

# German Text Embedding Clustering Benchmark

Silvan Wehrli<sup>1</sup> Bert Arnrich<sup>2</sup> Christopher Irrgang<sup>1</sup>

<sup>1</sup>Centre for Artificial Intelligence in Public Health Research (ZKI-PH)  
Robert Koch Institute, Berlin, Germany  
{WehrliS,IrrgangC}@rki.de

<sup>2</sup>Digital Health - Connected Healthcare  
Hasso Plattner Institute, University of Potsdam, Germany  
Bert.Arnrich@hpi.de

## Abstract

This work introduces a benchmark assessing the performance of clustering German text embeddings in different domains. This benchmark is driven by the increasing use of clustering neural text embeddings in tasks that require the grouping of texts (such as topic modeling) and the need for German resources in existing benchmarks. We provide an initial analysis for a range of pre-trained mono- and multilingual models evaluated on the outcome of different clustering algorithms. Results include strong performing mono- and multilingual models. Reducing the dimensions of embeddings can further improve clustering. Additionally, we conduct experiments with continued pre-training for German BERT models to estimate the benefits of this additional training. Our experiments suggest that significant performance improvements are possible for short text. All code and datasets are publicly available.

## 1 Introduction

Clustering is increasingly used in tasks requiring to group semantically similar text pieces. This includes, for instance, data selection (Aharoni and Goldberg, 2020), data exploration (Voigt et al., 2022), and neural topic modeling (Zhao et al., 2021). One approach for this kind of topic modeling is BERTopic (Grootendorst, 2022), which, in principle, uses generic clustering algorithms for text embeddings to find latent topics in text corpora. This is in stark contrast to more traditional topic modeling techniques using Latent Dirichlet Allocation (Blei et al., 2003) or Non-Negative Matrix Factorization (Févotte and Idier, 2011) and representing text as simple bag-of-words. The shift to embedding-based approaches is driven by the continuous development of neural language models,

successfully used in natural language understanding (NLU) tasks such as semantic textual similarity (Reimers and Gurevych, 2019; Gao et al., 2021) or retrieval and reranking (Huang et al., 2020; Yates et al., 2021). The availability of plug-and-play frameworks for the computation of vector representation only fosters this trend. One such framework is Sentence Transformers (Reimers and Gurevych, 2019), which is used by BERTopic. It provides an extensive collection of pre-trained transformer models and techniques to fine-tune models for similarity-focused language tasks.

Benchmarks help to understand the usefulness of these easily available language models, allowing to compare existing and newly developed models for language tasks of interest. The Massive Text Embedding Benchmark (MTEB, Muennighoff et al., 2023) provides such a benchmark for a wide range of embedding-based tasks (e.g., classification, clustering, or reranking) and datasets from different domains (e.g., online reviews, scientific publications, or social media). MTEB includes a wider range of tasks and focuses on more recent language models than other benchmarks (such as SentEval (Conneau and Kiela, 2018)). MTEB, offering an easy-to-use API, invites the evaluation of models and submissions to a publicly accessible leaderboard.<sup>1</sup>

However, MTEB only considers the English language for the evaluation of clustering. The inclusion of non-English data is important, as the performance of multilingual models may not equal their monolingual counterparts (Rust et al., 2021), and as a means to evaluate the potentially strong cross-lingual transfer capability of multilingual models

<sup>1</sup><https://huggingface.co/spaces/mteb/leaderboard>

(e.g., Huang et al., 2019). This work addresses this limitation by providing benchmark datasets and results for German. What is more, MTEB evaluates clustering performance on a single clustering algorithm. This is a suitable approach for such a broad benchmark as it simplifies the evaluation in terms of computational and content-related complexity. From a practical point of view, and specifically for clustering, the evaluation of different algorithms is helpful. Building on the MTEB API, we provide code and evaluation results for a broader range of clustering algorithms.

Finally, we conduct experiments with continued pre-training. The idea of this additional training is to adapt language models, typically trained on large and heterogeneous data collections, to the data of a specific domain or task, and has been shown to improve performance on downstream tasks (e.g., Howard and Ruder, 2018; Lee et al., 2019; Gururangan et al., 2020). We analyze the benefit of such adaptive training for clustering within this work. All code and datasets are publicly available.<sup>2</sup>

## 2 Datasets

### 2.1 MTEB Clustering

**Data sources** The MTEB clustering benchmark covers a range of topical domains and writing styles using data from different sources: arXiv, bioRxiv, and medRxiv for scientific publications (e.g., economics or medicine), Reddit for informal social media, Stack Exchange for topical online discussions (such as code), and the 20 Newsgroup dataset (Buitinck et al., 2013).

**Text length** MTEB contains two datasets for each data source: A sentence-to-sentence (S2S) dataset compares short texts, and a paragraph-to-paragraph (P2P) dataset compares relatively longer texts. For instance, in the case of arXiv, the S2S dataset only contains publication titles, and the P2P dataset contains the concatenation of titles and abstracts. The two datasets provide models with different amounts of information.

**Metric** The evaluation is based on the V-measure (Rosenberg and Hirschberg, 2007). Given a ground truth, the V-measure outputs a score between 0 and 1, measuring homogeneity (clusters contain only one class) and completeness (clusters contain all

class samples). MTEB uses topical categories derived from the data, such as the scientific discipline of a publication or newsgroup, as the ground truth.

**Data selection** Lastly, datasets in the MTEB clustering benchmark comprise up to 30 random samples of varying size and with a number of different classes drawn from all samples of a data source (splits).

### 2.2 German Additions

We follow MTEB’s design for German datasets, aiming to simulate a wide range of real-world scenarios by including different domains, text lengths, and clustering complexities (Subsection 2.1). Compared to English, fewer German open-source datasets seem to exist, which are suitable for this work. Furthermore, some of the open-source datasets in MTEB are generally less relevant for a German benchmark as they contain little to no German content. This includes, for instance, arXiv and Stack Exchange, both mostly English-only data sources.<sup>3</sup> We have identified three openly available German data sources relevant to this benchmark. In the following, we discuss these data sources and the constructed benchmark datasets in more detail (see Table 1 for a summary).<sup>4</sup>

**Blurbs** As a first data source, we use data from the GermEval 2019 shared task on hierarchical blurbs classification (Remus et al., 2019). This data consists of German book metadata, including titles, blurbs (short, promotional descriptions of books), and genres. Even though blurbs are not part of MTEB, the data is well-suited: it is open source and contains topical texts of different lengths (titles, blurbs). What is more, three levels of genres express different levels of detail. The most general genres, for instance, include *Sachbuch* (non-fiction) or *Literatur und Unterhaltung* (literature and entertainment). Secondary and tertiary genres are increasingly specific (e.g., *Fantasy* (fantasy) and *Historische Fantasy* (historical fantasy)). We use this information to evaluate a model’s ability to cluster at different granularity (i.e., the ground truth). We build two datasets, one

<sup>3</sup>There are only around 600 submissions on arXiv (following best practices for search) at the time of writing (<https://info.arxiv.org/help/faq/multilang.html>). Similarly, Stack Exchange content is almost exclusively in English: <https://meta.stackexchange.com/questions/13676/d-o-posts-have-to-be-in-english-on-stack-exchange>

<sup>4</sup>We provide the scripts used to create these datasets in the Hugging Face repositories linked in our GitHub repository.

<sup>2</sup><https://github.com/ClimSocAna/tecb-de>. Additionally, the datasets are officially part of the MTEB library, and results are available in the MTEB leaderboard.

