Applied Computing and Intelligence

https://www.aimspress.com/journal/aci

Applied Computing and Intelligence
6(2): 245–264.
DOI: 10.3934/aci.2026014
Received: 13 May 2026
Revised: 24 September 2026
Accepted: 24 September 2026
Published: 06 November 2026

Research article

Continued pretraining of FinBERT on Finnish histopathological reports: train-time signals and proxy downstream correlations

Rami Luisto[1,2,3,*], Liisa Petäinen[1], Tommi Grönholm[1], Jan Böhm[4], Maarit Ahtiainen[4], Tomi Lilja[4], Ilkka Pölönen[1] and Sami Äyrämö[1,5]

[1] Faculty of Information Technology, University of Jyväskylä, Jyväskylä, Finland
[2] Heart and Lung Center, Helsinki University Hospital, Helsinki, Finland
[3] Qutwo, Helsinki, Finland
[4] Central Finland Biobank, Jyväskylä, Finland
[5] Wellbeing Services County of Central Finland, Jyväskylä, Finland
* **Correspondence:** Email: rami.m.luisto@jyu.fi.

Academic Editor: Pasi Fränti

Abstract: In natural language processing (NLP) classification tasks where a lack of labeled data is an issue, continued pretraining (CPT) of transformer models on unlabeled data is an established approach. In this paper, we had two aims. (1) We described our observations from continued pretraining of the Finnish BERT transformer model (FinBERT) on a Finnish histopathological dataset (below, *the Histopathology data*). (2) Since the Histopathology data had no classification labels, we gathered public Finnish datasets as proxy data to analyze whether the signals observed in (1) were associated with downstream classification gains. We observed that CPT train-time loss curves differed strongly by domain, and that, in an exploratory analysis, certain CPT-derived features correlated with proxy classification improvement. In particular, this report contributes to the limited literature on NLP for Finnish healthcare data.

Keywords: continued pretraining; FinBERT; Finnish NLP; medical NLP; masked language modeling; representation similarity

1. Introduction

Our driving motivation here is Natural Language Processing (NLP) work in the healthcare sector, in particular with regard to Finnish medical text data. Finnish is a low-resource language with complex grammatical structures, and is therefore at a disadvantage in the NLP field when compared to larger languages. In particular, off-the-shelf multilingual models or translation-based approaches often fall

short [32]. Furthermore, a specific feature of any artificial intelligence (AI) work in the healthcare sector is that datasets are delicate and heavily regulated [3, 19]. It is not uncommon in a healthcare project to end up with only a partial dataset, e.g., only text without labels, either because of a mistaken data request or because labeling requires expert medical knowledge. The process of acquiring these labels can then take months of calendar time. In this project, we were interested in seeing whether we could make good use of such waiting time.

More specifically, our aim was to do a continued pretraining run on a histopathological dataset, obtained from the Central Finland Biobank, for which we lacked the means to obtain class labels. In this paper, we present our results on such continued pretraining and its effects. For contextualization and hypothesis generation, we ran a similar training with public datasets as well. We discuss in Section 4 what we were able to observe of the models during and after the training; the central observation is in Figure 1.

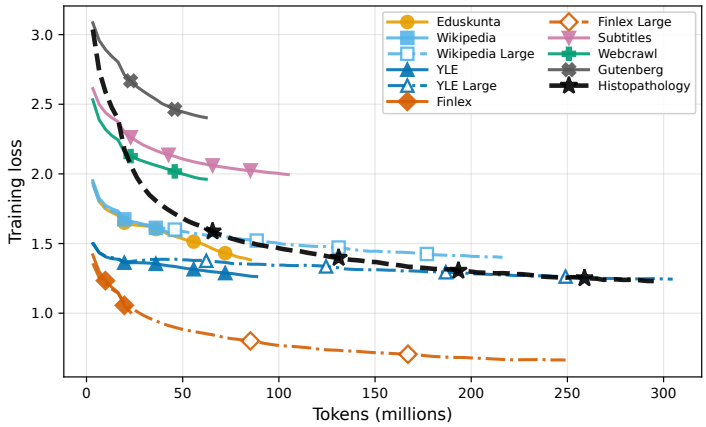


Figure 1. Training loss during continued pretraining of FinBERT on various datasets it is more or less "familiar" with. Markers represent epoch boundaries.

Besides continued pretraining, we ran classification tasks on the datasets where labels were available. Our aim was to generate hypotheses on possible correlations between train-time signals and classification performance. We found a few tentative associations, though a replication study is needed.

Modern NLP tends to separate the training of a language model into stages of increasing specialization. Typically, training starts with self-supervised pretraining on a large text corpus followed by possible domain fine-tuning and then task-specific training. Though this paradigm has no singular origin, it was popularized by universal language model fine-tuning (ULMFiT) [16], GPT[1] [40], and BERT[2] [8]. Such ideas were also demonstrated already by Dai and Le [7] and Peters et al. [35]. Regardless, this staged approach is particularly suitable for models based on the transformer architecture, see, e.g., [8,25,42,50]. Recent work has continued to study the benefits of the intermediate adaptation stage: Gururangan et al. [12] distinguished *domain-adaptive* from *task-adaptive* pretraining and showed that both approaches can improve downstream performance. In this paper, we do not make a differentiation at this resolution and just talk of *continued pretraining* (CPT).

[1]GPT technically stands for *Generative Pre-trained Transformer*, but it has turned into a lexicalized acronym like laser or radar.

[2]As with GPT, BERT technically stands for *Bidirectional Encoder Representations from Transformers*, but it has also turned into a lexicalized acronym.

While various types of pretraining have been extensively studied for English medical texts (e.g., BioBERT [22], ClinicalBERT [2]), applying these techniques to minority or low-resource languages can present special challenges. The relative scarcity of healthcare data limits the available tools, and historically multilingual models are not always competitive with language-specific models, see [51]. Recent literature also highlights the necessity of localized, in-domain data for effective adaptation. For instance, Myllylä et al. [32] demonstrated that when extracting information from unstructured Finnish radiology reports, translating the reports to English to leverage existing models yielded worse results than using native Finnish texts.

Similar observations have been made across other low-resource languages. In Turkish, Türkmen et al. [49] found that continuing pretraining on even a small, highly specific in-domain corpus was more effective for clinical text classification than utilizing broader, generic biomedical corpora. For European Portuguese, the MediAlbertina model [34] drastically reduced perplexity and improved information extraction simply by undergoing domain adaptation on local electronic medical records. Likewise, Bui et al. [6] showed that fine-tuning on compiled Vietnamese medical textbook and forum data significantly improved health communication performance, bypassing the heavy "English bias" inherent in the base models.

Especially in the realm of large language models (LLMs) it can be crucial to predict from smaller exploratory training runs what the optimal configuration of model size, training-data amount, and compute budget would be. The literature records various power laws that can be used to extrapolate model training behavior. Such power laws exist both for comparing models within a data domain (see, e.g., [15, 18] and references therein) and for comparing behavior across domains; see [5]. Though such work is typically focused on LLMs instead of small language models (SLMs) like BERT [8], the underlying transformer architecture is the same, and many of the effects transfer. We will not be fitting our train-time observations to such power laws, but they will provide us with a lens for qualitative analysis of the results.

