# What if fanfiction, but also coding: Investigating cultural differences in fanfiction writing and reviewing with machine learning methods
-

## Fine Tuning a BERT-based Pre-Trained Language Model for Named Entity Extraction within the Domain of Fanfiction

**Nathan Kindt**[1]

**Supervisors: Hayley Hung**[1]**, Chenxu Hao**[1]**, Ivan Kondyurin**[1]

[1]EEMCS, Delft University of Technology, The Netherlands

A Thesis Submitted to EEMCS Faculty Delft University of Technology,
In Partial Fulfilment of the Requirements
For the Bachelor of Computer Science and Engineering
January 26, 2025

Name of the student: Nathan Kindt
Final project course: CSE3000 Research Project
Thesis committee: Hayley Hung, Chenxu Hao, Ivan Kondyurin, Elmar Eisemann

**Abstract**

The introduction of Pretrained Language Models (PLMs) has revolutionised the field of Natural Language Processing (NLP) and paved the way for many new, exciting large-scale studies for various areas of research. One such field presents itself in the emerging digital literary corpus that is fanfiction, providing research opportunities within the fields of (NLP), Computational (Socio-) Linguistics, the Social Sciences and Digital Humanities. However, because of the unique linguistic characteristics of this literary domain many modern NLP solutions utilizing PLMs encounter difficulties when applied on fanfiction texts. This paper aims to indicate that the performance of various NLP tasks performed by PLMs on fanfiction texts can be improved by applying Domain Adaptive Pre-Training (DAPT) to PLMs. A case-study is performed to show that the performance of a BERT-based PLM can be improved for the downstream NLP task of Named Entity Recognition (NER) by applying supervised domain specific fine-tuning. While we gain a 6% increase in F1 score performance, we are sceptical about these results due to the limited amount of annotated data available leading to the model overfitting and show a lack of capacity to generalize to unseen data from the CoNLL NER dataset.

## 1 Introduction

Fanfiction is defined as fictional literary works produced by non-professional writers that expand on existing plot lines and/or characters within commercially published canonical works. It has been shown that this ecosystem of literary production presents opportunities for research within the fields of Natural Language Processing (NLP), Computational (Socio-) Linguistics, the Social Sciences and Digital Humanities [1], [2], [3], [4], [5], [6], [7]. Yet most NLP tools that have been developed make use of existing digital repositories containing 'regular' literature such as Google Books and Project Gutenberg [8].

Fanfiction has many advantages over these existing data sources, it provides a high volume of digitally available, annotated—through user interactions and other forms of metadata—literary texts [5], and are less influenced by factors such as marketing and the editorial process [3]. Another affordance comes from the deep, complex social structures which govern their creation; it has even been argued as a form of modern mythology [9]. Fanfiction communities such as fanfiction.net and Archive of Our Own (AO3) have been shown to feature strong participatory culture—where people actively engage with and contribute to the shared narratives—and were recognized to "highlight the motivations and desires of readers" [2]. All these factors signal fanfiction as a social phenomenon worthwhile for further investigation.

Despite this high potential, fanfiction has been indicated by scholars to be under-studied [1], [10]. One possible reason being that fanfiction is seen as an inferior form of literature, suffering from cultural stereotyping and being dismissed as "derivative, unedited, written mostly to elicit strong affective responses" [3]. While public—and thus academic—perception continues to increase as fanfiction gains more mainstream popularity, research is also limited by the available computational technologies needed to process this literary data. In research there often exists a trade-off between the quality and the quantity of data. NLP can be used to auto- matically annotate large datasets but—depending on the accuracy of the NLP technique(s) used—produce noisy results. Human annotation generally produces cleaner datasets, but this process is more expensive and thus often not feasible on larger scales. Therefore current research is either small-scale, or uses reliable NLP techniques such as Sentiment or Lexical Diversity Analysis [6], [10]. However, recent advancements in the field of NLP have allowed for more—and even new—kinds of research to be done [6].