Name	Target	Unique Samples	Splits	Size (per split)	Classes (per split)	Avg. chars (per sample)
BlurbsClusteringS2S	book titles	17,726	28	177 to 16,425	4 to 93	23
BlurbsClusteringP2P	blurbs (title and blurb)	18,084	28	177 to 16,425	4 to 93	664
TenKGnadClusteringS2S	news article titles	10,267	9	1,436 to 9,962	9	51
TenKGnadClusteringP2P	news article texts (title and text)	10,275	9	1,436 to 9,962	9	2,648
RedditClusteringS2S	submission titles	40,181	10	9,288 to 26,221	10 to 50	52
RedditClusteringP2P	submission descriptions (title and text)	40,305	10	9,288 to 26,221	10 to 50	902

Table 1: Summary of the German benchmark datasets for evaluating the clustering performance of neural language models. Numbers for *Avg. chars* are rounded.

that only includes book titles (BlurbClusteringS2S) and one that includes the concatenation of titles and blurbs (BlurbsClusteringP2P). The design is based on MTEB’s arXiv-based clustering tasks, which use arXiv’s two-level categorization (e.g., math and numerical analysis) to simulate cluster granularity.

More concretely, we create 10 splits (subsamples) that consider only the broadest category (coarse clustering) and, similarly, 10 splits that consider the second-level genre (fine-grained clustering across all top-level genres). We randomly selected between 10 and 100 percent of the available data for each split. Lastly, we create eight splits by splitting the data based on the top-level genre and considering the second-level genre (fine-grained clustering within a genre).<sup>5</sup>

*Der Krieg der Trolle (4)*

*Im Land zwischen den Bergen ist die Zeit des Friedens vorbei. Krieg liegt in der Luft, und dann taucht auch noch ein tödlich verwundeter Zwerg im südlichen Hochland von Wlachkis auf – Ereignisse, die wie ein dunkler Schatten auf dem Land liegen. [...]*

Example for a book title and blurb from the main category *Literatur & Unterhaltung* respectively *Fantasy* and *Abenteuer-Fantasy* (second level).

**News articles** As a second source, we use data from the One Million Posts Corpus (Schabus et al.,

<sup>5</sup>We only consider samples with one top-level and up to two second-level genres. If a sample has two second-level genres, we select the less frequent one (assuming it is more descriptive) to make the label selection less ambiguous.

2017), inspired by the 10kGNAD dataset<sup>6</sup>. 10kGNAD extracts news article information from the One Million Posts Corpus, which consists of annotated user comments (including the corresponding news articles) posted to an Austrian newspaper website. There are nine news categories such as *Wissenschaft* (science) or *Web*, and we use these categories as ground truth for the evaluation. We build two datasets: TenKGnadClusteringS2S, only using article titles, and TenKGnadClusteringP2P dataset, using the whole article texts. We follow MTEB’s TwentyNewsgroupsClustering (consisting of news article titles and newsgroups) data selection strategy and draw 10 random samples of varying sizes (selecting at least 10% of all data).

*Stoke holt Shaqiri von Inter*

*Arnautovic-Klub zahlt Rekordsumme für Schweizer*

*Stoke-on-Trent/Mailand – Xherdan Shaqiri wechselt von Inter Mailand zu Stoke City und wird damit Teamkollege von Marko Arnautovic. [...]*

Example of a news article consisting of the title, the subheadline and the article text from the *Sport* news section.

**Reddit** We use data from Reddit as a third data source which we retrieved from the official Reddit API<sup>7</sup>. More precisely, we have collected popular (i.e., hot and top) submissions to 80 German Subreddits such as *r/Bundesliga*, *r/Finanzen*, or

<sup>6</sup><https://tblock.github.io/10kGNAD>

<sup>7</sup><https://www.reddit.com/dev/api>

*r/reisende*.<sup>8</sup> We do not disclose the raw data. Instead, we provide the submission ids and scripts to reproduce the datasets in our GitHub repository. Additionally, Figure 3 in the appendix summarizes the collected data. Our approach is motivated by data privacy and sharing considerations, as discussed in detail in the [Ethics Statement](#).

In any case, we construct two datasets from the collected data: SubredditClusteringS2S, which only considers the submission titles, and SubredditClusteringP2P, which combines submission titles and texts. Subreddits We follow the data selection used for MTEB’s Reddit-based datasets and build 10 splits with submissions from 10 to 50 randomly selected Subreddits.

*Wieviel “Trinkgeld” für Lieferdienste?*

*Wie viel gebt ihr - sofern ihr Lieferdienste wie Lieferando etc. nutzt - den Fahrern Trinkgeld? Richtet ihr euch nach der 10% Faustregel in der Gastro?*

Example for a submission consisting of the title and text to the German Subreddit *r/Finanzen*.

### 3 Evaluation Setup

#### 3.1 Models

We select a range of transformer-based models (Vaswani et al., 2017) based on their ability to process German text, architecture, and pre-training methods.<sup>9</sup> For all models, similar to MTEB, we use the mean of a model’s output embeddings as text embedding (mean-pooling).

**Monolingual models** We include the monolingual GBERT and GELECTRA models (Chan et al., 2020), both based on the BERT (Devlin et al., 2019) architecture. GBERT uses whole word masking (WWM) for pre-training<sup>10</sup>, while GELECTRA uses the ELECTRA pre-training method (Clark et al., 2020), which aims to improve computational efficiency. The models are pre-trained on the German part of OSCAR (Ortiz Suárez et al., 2019),

<sup>8</sup>A Subreddit is a topic-specific forum on Reddit, and a submission is a post to a Subreddit. We select active German Subreddit based on desk research and filter for German submissions if Subreddits also contain non-German submissions.

<sup>9</sup>Table 4 in the appendix lists the repositories of all models.

<sup>10</sup>The originally proposed BERT architecture (Devlin et al., 2019) masks subword tokens during pre-training. The authors introduced whole word masking after the publication: <https://github.com/google-research/bert/commit/0fce55>.

a set of monolingual corpora based on Common Crawl (Wenzek et al., 2020), which is a repository for multilingual web crawl data. Additionally, the pre-training data includes, in smaller parts, dumps from Wikipedia and text from a range of domains such as court decisions, movie subtitles, speeches, or books. GottBERT (Scheible et al., 2020) presents another BERT-flavoured German language model. It is trained on the German part of OSCAR. It uses the RoBERTa pre-training setup (Liu et al., 2019), aiming to optimize the training setup (e.g., hyperparameter values) of the original BERT setup (Devlin et al., 2019).

**Multilingual models** We choose competitive multilingual models based on the MTEB leaderboard. This includes two pre-trained English models (MiniLM-L12-v2-ml, MPNet-base-v2-ml) fine-tuned using multilingual knowledge distillation (Reimers and Gurevych, 2020). Another model, USE-CMLM-ml, uses an adapted masked language modeling technique for training (Yang et al., 2021). SRoBERTa-cross, a Hugging Face community model, is based on the small variant of XLM-RoBERTa, a RoBERTa model trained on data in over 100 languages from Common Crawl, and then fine-tuned for German-English sentence similarity. We also use Sentence-T5 (ST5) encoders (Ni et al., 2022). These models are based on the multilingual general-purpose T5 encoder-decoder model (Raffel et al., 2020) and fine-tuned on a large English dataset for sentence similarity. All of these models use training techniques designed to improve short text representations. Therefore, we also select XLM-RoBERTa-large as a more general-purpose multilingual model. It uses the masking technique from BERT (Devlin et al., 2019) for pre-training.

