LTV: Labeled Topic Vector

Daniel Baumartz **Tolga Uslu** **Alexander Mehler**
Text Technology Lab
Goethe University Frankfurt
Frankfurt, Germany
{baumartz,uslu,mehler}@em.uni-frankfurt.de
https://www.texttechnologylab.org/

Abstract

In this paper, we present *LTV*, a website and an API that generate labeled topic classifications based on the *Dewey Decimal Classification* (DDC), an international standard for topic classification in libraries. We introduce *nnDDC*, a largely language-independent neural network-based classifier for DDC-related topic classification, which we optimized using a wide range of linguistic features to achieve an F-score of 87,4%. To show that our approach is language-independent, we evaluate *nnDDC* using up to 40 different languages. We derive a topic model based on *nnDDC*, which generates probability distributions over semantic units for any input on sense-, word- and text-level. Unlike related approaches, however, these probabilities are estimated by means of *nnDDC* so that each dimension of the resulting vector representation is uniquely labeled by a DDC class. In this way, we introduce a neural network-based *Classifier-Induced Semantic Space* (*nnCISS*).

1 Introduction

We present a model for calculating neural network-based *Classifier-Induced Semantic Space*s (*nnCISS*) using the *Dewey Decimal Classification* (DDC), that is, an international standard for topic classification in libraries. Based on this model, input units on the sense-, word-, sentence- or text level can be mapped onto the same feature space to compute, for example, their semantic similarity (Bär et al., 2012; Pilehvar and Navigli, 2015). Such an approach is needed whenever multiresolutional semantic information has to be processed to interrelate, for example, units of different levels of linguistic resolution (e.g., words or phrases to texts).

Contrary to related approaches (Landauer and Dumais, 1997; Blei et al., 2003) we use classifiers to define the dimensions of CISS, which are directly labeled by the underlying target class. This has the advantage that embeddings of linguistic units in semantic spaces can be interpreted directly in relation to the class labels.

In order to demonstrate the expressiveness of *nnCISS*, we conduct two classification tasks and show that using *nnCISS*-based feature vectors improve any of these classifications.

We generate several DDC corpora by exploring information from *Wikidata*, *Wikipedia* and the *Integrated Authority File* (*Gemeinsame Normdatei* – GND) of the German National Library. Any Wikipedia article in such a corpus is linked to an entry in Wikidata, which contains a property[1] attribute referring to the DDC, or to a GND page containing a corresponding DDC tag[2]. Since many Wikipedia articles refer to Wikidata or the GND, we were able to explore these articles as training examples of the corresponding DDC classes. The DDC includes three levels of thematic resolution: The first level distinguishes 10 main topics, each of which is subdivided into maximally 10 topics on the 2nd level (99 classes), which in turn are subdivided into maximally 10 topics on the 3rd level (915 classes). We use the 2nd and 3rd level of DDC as two alternative classification schemes.

[1] https://www.wikidata.org/wiki/Property:P1036

[2] e.g., https://d-nb.info/gnd/4176546-1

Proceedings of the 27th International Conference on Computational Linguistics: System Demonstrations, pages 142–145
Santa Fe, New Mexico, USA, August 20-26, 2018.

Wikipedia is offered for a wide range of languages, which allows us to create such corpora for different languages. In addition, translations provided by both Wikipedia and Wikidata enable the creation of language-specific training corpora by evaluating translation relationships between articles assigned to the DDC and articles for which these assignments do not exist. In this paper, we focus on Arabic, English, French, German, Spanish, and Turkish while performing a deeper analysis by example of the German corpus (#articles 15 136, #tokens per article 1 228, #classes 2nd level 98 and #classes 3rd level 641). Additionally we select more Wikipedias from the *List of Wikipedias*[3], where $depth >= 50$ and $\#articles >= 10\,000$, to be available through our *LTV* API.

2 Classification Model

The architecture of the *LTV* framework consists of four steps:

1. We use TextImager (Hemati et al., 2016) for preprocessing (lemmatization, part of speech tagging) the German Wikipedia and perform *Word Sense Disambiguation* (WSD) by means of fastSense (Uslu et al., 2018a), a WSD tool that is trained on the entire German Wikipedia. Our approach is in line with (Pilehvar and Navigli, 2015) and, thus, disambiguates input words to obtain sense representations as input for calculating sense embeddings.

2. The disambiguated Wikipedia corpus is then used to create sense embeddings by means of word2vec (Mikolov et al., 2013) using all sentences as input.

3. The aim is to obtain disambiguated articles and sense embeddings for training a DDC classifier and thus generating *nnDDC*. For this we enrich the disambiguated Wikipedia articles with DDC information using Wikidata/GND. We use (Uslu et al., 2018b) to classify an input on the sense-, word, sentence- or document-level regarding the DDC as the target classification. In this paper, we optimize this classifier with respect to feature selection and extend it by alternatively using sense embeddings combined with a disambiguated corpus.