Since the introduction of Transformers by Vaswani et al. [11], many NLP techniques have been revolutionized by utilizing Pre-trained Language Models (PLMs) based on this transformer architecture[1] (Figure 1) [12], [13], [14], [15]. Numerous different PLMs have been developed with varying architectures[2], all with their own (potential) applications, challenges and limitations which are an active area of research [17]. For applications in the field of NLP, three distinct paradigms have been identified: "fine-tuning for the task of interest, prompting the PLMs to perform the desired task, or reformulating the task as a text generation problem" [15]. While for a given NLP task 'the best' performing model—a combination of architecture, training strategies and paradigms—depends on multiple factors such as the type of task, computational resources and available training data, in general certain architectures and paradigms lend themselves better for certain tasks than others. But they all share that they undergo unsupervised pre-training on large amounts of textual data in order to gain a general 'understanding' of human language by modeling the context dependent semantic meaning of words. Afterwards, models can be refined through fine-tuning which adapts them to a specific task and/or domain.

Depending on the dataset on which they are pre-trained, and if and how fine-tuning is applied, models can be categorised as being either a generalist or specialist. Where generalist models attempt to achieve good performance across multiple domains and tasks, specialist models are trained on data for one specific (or a few related) tasks often within a specific domain. Examples among many within NLP include models such as LEGAL-BERT [18] or BioBERT [19], which were specifically developed to increase performance of NLP tasks within their respective domains (the legal and biomedical domain). These studies have indicated that models tailored to their target domain have the potential to perform better—both in accuracy and efficiency—than generalist models, depending on the adaptation strategy used. Another study has shown that while there are some benefits to the addition of generalist data to the training set of PLMs, generalist models currently lag behind specialist models across all tasks [20].

[1] We note the distinction between the term *Pre-trained Language Model* and the more commonly used term *Large Language Model* (LLM), a subset of Language Models which are '*Large*'. The adjective '*Large*' is subjective and thus its meaning has shifted over time when models grew in size. While models like BERT were in comparison large when developed with hundreds of millions of parameters and have thus been called as such, modern variants of e.g. GPT have hundreds of billions to trillions of parameters, making the term LLM ambiguous. What is often meant in practice with 'LLMs' are Language Models (LMs) based on the Transformer architecture that are pre-trained on large amounts of data, but LMs exist that are not pre-trained and/or depend on a different type of architecture. From here on out, we will use the term PLM specifically to refer to transformer-based PLMs.

[2] Readers interested in gaining a better understanding of PLMs and a review of their development and origins, we refer to M. Mars [16]. For a more general overview of the applications of PLMs and their state-of-the-art outside of NLP, see Hadi et al. [17].

The success of PLMs specialised for specific domains begs the question if fanfiction, as an unique literary domain, can also benefit from domain adaptation. We hypothesize that fanfiction writing exhibits significant distinct linguistic characteristics in regards to regular fictional writing. Therefore NLP techniques utilizing PLMs developed for regular fictional literature under-perform when applied to the domain of fanfiction. Because fanfiction as an ecosystem of literary production presents opportunities for research within various fields, we argue for the adaptation of PLMs to this domain for the benefit of interdisciplinary studies utilizing NLP techniques.

To this end, we developed a PLM fine-tuned to the domain of fanfiction for the Named Entity Recognition (NER) task. The NER task consists of extracting all named entities (e.g. Persons, Locations, Organisations) out of unstructured text (see Figure 1). NER is a fundamental problem within Information Extraction (IE), a subfield of NLP [21]. Many downstream IE tasks depend on NER as a first step such as Entity Linking, Relation Extraction and Coreference Resolution [21], [22]. Improvements of NER on fanfiction texts could for example lead to improved querying within fanfiction corpora and better automated quote attribution pipelines for studies.

**Contributions.** This research presents BERT-fanfic-NER, a PLM that is fine-tuned on NER within the domain of fanfiction, gaining an improvement in performance of 6% F1 score. We also investigated the current limitations of NLP by PLMs within fanfiction for the NER task and found that existing state-of-the-art PLMs fine-tuned for the NER task (BERT-NER) under-performed when applied to the domain of fanfiction.

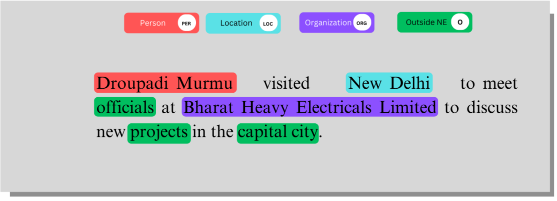

Person PER
Location LOC
Organization ORG
Outside NE O
Droupadi Murmu visited New Delhi to meet officials at Bharat Heavy Electricals Limited to discuss new projects in the capital city.