#### 3.2 Clustering

**Algorithms** Like MTEB, we use Minibatch k-Means (Buitinck et al., 2013) and V-measure as an evaluation metric. Additionally, we perform analyses for Agglomerative Clustering (Buitinck et al., 2013), DBSTREAM (Montiel et al., 2021), and HDBSCAN (McInnes et al., 2017). We select these algorithms based on their ease of use (e.g., pip-ready package), popularity, and abilities: Agglomerative Clustering is distance-based, similar to Minibatch k-Means. However, it may be more suited for modeling clusters of varying shapes and sizes. HDBSCAN, a density-based clustering algorithm, is used per default by the increas-

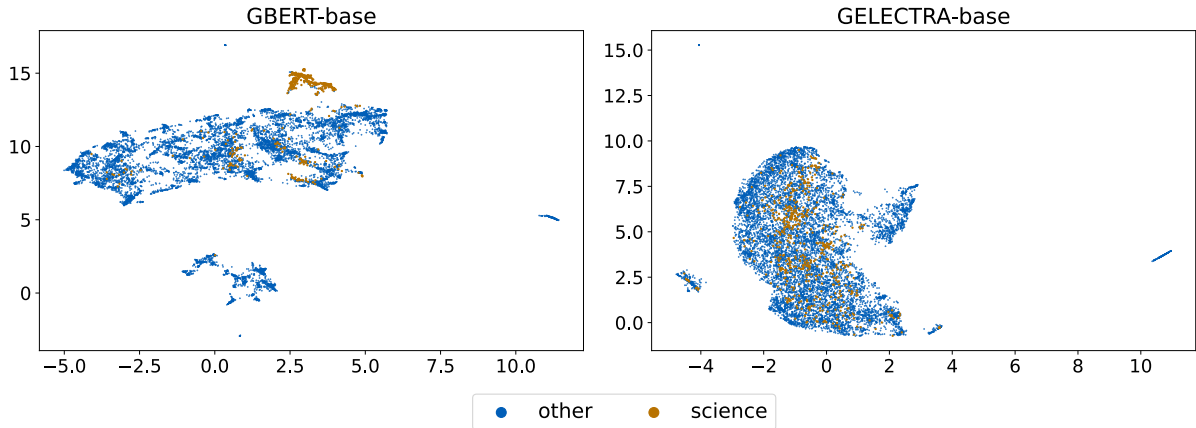


Figure 1: Word embeddings of all texts from the TenKgnadClusteringP2P dataset reduced to two dimensions with UMAP. Texts belonging to the news category *Wissenschaft* (science) are highlighted by color and size.

ingly popular BERTopic. Lastly, DBSTREAM is a density-based algorithm for evolving data streams. In principle, DBSTREAM could cluster documents for real-time analysis (e.g., news monitoring). We use default parameters for all algorithms, focusing on the out-of-the-box performance. DBSTREAM does not support setting the number of clusters. In all other cases, we provide clustering models with this information (similar to MTEB).

**Dimensionality reduction** We also experiment with dimensionality reduction to cluster lower dimensional data, motivated by the curse of dimensionality (Beyer et al., 1999; Aggarwal et al., 2001). We reduce the embedding vectors to two dimensions for every language model using PCA (Buitinck et al., 2013), a standard technique and UMAP (McInnes et al., 2018), which BERTopic suggests.

### 3.3 Adaptive Pre-training

We conduct experiments with adaptive pre-training separately for each dataset described in Subsection 2.2. In general, we assume an application scenario, where clustering is used to unsupervisedly analyze an ongoing text-based information feed, e.g., Twitter. Here, the focus lies not on the extrapolation capability towards new unseen data but on a consistent clustering of the entire text body. Thus, we use the evaluation data simultaneously as training data. This setup allows us to draw real-world conclusions, whether continued pre-training as an additional step before clustering can reliably improve clustering outcomes. We experiment with two pre-training techniques and evaluate them through V-measure. Firstly, we train with the general-purpose WWM technique (simi-

lar to GBERT). Given the relatively small training datasets, we follow the parameter setup for task-adaptive pre-training suggested in Gururangan et al. (2020). We also experiment with the Transformers and Sequential Denoising Auto-Encoder (TSDAE) method (Wang et al., 2021), a state-of-the-art unsupervised training method for improving sentence embeddings, and we use the suggested parameter setup by the authors.

We use GBERT for these experiments. GBERT perform competitively (as discussed in Section 4), and we are interested in the potential improvements for such strong models. What is more, GBERT uses the general-purpose WWM for pre-training, allowing it to evaluate the effect of the more task-specific TSDAE training. Generally, we want to provide some intuition for the potential use of these training methods specifically for clustering.

## 4 Results and Discussion

### 4.1 Baseline: Minibatch k-Means

**Monolingual models** GBERT models perform better than the other monolingual GELECTRA and GottBERT models, as shown in Table 2. GBERT-large ranks second best of all evaluated models. All models perform relatively better on P2P datasets compared to the S2S counterparts. This is an intuitive result, considering that models are presented with more information in these tasks. The weak performance of the GELECTRA models is surprising, given the strong results on downstream tasks reported in Chan et al. (2020). Figure 1 provides some visual intuition for this performance lack. For the news articles from the TenKgnadClusteringP2P dataset, GELECTRA produces more

Model	Blurbs		TenkGnad		Reddit		Avg.
	S2S	P2P	S2S	P2P	S2S	P2P	
GBERT-base	11.27	35.36	24.23	37.16	28.57	35.30	28.65
GBERT-large	13.38	39.30	<b>34.97</b>	41.69	34.47	44.61	34.74
GELECTRA-base	7.74	10.06	4.11	9.02	6.59	7.73	7.54
GELECTRA-large	7.57	13.96	3.91	11.49	7.59	10.54	9.18
GottBERT	8.37	34.49	9.34	33.66	16.07	19.46	20.23
MiniLM-L12-v2-ml	14.33	32.46	22.26	36.13	33.34	44.59	30.52
MPNet-base-v2-ml	15.81	34.38	22.00	35.96	36.39	48.43	32.16
SROBERTa-cross	12.69	30.82	10.94	23.50	27.98	33.01	23.16
USE-CMLM-ml	15.24	29.63	25.64	37.10	33.62	49.70	31.82
ST5-base	11.57	30.59	18.11	<b>44.88</b>	31.99	45.80	30.49
ST5-xxl	<b>15.94</b>	<b>39.91</b>	19.69	43.43	<b>38.54</b>	<b>55.90</b>	<b>35.57</b>
XLM-RoBERTa-large	7.29	29.84	6.16	32.46	10.19	23.50	18.24

Table 2: V-measure scores for the benchmark results of all evaluated models using the Minibatch k-Means algorithm. Results are multiplied by 100 and rounded to two decimals. **Bold** numbers indicate best column-wise result.

