

# Domain and Task Adaptive Pretraining for Language Models

Leonard Konle<sup>a</sup>, Fotis Jannidis<sup>a</sup>

<sup>a</sup>Würzburg University, Germany

## Abstract

All current state-of-the-art systems in NLP utilize transformer based language models trained on massive amounts of text. This paper discusses strategies to adapt these models to historical domains and tasks, typical for research in the Computational Humanities. Using two task-specific corpora from the same domain (literary texts from the 19th Century) and Bert [6] resp. Distilbert [22] as baselines, we can confirm results from a recent study that continuing pretraining on the domain and the task data substantially improves task performance. Training a model from scratch using Electra [5] is not competitive for our data sets.

## Keywords

NLP, Machine Learning

## 1. Introduction

A typical task in Computational Humanities may look like this: we want to create social networks based on character appearances within novels. As a data basis we need all character mentions within the text. This requirement seems easy to fulfill given the large number of tools for Named Entity Recognition, and in fact, most available systems will produce acceptable results. But no matter how sophisticated these models are and what results they achieve in common benchmarks, there will always be a loss of accuracy due to the difference between training data consisting of contemporary language (e.g. news) and our domain of interest (e.g. literature or historical language).

If we want to improve our data basis for social networks further, it is also necessary to distinguish whether a figure is actually present or only mentioned in dialogue. For this reason, we need a second model capable of identifying different forms of speech rendition. It is more challenging to identify a suitable model for such a task since certain types of speech, such as free indirect speech, typically only appear in literature.

In the past, such problems were solved by creating training data in elaborate annotation projects. But the use of pretrained models has changed the landscape of natural language processing in recent years. While traditional approaches like SVM or Linear Regression only require the annotated data set for training, the two-step methods used today consist of a pretraining step on large amounts of text data without annotations and a subsequent fine-tuning step, which describes the training on the actual task (see fig. 1). Using pretrained models like Bert [6] allows to work with rich semantic (and syntactic and probably even general

---


*CHR 2020: Workshop on Computational Humanities Research, November 18–20, 2020, Amsterdam, The Netherlands*

✉ leonard.konle@uni-wuerzburg.de (L. Konle); fotis.jannidis@uni-wuerzburg.de (F. Jannidis)

🆔 0000-0001-6944-6113 (F. Jannidis)

© 2020 Copyright for this paper by its authors.

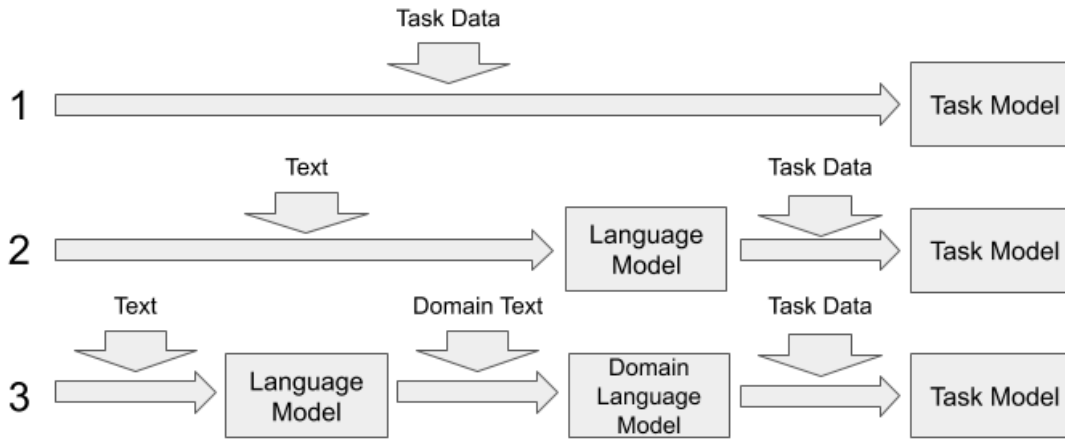
Use permitted under Creative Commons License Attribution 4.0 International (CC BY 4.0).