Figure 1: The NER Problem [22]

## 2 Background

### A. Pretrained Language Models for NER

BERT is a PLM that has revolutionized NLP within the area of Natural Language Understanding (NLU) because of its ability to deeply understand context through its bidirectional attention mechanism [23]. At the time of its creation, BERT greatly improved the scores of popular NLU benchmarks such as GLUE [24]. It utilizes an encoder-only transformer architecture, converting the text into a context-dependent representation and enabling predictions based on these representations. A great benefit of the BERT model was the pre-train then fine-tune paradigm. First the Language Model is trained on massive, heterogeneous, unlabeled corpora in order to gain a general understanding of human language resulting in the Pretrained LM. Then it could be fine-tuned on specific downstream tasks using labeled data, achieving state-of-the-art performance. The pretraining is a computationally expensive step; this paradigm allows for the same PLM parameters resulting from the first step to be fine-tuned into multiple models for specific downstream tasks, making model creation more efficient [23].

Other architectures include decoder-only models, which generate text by predicting the next word (or token) based on previous words using autoregression. The most influential of these is no doubt the Generative Pre-trained Transformer (GPT) [25]. Although this makes it ideal for text generation tasks, it limits the model to an unidirectional (left-to-right) architecture in which the context window only attends to previous tokens, thereby losing critical information for Token-Level NLU and/or IE tasks such as NER and Question Answering (QA) [23].

### B. Named Entity Extraction

The NER task involves extracting named entities out of unstructured text. Entities are each assigned a class label and while different classes of entity types can be defined, in the Standard-IE setting these classes are Person (PER), Organization (ORG) and Location (LOC). For other types of entities the Miscellaneous (MISC) class is added and tokens not labeled as any entity are given the 'Outside entity class' (O) label.

PLMs like BERT split words into standardized units called Tokens in order to handle words not known to the models vocabulary, using '##' to indicate separated words. In order to accurately label words split into multiple tokens this way or names consisting of multiple words (first and last name), Beginning (B-) or Inside (I-) type of labels defined as shown in Table 1.

BERT-NER is a family of BERT-based models specifically fine-tuned on NER [22]. They are trained on the CoNLL NER dataset [26] which consists of annotated news stories for NER, and achieve a state-of-the-art F1 score of 93.79%. While generative PLMs have shown incredible zero-shot performance at various NLP tasks it has been shown that for Standard-IE settings BERT-based models outperform generative models [20]. There does exist promising research into increasing their NLU performance by reformulating these inherently non-generative NLP tasks into text generation problems with prompt engineering [25].

| | | | | | | |
|---|---|---|---|---|---|---|
| Harry | Potter | entered | Ho | ##g | ##wart | ##s |
| B-PER | I-PER | O | B-LOC | I-LOC | I-LOC | I-LOC |

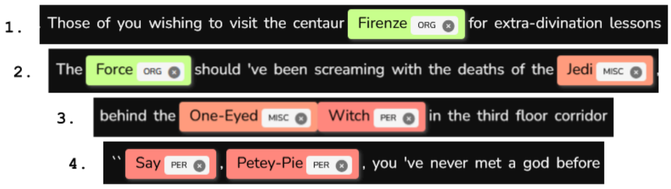
Table 1: Tokenized Text labeled for NER

### C. Fanfiction as Unique Domain

Fanfiction is a highly diverse form of literature. While its characteristics provide many advantages, these also come with their unique challenges and limitations. The lack of an editorial process or marketing influences allows them to reflect a wider range of social movements and writing styles, but it also makes it harder to generalise NLP techniques. Inconsistent use of capitalization, story structure and the use of editor notes within the text all make creating a fully automatic NLP pipeline a difficult endeavor.

For PLM based approaches another challenge arises from the inherent bias of these models originating from their training data. Since most PLMs are pre-trained on corpora containing (english) literature and/or Wikipedia, these models will contain 'knowledge' from texts contained in their pre-training set. Research has shown that PLMs perform better on IE tasks for books contained in their pre-training set than on those that are not [27]. Fanfiction often makes changes to the source canon; changing properties of characters such as gender or relationships, placing characters in different literary universes (cross-universe), or changing the original genre of the text. When PLMs are applied for NLP tasks to fanfiction their answers could therefore be biased and based on knowledge learned from pretraining instead of the given context.

Figure 2 shows some examples of problems for NER within fanfiction, specifically where fictional entities are being misclassified. While the reader presumably has knowledge of the original source text and thus is familiar with this entity, the BERT model does not have this knowledge since the original works are not contained in the training data of BERT (with the exception of information contained on Wikipedia which is in BERTs training data).