Algorithm	Blurbs		TenKGnad		Reddit		Avg.
	S2S	P2P	S2S	P2P	S2S	P2P	
Minibatch k-Means	11.77	30.07	16.78	32.21	25.44	34.88	25.19
<i>PCA-reduced embeddings</i>	<i>9.40</i>	<i>23.50</i>	<i>11.41</i>	<i>20.56</i>	<i>11.95</i>	<i>16.10</i>	<i>15.49</i>
<i>UMAP-reduced embeddings</i>	<i>12.65</i>	<i>29.58</i>	<i>21.76</i>	<i>39.73</i>	<i>28.56</i>	<i>41.28</i>	<i>28.93</i>
Agglomerative Clustering	12.45	30.40	17.25	34.18	25.74	35.86	25.98
<i>PCA-reduced embeddings</i>	<i>9.33</i>	<i>23.30</i>	<i>11.03</i>	<i>20.24</i>	<i>11.89</i>	<i>16.67</i>	<i>15.41</i>
<i>UMAP-reduced embeddings</i>	<i>12.72</i>	<i>32.88</i>	<i>21.45</i>	<i>39.94</i>	<i>28.65</i>	<i>41.50</i>	<b><u>29.52</u></b>
HDBSCAN			n/a				n/a
<i>PCA-reduced embeddings</i>	<i>9.68</i>	<i>13.12</i>	<i>7.08</i>	<i>10.60</i>	<i>14.58</i>	<i>16.83</i>	<i>11.98</i>
<i>UMAP-reduced embeddings</i>	<i>14.98</i>	<i>22.51</i>	<i>14.46</i>	<i>27.95</i>	<i>24.19</i>	<i>30.61</i>	<i>22.45</i>
DBSTREAM			n/a				n/a
<i>PCA-reduced embeddings</i>	<i>6.41</i>	<i>14.19</i>	<i>7.79</i>	<i>12.46</i>	<i>8.38</i>	<i>10.68</i>	<i>9.99</i>
<i>UMAP-reduced embeddings</i>	<i>12.93</i>	<i>31.41</i>	<i>22.56</i>	<i>38.27</i>	<i>28.61</i>	<i>36.59</i>	<i>28.40</i>

Table 3: Average V-measure score of all evaluated models using different clustering algorithms and reduced embeddings as input (in *italic*). Results are multiplied by 100 and rounded to two decimals. The **bold** number indicates the best overall result and underlined results the best result per clustering algorithm.

evenly-spread embeddings than GBERT, and embeddings belonging to the same topic (such as science) tend to be more spread. GottBERT lies in the middle between the GELECTRA and GBERT models. The gap to GBERT models is likely caused by GottBERT’s smaller and less diverse training data. As discussed in Subsection 3.1, GBERT models contain training data that is more similar to the characteristics of the evaluation datasets (e.g., books and shorter text sequences such as movie subtitles).

**Multilingual models** Apart from SROBERTa-cross and XLM-RoBERTA-large, multilingual models perform competitively with scores close to the monolingual GBERT-base and GBERT-large. ST5-xxl is the best-performing model overall, scor-

ing best on five out of six datasets (Table 2). Moreover, the scaled ST5-xxl model (4.8B parameters) shows clear performance gains compared to its base variant (ST5-base, 110M parameters). ST5 models’ fine-tuning data includes Reddit data, which may explain the strong performance on German Reddit datasets, i.e., a robust cross-lingual transfer. The relatively weak results for XLM-RoBERTA-large are likely caused by less diverse training data and more general pre-training compared to the other multilingual models. The performance of SROBERTa-cross, based on the smaller version of the XLM-RoBERTA-large model and fine-tuned for sentence similarity, also points in this direction, performing better on five out of six datasets than XLM-RoBERTA-large.

## 4.2 Beyond k-Means

Table 3 reports the average V-measure score of all evaluated models for different clustering algorithms. We do not report results for DBSTREAM and HDBSCAN for non-reduced embeddings, as we observed very poor computational performance for high-dimensional data during our experiments. In any case, results for Minibatch k-Means and Agglomerative Clustering suggest possibly better performance with reduced embeddings: Using UMAP-reduced embeddings improves the Minibatch k-Means and Agglomerative Clustering scores, on average, by around +13-15% (compared to not using any reduction at all). What is more, clustering in low dimensions may benefit the explainability of models as it allows to visually analyze results (e.g., Figure 1). However, this does not hold for clustering with PCA-reduced embeddings, which show the worst results by far. This limits its use for text-based clustering.

Overall, and based on the results for clustering with UMAP-reduced embeddings, Minibatch k-Means and Agglomerative Clustering perform very similarly. DBSTREAM performs slightly worse on average, caused by relatively weaker results for P2P datasets.<sup>11</sup> HDBSCAN performs worst on five out of six datasets. We suspect the weaker results for HDBSCAN are caused by its sensitivity to classifying data points as noise. In our experiments, we observe that in some cases, more than 30% of the data is labeled as noise. A different configuration for HDBSCAN would likely improve results. However, the usefulness of such sensitive algorithms may also depend on the use case (e.g., whether any text data is considered as noise).

## 4.3 Adaptive Pre-training with GBERT

The evaluation for GBERT-base and adaptive pre-training with WWM and TSDAE, reported in Figure 2, shows clear performance improvements for all benchmark datasets.<sup>12</sup> The improvements on the S2S datasets are considerably more significant for both pre-training methods: After around one epoch of TSDAE training, V-measures improve by around +31% on average. After around 10 epochs of WWM training, V-measures improve by around +30% on average. For P2P datasets, the improve-

<sup>11</sup>Shortly before the final submission, we found a small bug in the DBSTREAM implementation we used: <https://github.com/online-ml/river/issues/1265>. However, we do not expect a significant influence on the overall results.

<sup>12</sup>We provide exact numbers in Table 5 in the appendix.

ments are relatively smaller, and models profit from more extended WWM training (average improvement of around +15% after 30 epochs). The benefit of TSDAE training seems less clear, requiring more extended training to compensate for initial performance drops and different training times to reach maximum improvements. Overall, and based on the results for 10 epochs of WWM training (Table 5), GBERT-base converges to the performance of the larger GBERT-large (33.70 vs. 34.74) and ranks second best out of all models for all S2S dataset. We suspect the more significant improvements on smaller text sequences (S2S) are likely caused by the fact that GBERT models are generally pre-trained on much longer sequences (maximum of 512 subword tokens per sample), considering that S2S samples are, on average, only around up to 50 characters long (Table 1). This may also explain why the improvements for P2P datasets are relatively minor, as these texts more closely resemble the pre-training data of the unadopted GBERT (in terms of text length).

We performed similar experiments for GBERT-large, as shown in Figure 4 in the appendix. In most cases, and for both pre-training methods, the performance decreases significantly (e.g., TenKGnadClusteringS2S with TSDAE) or stays relatively unchanged (e.g., TenKGnadClusteringS2S with WWM). The training stability seems low as different training with different seeds may result in different performances (see also Table 5). We suspect the relatively low batch sizes (256 for WWM and eight for TSDAE compared to 2,048 for GBERT-large’s previous pre-training) lead to these training instabilities, as parameter updates are too aggressive. The parameter setups we used are based on experiments with BERT models similar to GBERT-base in terms of parameters (Gururangan et al., 2020; Wang et al., 2021). Our results suggest that these setups are unsuitable for larger models.