In this work, our goal is twofold. First, we study the effect of CPT on a dataset without training labels and report on how various tools fare in observing such effects. Second, we replicate classical studies on how CPT can affect classification task performance in the setting of the Finnish BERT model. We emphasize that our aim is to observe these phenomena and make qualitative assessments. In particular, we do not aim for optimal classification algorithms as we wish to observe performance differentials rather than absolute levels. Finally, a considerable part of the driving motivation of this work has been the scarcity of reports and research on NLP in the domain of Finnish-language healthcare data. With access to such data being highly regulated, these kinds of reports can prove to be valuable in future work.

2. Data collection

Our target model here is the Finnish-language BERT variant introduced by Virtanen et al. [51], `TurkuNLP/bert-base-finnish-cased-v1`, which we refer to here as *FinBERT*. To discuss CPT, we first need to describe the data used for the original pretraining of FinBERT, as CPT behaves differently depending on whether the model has already seen the continued-pretraining data (see, e.g., [37]).

2.1. The FinBERT pretraining data

The FinBERT model was trained on a corpus of data combining three main sources: news, discussions, and webcrawl. The news data was gathered from the YLE[3] corpus[4] and the STT[5] corpus[6], the discussions were from the Suomi24 corpus[7] and the webcrawl data was a combination of [26], a (Finnish) subset of the Common Crawl project[8], and a supplement from Finnish Wikipedia. We refer to [51] for the details of the data gathering and processing, and just replicate their data distribution in Table 1.

Table 1. FinBERT pretraining text source statistics as reported in [51].

	Docs	Sents	Tokens	Chars
News	4M	68M	0.9B	6B
Discussion	83M	351M	4.5B	28B
Crawl	11M	591M	8.1B	55B
Total	98M	1 010M	13.5B	89B

In particular, we emphasize that the model has not been trained specifically on medical data. We presume the datasets contain some medical text, especially through Wikipedia and webcrawls, but medical text was not a specific focus.

2.2. Data used in this project

In this project, we used both a private medical dataset of histopathological reports, provided by the Central Finland Biobank and hosted in the Acamedic[9] secure cloud environment, and a collection of publicly available datasets. The Histopathology dataset contained about 43M tokens' worth of Finnish medical patient records describing both patient visits and clinical sample analysis statements.

As for the publicly available data, the amount of freely available Finnish text data is much smaller than, for example, English. In particular, it is not easy to find other large sources not already included in the FinBERT pretraining dataset. We did, however, want to gather a collection of publicly available datasets containing both data that resembles the original pretraining data and data from sources that were unrepresented, or at least underrepresented, in the pretraining dataset. Our collection of ten dataset variants from seven domains plus our Histopathology data is listed in Table 2 and described in more detail in Appendix A. In the analyses, small and large versions of the same source corpus are treated as separate dataset variants. Unless otherwise noted, each text corpus was split into sets A/B/C in a 70/15/15 distribution. In such a split, set A was used for CPT training, set B for CPT evaluation, then set B again for classification training and set C for classification testing.

Of these datasets, the YLE variants are the only ones whose exact texts we know were included in

[3]Finland's national public broadcasting company.
[4]http://urn.fi/urn:nbn:fi:lb-2017070501
[5]Finland's national news and picture agency.
[6]http://urn.fi/urn:nbn:fi:lb-2019041501
[7]http://urn.fi/urn:nbn:fi:lb-2019010801
[8]https://commoncrawl.org
[9]We note that "Acamedic" is the name of a secure cloud environment built by the HUS Helsinki University Hospital rather than a misspelling of "academic".

the FinBERT pretraining. We sampled the Wikipedia data from the November 2023 dump (extraction run in 2025), so it should be at least somewhat different from what FinBERT was shown during its training in 2019. The other datasets, except for the Histopathology data, are ones the FinBERT model *might* have seen at least partially during pretraining through the webcrawled data. Note that the Eduskunta dataset is a strong outlier in that it has very short texts.

Table 2. Training split statistics and token length distribution for each dataset. The < 256 column lists what percentage of samples had fewer than 256 tokens, and the Labels column gives the number of classification classes ("No labels" where none were available). Histopathology values are approximate as statistics were not recorded at a detailed level during training.

Dataset	Training Samples	Tokens (Millions)	Median Tokens	Mean Tokens	< 256 (%)	Labels (classes)
Histopathology	~125 000	~30	n/a	~243	n/a	No labels
Eduskunta	34 374	1	24	26.6	100.0	15
Finlex	16 912	8	1 213	6 397.5	28.8	2
Finlex Large	162 627	82	514	510.1	0.0	2
Gutenberg	42 000	21	490	501.9	0.2	7
Subtitles	41 262	20	495	488.9	2.1	No labels
Webcrawl	42 000	12	284	289.7	41.7	8
Wikipedia	33 985	9	241	263.0	53.5	8
Wikipedia Large	85 234	22	232	254.5	56.7	8
YLE	34 997	9	237	259.0	55.7	8
YLE Large	120 214	31	241	260.5	54.7	8

3. Model and methods

Our work in this paper revolves around the FinBERT model. In many cases the more modern SLM variants like RoBERTa [25] or especially ModernBERT [52] would be more suitable for this kind of testing, but for the low-resource language of Finnish, language-specific models have historically been important. Indeed, Virtanen et al. [51] showed that a Finnish BERT clearly outperformed the multilingual BERT of its time. More recent models have narrowed this gap, and the contemporary balance depends on the task, domain, and scale. Even though FinBERT might not be the best-performing model for this task, it provides a natural choice for a base model for observing the effects of CPT. For more details of the BERT architecture we refer the reader to [8, 51].

For the purposes of this paper, we note that FinBERT ingests as input a text string, splits it into a sequence of substrings called *tokens*,[10] and processes this sequence into a sequence of so-called *embedding vectors*; for FinBERT these vectors are 768-dimensional. This sequence of embedding vectors then goes through 12 *transformer blocks* to produce a final output sequence of embedding vectors. Crucially, the FinBERT model always prepends a special `[CLS]` token to any token sequence. The embedding vector of this token aims to capture the semantic content of the whole input string, and

[10]These are typically short common words or syllables.

it is this embedding vector we mostly work with in this paper. Thus, any set of input texts is converted to a set of 768-dimensional `[CLS]` embeddings, which we call the (embedding) point cloud. These points have natural distance and similarity measures like the Euclidean metric or the cosine similarity of vectors. We also note that the FinBERT model has a maximum "token window size" of 512 tokens, meaning that any text longer than that needs to be cut down to 512 tokens.

From the FinBERT core model we train CPT-adapted versions (which we call the *CPT models*) on various datasets, as described in Section 3.1. These CPT models are compared to the base FinBERT model both through unlabeled embedding-vector comparisons and through their performance in a labeled classification task. For the classification task we ran several variants, but for the sake of brevity we limit ourselves to discussing approaches where we trained a simple logistic regression or a k-nearest neighbors (k-NN) classifier on small subsets of embedding vectors generated by the model; see Section 4.2 for details.

Our training runs were executed on one of two machines: a MacBook Pro (M4 Max, 36 GB RAM) or a Tesla V100-equipped virtual machine running Linux. The various CPT runs usually took several hours each and our classification runs at most a few minutes.