1. Those of you wishing to visit the centaur Firenze ORG for extra-divination lessons
2. The Force ORG should 've been screaming with the deaths of the Jedi MISC
3. behind the One-Eyed MISC Witch PER in the third floor corridor
4. `` Say PER , Petey-Pie PER , you 've never met a god before

Figure 2: Examples of fanfiction texts annotated by the `dslim/bert-base-NER` model.

## 3 Related Research

Many studies on fanfiction text using Data Science and/or NLP have followed since the paper on a Computational Analysis of Fanfiction by Milli and Bamman [5], such as [3], [4], [6]. However, these studies use existing NLP instead of researching their effectiveness in this domain. There does exist one study for this topic, namely FanfictionNLP [8]. Yoder et al. created a NLP pipeline specifically for Coreference Resolution (e.g. matching he/she with character names), Quote and Assertion Attribution. The BERT model variant called spanBERT [28] is used for these IE tasks, since its architecture allows it to perform better on these span selection IE tasks. SpanBERT is pretrained on the same dataset as BERT-base, so we hypothesise that the performance of IE tasks on fanfiction text of spanBERT, and thus FanfictionNLP, could be improved through domain adaptation.

This paper is also inspired by the work of Gururanganet al. [29], which investigated the potential of additional pretraining to better adapt PLMs to other domains and target tasks through either Domain Adaptive Pre-Training (DAPT), Task Adaptive Pre-Training (TAPT) or the combination of both (see Figure 3). These methods are performed before fine-tuning and don't require annotated datasets, which is advantageous for domains or tasks where annotated data is limited or the cost of annotating data is too high.

The success of domain and task adaptation of BERT has been demonstrated numerous times. Examples of domain adaptation include LEGAL-BERT [18] or BioBERT [19], which increase performance of IE tasks within the legal and biomedical domain respectively. For BioBERT first DAPT is applied on unannotated Biomedical texts, followed by task-specific fine-tuning on 3 different IE tasks. LEGAL-BERT follows the same methodology, but also investigates the potential of pre-training entirely on domain-specific corpora.

Examples of task adaptation include BERT-NER [22], which is a BERT-base model with a token classification layer added fine-tuned on annotated NER data, namely the CoNLL dataset [26]. This dataset is a widely recognized benchmark dataset for NER and consists of annotated english news stories. A further review of the application of PLMs for NER can be found at [21], [30].

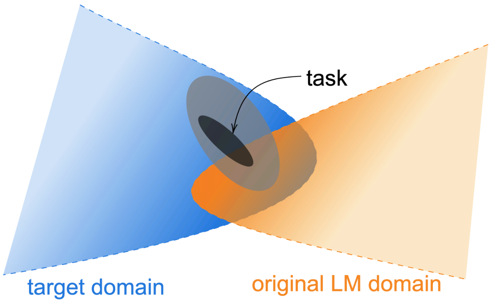


Figure 3: An illustration of data distributions. DAPT is performed on data from the target domain, and TAPT is done on data from the task domain. [29]

## 4 Methodology

### *A. Model Selection*

The `bert-base-cased`[3] model is chosen over other BERT-based architectures for its state-of-the-art performance on NER as shown by [30]. This also allows for a more meaningful evaluation of the resulting `bert-fanfic-NER`, since the resulting model can be compared to the existing BERT-NER (specifically `bert-base-NER`[4]) which is also derived from the pretrained BERT-base model.

### *B. Dataset Collection*

In order to train and evaluate the models for NER, 10 fanfiction text were collected from *Archive Of Our Own*[5]. The stories were semi-randomly (curated) selected in order to guarantee that they are English, between 500 and 2000 words, and each based on a different source canon to reduce overfitting onto a single genre or canon.

Unsupervised annotations were generated using an existing PLM and then verified and corrected by a human annotator using a NER annotator tool[6] adhering to the Universal NER Annotation Guidelines[7]. The Label Counts of the resulting dataset can be seen in Table 1.

Since these texts are of greater length than the model's context window, the texts are chunked into parts of 256 Tokens. In order to preserve context for Token Classification, these chunks are padded to the left and right with the tokens of the context up to the models maximum context window size of 512. The resulting dataset consists of 160 entries.