## 5 Conclusion

This work introduces German benchmark datasets for the evaluation of embedding-based clustering, building on the monolingual clustering benchmark from Massive Text Embedding Benchmark (MTEB). We introduce six datasets from three sources (blurbs, news articles, and Reddit). Additionally, we evaluate the out-of-the-box performance of different clustering algorithms and show that UMAP-reduced embeddings improve cluster-

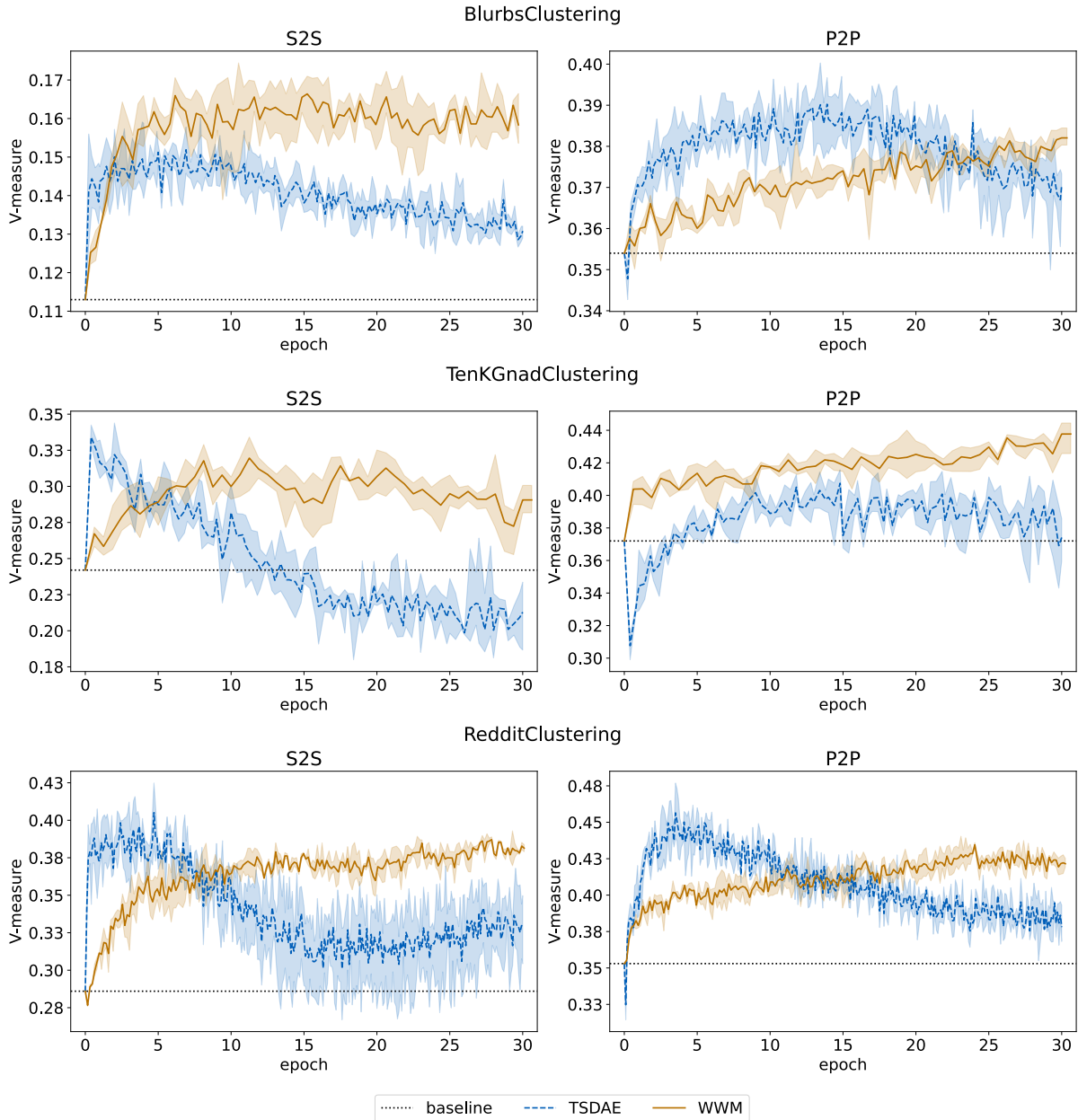


Figure 2: Change of the V-measure score with continued pre-training for GBERT-base comparing WWM and TSDAE pre-training methods for Minibatch k-Means clustering. Lines represent the average of three model runs with different seeds, and filled areas indicate minimum and maximum V-measure scores. *baseline* indicates results without additional pre-training.

ing outcomes and simplify the visual analysis simultaneously.

In total, we evaluate 12 language models. Results are mixed as there are both strong (GBERT-large, ST5-xxl) and weak (GELECTRA-base, XLM-RoBERTa-large) monolingual and multilingual models (Table 2). The selected models cover a wide range of different pre-training data, model sizes, and pre-training methods. A thorough investigation of how these factors influence clustering outcomes could build on this work.

Lastly, we experiment with adaptive pre-training for GBERT models. We show that for GBERT-base, TSDAE and WWM pre-training drastically improves the performance for short texts and relatively modestly for longer texts. Results for the larger GBERT model are inconsistent and only show improvements in one case, which we suspect is caused by a too aggressive hyperparameter configuration. This leaves room for future experimentation, which would ideally include larger datasets.



## Limitations

**Diversity of datasets** Compared to the MTEB clustering benchmark, our proposed German benchmark is less diverse. For instance, it does not contain formal writing (e.g., scientific papers). Moreover, the proposed datasets are relatively small with a maximum split size of around 26k samples (Table 1). Real-world applications may involve larger data (possibly hundreds of thousands of data samples) with a high degree of semantic variability (e.g., hundreds of topics), forcing models to perform extremely fine-grained clustering.

**Pre-training experiments** Given the relatively small training datasets, our experiments do not allow us to conclude the possible benefits of more extensive data. In the case of larger available data, longer pre-training might be beneficial. Furthermore, the experiments focus on monolingual BERT-based model architecture. Benefits of continued pre-training may differ for, e.g., multilingual models or pre-trained models with smaller pre-training datasets (such as GottBERT).

**Beneficial model properties** As discussed in Section 4, some models perform very differently, although trained on similar data. From a practical point of view, a more thorough analysis of performance-increasing factors would be helpful (i.e., model size and architecture, pre-training method, and training data). Moreover, it would also be interesting to better understand how models assess the similarity of text. This could affect how well models are suited for specific clustering tasks (e.g., how models deal with words with specific grammatical functions or unseen words).

**Large language models** The rise of generative large language models (LLMs), such as GPT-4 (OpenAI, 2023), and primarily open-source models, such as LLaMA (Touvron et al., 2023), are not represented in this work. While the benefit of generative models for NLU may not yet be fully understood, preliminary work suggests strong performance (e.g., Neelakantan et al., 2022; Muenighoff, 2022). However, this work focuses on well-established models and training techniques that can be easily used with decent resources (e.g., a single GPU) and thus benefit the open-source community the most.

## Ethics Statement

We acknowledge the ACL Code of Ethics<sup>13</sup> as an essential instrument in ensuring that research in computer science serves the public good. In the context of this work, we want to discuss our approach to share user-owned social media data responsibly. Social media has become an integral part of everyday life and an important data source in many research fields. For instance, social media can be used to address information voids during health emergencies (Boender et al., 2023). Consequently, the use of social media data in NLU research and applications has increased, rendering the inclusion of such data in this benchmark (i.e., Reddit) essential.

We use Reddit data for this benchmark for comparability (to MTEB) and because Reddit has open API access (meaning that any interested user can reproduce the published results). In fact, Reddit data is also available in large amounts without registration to the Reddit services: The Pushshift dataset (Baumgartner et al., 2020) has been collecting any public Reddit data for over a decade, providing the collected data to anyone and without any form of authentication. We believe this is a problematic approach as the data is distributed without requiring parties to accept the Reddit API terms of use. Specifically, and as per the current terms, Reddit data is owned by Reddit users (and not the platform itself), allowing users to delete accounts and content.<sup>14</sup> Specifically, the General Data Protection Regulation (GDPR), applicable to member states of the European Union (EU), mandates the right of the deletion of personal data ("the right to be forgotten").<sup>15</sup> Deliberately sharing user-owned data to anonymous parties makes it practically impossible for users to invoke their rights. Instead, data should only be obtained through the official Reddit API, which can be used to obtain and update Reddit data. Therefore, we do not disclose the raw data and instead, only share data identifiers and advise interested researchers to use the official channels.