3.1. Continued pretraining

For CPT we use the masked language modeling (MLM) task[11] in which the model predicts randomly masked tokens with a cross-entropy loss; see [8]. For each dataset variant (described in Section 2) we continued training the FinBERT model on the MLM objective with the standard masking probability of 15% (of the masked tokens, 80% are replaced by `[MASK]`, 10% by a random token, and 10% are left unchanged). All runs shared the following configuration: AdamW optimizer ($\beta_1 = 0.9$, $\beta_2 = 0.999$, $\varepsilon = 10^{-8}$) with a learning rate of 5×10^{-5}, linear warmup over the first 500 optimizer steps followed by linear decay, weight decay of 0.01, gradient-norm clipping at 1.0, an effective batch size of 64 sequences, a maximum sequence length of 512 tokens (longer inputs truncated, shorter inputs padded to full length), and random seed 42. We evaluated on a validation set every 500 steps and kept the checkpoint with the lowest evaluation loss; no early stopping was used. The number of epochs varied between three and five across datasets; Table 3 lists the realized epochs, optimizer steps, and best evaluation losses per dataset. This configuration is similar to what is used in the original FinBERT pretraining [51], though the original training also used the extra task of Next Sentence Prediction (NSP), which we decided to omit as it seems to provide little extra benefit according to the literature, see [25].

Most runs used BF16 mixed precision[12] on the M4 Max machine with a per-device batch size of 16 and gradient accumulation over 4 batches. The Finlex, Finlex Large, and Gutenberg runs were instead executed on the V100 machine, which lacks hardware BF16 support, using FP16 mixed precision, gradient checkpointing, and smaller per-device batches with correspondingly larger gradient accumulation. The effective batch size of 64 was identical across all runs. The Histopathology run, executed inside the Acamedic environment, used the same hyperparameters as the M4 Max runs.

[11] Also known as the *Cloze task* in the more classical literature.

[12] In *mixed-precision training* part of the computation is carried out in 16-bit floating-point numbers instead of the usual 32-bit ones, which reduces memory use and speeds up training. BF16 and FP16 are two such 16-bit formats: BF16 (*brain floating point 16-bit*) spans the same range of magnitudes as a 32-bit float but with fewer significant digits, while FP16 (*floating point 16-bit*) is more precise over a narrower range. Which of the two is available depends on the hardware.

Table 3. Realized CPT training runs per dataset variant. Histopathology values are approximate as exact numbers were not recorded during training.

Dataset	Train samples	Epochs	Optimizer steps	Best eval loss
Histopathology	∼125 000	5	∼9 000	1.168
Eduskunta	34 374	5	2 690	1.403
Finlex	16 912	3	795	0.884
Finlex Large	162 627	3	7 626	0.595
Gutenberg	42 000	3	1 971	2.281
Subtitles	41 262	5	3 225	1.902
Webcrawl	42 000	3	1 971	1.858
Wikipedia	33 985	3	1 596	1.453
Wikipedia Large	85 234	5	6 660	1.335
YLE	34 997	5	2 735	1.208
YLE Large	120 214	5	9 395	1.176

3.2. Measuring embedding changes

Our core idea was to quantify how much CPT affected a model by measuring changes in text embedding vectors. In particular, after running CPT with our training data (set A), we extracted the `[CLS]` embedding vectors of the validation dataset (set B) from each of FinBERT's 1+12 transformer layers[13]. Thus from the validation dataset we created 13 embedding point clouds from both the base and CPT models, which we denote $E_0^{\text{base}}, \ldots, E_{12}^{\text{base}}$ and $E_0^{\text{CPT}}, \ldots, E_{12}^{\text{CPT}}$. We then observed how much the point clouds E_j^{base} and E_j^{CPT} resembled each other for each $j = 0, \ldots, 12$.

The following techniques are common tools for measuring the similarity of two point clouds. Instead of providing detailed definitions, we give brief descriptions and refer the reader to standard sources. See [4, 9, 55] for further comparative studies on how they can be used in analyzing neural networks in general, and how they (and other approaches) differ. See also [54] for a more foundational approach to measuring "neural representations" in general.

- **Centered Kernel Alignment (CKA)**: A rotation-invariant similarity metric that compares the geometric structure of two sets of vectors based on the distribution of cosine similarities within each set [20, 28, 36].
- **Procrustes Analysis**: Measures the optimal alignment between two shapes. We compute the Procrustes distance (or similarity) after finding the best orthogonal rotation to map the CPT embeddings to the base embeddings [13, 46].
- **Representational Similarity Analysis (RSA)**: Compares the correlation structures of the two spaces. We calculate the distance matrices of both spaces and then compute the correlation between these matrices [10, 21].

We also measured the internal geometry of the point clouds E_{12}^{base} and E_{12}^{CPT}. The following approaches measure each point cloud separately and then compare the results.

[13]The BERT architecture has 12 *transformer blocks* or *layers*. We denote the collection of context-free embeddings for the input tokens as layer 0 embeddings. See again [51] or [8] for details.

- **Isotropy**: We use the *Partition Function* $Z(\tau)$ and *Effective Rank* as metrics to measure how uniformly the embeddings occupy the latent space [31, 44, 45].
- **Clustering**: We applied k-means clustering, choosing k by the Silhouette score, and, where class labels were available, quantified how well the clusters align with the labels using the adjusted Rand index (ARI) and normalized mutual information (NMI) [17, 43, 47].

3.3. Classification tasks

The typical approach when using BERT-based models for classification, as already advocated in [8], is to fine-tune the whole model on the classification task. Indeed, the "raw" `[CLS]` embeddings are known to capture semantic content poorly, see, e.g., [23] and references therein. Even so, we deliberately depart from this approach, opting instead to train simple classifiers on the frozen `[CLS]` embeddings. Our core rationale is that our aim is not to evaluate the `[CLS]` embedding quality, nor to optimize the classification results, but to observe the effects of CPT. In an exploratory stage of the experiments we did indeed run a few "full" classification training runs, but once the whole model is trained on the task, the base and CPT models converge to similar performance and the CPT effect is largely washed out.

Furthermore, both to simulate limited-data situations and to enhance the signal, we limited our classification training-set size. In particular, for our classification task we sampled a limited subset (e.g., 500, 1 000, or 2 000 samples) of set B. For classifiers we used either a simple logistic regression or a k-NN classifier.

We trained the classifiers on the raw 768-dimensional final-layer `[CLS]` embeddings without any normalization, standardization, or dimensionality reduction. For logistic regression we used L2 regularization, the limited-memory Broyden–Fletcher–Goldfarb–Shanno (L-BFGS) solver, a maximum of 2 000 iterations, and inverse class-frequency weights to counter class imbalance; the regularization strength C was selected by 5-fold cross-validated grid search over 20 logarithmically spaced values in $[10^{-4}, 10^{4}]$, optimizing macro-F1. For the k-NN classifier we likewise selected the hyperparameters by stratified 5-fold cross-validated grid search (optimizing macro-F1) over $k \in \{3, 5, 7, 11, 15, 21\}$, uniform versus distance weighting, and cosine versus Euclidean distance, independently for every combination of dataset, training-subset size, random seed, and model (base or CPT). Each experiment drew its labeled subset of the training data with class-proportional allocation, trained on it, and evaluated on a held-out dataset; we repeated every experiment with five random seeds (42, 123, 456, 789, 1024) and report means over the repetitions. The same sampled subsets were used for the base and the CPT model, so the comparison between the two is paired.