[3] https://huggingface.co/google-bert/bert-base-cased
[4] https://huggingface.co/dslim/bert-base-NER
[5] https://archiveofourown.org/
[6] https://github.com/tecoholic/ner-annotator
[7] http://www.universalner.org/guidelines/

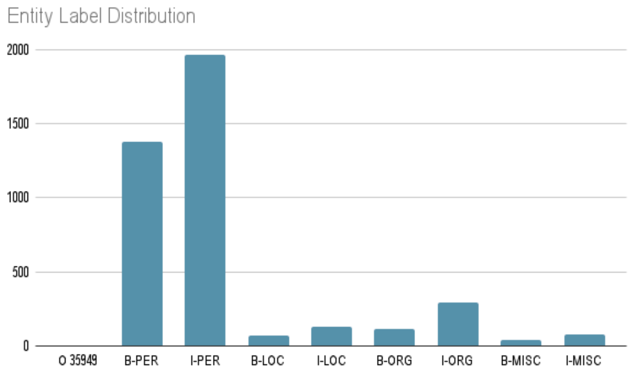


Table 1: Label Counts for the 10 human annotated fanfiction stories

To accurately evaluate the model during and after training, the dataset was randomly divided into a *training* split, a *validation* split and a *test* split. The *validation* set is used during the training process to monitor the progress and adjust hyperparameters such as learning rate. This split is important to measure the model's ability to generalize to unseen data during training and thus reduce the risk of overfitting. The final model performance is measured on the *test* set.

However, since each story was divided into multiple chunks, each split would contain entries of all different stories thereby potentially 'contaminating' the test set with entities the model was directly trained on. This would limit the final evaluation to accurately measure the models ability to generalise to unseen data. A second test set was therefore created in order to detect overfitting by leaving out entries of one fanfiction text of a different topic called *overfit_test*, on which `bert-fanfic-NER`'s generalisation ability can be more accurately evaluated. An overview of the resulting dataset and splits can be seen in Table 3.

| dataset split | size |
|---|---|
| *training* | 120 |
| *evaluation* | 14 |
| *test* | 15 |
| *overfit test* | 11 |

Table 3: Overview of the for NER annotated fanfiction dataset and the splits thereof used during training and evaluation.

### *C. Training Strategy*

The training was performed on using the Hugging Face Transformers[8] open-source Python Library, which uses the Adam optimizer for gradient-based optimization during training. Initially a learning rate of 0.00005 was selected. The classification accuracy during training is calculated as the proportion of correctly classified entity labels within an entry, whereby the padded context tokens of length 128 to the left and right are ignored. The model is then trained for a maximum of 16 epochs (a complete run through all data in the *training* split), or until no improvement of performance on the validation set is measured for 3 consecutive epochs. This is called *Early Stopping*, and is also a measure to avoid overfitting. The model was trained on a single Nvidia Quadro P1000 GPU.

### *D. Evaluation*

The final performance of `bert-fanfic-NER` is tested on the *test* split and the separated overfitting control set *overfit_test*. The Huggingface Evaluator package is used to calculate the performance scores. This class measures the precision, recall and F1 score for each class, where the F1 score is calculated as the harmonic mean of precision and recall thus ignoring True Negatives. This is a more meaningful metric than regular accuracy since for NER missed entities are as important as incorrectly labeled ones. Secondly the overall precision score across all classes is heavily skewed towards the accuracy of the far more numerous 'O' label, making this score less insightful then the overall F1 score.

The model is also tested on the CoNLL dataset for further insight into its generalization ability. As a baseline for the metrics, the performance of the BERT-NER model is used.

## 4 Results

### *A. Training*

The training of the model took a total of 32 minutes and all 16 epochs (1 epoch = 7.5 training steps), no early stopping was reached. The F1 score and loss measured on the *evaluation* split can be seen in tables 5&6.