<sup>13</sup><https://www.aclweb.org/portal/content/acl-code-ethics>

<sup>14</sup>Reddit recently updated the API terms, which became effective on June 19, 2023 (<https://www.redditinc.com/policies/data-api-terms>). The updated terms define a less permissive use of Reddit data for artificial intelligence applications, and interested researchers should carefully consider these terms. This work was performed under the old, more permissive API terms.

<sup>15</sup><https://gdpr.eu/right-to-be-forgotten>

## References

- Charu C. Aggarwal, Alexander Hinneburg, and Daniel A. Keim. 2001. On the surprising behavior of distance metrics in high dimensional space. In *Database Theory — ICDT 2001*, pages 420–434, Berlin, Heidelberg. Springer Berlin Heidelberg.
- Roei Aharoni and Yoav Goldberg. 2020. [Unsupervised domain clusters in pretrained language models](#). In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 7747–7763, Online. Association for Computational Linguistics.
- Jason Baumgartner, Savvas Zannettou, Brian Keegan, Megan Squire, and Jeremy Blackburn. 2020. [The pushshift reddit dataset](#). *Proceedings of the International AAAI Conference on Web and Social Media*, 14(1):830–839.
- Kevin Beyer, Jonathan Goldstein, Raghu Ramakrishnan, and Uri Shaft. 1999. When is “nearest neighbor” meaningful? In *Database Theory — ICDT’99*, pages 217–235, Berlin, Heidelberg. Springer Berlin Heidelberg.
- David M. Blei, Andrew Y. Ng, and Michael I. Jordan. 2003. Latent dirichlet allocation. *J. Mach. Learn. Res.*, 3:993–1022.
- T Sonia Boender, Paula Helene Schneider, Claudia Houareau, Silvan Wehrli, Tina D Purnat, Atsuyoshi Ishizumi, Elisabeth Wilhelm, Christopher Voegeli, Lothar Wieler, and Christina Leuker. 2023. [Establishing infodemic management in germany: a framework for social listening and integrated analysis to report infodemic insights at the national public health institute](#). *JMIR Infodemiology*.
- Lars Buitinck, Gilles Louppe, Mathieu Blondel, Fabian Pedregosa, Andreas Mueller, Olivier Grisel, Vlad Niculae, Peter Prettenhofer, Alexandre Gramfort, Jaques Grobler, Robert Layton, Jake VanderPlas, Arnaud Joly, Brian Holt, and Gaël Varoquaux. 2013. API design for machine learning software: experiences from the scikit-learn project. In *ECML PKDD Workshop: Languages for Data Mining and Machine Learning*, pages 108–122.
- Branden Chan, Stefan Schweter, and Timo Möller. 2020. [German’s next language model](#). In *Proceedings of the 28th International Conference on Computational Linguistics*, pages 6788–6796, Barcelona, Spain (Online). International Committee on Computational Linguistics.
- Kevin Clark, Minh-Thang Luong, Quoc V. Le, and Christopher D. Manning. 2020. [Electra: Pre-training text encoders as discriminators rather than generators](#). In *International Conference on Learning Representations*.
- Alexis Conneau and Douwe Kiela. 2018. [SentEval: An evaluation toolkit for universal sentence representations](#). In *Proceedings of the Eleventh International Conference on Language Resources and Evaluation (LREC 2018)*, Miyazaki, Japan. European Language Resources Association (ELRA).
- Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. [BERT: Pre-training of deep bidirectional transformers for language understanding](#). In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 4171–4186, Minneapolis, Minnesota. Association for Computational Linguistics.
- Cédric Févotte and Jérôme Idier. 2011. [Algorithms for nonnegative matrix factorization with the  \$\beta\$ -divergence](#). *Neural Computation*, 23(9):2421–2456.
- Tianyu Gao, Xingcheng Yao, and Danqi Chen. 2021. [SimCSE: Simple contrastive learning of sentence embeddings](#). In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 6894–6910, Online and Punta Cana, Dominican Republic. Association for Computational Linguistics.
- Maarten Grootendorst. 2022. Bertopic: Neural topic modeling with a class-based tf-idf procedure. *arXiv preprint arXiv:2203.05794*.
- Suchin Gururangan, Ana Marasović, Swabha Swayamdipta, Kyle Lo, Iz Beltagy, Doug Downey, and Noah A. Smith. 2020. [Don’t stop pretraining: Adapt language models to domains and tasks](#). In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 8342–8360, Online. Association for Computational Linguistics.
- Jeremy Howard and Sebastian Ruder. 2018. [Universal language model fine-tuning for text classification](#). In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 328–339, Melbourne, Australia. Association for Computational Linguistics.
- Haoyang Huang, Yaobo Liang, Nan Duan, Ming Gong, Linjun Shou, Daxin Jiang, and Ming Zhou. 2019. [Unicoder: A universal language encoder by pre-training with multiple cross-lingual tasks](#). In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 2485–2494, Hong Kong, China. Association for Computational Linguistics.
- Jui-Ting Huang, Ashish Sharma, Shuying Sun, Li Xia, David Zhang, Philip Pronin, Janani Padmanabhan, Giuseppe Ottaviano, and Linjun Yang. 2020. [Embedding-based retrieval in facebook search](#). In *Proceedings of the 26th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining, KDD ’20*, page 2553–2561, New York, NY, USA. Association for Computing Machinery.