4. Next we utilize *nnDDC* to generate *nnCISS* for a given input in this way, that each input unit on the sense-, word- or text-level can be mapped onto an n-dimensional feature vector whose dimensions correspond to DDC classes. *nnCISS* generates a probability distribution over the DDC classes (of either the 2nd or 3rd level).

3 Evaluation

3.1 Evaluating *nnDDC*

We evaluate *nnDDC* regarding the question which features are most successful in DDC-oriented text classification.

We have trained and evaluated different document inputs (articles, sections, paragraphs and sentences as well as disambiguations and embeddings) and features like lemmatization of input token, included POS info, removed function words, sub-word units or n-gram features. We have also conducted a parameter study on various training hyperparameters like number of epochs and learning rate. In this way, we have increased the F-score to 87,4%.

Language	DDC 2	DDC 3
German	87,4%	78,1%
English	79,8%	72,6%
Arabic	79,8%	68,8%
Turkish	78,9%	67,5%
French	79,4%	68,1%
Spanish	79,7%	70,5%

Figure 1: F-scores for different languages for 2nd and 3rd level DDC.

Table 1 shows that though *nnDDC* performs worse in the case of the other languages compared to German, the results for the 2nd level of the DDC are nevertheless close to 80%. Evaluating the about 40 more languages we achieve an average score of 71%. Since corpus generation for these languages is

[3] https://en.wikipedia.org/wiki/List_of_Wikipedias#Detailed_list

straightforward, this also demonstrates that our approach is largely language independent at least what concerns languages that are sufficiently manifested by language specific releases of Wikipedia.

Switching to the 3rd level of DDC, we observe a drop in F-score, while in case of the German Wikipedia we still perform at about 78% and any topic vector is now enriched by providing more detailed information.

3.2 Evaluating *nnCISS*

To show that our DDC-based topic model improves classification, we have performed classification tasks on two data sets: The *DBpedia Ontology Classification Dataset*[4] and the *AG's news corpus*[5]. To be independent of the classifier, this experiment was conducted by means of StarSpace (Wu et al., 2017). Table 2 shows the results and the impact of *nnCISS*, and while the improvements are not very large, with such a high classification quality every percentage is important.

Input	**DBpedia**	**AG News**
Text without *nnCISS*	97,89%	89,88%
Text + *nnCISS* (DDC 2)	98,00%	90,18%
Text + *nnCISS* (DDC 3)	98,06%	90,33%

Figure 2: F-Scores in the DBpedia and AG News classification tasks.

4 *LTV* Software Demonstration

We offer the classifier (*nnDDC*) and the DDC topic model (*nnCISS*) for all above mentioned languages on `https://textimager.hucompute.org/DDC/`. It is directly accessible as a REST API or via the UI on the website. We have implemented the classifier for *LTV* as an UIMA annotator, this allows us to seamlessly integrate into TextImager and utilize the pipeline feature to process the input text. In the pipeline we first preprocess the text in exactly the way we prepared our training data and then perform the classification via our annotator. This eliminates the need for the user to preprocess the input and also makes the results reproducible. To use the API one performs a `POST` request which contains the input text to classify as well as some information about the format and the pipeline to use. All available pipelines are listed on the site. For example:

```
{ ”inputText”: ”Beispiel über Angela Dorothea Merkel, ...”,
  ”inputFormat”: ”plain”, ”outputFormat”: ”ddc_json”, ”options”: [
   { ”de”: [
      ”LanguageToolSegmenter”, ”ParagraphSplitter”,
      ”MarMoTLemma”, ”MarMoTTagger”,
      ”FastTextDDC2LemmaNoPunctPOSNoFunctionwordsWithCategories
        TextImagerService” ] } ] }
```

This request returns an `JSON` object containing:

```
{ ”ddc”:[
     {”prob”:0.990234,”label”:”__label_ddc__320”,”tags”:[”ddc2”]},
     ... ],
  ”success”: true, ”language”: ”de” }
```