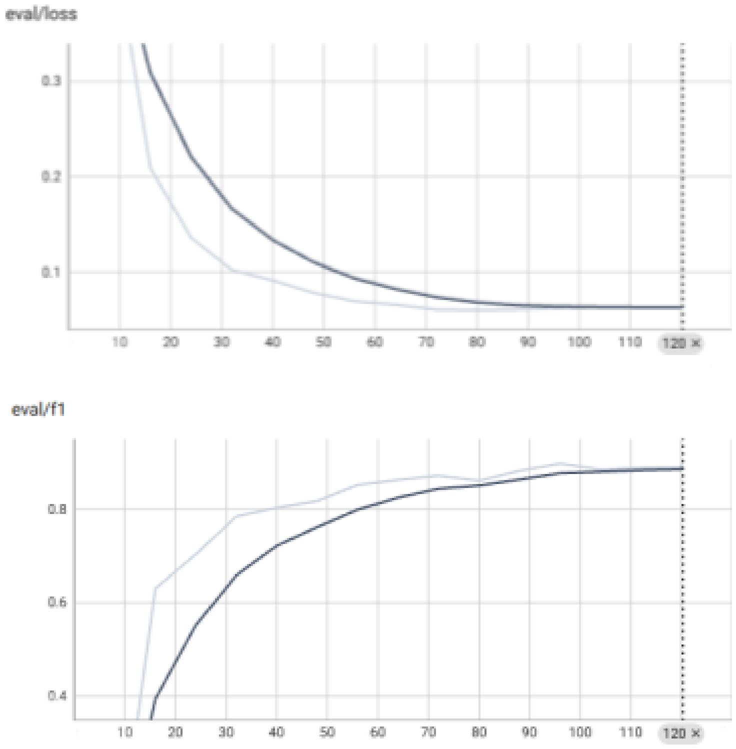


Table 5&6: The F1 accuracy score and loss of `bert-fanfic-NER` during training

### *B. Results*

The results are shown below (Table 7&8). The reported performance of BERT-NER corresponds to the performance measured during the experiment. When evaluated on the fanfiction dataset (`ALL FANFIC`) containing all annotated fanfiction texts, a slight decrease in performance is observed

[8] https://huggingface.co/docs/transformers/en/index

for the PER class and a major loss of performance on all other labels, resulting in a significant drop in overall F1 score of 7%.

BERT-fanfic-NER's final overall F1 score evaluated on the test set is 89%, which is 6% higher than BERT-NER. The evaluation on the overfit test resulted in a higher accuracy of 90% total F1 score. Noteworthy is the poor performance of BERT-fanfic-NER on the CoNLL dataset with an F1 score of 24%.

*Evaluation Scores of BERT-NER and BERT-FANFIC-NER*

| Dataset<br>Model | CoNLL | | TEST-SPLIT | | OVERFIT-TEST | | ALL FANFIC | |
|---|---|---|---|---|---|---|---|---|
| | B-NER | B-FF | B-NER | B-FF | B-NER | B-FF | B-NER | B-FF |
| PER: Precision | 0.96 | 0.40 | 0.95 | 0.96 | 0.92 | 0.92 | 0.93 | - |
| Recall | 0.96 | 0.55 | 0.87 | 0.97 | 0.89 | 0.97 | 0.90 | - |
| F1 | 0.96 | 0.46 | 0.92 | 0.96 | 0.90 | 0.95 | 0.91 | - |
| LOC: Precision | 0.93 | 0.44 | 0.27 | 0.45 | 1.0 | 1.0 | 0.68 | - |
| Recall | 0.93 | 0.02 | 0.71 | 0.71 | 0.57 | 0.42 | 0.69 | - |
| F1 | 0.93 | 0.03 | 0.40 | 0.56 | 0.73 | 0.60 | 0.68 | - |
| ORG: Precision | 0.89 | 0.25 | 0.21 | 0.40 | 0.0 | 0.0 | 0.37 | - |
| Recall | 0.91 | 0.04 | 0.14 | 0.48 | 0.0 | 0.0 | 0.36 | - |
| F1 | 0.90 | 0.07 | 0.17 | 0.44 | 0.0 | 0.0 | 0.37 | - |
| MISC: Precision | 0.78 | 0.0 | 0.43 | 0.34 | 0.66 | 0.0 | 0.25 | - |
| Recall | 0.83 | 0.0 | 0.64 | 0.14 | 1.0 | 0.0 | 0.57 | - |
| F1 | 0.80 | 0.0 | 0.51 | 0.20 | 0.8 | 0.0 | 0.35 | - |
| | | | | | | | | |
| overall precision | 0.90 | 0.39 | 0.84 | 0.89 | 0.87 | 0.92 | 0.84 | - |
| overall recall | 0.92 | 0.17 | 0.82 | 0.90 | 0.86 | 0.89 | 0.84 | - |
| overall f1 | 0.91 | **0.24** | 0.83 | **0.89** | 0.86 | **0.90** | 0.84 | - |
| overall accuracy | 0.98 | 0.87 | 0.98 | 0.99 | 0.99 | 0.99 | 0.99 | - |