Our datasets had widely differing label counts, label balances, and general data quality. Simply comparing e.g., the increase in accuracy from CPT for a given classification task would bias the comparison toward domains with lower baseline accuracy. This is because increasing a classifier's accuracy from 90% to 95% is much harder than from 40% to 45%—not to mention the fact that from the starting point of 50% one can improve by 15 percentage points, but starting from 92% one cannot. Thus we had a few different measures that we used for both accuracy and F1 scores. Here s_{CPT} stands for the accuracy or F1 score of the CPT model, while s_{BL} stands for the baseline score of the base model.

(1) Error reduction rate (ERR): what fraction of remaining error was removed $(s_{\mathrm{CPT}} - s_{\mathrm{BL}})/(1 - s_{\mathrm{BL}})$.
(2) Logit delta: $\mathrm{logit}(s_{\mathrm{CPT}}) - \mathrm{logit}(s_{\mathrm{BL}})$, where $\mathrm{logit}(p) = \log(p/(1-p))$.

4. Results

We discuss our results across dataset variants in parts: CPT training, the classification task, and then the observed correlations between CPT and classification performance. We note that the Eduskunta dataset was an outlier in many measurements, likely because its texts are very short; see again Table 2 for the distribution. Note that text length can have strong effects on NLP learning [33].

4.1. Continued pretraining

In Figure 1 we show the train-time loss curves of the CPT. What we find most striking is that the two YLE datasets show very little change in the loss, while the Finlex and Histopathology datasets show a drastic change. We interpret this as evidence that the YLE data was already "familiar" to the FinBERT model and produced very little new adaptation need, while the datasets that were more novel to the model, like Histopathology or Finlex, caused drastic changes.

This type of interpretation can be studied through the lens of the scaling laws used to predict LLM train- and test-time behavior. The various classical sources, e.g., [5, 15, 18], advocate for slightly different variations of power laws. For the sake of simplicity, we view our findings through the one used by Hoffmann et al. in [15]. It describes the power law[14] for the pretraining loss L as

$$L(N, D) = E + \frac{A}{N^{\alpha}} + \frac{B}{D^{\beta}}, \tag{1}$$

where E is the underlying entropy of the text domain from which the training data is sampled (see e.g., [27]), N is the number of model parameters, D is the number of training tokens, and the rest are constants that depend on the model and data context. The term $AN^{-\alpha}$ describes the *approximation error* we get from fitting a model with limited parameters and capability to our data, whereas the term $BD^{-\beta}$ describes the *estimation error* we get from having only a small sampling of all possible data.

Since we are working with a fixed model, we do not vary N, and so for us the power law takes the form $L(D) = C + BD^{-\beta}$. With this as our theoretical framework we can suggest an explanation for the shape of the loss curves in Figure 1. The horizontal asymptotes of the train loss curves represent a combination of the underlying entropy of the particular dataset and the approximation error we get from using FinBERT rather than a perfect oracle model.

To analyze the magnitude of the loss-curve *drop*, it is important to note that the original power laws were built for the *pretraining* stage and not for CPT. In particular, the almost flat shapes of the training loss curves of the YLE datasets probably reflect the fact that the FinBERT model has already seen the data. We are essentially seeing here the flatter tail end of a larger loss curve that "started" during the FinBERT pretraining. For the other domains, especially Finlex and Histopathology, the very strong drop indicates that such data was most likely not present in the pretraining data, and the magnitude of the drop is then a sign of how much estimation error we were able to reduce.

CKA, RSA, and Procrustes measured how the `[CLS]` embedding point clouds of the set B texts differed between the CPT and base models; the results were very similar across the three methods. Figure 2 displays the CKA result. This should reflect not only the change in the weights of the model due to CPT, but more crucially how the model's "understanding" of this domain's text content changed.

[14] We emphasize here that we are not aiming to fit a power law to our data, but rather use this as a tool for a qualitative analysis.

We note that the changes do not seem to directly correlate with our train-time loss dynamics. In Figure 2, the Subtitles dataset is almost an outlier, showing very large changes, while the Finlex dataset shows the least change. There seems to be a nonlinear connection between how much the model adapted to the text and how much change in the embedding vectors the learning required. We note, however, that these results are highly dependent on the text content of the particular domain; we are not probing the model layers directly but rather the embeddings of texts in set B.

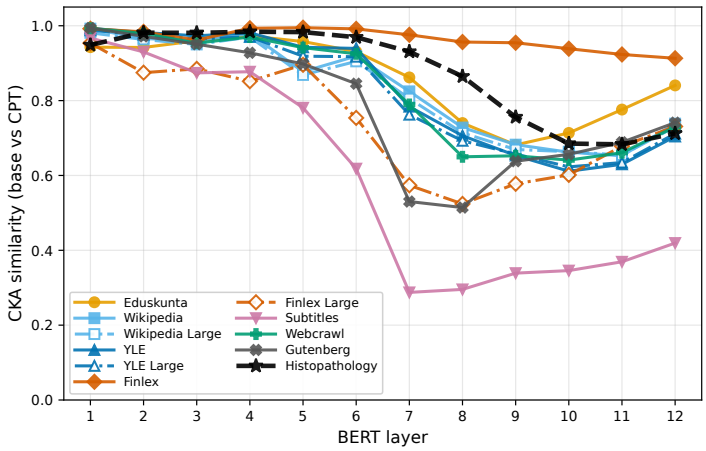


Figure 2. CKA similarity between the base and CPT models across FinBERT layers, computed on the `[CLS]` embeddings of set B; higher values indicate greater similarity. Layer 0 (the context-free token embeddings) is omitted as the CKA scores were degenerate.

4.2. Classification task

Figure 3 shows one of our classification macro-F1 results. We note that the classification macro-F1 scores vary considerably across dataset variants. To demonstrate how raw accuracy- and macro-F1-score deltas differ from the relative versions discussed in Section 3.3, we have listed the raw deltas next to logit deltas and ERR scores in Table 4.

The results are generally as expected. The CPT training tended to improve the classification results, though with the YLE datasets the effects were close to zero. Once again, the YLE datasets were part of the original FinBERT training data. We note that the different datasets are quite heterogeneous with respect to their amenability to classification.

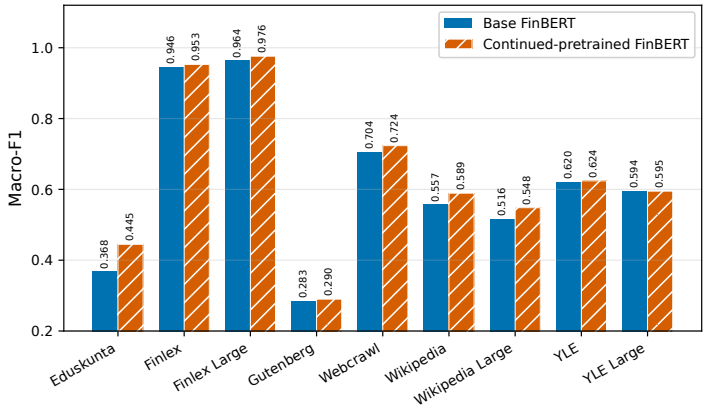


Figure 3. Macro-F1 of the base and CPT models for the logistic regression classifier across the labeled dataset variants, trained on 500 labeled samples.

Table 4. Accuracy- and macro-F1-based improvement measures for the logistic-regression classifier trained on 500 labeled samples: the delta, the logit delta, and ERR between the CPT and base models, computed from the per-domain mean scores over the five random seeds. The largest value in each row is shown in bold.