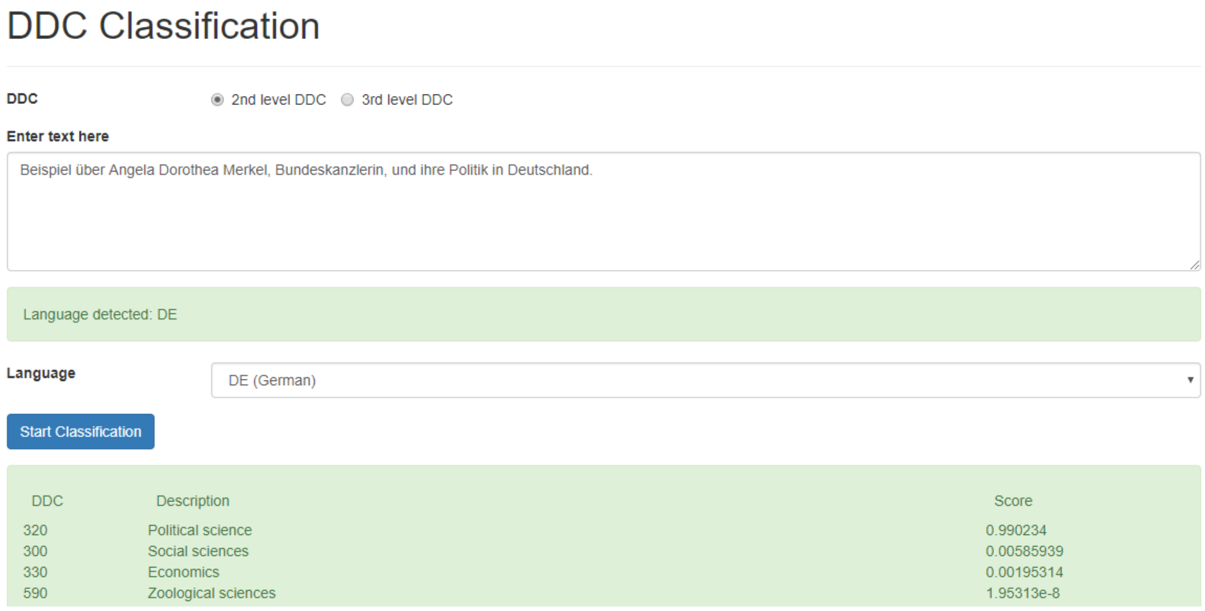
The website provides an easy access to the API, requiring no programming skills to use. Users can paste text to classify and select the DDC level and language (it also tries to autodectect the language of the input text and selects a suitable pipeline for you). The UI then displays the results providing the DDC description, see Figure 3.

5 Conclusion

We presented a website and API to access and use a neural network based classifier to categorize DDC classes. For this we have used various features and resources to achieve the best possible classification,

[4] `www.wiki.dbpedia.org/data-set-2014`

[5] `www.di.unipi.it/~gulli/AG_corpus_of_news_articles.html`



DDC Classification

DDC ● 2nd level DDC ○ 3rd level DDC

Enter text here

Beispiel über Angela Dorothea Merkel, Bundeskanzlerin, und ihre Politik in Deutschland.

Language detected: DE

Language: DE (German)

DDC	Description	Score
320	Political science	0.990234
300	Social sciences	0.00585939
330	Economics	0.00195314
590	Zoological sciences	1.95313e-8

Figure 3: Screenshot of the *LTV* website

managing to achieve a quality of over 87% (and considering the top three classes, we even exceed 96%). For a given text, the classifier generates a probability distribution over the DDC classes and thus a vector. This vector can be used as input for other classification tasks and we have shown that improvements can be achieved.

References

Daniel Bär, Chris Biemann, Iryna Gurevych, and Torsten Zesch. 2012. UKP: Computing semantic textual similarity by combining multiple content similarity measures. In *Proc. of SemEval '12*, pages 435–440, Stroudsburg.

David M Blei, Andrew Y Ng, and Michael I Jordan. 2003. Latent dirichlet allocation. *Journal of machine Learning research*, 3(Jan):993–1022.

Wahed Hemati, Tolga Uslu, and Alexander Mehler. 2016. Textimager: a distributed uima-based system for nlp. In *Proceedings of COLING 2016, the 26th International Conference on Computational Linguistics: System Demonstrations*, pages 59–63.

Thomas K. Landauer and Susan T. Dumais. 1997. A solution to Plato's problem: The latent semantic analysis theory of acquisition, induction, and representation of knowledge. *Psychological Review*, 104(2):211–240.

Tomas Mikolov, Kai Chen, Greg Corrado, and Jeffrey Dean. 2013. Efficient estimation of word representations in vector space. *arXiv preprint arXiv:1301.3781*.

Mohammad Taher Pilehvar and Roberto Navigli. 2015. From senses to texts: An all-in-one graph-based approach for measuring semantic similarity. *Artificial Intelligence*, 228:95–128.

Tolga Uslu, Alexander Mehler, Daniel Baumartz, Alexander Henlein, and Wahed Hemati. 2018a. fastsense: An efficient word sense disambiguation classifier. In *Proceedings of the 11th edition of the Language Resources and Evaluation Conference, May 7 - 12*, LREC 2018, Miyazaki, Japan. accepted.

Tolga Uslu, Alexander Mehler, Andreas Niekler, and Wahed Hemati. 2018b. Towards a DDC-based topic network model of wikipedia. In *Proceedings of 2nd International Workshop on Modeling, Analysis, and Management of Social Networks and their Applications (SOCNET 2018), February 28, 2018*. accepted.

Ledell Wu, Adam Fisch, Sumit Chopra, Keith Adams, Antoine Bordes, and Jason Weston. 2017. Starspace: Embed all the things! *CoRR*, abs/1709.03856.