Table 7: Evaluation Scores of BERT-NER (B-NER) and BERT-FANFIC-NER (B-FF) on 4 different datasets. The CoNLL general NER dataset (CoNLL), The test split of the NER annotated fanfiction dataset (TEST-SPLIT), a separate differing topic fanfiction text dataset to test overfitting onto specific entities present in the training (OVERFIT-TEST), and the complete annotated fanfiction text dataset for establishing a baseline performance of BERT-NER.

| Dataset | CoNLL-TEST | TEST-SPLIT | OVERFIT-TEST | ALL FANFIC |
|---|---|---|---|---|
| O | 38323 | 6291 | 2449 | 34291 |
| PER | 1617 | 265 | 75 | 1371 |
| LOC | 1661 | 7 | 7 | 70 |
| ORG | 1668 | 21 | 1 | 112 |
| MISC | 702 | 14 | 2 | 44 |
| Total Entities | 5648 | 307 | 85 | 1597 |

Table 8: Label count of the different datasets shown in Table 7.

Table 8 shows the total amount of labels present in each dataset. The label distribution of ALL FANFIC is added as comparison, since BERT-fanfic-NER is trained on a subset of this data. The counts of the CoNLL dataset shown are more specifically from the 'test' split, which constitutes around 10% of the entire dataset.

## 5 Discussion

### *A. Fanfiction as Unique Literary Domain*

The baseline provided by BERT-NER suggests that fanfiction texts indeed differ from other types of texts, since the performance of BERT-NER dropped by 7% (from 91% F1 score to 84%) when evaluated on the fanfiction dataset. This suggests our hypothesis that fanfiction writing exhibits significant distinct linguistic characteristics in regards to regular fictional writing, and therefore NLP techniques utilizing PLMs developed for regular fictional literature under-perform when applied to the domain of fanfiction is correct. We note specifically a drop in performance on ORG, LOC and MISC entities. This could be the result of the fictional nature of these names. Since the BERT-NER model is fine-tuned on non-fictional texts for NER it could lack training in recognizing fictional entities. The high score of the PER however indicates that this is not the case, causing the reason for this drop of performance to be inconclusive. We suggest further research should be conducted to investigate these findings.

### *B. Data Diversity and Contamination*

While results initially look promising, some skepticism is in order because of the relatively small dataset size. In comparison, the BERT-NER model was trained on 14041 entries containing 23499 labeled entities, BERT-fanfic-NER was trained on only 120 entries containing around 2500 labeled entities. Due to the limited amount of fanfiction stories, there also exists a general lack of diversity in the entity types. This endangers the model to overfitting onto these specific entities instead of generalizing to different data. The poor performance of BERT-fanfic-NER on CoNLL implies that the model hasn't generalised well to other data outside of fanfiction. Yet on the overfit test BERT-fanfic-NER performed better than BERT-NER, with an overall increase in F1 score of 4%. This leads to the conclusion that while BERT-fanfic-NER did not generalise to CoNLL, it was able to generalize to other fanfiction data. However, due to the small dataset size of the overfit-test we deem these results inconclusive. For example, of the 75 PER labels of the overfit test, a large portion consists of identical PER entities, namely the protagonists. When the model performs well on this name, its overall performance on that story will significantly improve while performance on other texts might be poor.

In addition, the final performance of BERT-fanfiction-NER was measured on the test-set. This set contained a random selection of sentences from the nine fanfiction stories, meaning the model was directly trained on most of the entity names contained in the test-set. This 'contamination' of the dataset is less of an issue for larger datasets when the pool of entities is larger, such as in the CoNLL dataset.

### C. Erroneous Annotations and Implementations

The possibility of erroneous annotations and implementations must also be considered. While the effect of several mis-labeled entities should be negligible, in cases with small amounts of total labels (e.g. only 7 locations in the test-set) these could account for large differences in evaluation scores. For this experiment multiple data processing steps have been implemented in order to e.g. convert the json SpaCy annotations files into huggingface datasets suitable for model training and evaluation. Small implementation errors could e.g. cause misalignment in token labels sabotaging training and evaluation, leading to erratic results.