 CEUR Workshop Proceedings (CEUR-WS.org)

knowledge) representations, which considerably improves every natural language processing task. The usual approach supported by libraries like Huggingface [26] is to pretrain a model from scratch on a very large corpus and finetune it on a much smaller collection for a specific task. Huggingface also provides a collection of pretrained models in many languages that can be finetuned for various standard tasks in a few lines of code. But their use in the Computational Humanities context is hindered by the lack of historical text corpora to train them from scratch.

There is an important difference between machine learning as it is usually done and the kind of setup we discuss here: While we have - as usual - only labels for a small set of our population, we have a closed set of data as our population and we have access to this set. We do not expect our model to handle yet unseen instances, but we try to achieve the best results for those instances we have in our corpus/population. This has consequences for our attitude towards aspects like overfitting and forgetting.

Our paper explores different setups to achieve the best results for this kind of setup. Encouraged by the paper “Don’t stop Pretraining” [8] on domain adaptation via continued pretraining we adopt their experimental design and apply it on two datasets exemplary for Computational Humanities.



**Figure 1:** Training schemes. 1: Traditional approaches like SVM with one training step on the task dataset. 2: Bert-like language model approach with one pretraining step on unlabeled texts and a fine-tuning step on task data. 3: Bert-like approach with two consecutive pretraining steps on unlabeled text and text from the task domain and a fine-tuning step on task data.

## 2. Domain Adaptation

Due to the sheer mass of text data required to generate a language model, its foundation is often a heterogeneous collection of large, publicly available domains (Newspaper, Wikipedia, etc.). On the one hand, this results in a general language representation and is not limited to these domains, as is confirmed by the performance of these models on other text types. On the other hand, studies show that substantially better results are achieved if the text type of a task has already been present in pretraining [25]. Since the generation of a new language model enriched with texts of a domain makes neither economic nor ecological sense, the continuation of pretraining an already existing model seems to be a promising alternative [14].

## 3. Task Description

### 3.1. Named Entity Recognition

The German Corpus of Novels (DROC) [13] brings together 90 annotated fragments of German-language novels (each containing about 200 sentences) mainly from the 19th century. While It contains over 50,000 manually created annotations on named entities, coreference, direct speech, as well as speakers and addressees of this direct speech, we just use the tags for named entities and appellatives (like the ‘butcher’ or the ‘ugly’) as training data for our finetuning step. This kind of task is used in the context of computational literary studies if all mentions of characters are to be extracted (with the exception of pronouns), for example in the context of character networks.

### 3.2. Speech Rendition

The second task on Speech Rendition uses the labeled corpus “Redewiedergabe“ [2], a collection of segments and full documents from literary texts, news and journal texts, all from the 19th Century annotated with four types of speech: direct, indirect, free indirect, and reported. We only use those segments categorized as fictional and formalized the task to a token classification with five possible classes (including non-speech). In Computational Literary Studies it is often important to know, whether some part of the text, for example the evaluative description of a character, can be attributed directly in the form of direct speech or indirectly, for example in the form of reported speech, to the narrator and therefore is reliable (if the narrator is reliable) or to another character, and therefore tells the reader at least as much about the speaker as about the object. On average fiction consists of 20 to 30% of direct speech; add to that the large amount of other forms of speech rendition and it becomes clear that an astonishingly large part of fiction is the communication of communication.

## 4. Related Research

Domain adaptation via continued pretraining is a frequently focused subarea in NLP research centered around BERT-like language models. The general benefit of adaptation has been shown in diverse domains, most prominently medicine [16, 10, 1] and biology [14, 18, 7].

All recent approaches on Named Entity Recognition [15] and Speech Rendition make use of pretrained Language Models. Brunner et al. (2020) [3] train a model for Speech Rendition with custom fastText and FLAIR embeddings with an additional CRF-Layer which is superior to a generic German BERT model. Since this paper’s objective is not about finding the best model for this task, but to show the effect of continued language modeling, we will focus on BERT-based approaches.

## 5. Corpora

We use a freely available German BERT Model from the Huggingface modelhub trained on the large text collection Corpus *B*. The texts from this collection come from various resources, mainly webcrawls and Wikipedia. In contrast, we composed Corpus *S* and used just texts that either come from the same domain as our task data (narratives) or their period (19th century). Corpus *D* is a subset of Corpus *S* containing only texts matching both time period

and domain with our tasks. Corpus  $T$  and  $R$  are the two corpora discussed above in section ‘task description’ used to finetune models but are also listed here because they are also used without labels for pretraining.