	Eduskunta	Finlex	Finlex Large	Gutenberg	Webcrawl	Wikipedia	Wikipedia Large	YLE	YLE Large
Accuracy delta	**0.074**	0.005	0.004	0.006	0.019	0.035	0.031	0.007	0.004
Accuracy logit delta	0.296	0.146	**0.410**	0.031	0.091	0.145	0.125	0.029	0.016
Accuracy ERR	0.143	0.132	**0.334**	0.009	0.061	0.082	0.069	0.018	0.010
Macro-F1 delta	**0.077**	0.007	0.012	0.007	0.019	0.032	0.033	0.005	0.001
Macro-F1 logit delta	0.321	0.149	**0.430**	0.032	0.094	0.131	0.131	0.019	0.004
Macro-F1 ERR	0.122	0.132	**0.341**	0.009	0.065	0.073	0.067	0.012	0.003

4.3. MLM-classification correlations

Here we discuss correlations between signals observed during CPT and performance changes in the classification tasks between models. We emphasize that the analysis is exploratory and post hoc. In particular, for the sake of transparency we note that *we did not record beforehand which features and targets to use in the comparison*. Indeed, we found some features that seem to correlate with better classification performance, but these were selected from a set of 91 features and 129 targets[15]. We did not apply a correction for multiple comparisons and therefore treat the observed correlations as hypothesis-generating patterns rather than confirmatory evidence.

In Figure 4 we show a subset of this heatmap with our most relevant features and targets. Each cell corresponds to a scatter plot of the nine labeled dataset variants—with the given feature and target as the x- and y-axes—and a line fitted to it (see, e.g., Figure 5). The colors encode Pearson correlation coefficients, with dark red indicating strong positive correlation and dark blue indicating strong negative correlation.

From this heatmap of correlations, we note that there are isolated large-magnitude cells, which we presume to reflect spurious correlations. However, there are several strong horizontal strips which seem to persist over different target measures; these are, in an approximate order of prominence:

(1) *Train loss relative improvement*. This is simply the drop in the CPT train-time loss divided by the loss at the start (see again Figure 1).
(2) *Isotropy effective rank delta and Isotropy partition function delta.* In particular, the more isotropic (more evenly distributed) the embeddings became, the better the classifier performed.
(3) *Train first epoch loss relative improvement.* Same as the first point, but restricted to just the first epoch. It seems to be less descriptive than the full relative loss change, though still prominent.
(4) *CKA Max and CKA Layers 1–3.* This also persists for the RSA and Procrustes measures. Perhaps large changes in the earlier layers of the model are an indicator that we had a lot of data for training, and thus the changes propagated strongly to the earlier layers as well?
(5) *Clustering ARI.* When clustering improves during CPT, it seems that the clusters tend to be more supportive of our classification tasks, regardless of whether we are using k-NN or logistic

[15]The full heatmap of correlations between all of the features and targets we had is available online at https://github.com/ramiluisto/CPT_analysis_companion/blob/main/figures/heatmap.pdf.

regression.

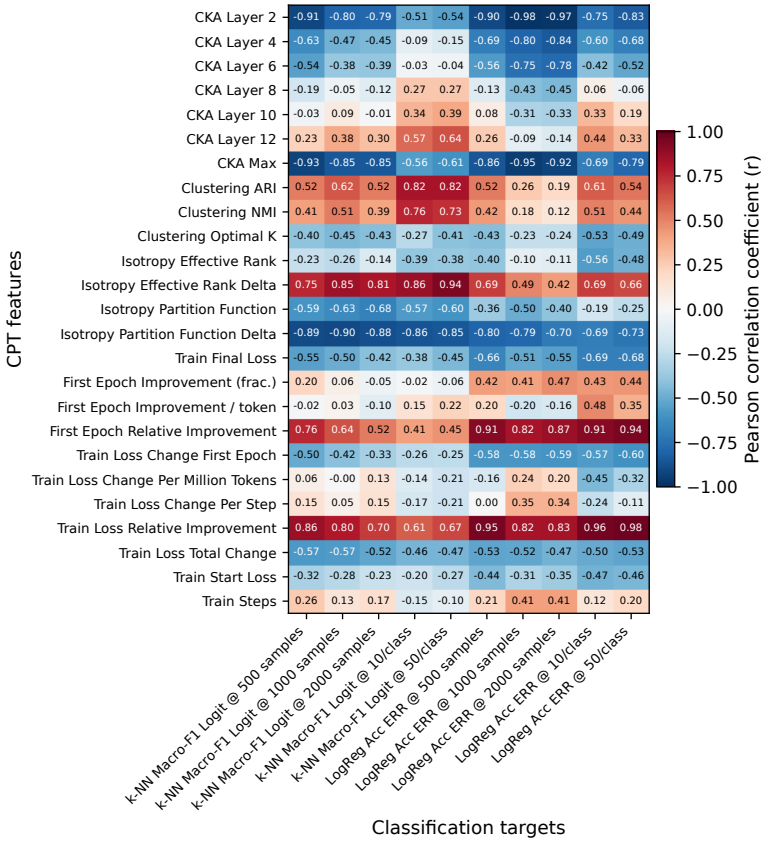


Figure 4. Subset of the correlation heatmap between CPT features and classification task targets across labeled dataset variants.

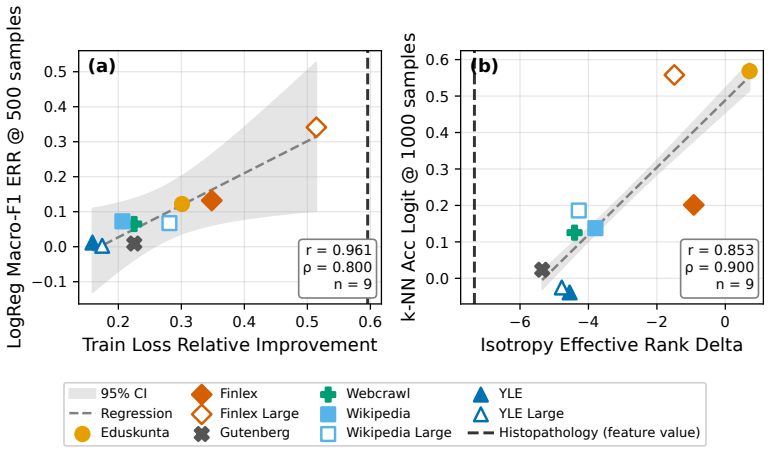


Figure 5. Example correlations between CPT signals and classification improvement. (a) Train-loss relative improvement vs. the logistic-regression macro-F1 error-reduction rate (500 training samples). (b) Isotropy effective-rank delta vs. the *k*-NN accuracy logit delta (1 000 training samples). The vertical dashed lines mark the Histopathology dataset's feature values; they are feature-side references only, as no classification labels exist for that dataset.

Separately from the multiple-comparisons issue, it is not clear whether these different features are correlated or whether they all reflect a common "considerable deep changes during CPT" confounder. There could be several distinct effects at play. We also note that the amount of change in the later layers of the model does not seem to strongly correlate with classification performance, which we find curious.