- Jinhyuk Lee, Wonjin Yoon, Sungdong Kim, Donghyeon Kim, Sunkyu Kim, Chan Ho So, and Jaewoo Kang. 2019. [BioBERT: a pre-trained biomedical language representation model for biomedical text mining](#). *Bioinformatics*, 36(4):1234–1240.
- Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. 2019. Roberta: A robustly optimized bert pretraining approach. *arXiv preprint arXiv:1907.11692*.
- Leland McInnes, John Healy, and Steve Astels. 2017. hdbscan: Hierarchical density based clustering. *The Journal of Open Source Software*, 2(11):205.
- Leland McInnes, John Healy, Nathaniel Saul, and Lukas Grossberger. 2018. Umap: Uniform manifold approximation and projection. *The Journal of Open Source Software*, 3(29):861.
- Jacob Montiel, Max Halford, Saulo Martiello Mastelini, Geoffrey Bolmier, Raphael Sourty, Robin Vaysse, Adil Zouitine, Heitor Murilo Gomes, Jesse Read, Talel Abdesslem, and Albert Bifet. 2021. River: Machine learning for streaming data in python. *J. Mach. Learn. Res.*, 22(1).
- Niklas Muennighoff. 2022. Sgpt: Gpt sentence embeddings for semantic search. *arXiv preprint arXiv:2202.08904*.
- Niklas Muennighoff, Nouamane Tazi, Loic Magne, and Nils Reimers. 2023. [MTEB: Massive text embedding benchmark](#). In *Proceedings of the 17th Conference of the European Chapter of the Association for Computational Linguistics*, pages 2014–2037, Dubrovnik, Croatia. Association for Computational Linguistics.
- Arvind Neelakantan, Tao Xu, Raul Puri, Alec Radford, Jesse Michael Han, Jerry Tworek, Qiming Yuan, Nikolas Tezak, Jong Wook Kim, Chris Hallacy, et al. 2022. Text and code embeddings by contrastive pre-training. *arXiv preprint arXiv:2201.10005*.
- Jianmo Ni, Gustavo Hernandez Abrego, Noah Constant, Ji Ma, Keith Hall, Daniel Cer, and Yinfei Yang. 2022. [Sentence-t5: Scalable sentence encoders from pre-trained text-to-text models](#). In *Findings of the Association for Computational Linguistics: ACL 2022*, pages 1864–1874, Dublin, Ireland. Association for Computational Linguistics.
- OpenAI. 2023. Gpt-4 technical report. *arXiv preprint arXiv:2303.08774*.
- Pedro Javier Ortiz Suárez, Benoît Sagot, and Laurent Romary. 2019. [Asynchronous Pipeline for Processing Huge Corpora on Medium to Low Resource Infrastructures](#). In *7th Workshop on the Challenges in the Management of Large Corpora (CMLC-7)*, Cardiff, United Kingdom. Leibniz-Institut für Deutsche Sprache.
- Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, and Peter J. Liu. 2020. Exploring the limits of transfer learning with a unified text-to-text transformer. *J. Mach. Learn. Res.*, 21(1).
- Nils Reimers and Iryna Gurevych. 2019. [Sentence-BERT: Sentence embeddings using Siamese BERT-networks](#). In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 3982–3992, Hong Kong, China. Association for Computational Linguistics.
- Nils Reimers and Iryna Gurevych. 2020. [Making monolingual sentence embeddings multilingual using knowledge distillation](#). In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 4512–4525, Online. Association for Computational Linguistics.
- Steffen Remus, Rami Aly, and Chris Biemann. 2019. Germeval 2019 task 1: Hierarchical classification of blurbs. In *Proceedings of the 15th Conference on Natural Language Processing (KONVENS 2019)*, pages 280–292, Erlangen, Germany. German Society for Computational Linguistics & Language Technology.
- Andrew Rosenberg and Julia Hirschberg. 2007. [V-measure: A conditional entropy-based external cluster evaluation measure](#). In *Proceedings of the 2007 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning (EMNLP-CoNLL)*, pages 410–420, Prague, Czech Republic. Association for Computational Linguistics.
- Phillip Rust, Jonas Pfeiffer, Ivan Vulić, Sebastian Ruder, and Iryna Gurevych. 2021. [How good is your tokenizer? on the monolingual performance of multilingual language models](#). In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 3118–3135, Online. Association for Computational Linguistics.
- Dietmar Schabus, Marcin Skowron, and Martin Trapp. 2017. [One million posts: A data set of german online discussions](#). In *Proceedings of the 40th International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR)*, pages 1241–1244, Tokyo, Japan.
- Raphael Scheible, Fabian Thomczyk, Patric Tippmann, Victor Jaravine, and Martin Boeker. 2020. [Gottbert: a pure german language model](#). *arXiv preprint arXiv:2012.02110*.
- Hugo Touvron, Thibaut Lavril, Gautier Izacard, Xavier Martinet, Marie-Anne Lachaux, Timothée Lacroix, Baptiste Rozière, Naman Goyal, Eric Hambro, Faisal Azhar, Aurelien Rodriguez, Armand Joulin, Edouard

- Grave, and Guillaume Lample. 2023. [Llama: Open and efficient foundation language models](#).
- Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. 2017. [Attention is all you need](#). In *Advances in Neural Information Processing Systems*, volume 30. Curran Associates, Inc.
- Henrik Voigt, Monique Meuschke, Sina Zarrieß, and Kai Lawonn. 2022. [KeywordScape: Visual document exploration using contextualized keyword embeddings](#). In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, pages 137–147, Abu Dhabi, UAE. Association for Computational Linguistics.
- Kexin Wang, Nils Reimers, and Iryna Gurevych. 2021. [TSDAE: Using transformer-based sequential denoising auto-encoder for unsupervised sentence embedding learning](#). In *Findings of the Association for Computational Linguistics: EMNLP 2021*, pages 671–688, Punta Cana, Dominican Republic. Association for Computational Linguistics.
- Guillaume Wenzek, Marie-Anne Lachaux, Alexis Conneau, Vishrav Chaudhary, Francisco Guzmán, Armand Joulin, and Edouard Grave. 2020. [CCNet: Extracting high quality monolingual datasets from web crawl data](#). In *Proceedings of the Twelfth Language Resources and Evaluation Conference*, pages 4003–4012, Marseille, France. European Language Resources Association.
- Ziyi Yang, Yinfei Yang, Daniel Cer, Jax Law, and Eric Darve. 2021. [Universal sentence representation learning with conditional masked language model](#). In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 6216–6228, Online and Punta Cana, Dominican Republic. Association for Computational Linguistics.
- Andrew Yates, Rodrigo Nogueira, and Jimmy Lin. 2021. [Pretrained transformers for text ranking: BERT and beyond](#). In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies: Tutorials*, pages 1–4, Online. Association for Computational Linguistics.
- He Zhao, Dinh Phung, Viet Huynh, Yuan Jin, Lan Du, and Wray Buntine. 2021. [Topic modelling meets deep neural networks: A survey](#). In *Proceedings of the Thirtieth International Joint Conference on Artificial Intelligence, IJCAI-21*, page 4713–4720. International Joint Conferences on Artificial Intelligence Organization. Survey Track.

## A Additional Results

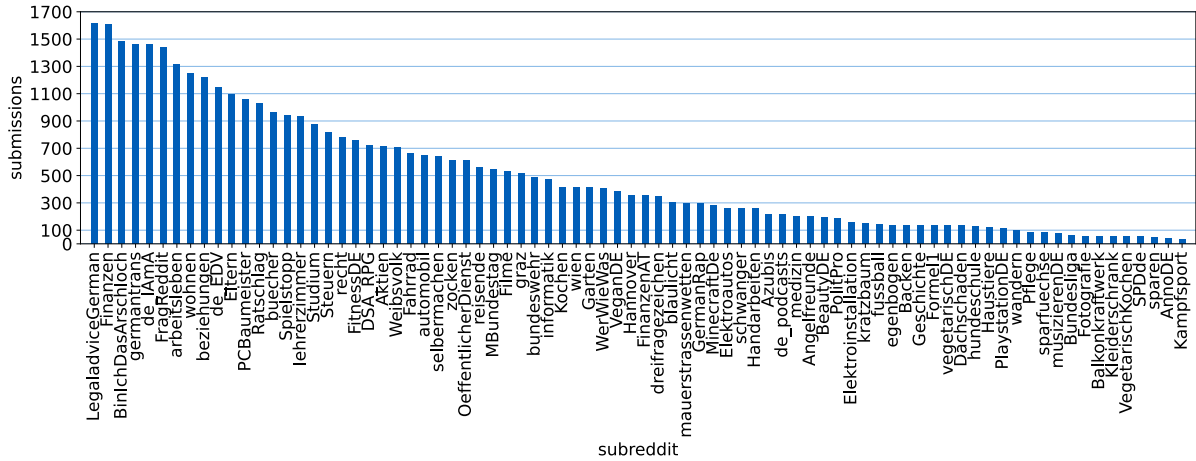


Figure 3: Distribution of German Reddit dataset used for RedditClusteringS2S and RedditClusteringP2P.