**Table 1**  
Corpora characteristics

	Content	Token
Corpus $B$	Wikipedia dump, EU Bookshop corpus, Open Subtitles, CommonCrawl, ParaCrawl and News Crawl	2350M
Corpus $S$	Fiction and newspapers from the 19th and 20th Century	1138M
Corpus $D$	19th Century fiction	739M
Corpus $T$	19th Century fiction from the DROC Corpus [13]	140k
Corpus $R$	19th Century fiction from the Speech Rendition Corpus [2]	130k

### 5.1. Domain Similarity

From the history of the German language we know that though the difference between German today and the 19th Century is not as big as compared to Early High German, there is nevertheless a lot of systematic differences on all levels of the language system like inflectional morphology, verbal inflection, morphosyntax, tempus use of verbs, syntax and more [19]. In the 19th Century, before the language crisis around 1900, the language of literature usually followed the precepts of the middle style (genus medium), which was, however, enriched with pathos, quotes and rhetorical decoration. Since the language crisis, on the other hand, literary German has been difficult to describe as a common unit, but is something specific to certain groups or individuals. At the same time, it often defines itself in a frontal position against the everyday style [20].

To gain an insight how large the difference between our general language Corpus  $B$  and the domain specific corpora  $D, T$  and  $R$  are, we take the 10.000 most frequent words of the task datasets (Corpus  $T$  and Corpus  $R$ ) and draw samples from the domain adaptation data and the data used to train Bert and DistilBert from scratch (Corpus  $D$  and Corpus  $B$ ). The samples hold the same number of different documents and the same document length as the task datasets they are compared with. We sampled 20 times and calculated the vocabulary overlap in the 10.000 most frequent words. This simple approach is sufficient in our binary setting, but in other cases more distinguished measures [24] for domain similarity should be used. The results from Table 2 confirms the assumption that Corpus  $T$  and  $R$  are more similar to Corpus  $D$  than to Corpus  $B$ .

**Table 2**  
Domain Similarity measured by shared vocabulary of 10.000 most frequent words.

	Corpus $D$	Corpus $B$
Corpus $T$	.537 (.016)	.295 (.003)
Corpus $R$	.459 (.024)	.224 (.002)

## 6. Methods

We use BERT<sup>1</sup> [6] trained on german texts from diverse domains (Corpus *B*) as a baseline for both tasks. In addition, we test Distilbert<sup>2</sup> [22] based on the mentioned BERT model. This model is expected to perform slightly worse than BERT itself but is handy due to its small size, which results in faster computation time (pretraining and fine-tuning) and lower hardware requirements.

The setup of our experiments includes a two-step procedure, which in the first step, as proposed by Gururangan et al. (2020) [8] continues the pretraining of a model on texts of the target domain (domain adaptive pretraining, see Fig.1), on the texts of the downstream task itself (task adaptive pretraining), or both. In the second step, the pretrained models are finetuned for two different tasks: Named Entity Recognition and Speech Rendition).

As an alternate approach, we train the recently introduced ELECTRA [5] from scratch. We choose ELECTRA because its pretraining task requires less time and data and it is supposed to achieve BERT-like results. (Code available on Github<sup>3</sup>) The finetuning process is in all setups performed over 10 epochs.

### 6.1. Pre-Training Parameters

Domain adaptive pretraining is performed by iterating 5 epochs over Corpus *D* with a learning rate of 1e-4. Task adaptive pretraining is performed by iterating 50 epochs over Corpus *T* with the same learning rate. For the training from scratch, we use Corpus *S* to train a new ELECTRA instance. We adopt the settings of ELECTRA-small from its Github Repository<sup>4</sup> and trained for 1.6Mio Steps with a batchsize of 64 over 20Mio examples. ELECTRA is used as an alternate approach where all labeled data and texts from the relevant domain are used to perform one large pretraining step. For this reason, ELECTRA is not used for task or domain adaptations.

## 7. Results

Table 3 shows the results for pretraining Bert and DistilBert on domain and task specific texts or both and includes the baseline for both tasks. We use approximate randomized testing [17] recommended by Jurafsky (2020) [11] to verify our results. The combined approach of Task and Domain Adaptation shows a significant improvement in both NER ( $a < .0001$ , 10k samples) and Speech Rendition ( $a < .01$ , 10k samples) compared to the baseline.