5. Conclusions, limitations, and future work

In this paper, we set out to (1) measure the effects of CPT on the FinBERT model with histopathological reports and (2) test, on public proxy datasets, whether any CPT signals correlate with downstream classification performance. For (1) we observed clearly differing train-time loss-curve behavior between the Histopathology data, which is certainly novel to the model, and the YLE data, which was present in the FinBERT pretraining data. For (2) our exploratory analysis demonstrated tentative associations between CPT signals and downstream performance. In particular, the connection between the different features extracted from the training, especially loss curve geometry, layerwise embedding changes, and embedding isotropy, is suggestive.

None of our tentative findings are statistically established, and they were in line with expectations. We nevertheless consider this work a useful contribution to the sparse literature on Finnish healthcare NLP. At minimum, we provide informed priors for planning CPT when labeled data is lacking.

For future work we see both theoretical possibilities and further practical experiments. On the theoretical side, these types of analyses would also benefit from a mathematical foundation for "capped power laws for CPT" that would account for the fact that a model has already seen large amounts of data. Work in this direction already exists; see, e.g., the transfer scaling laws in [14] or the domain-continual pretraining (D-CPT) law in [39]. We feel there is room for exploration and advances in this area, especially in the domain of low-resource languages and small models.

On the practical side, another way to obtain CPT signals would be to apply the measures described here to the model weights, neural activations, or attention patterns rather than embedding vectors, see e.g., [11, 30, 41] for more involved approaches. Replication of our results with other datasets would also be crucial.

Finally, we note that we have omitted from this report some experiments we did not find particularly informative. For the MLM task we also plotted various principal component analysis (PCA), independent component analysis (ICA), t-distributed stochastic neighbor embedding (t-SNE), and uniform manifold approximation and projection (UMAP) projections of the resulting `[CLS]` embedding point clouds of the set B texts. There were visually clear differences in the images between the base and CPT models, but nothing we found worth reporting.

Acknowledgments

We gratefully acknowledge the Central Finland Biobank. We thank the Wellbeing Services County of Central Finland for research services. We also thank members of the GenAID research group for medical expert consultation on oncological matters.

This study is one part of the Central Finland AI hub II project that has received funding from the Regional Council of Central Finland (https://www.keskisuomi.fi/) and the European Regional

Development Fund (ERDF) (https://ec.europa.eu/regional_policy/funding/erdf_en).

Parts of this work were supported by the Strategic Research Council at the Research Council of Finland (grant number 372591) under GAINS ("Generative AI and Digital Solutions: Enhancing Effectiveness and Productivity of Healthcare Services") and by Business Finland (1562/31/2024) through the GenAID research project.

Use of AI tools declaration

Throughout this project the first author has used various AI tools, namely Cursor, Claude Code, and ChatGPT, to assist in some of the more mundane parts of programming. Most scripts have been created in collaboration with coding agents, and AI tools have been used for critical feedback on the manuscript. Any work by AI tools has been validated by a human.

The Histopathology dataset resided in the Acamedic environment, and no AI system besides the FinBERT model trained there interacted with the data.

The textual content of this article was written by humans, and the authors take full academic responsibility for its contents.

Data and code availability

The public corpora used in this study are available from their original distributors: Finnish Wikipedia (Wikimedia dumps), the YLE Finnish News Archive (Kielipankki, restricted license), Finlex Open Data (Ministry of Justice), the Eduskunta-VKK dataset (GitHub), Project Gutenberg (Gutendex), OpenSubtitles (OPUS/Kielipankki), and `Finnish-NLP/mc4_fi_cleaned` (HuggingFace). Our extraction, preprocessing, and split-construction scripts, together with the configuration files and random seeds needed to reconstruct our datasets and train/validation/test splits from these sources, are published as a public code release accompanying this paper at https://github.com/ramiluisto/CPT_analysis_companion. The YLE corpus license does not permit redistribution of derived text data; for that corpus the release provides the scripts and sample identifiers needed to reconstruct our datasets from the original Kielipankki distribution. The Histopathology dataset contains sensitive patient information and cannot be shared; it was accessed and processed exclusively within the Acamedic secure environment.

Ethics statement

The Histopathology dataset was obtained from the Central Finland Biobank under a biobank data agreement, in accordance with the Finnish Biobank Act. The data was pseudonymized and was accessed and processed solely within the Acamedic secure computing environment; no patient-level data left the environment, and only aggregate training statistics were exported.

Conflict of interest

All authors declare no conflicts of interest in this paper.

References

1. *A. Ajanki, Lauseluokitteludatasetti: ministerien vastaukset kirjallisiin kysymyksiin*, GitHub, 2019. Available from: https://github.com/aajanki/eduskunta-vkk.
2. E. Alsentzer, J. Murphy, W. Boag, W. H. Weng, D. Jindi, T. Naumann, et al., Publicly available clinical bert embeddings, *Proceedings of the 2nd Clinical Natural Language Processing Workshop*, 2019, 72–78. https://doi.org/10.18653/v1/W19-1909
3. W. Bani Issa, I. Al Akour, A. Ibrahim, A. Almarzouqi, S. Abbas, F. Hisham, et al., Privacy, confidentiality, security and patient safety concerns about electronic health records, *Int. Nurs. Rev.*, **67** (2020), 218–230. https://doi.org/10.1111/inr.12585
4. Y. Bo, A. Soni, S. Srivastava, M. Khosla, Evaluating representational similarity measures from the lens of functional correspondence, arXiv: 2411.14633. https://doi.org/10.48550/arXiv.2411.14633
5. D. Brandfonbrener, N. Anand, N. Vyas, E. Malach, S. M. Kakade, Loss-to-loss prediction: Scaling laws for all datasets, arXiv: 2411.12925. https://doi.org/10.48550/arXiv.2411.12925
6. N. Bui, G. Nguyen, N. Nguyen, B. Vo, L. Vo, T. Huynh, et al., Fine-tuning large language models for improved health communication in low-resource languages, *Comput. Meth. Prog. Bio.*, **263** (2025), 108655. https://doi.org/10.1016/j.cmpb.2025.108655
7. A. M. Dai, Q. V. Le, Semi-supervised sequence learning, *Proceedings of the 29th International Conference on Neural Information Processing Systems*, 2015, 3079–3087.
8. J. Devlin, M. W. Chang, K. Lee, K. Toutanova, BERT: pre-training of deep bidirectional transformers for language understanding, *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (NAACL-HLT)*, 2019, 4171–4186. https://doi.org/10.18653/v1/N19-1423
9. F. Ding, J. S. Denain, J. Steinhardt, Grounding representation similarity through statistical testing, *Proceedings of the 35th International Conference on Neural Information Processing Systems*, 2021, 1556–1568.
10. K. Dwivedi, G. Roig, Representation similarity analysis for efficient task taxonomy & transfer learning, *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2019, 12387–12396. https://doi.org/10.1109/CVPR.2019.01267
11. G. Eilertsen, D. Jönsson, T. Ropinski, J. Unger, A. Ynnerman, Classifying the classifier: Dissecting the weight space of neural networks, *Proceedings of the 24th European Conference on Artificial Intelligence*, 2020, 1119–1126. https://doi.org/10.3233/FAIA200209
12. S. Gururangan, A. Marasović, S. Swayamdipta, K. Lo, I. Beltagy, D. Downey, et al., Don't stop pretraining: adapt language models to domains and tasks, *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics (ACL)*, 2020, 8342–8360. https://doi.org/10.18653/v1/2020.acl-main.740
13. J. V. Haxby, J. S. Guntupalli, A. C. Connolly, Y. O. Halchenko, B. R. Conroy, M. I. Gobbini, et al., A common, high-dimensional model of the representational space in human ventral temporal cortex, *Neuron*, **72** (2011), 404–416. https://doi.org/10.1016/j.neuron.2011.08.026

14. D. Hernandez, J. Kaplan, T. Henighan, S. McCandlish, Scaling laws for transfer, arXiv: 2102.01293. https://doi.org/10.48550/arXiv.2102.01293

15. J. Hoffmann, S. Borgeaud, A. Mensch, E. Buchatskaya, T. Cai, E. Rutherford, et al., Training compute-optimal large language models, *Proceedings of the 36th International Conference on Neural Information Processing Systems*, 2022, 30016–30030.