Model	Hugging Face Repository
GBERT-base	<a href="https://huggingface.co/deepset/gbert-base">https://huggingface.co/deepset/gbert-base</a>
GBERT-large	<a href="https://huggingface.co/deepset/gbert-large">https://huggingface.co/deepset/gbert-large</a>
GELECTRA-base	<a href="https://huggingface.co/deepset/gelectra-base">https://huggingface.co/deepset/gelectra-base</a>
GELECTRA-large	<a href="https://huggingface.co/deepset/gelectra-large">https://huggingface.co/deepset/gelectra-large</a>
GottBERT	<a href="https://huggingface.co/uklfr/gottbert-base">https://huggingface.co/uklfr/gottbert-base</a>
MiniLM-L12-v2-ml	<a href="https://huggingface.co/sentence-transformers/paraphrase-multilingual-MiniLM-L12-v2">https://huggingface.co/sentence-transformers/paraphrase-multilingual-MiniLM-L12-v2</a>
MPNet-base-v2-ml	<a href="https://huggingface.co/sentence-transformers/paraphrase-multilingual-mpnet-base-v2">https://huggingface.co/sentence-transformers/paraphrase-multilingual-mpnet-base-v2</a>
SRoBERTa-cross	<a href="https://huggingface.co/T-Systems-onsite/cross-en-de-roberta-sentence-transformer">https://huggingface.co/T-Systems-onsite/cross-en-de-roberta-sentence-transformer</a>
USE-CMLM-ml	<a href="https://huggingface.co/sentence-transformers/use-cnml-multilingual">https://huggingface.co/sentence-transformers/use-cnml-multilingual</a>
ST5-base	<a href="https://huggingface.co/sentence-transformers/sentence-t5-base">https://huggingface.co/sentence-transformers/sentence-t5-base</a>
ST5-xxl	<a href="https://huggingface.co/sentence-transformers/sentence-t5-xxl">https://huggingface.co/sentence-transformers/sentence-t5-xxl</a>
XLM-RoBERTa-large	<a href="https://huggingface.co/xlm-roberta-large">https://huggingface.co/xlm-roberta-large</a>

Table 4: Hugging Face repositories for all evaluated language models.

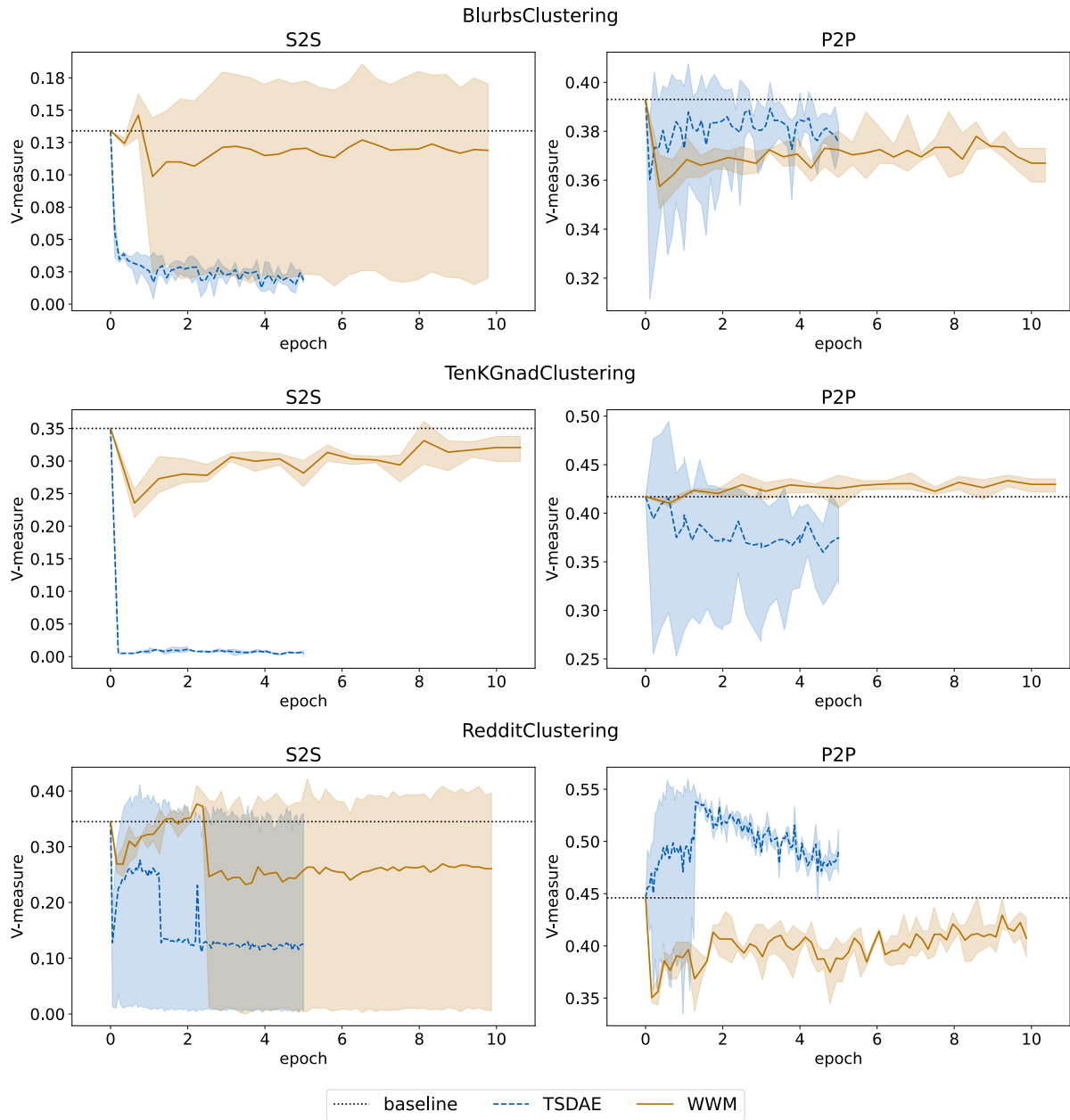


Figure 4: Change of the V-measure score with continued pre-training for GBERT-large comparing WWM and TSDAE pre-training methods for Minibatch k-Means clustering. Lines represent the average of three model runs with different seeds, and filled areas indicate minimum and maximum V-measure scores. *baseline* indicates results without additional pre-training.

Method	Blurbs			TenKGnad			Reddit													
	SD	S2S Avg.	$\Delta$	SD	S2S Avg.	$\Delta$	SD	S2S Avg.	$\Delta$	SD	P2P Avg.	$\Delta$	Avg.	$\Delta$						
<b>base</b>																				
TSDAE (1)	0.46	14.37	+3.10	0.29	37.08	+1.72	1.40	32.87	+8.64	0.79	34.73	-2.43	0.42	37.29	+8.72	0.33	39.03	+3.73	32.56	+3.91
WWM (10)	0.80	15.71	+4.44	0.41	36.83	+1.47	1.12	30.81	+6.58	0.27	41.47	+4.31	0.53	36.48	+7.91	0.87	40.90	+5.60	33.70	+5.05
WWM (30)		n/a		0.18	38.20	+2.84		n/a		0.85	43.77	+6.61		n/a		0.25	42.16	+6.86	41.38	+12.73
<b>large</b>																				
TSDAE (1)	0.87	2.57	-10.81	1.35	38.48	-0.82	0.36	0.82	-34.15	7.47	35.60	-6.09	17.63	25.90	-8.57	8.65	48.89	+4.28	25.38	-9.36
WWM (10)	6.98	11.89	-1.49	0.57	36.70	-2.60	1.60	32.07	-2.90	0.58	42.98	+1.29	17.99	26.04	-8.43	1.53	40.74	-3.87	31.74	-3.00

Table 5: V-measure scores for GBERT-base (**base**) and GBERT-large (**large**) with continued pre-training for TSDAE and WWM methods. Evaluated is based on Minibatch k-Means clustering, and reported V-measure scores are multiplied by 100. Results are multiplied by 100 and rounded to two decimals. Numbers in brackets denote the number of epochs. Avg. represents the average of three training runs with different seeds and SD the standard deviation of V-measures.  $\Delta$  reports the absolute difference to the baseline, i.e., the model results without additional pre-training.