## 8. Discussion

Our results show that continuing pretraining improves performance substantially and should be integrated into machine learning tasks. Domain and task adaptation show positive effects and even better results are achieved with a combination of both.

At the moment pretraining existing models, in terms of computation effort and prediction quality, seems to be more effective than training models from scratch. While it will not always

---

<sup>1</sup>Model page: <https://huggingface.co/bert-base-german-dbmdz-cased>

<sup>2</sup>Model page: <https://huggingface.co/distilbert-base-german-cased>

<sup>3</sup>Paper Repository: [https://github.com/fotisj/efficient\\_training](https://github.com/fotisj/efficient_training)

<sup>4</sup><https://github.com/google-research/electra>

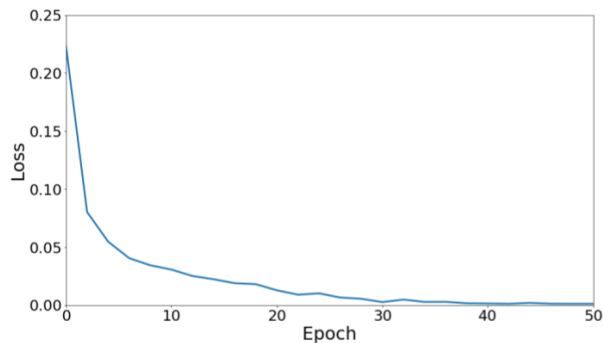
**Table 3**

Results of 10-fold cross-validation on the DROC Named Entity Recognition Task per model. Base refers to models finetuned without additional pretraining steps. Domain models are pretrained on Corpus *D* and Task models on either Corpus *T* or Corpus *R*. Task+Domain refers to the combination of both.

DROC			
Model	Bert	DistilBert	Electra
Base F1 (std)	.859 (.032)	.838 (.32)	.791 (0.03)
Domain	.863 (.041)	.854 (.028)	n.a.
Task	.882 (.039)	.853 (.031)	n.a.
Task+Domain	.927 (.029)	.893 (.043)	n.a.
Speech Rendition			
Model	Bert	DistilBert	Electra
Base F1 (std)	.538 (.086)	.455 (.04)	.373 (.057)
Domain	.637 (.234)	.557 (.356)	n.a.
Task	.587 (.403)	.568 (.18)	n.a.
Task+Domain	.691 (.136)	.655 (.132)	n.a.

be an option to perform a domain adaptation due to the lack of suitable data, task adaptation can be performed in every setup.

Research on distilling language models shows that it is possible to create smaller versions of large pretrained models with minor performance loss [21]. We show that adaptation steps can enhance those models (DistilBert) to produce results comparable to their teacher models (BERT).

**Figure 2:** Evaluation loss for every epoch in pretraining BERT with Corpus *T*

From the perspective of traditional machine learning paradigms, 50 iterations on a small chunk of data might seem like a dangerous case of overfitting. But in this setup, the model cannot memorize the correct solution, but only build a better representation of the task data. The language model is basically overfitted on the task data (see fig.2), and the finetuned model performs well on this basis. We assume that this setup may be specifically helpful in the context of Computational Humanities research with limited and closed data sets. A typical danger in the application of ML in companies and similar contexts is that new data is drawn from a different distribution and therefore, the performance of the model decreases or it fails.

In the context of closed data sets it is feasible to pretrain on the task data to achieve the best performance.

## 9. Future Work

In our experiment, the task-specific corpus represented the collection a researcher wants to do prediction on. Following experiments will test what happens when we enlarge this collection. The overfitting of the language model should still be helpful for the task on a similar level, but this probably depends on questions like the size of the collection and its similarity to the task corpus.

This paper applied a rather straightforward domain adaptation considering just one collection of texts (Corpus  $D$ ) from the same domain as the task. But in many cases, specifically, when dealing with older texts or rare text types, there is no corpus large enough fulfilling both the text type and the time constraint. For this reason, we will explore domain adaptation with multiple corpora and how they should be arranged in size, training time, and order. An adaptation of a model trained from Wikipedia to baroque poems could be done by training on a large collection of poems and in a second step on as many baroque texts as possible or vice-versa. Likewise promising is the idea to train a model from contemporary language back to a certain point in literary history by pretraining on slices of 10-50 years. Such a fine-grained selection would lead to settings with very small datasets and raises the question if these can still be used for pretraining and the relation between performance gain and corpus size in general.