16. J. Howard, S. Ruder, Universal language model fine-tuning for text classification, *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (ACL)*, 2018, 328–339. https://doi.org/10.18653/v1/P18-1031

17. L. Hubert, P. Arabie, Comparing partitions, *J. Classif.*, **2** (1985), 193–218. https://doi.org/10.1007/BF01908075

18. J. Kaplan, S. McCandlish, T. Henighan, T. B. Brown, B. Chess, R. Child, et al., Scaling laws for neural language models, arXiv: 2001.08361. https://doi.org/10.48550/arXiv.2001.08361

19. I. Keshta, A. Odeh, Security and privacy of electronic health records: concerns and challenges, *Egypt. Inform. J.*, **22** (2021), 177–183. https://doi.org/10.1016/j.eij.2020.07.003

20. S. Kornblith, M. Norouzi, H. Lee, G. Hinton, Similarity of neural network representations revisited, *Proceedings of the 36th International Conference on Machine Learning*, 2019, 3519–3529.

21. N. Kriegeskorte, M. Mur, P. A. Bandettini, Representational similarity analysis-connecting the branches of systems neuroscience, *Front. Syst. Neurosci.*, **2** (2008), 249. https://doi.org/10.3389/neuro.06.004.2008

22. J. Lee, W. Yoon, S. Kim, D. Kim, S. Kim, C. H. So, et al., BioBERT: a pre-trained biomedical language representation model for biomedical text mining, *Bioinformatics*, **36** (2020), 1234–1240. https://doi.org/10.1093/bioinformatics/btz682

23. B. Li, H. Zhou, J. He, M. Wang, Y. Yang, L. Li, On the sentence embeddings from pre-trained language models, *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, 2020, 9119–9130. https://doi.org/10.18653/v1/2020.emnlp-main.733

24. P. Lison, J. Tiedemann, OpenSubtitles2016: extracting large parallel corpora from movie and TV subtitles, *Proceedings of the 10th International Conference on Language Resources and Evaluation (LREC 2016)*, 2016, 923–929.

25. Y. Liu, M. Ott, N. Goyal, J. Du, M. Joshi, D. Chen, et al., Roberta: a robustly optimized bert pretraining approach, arXiv: 1907.11692. https://doi.org/10.48550/arXiv.1907.11692

26. J. Luotolahti, J. Kanerva, V. Laippala, S. Pyysalo, F. Ginter, Towards universal web parsebanks, *Proceedings of the Third International Conference on Dependency Linguistics (Depling 2015)*, 2015, 211–220.

27. D. J. MacKay, *Information theory, inference and learning algorithms*, Cambridge: Cambridge University Press, 2003.

28. N. Maheswaranathan, A. Williams, M. Golub, S. Ganguli, D. Sussillo, Universality and individuality in neural dynamics across large populations of recurrent networks, *Proceedings of the 33rd International Conference on Neural Information Processing Systems*, 2019, 15629–15641.

29. *Ministry of Justice, Finland, Finlex open data—Finnish legislation and treaties*, Finlex Avoin Data portal, 2025. Available from: https://www.finlex.fi/fi/avoin-data/lataa-aineistoja.

30. A. Morcos, M. Raghu, S. Bengio, Insights on representational similarity in neural networks with canonical correlation, *Proceedings of the 32nd International Conference on Neural Information Processing Systems*, 2018, 5732–5741.

31. J. Mu, P. Viswanath, All-but-the-top: simple and effective postprocessing for word representations, *Proceedings of the 6th International Conference on Learning Representations (ICLR)*, 2018, 1–25.

32. E. Myllylä, P. Siirtola, A. Isosalo, J. Reponen, S. Tamminen, O. Laatikainen, Extracting information from unstructured medical reports written in minority languages: a case study of finnish, *Data*, **10** (2025), 104. https://doi.org/10.3390/data10070104

33. K. Nagatsuka, C. Broni-Bediako, M. Atsumi, Length-based curriculum learning for efficient pre-training of language models, *New Gener. Comput.*, **41** (2023), 109–134. https://doi.org/10.1007/s00354-022-00198-8

34. M. Nunes, J. Boné, J. C. Ferreira, P. Chaves, L. B. Elvas, Medialbertina: an european portuguese medical language model, *Comput. Biol. Med.*, **182** (2024), 109233. https://doi.org/10.1016/j.compbiomed.2024.109233

35. M. E. Peters, M. Neumann, M. Iyyer, M. Gardner, C. Clark, K. Lee, et al., Deep contextualized word representations, *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (NAACL-HLT)*, 2018, 2227–2237. https://doi.org/10.18653/v1/N18-1202

36. J. Phang, H. Liu, S. Bowman, Fine-tuned transformers show clusters of similar representations across layers, *Proceedings of the Fourth BlackboxNLP Workshop on Analyzing and Interpreting Neural Networks for NLP*, 2021, 529–538. https://doi.org/10.18653/v1/2021.blackboxnlp-1.42

37. S. J. Prince, *Understanding deep learning*, Cambridge: MIT press, 2023.

38. *Project Gutenberg, Project gutenberg electronic book collection—Finnish subset*, GitHub, Inc., 2025. Available from: https://gutendex.com.

39. H. Que, J. Liu, G. Zhang, C. Zhang, X. Qu, Y. Ma, et al., D-CPT law: domain-specific continual pre-training scaling law for large language models, *Proceedings of the 38th International Conference on Neural Information Processing Systems*, 2024, 90318–90354.

40. A. Radford, K. Narasimhan, T. Salimans, I. Sutskever, *Improving language understanding by generative pre-training*, Technical report, 2018.

41. M. Raghu, J. Gilmer, J. Yosinski, J. Sohl-Dickstein, SVCCA: singular vector canonical correlation analysis for deep learning dynamics and interpretability, *Proceedings of the 31st International Conference on Neural Information Processing Systems*, 2017, 6078–6087.

42. A. Rogers, O. Kovaleva, A. Rumshisky, A primer in BERTology: what we know about how BERT works, *Transactions of the Association for Computational Linguistics*, **8** (2020), 842–866. https://doi.org/10.1162/tacl_a_00349

43. P. J. Rousseeuw, Silhouettes: a graphical aid to the interpretation and validation of cluster analysis, *J. Comput. Appl. Math.*, **20** (1987), 53–65. https://doi.org/10.1016/0377-0427(87)90125-7

44. O. Roy, M. Vetterli, The effective rank: a measure of effective dimensionality, *Proceedings of the 15th European Signal Processing Conference*, 2007, 606–610.

