0.31
BERT	p: 0.53, r: 0.44, F1: 0.46	p: 0.44, r: 0.45, F1: 0.44	p: 0.56, r: 0.38, F1: 0.42

First Results – Level 5 – classes removed

Accuracy

	LR	SVM	RF
CountVectorizer	0.63	0.65	0.57
TFIDF	0.61	0.65	0.59
Word2Vec	0.44	0.38	0.49
Doc2Vec	0.27	0.35	0.45
BERT	0.62	0.63	0.55

Precision (p), Recall (r), F1 - Macro

	LR	SVM	RF
CountVectorizer	p: 0.71, r: 0.59, F1: 0.63	p: 0.67, r: 0.62, F1: 0.63	p: 0.63, r: 0.55, F1: 0.57
TFIDF	p: 0.72, r: 0.57, F1: 0.61	p: 0.67, r: 0.63, F1: 0.63	p: 0.63, r: 0.56, F1: 0.58
Word2Vec	p: 0.53, r: 0.38, F1: 0.42	p: 0.38, r: 0.32, F1: 0.31	p: 0.49, r: 0.46, F1: 0.46
Doc2Vec	p: 0.41, r: 0.15, F1: 0.17	p: 0.36, r: 0.3, F1: 0.31	p: 0.53, r: 0.37, F1: 0.41
BERT	p: 0.66, r: 0.58, F1: 0.60	p: 0.61, r: 0.62, F1: 0.60	p: 0.64, r: 0.49, F1: 0.53

B. Limitations

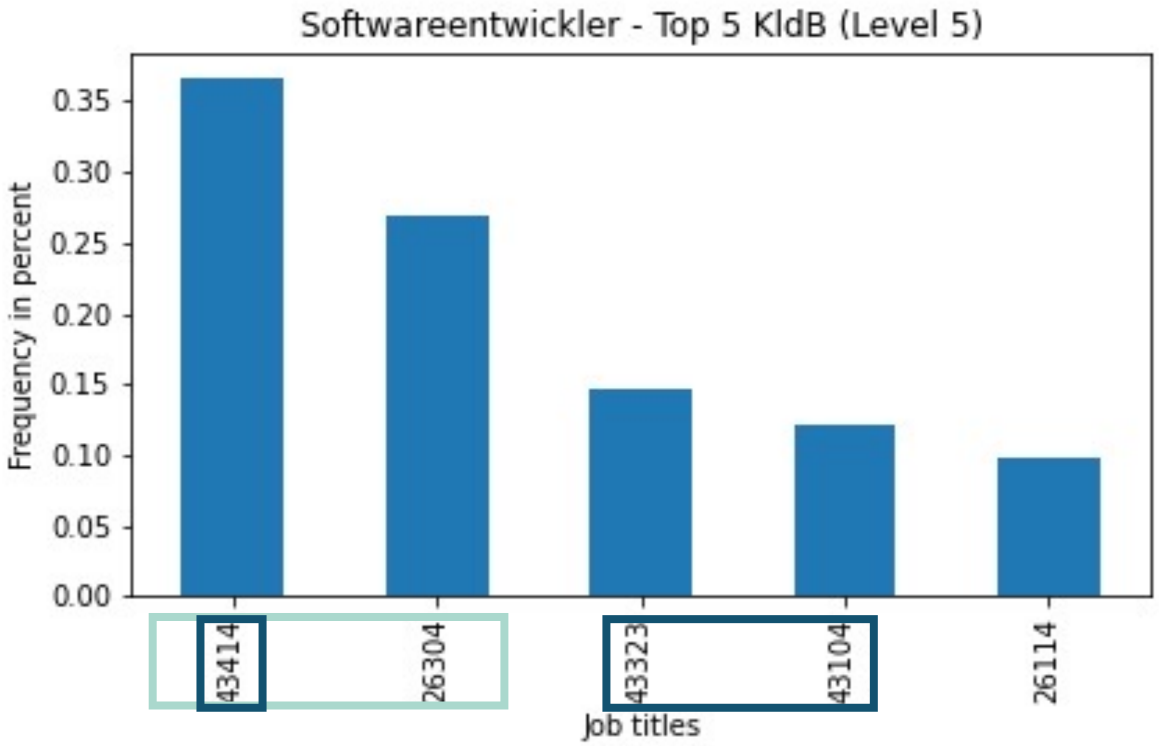
Discussion/Limitation

Occupation: „Softwareentwickler“

Ambiguity between Berufsbereichen
- 43414: **Naturwissenschaft, Geografie und Informatik**
- 26304: **Rohstoffgewinnung, Produktion und Fertigung**

Ambiguity within Berufsbereichs/ Berufshauptgruppe
- **43323**
- **43104**
- **43414**

Informatik- Informations- und Kommunikationstechnologieberufe



C. Definitions

Vectorization – sparse vectors

Example - Count Vectorizer

- Assume that the job titles can only be constructed using the following four words: Java, Python, developer and Entwickler. This is our vocabulary, which has in our case the length 4
- We construct for the vectorization a vector of length 4 for each job title
- The order of the words in the titles do not matter.
- For each title we go through each word in the word vector and replace it with a one, if the word occurs in the title.
- These vectors are called sparse because as the vocabulary increases, the length of the vector increases, so we get a vector with many zeros and a high dimensional space

0	0	0	0
Java	Python	developer	Entwickler

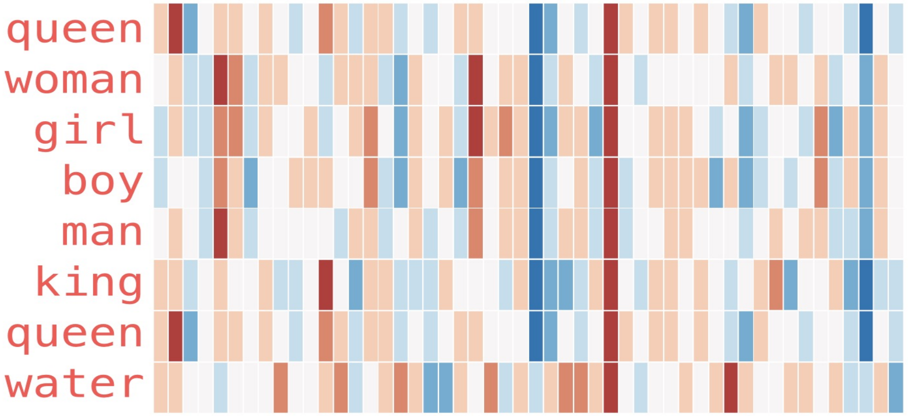
Example 1: „Java developer“

1	0	1	0
Java	Python	developer	Entwickler

Vectorization – dense vectors/word embeddings

abstract Example

- assume each word is represented by a coloured vector
- in contrast to count vectorizer word embeddings learn the similarity between words
- each column represent a dimension
- if the colour is the same for two words in the same dimension this means that the words are similar in that dimension



Source: Alammar, J. The illustrated Word2vec. 2019 <https://jalammar.github.io/illustrated-word2vec/> accessed: 09.12.2021