Another issue we came across is early stopping. When training a model from scratch additional steps through training examples will always enhance performance with smaller effects the longer a training continues, because the learning rate decreases. In such a setting early stopping is used to stop the process at a point where the performance gain can no longer justify the computational effort. In this small study, we simply adopted the number of trained epochs from Gururangan et al. (2020) [8], but the better approach would be to bind the early stopping mechanism to performance gain at the task we actually try to adapt to. On the other hand, this would increase training time, because it requires the execution of the whole task training for every epoch or  $n$  steps. This would add an unpleasant complexity, especially when considering training task hyperparameters and multiple pretraining corpora.

All of these decisions need to face the fact that neural language models can forget data they have been trained on a few steps ago. This phenomenon is referred to as catastrophic forgetting [12] and there are techniques to reduce memory loss like negative transfer [4], hard attention [23], weight freezing [9] or tuning the learning rate decay [27], these approaches lower forgetting in general, but show mixed results for specific datasets and need to be tested under low resource domain adaptation settings.

These tests should be preceded by considerations on the measurement of domain similarity, since, in addition to hyperparameters and the amount of available text, these are crucial for the generalization of results and best practice recommendations.

The receptivity of a model plays a special role for task adaptation scenarios and needs further research: if a model is adapted to a labeled dataset over several epochs and shows a strong performance there, it is necessary to include the research dataset (see Discussion) with the same intensity in the pretraining to make sure predictions there meet the expected quality. We can outline two possible scenarios: The first and more simple approach would include the complete

research data in the task adaptation step. But if this leads to decreasing performance, because the model is not capable to “remember” the whole dataset, we would propose to move a blocking window over the research data, add its content to task adaptation, perform a prediction after training and move on to the next segment. The efforts to stop forgetting in neural networks are strongly motivated by the idea of lifelong learning models, capable of solving every task they have been trained on no matter how many steps ago. This idea is not only relevant to domain adaptation but also to task adaptation, which opens up a parallel field with similar questions. Can the knowledge obtained in a NER Task on newspapers passed through domain adaptation pretraining still help solve the same task in the new domain? And if this is the case, does it have to be the same task or is a similar task sufficient (e.g. POS Tagging and Dependency Parsing or Sentiment and Emotion Analysis)?

Most of our measured results just differ in small numbers, a finding prevalent in machine learning research and language model approaches in particular. To give at least some idea of what is an effect of pretraining and what is just random fluctuation we decided to use cross-validation and provide standard deviation of evaluation scores. This is more reliable than single values and prevents cherry-picking best results, but still does not solve the problem in general.

## References

- [1] E. Alsentzer et al. “Publicly available clinical BERT embeddings”. In: *arXiv preprint arXiv:1904.03323* (2019).
- [2] A. Brunner et al. “Das Redewiedergabe-Korpus. Eine neue Ressource”. In: *DHd 2019 Conference Abstracts*. 2019, pp. 103–106. DOI: <https://doi.org/10.5281/zenodo.2600812>.
- [3] A. Brunner et al. “To BERT or not to BERT - Comparing Contextual Embeddings in a Deep Learning Architecture for the Automatic Recognition of four Types of Speech, Thought and Writing Representation”. In: *Proceedings of the 5th Swiss Text Analytics Conference and the 16th Conference on Natural Language Processing, SwissText/KONVENS 2020, Zurich, Switzerland, June 23-25, 2020 [online only]*. Ed. by S. Ebling et al. Vol. 2624. CEUR Workshop Proceedings. CEUR-WS.org, 2020, pp. 29–40. URL: <http://ceur-ws.org/Vol-2624/paper5.pdf>.
- [4] X. Chen et al. “Catastrophic forgetting meets negative transfer: Batch spectral shrinkage for safe transfer learning”. In: *Advances in Neural Information Processing Systems*. 2019, pp. 1908–1918.
- [5] K. Clark et al. “Electra: Pre-training text encoders as discriminators rather than generators”. In: *arXiv preprint arXiv:2003.10555* (2020).
- [6] J. Devlin et al. “Bert: Pre-training of deep bidirectional transformers for language understanding”. In: *arXiv preprint arXiv:1810.04805* (2018).
- [7] Y. Du et al. “Deep scaled dot-product attention based domain adaptation model for biomedical question answering”. In: *Methods* 173 (2020), pp. 69–74.
- [8] S. Gururangan et al. “Don’t Stop Pretraining: Adapt Language Models to Domains and Tasks”. In: *arXiv preprint arXiv:2004.10964* (2020).
- [9] J. Howard and S. Ruder. “Universal language model fine-tuning for text classification”. In: *arXiv preprint arXiv:1801.06146* (2018).