45. W. Rudman, N. Gillman, T. Rayne, C. Eickhoff, Isoscore: measuring the uniformity of embedding space utilization, *Proceedings of the Findings of the Association for Computational Linguistics: ACL 2022*, 2022, 3325–3339. https://doi.org/10.18653/v1/2022.findings-acl.262

46. S. L. Smith, D. H. P. Turban, S. Hamblin, N. Y. Hammerla, Offline bilingual word vectors, orthogonal transformations and the inverted softmax, *Proceedings of the 5th International Conference on Learning Representations (ICLR)*, 2017, 1–10.

47. A. Strehl, J. Ghosh, Cluster ensembles—a knowledge reuse framework for combining multiple partitions, *J. Mach. Learn. Res.*, **3** (2003), 583–617. https://doi.org/10.1162/153244303321897735

48. *R. Toivanen, A. Tanskanen, T. Vehviläinen, Finnish-NLP/mc4_fi_cleaned*, Hugging Face, 2022. Available from: https://huggingface.co/datasets/Finnish-NLP/mc4_fi_cleaned.

49. H. Türkmen, O. Dikenelli, C. Eraslan, M. C. Ţallı, S. S. Özbek, BioBERTurk: exploring turkish biomedical language model development strategies in low-resource setting, *J. Healthc. Inform. Res.*, **7** (2023), 433–446. https://doi.org/10.1007/s41666-023-00140-7

50. A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, et al., Attention is all you need, *Proceedings of the 31st International Conference on Neural Information Processing Systems*, 2017, 6000–6010.

51. A. Virtanen, J. Kanerva, R. Ilo, J. Luoma, J. Luotolahti, T. Salakoski, et al., Multilingual is not enough: BERT for finnish, arXiv: 1912.07076. https://doi.org/10.48550/arXiv.1912.07076

52. B. Warner, A. Chaffin, B. Clavié, O. Weller, O. Hallström, S. Taghadouini, et al., Smarter, better, faster, longer: A modern bidirectional encoder for fast, memory efficient, and long context finetuning and inference, *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics*, 2025, 2526–2547. https://doi.org/10.18653/v1/2025.acl-long.127

53. *Wikimedia Foundation, Wikimedia downloads*, Wikipedia, 2023. Available from: https://dumps.wikimedia.org.

54. A. H. Williams, E. Kunz, S. Kornblith, S. Linderman, Generalized shape metrics on neural representations, *Proceedings of the 35th International Conference on Neural Information Processing Systems*, 2021, 4738–4750.

55. J. Wu, S. Saha, Y. Bo, M. Khosla, Measuring the measures: discriminative capacity of representational similarity metrics across model families, arXiv: 2509.04622. https://doi.org/10.48550/arXiv.2509.04622

56. *Yleisradio, Yle Finnish news archive 2011–2018, source version*, Kielipankki, 2017. Available from: http://urn.fi/urn:nbn:fi:lb-2017070501.

A. Details on our datasets

All corpora were cleaned by removing samples with missing text, exact duplicate texts, and texts shorter than 50 characters. Unless stated otherwise, each corpus was then split into sets A/B/C with ratios 70/15/15 (stratified by class label where labels exist) using a fixed random seed (42); token counts throughout refer to the FinBERT tokenizer.

(1) **Histopathological reports:** This was a dataset provided by the Central Finland Biobank, containing medical descriptions of patients who underwent histopathological examination. Labels were not readily available[16], so we used this dataset purely for continued pretraining.
Size & Splits: Approximately 179 000 samples of text, with an average length of 243 tokens (80% of the samples had between 30 and 500 tokens).
Labels: None available.
Reference: Central Finland Biobank (Keski-Suomen Biopankki). Histopathology Reports—private medical dataset (unpublished, used with permission for this study). Central Finland Biobank, Jyväskylä, Finland. (No public access available.)

(2) **YLE:** We used the same YLE corpus as in the FinBERT pretraining, and sampled both a small and a large variant of that data. This is data that the FinBERT model has already seen in the pretraining, but a very small part of the full 13.5B tokens used by the FinBERT authors.[17] In particular, continuing the pretraining of the FinBERT model on this dataset should "teach" the model little that is new, but perhaps narrow its focus.
Size & Splits: Small variant: ∼50 000 samples (35k train / 7.5k val / 7.5k test). Large variant: 172 000 samples (120k train / 26k val / 26k test).
Labels: 8 different topical classes (e.g., Politics, Economy, Sports, Culture).
Reference: [56].

(3) **Wikipedia:** We sampled two datasets from the HuggingFace distribution of the Finnish Wikipedia dump of November 1, 2023. The Wikipedia data is something that the FinBERT pretraining also used, but we note that we drew a fresh sample. In all likelihood the data has at least drifted from the specific text that FinBERT saw during the pretraining.
Size & Splits: Small variant: 49 000 samples (34k train). Large variant: 122 000 samples (85k train).
Labels: 8 topic classes assigned by a keyword-matching heuristic over title and text (7 topics like *Science* or *Politics*, plus one catch-all *Other*). We note that these categories are not Wikipedia's own metadata.
Reference: [53].

(4) **Finlex:** We extracted from the Finlex database Finnish legal documents (legislation and international treaties). This is again text that the pretraining might have seen through the webcrawl.

[16]With expert advice it might have been possible to extract some labels from the data, but not necessarily at scale.

[17]The authors of FinBERT [51] state that the YLE dataset was about 20% of the news section of their corpus, which would put the full YLE data at about 1.3% of their pretraining data.

Size & Splits: Small variant: roughly 24 000 samples (17k train), with full documents kept intact (average length ~6 000 tokens). Large variant: 232 000 samples (162k train) created by sampling disjoint 512-token blocks from the source documents.
Labels: 2 document-type classes: treaty vs. statute (ratio ~1:3.4).
Reference: [29].

(5) **Eduskunta:** The Finnish parliament stores records of written questions and answers. We retrieved a dataset of these records for our experiments.
Size & Splits: 55 106 sentences in the official splits (49 106 train / 3 000 dev / 3 000 test); the MLM training set is a label-stratified 70% subset (34 374 sentences) of the official training split.
Labels: 15 responsible ministries (e.g., Ministry of Finance, Ministry of Education).
Reference: [1].

(6) **Subtitles:** The OpenSubtitles project provides a dedicated collection of Finnish subtitles. We sampled a dataset from this project, which was used for CPT experiments only.
Size & Splits: 59 000 samples (41k train).
Labels: None (we were not able to find reasonable classes for them).
Reference: [24]. (Finnish subtitles extracted via OPUS.)

(7) **Gutenberg:** We scraped Finnish books from Project Gutenberg.
Size & Splits: 60 000 samples (42k train).
Labels: 7 genres: Fiction, Drama, Poetry, Folklore, Humor, Non-fiction, and Children's literature.
Reference: [38].

(8) **Webcrawl:** We used an existing Finnish webcrawl dataset from Toivanen et al.
Size & Splits: 60 000 samples (42k train), about 12M tokens.
Labels: 8 source-type classes derived from the URL host (News, Government, Educational, Commercial, Forum, Blog, Wikipedia, Other).
Reference: [48].



AIMS Press

© 2026 the Author(s), licensee AIMS Press. This is an open access article distributed under the terms of the Creative Commons Attribution License (http://creativecommons.org/licenses/by/4.0)