- [10] B. Hur et al. “Domain Adaptation and Instance Selection for Disease Syndrome Classification over Veterinary Clinical Notes”. In: *Proceedings of the 19th SIGBioMed Workshop on Biomedical Language Processing*. 2020, pp. 156–166.
- [11] D. Jurafsky and J. H. Martin. *Speech and Language Processing (3rd ed. Draft)*. 2020.
- [12] J. Kirkpatrick et al. “Overcoming catastrophic forgetting in neural networks”. In: *Proceedings of the national academy of sciences* 114.13 (2017), pp. 3521–3526.
- [13] M. Krug et al. “Description of a Corpus of Character References in German Novels - DROC [Deutsches ROman Corpus]”. In: *DARIAH Working Paper 27* (2018).
- [14] J. Lee et al. “BioBERT: a pre-trained biomedical language representation model for biomedical text mining”. In: *Bioinformatics* 36.4 (2020), pp. 1234–1240.
- [15] J. Li et al. “A survey on deep learning for named entity recognition”. In: *IEEE Transactions on Knowledge and Data Engineering* (2020).
- [16] C. Lin et al. “Does BERT need domain adaptation for clinical negation detection?” In: *Journal of the American Medical Informatics Association* 27.4 (2020), pp. 584–591.
- [17] E. W. Noreen. *Computer Intensive Methods for Testing Hypothesis*. Wiley New York, 1989.
- [18] N. Poerner, U. Waltinger, and H. Schütze. “Inexpensive Domain Adaptation of Pre-trained Language Models: A Case Study on Biomedical Named Entity Recognition”. In: *arXiv preprint arXiv:2004.03354* (2020).
- [19] P. von Polenz. “Deutsche Sprachgeschichte Bd. III 19. und 20. Jahrhundert”. In: Berlin, New York, 1999, pp. 338–390.
- [20] P. von Polenz. “Deutsche Sprachgeschichte Bd. III 19. und 20. Jahrhundert”. In: Berlin, New York, 1999, pp. 473–484.
- [21] H. Sajjad et al. “Poor Man’s BERT: Smaller and Faster Transformer Models”. In: *arXiv preprint arXiv:2004.03844* (2020).
- [22] V. Sanh et al. “DistilBERT, a distilled version of BERT: smaller, faster, cheaper and lighter”. In: *arXiv preprint arXiv:1910.01108* (2019).
- [23] J. Serra et al. “Overcoming catastrophic forgetting with hard attention to the task”. In: *arXiv preprint arXiv:1801.01423* (2018).
- [24] V. Van Asch and W. Daelemans. “Using Domain Similarity for Performance Estimation”. In: *Proceedings of the 2010 Workshop on Domain Adaptation for Natural Language Processing*. Uppsala, Sweden: Association for Computational Linguistics, July 2010, pp. 31–36. URL: <https://www.aclweb.org/anthology/W10-2605>.
- [25] A. Wang et al. “Superglue: A stickier benchmark for general-purpose language understanding systems”. In: *Advances in Neural Information Processing Systems*. 2019, pp. 3266–3280.
- [26] T. Wolf et al. “HuggingFace’s Transformers: State-of-the-art Natural Language Processing”. In: *ArXiv* (2019), arXiv–1910.
- [27] Y. Xu et al. “Forget Me Not: Reducing Catastrophic Forgetting for Domain Adaptation in Reading Comprehension”. In: *arXiv preprint arXiv:1911.00202* (2019).