### D. Recommendations

The evaluation of BERT-NER on fanfiction text implies the need for further investigation of the performance of NLP tasks by PLMs on fanfiction texts. Our findings suggest that PLMs perform worse when applied within the domain of fanfiction. They are used in numerous interdisciplinary studies on fanfiction for NLP tasks, and the results of those studies might be affected by a worse-than-reported accuracy of these models. Due to computational limitations of this study, investigating the potential of Domain-Adaptive Pre-Training (DAPT) on fanfiction text for a PLM was out of scope for this research. Yet we indicate such a study to be worthwhile, since DAPT itself does not require annotated data. Annotated data is only needed for the final fine-tuning step, and it can be investigated if existing datasets such as CoNLL could be used for this. Meaning the performance of NLP by PLMs could be improved by creating a DAPT model which can be fine-tuned with existing task-specific data for downstream tasks.

### E. Responsible Research

With the widespread applications and popularity of PLMs, a great deal of concern has also been raised about the ethicality and safety of PLMs [12], as well as cautionary messages about their high training power costs and thus, depending on the power source, generate a large amount of carbon emissions. This study only fine-tuned PLMs on limited datasets, and while these power costs are still significant they are several magnitudes lower than the power requirements of pre-training an entire LLM from scratch [31].

Furthermore, when evaluating PLMs there exists the risk of data contamination, where models are evaluated on data contained in their training set. LLMs such as GPT are therefore unsuitable for a wide range of research, since the contents of its training data is not disclosed. In this study the BERT model was used which has disclosed its training set, and it was verified that fanfiction texts of any kind were not included in this training data. For future researchers the code, model and (privatised) dataset will be published for replicability of this study.

## 7 Conclusion

This research paper found that existing state-of-the-art PLMs fine-tuned for the NER task (BERT-NER) under-performed when applied to the domain of fanfiction. This implies that fanfiction writing exhibits significant distinct linguistic characteristics in regards to regular fictional writing, and signals the need for further investigation of the performance of other PLMs when applied to fanfiction texts. To investigate the potential of domain adaptation to fanfiction of PLMs, a BERT-based PLM was fine-tuned with annotated fanfiction texts for the downstream NLP task of Named Entity Recognition (NER). We show that the resulting model, BERT-fanfic-NER, achieves an improvement in F1 score of 6% in regards to BERT-NER. However, due to the limited dataset size, we are critical of these results and show overfitting occurs. Indicating future efforts to adapt PLM to the domain of fanfiction for NLP either require larger annotated datasets or should investigate the potential of different training approaches such as DAPT.

## 8 References

[1] J. L. Barnes, “Fanfiction as imaginary play: What fan-written stories can tell us about the cognitive science of fiction,” *Poetics*, vol. 48, pp. 69–82, Feb. 2015, doi: 10.1016/j.poetic.2014.12.004.

[2] Bronwen Thomas, “What Is Fanfiction and Why Are People Saying Such Nice Things about It??,” *Storyworlds J. Narrat. Stud.*, vol. 3, p. 1, 2011, doi: 10.5250/storyworlds.3.2011.0001.

[3] M. Jacobsen, Y. Bizzoni, P. F. Moreira, and K. L. Nielbo, “Patterns of Quality: Comparing Reader Reception Across Fanfiction and Commercially Published Literature,” in *Proceedings of the Computational Humanities Research Conference 2024*, W. Haverals, M. Koolen, and L. Thompson, Eds., in CEUR Workshop Proceedings, vol. 3834. Aarhus, Denmark: CEUR, Dec. 2024, pp. 718–739. Accessed: Jan. 07, 2025. [Online]. Available: https://ceur-ws.org/Vol-3834/#paper106

[4] A. Mattei, D. Brunato, and F. Dell’Orletta, “The Style of a Successful Story: a Computational Study on the Fanfiction Genre,” in *Proceedings of the Seventh Italian Conference on Computational Linguistics CLiC-it 2020*, F. Dell’Orletta, J. Monti, and F. Tamburini, Eds., Accademia University Press, 2020, pp. 284–289. doi: 10.4000/books.aaccademia.8718.

[5] S. Milli and D. Bamman, “Beyond Canonical Texts: A Computational Analysis of Fanfiction,” in *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, Austin, Texas: Association for Computational Linguistics, 2016, pp. 2048–2053. doi: 10.18653/v1/D16-1218.

[6] D. Nguyen, S. Zigmond, S. Glassco, B. Tran, and P. J. Giabbanelli, “Big data meets storytelling: using machine learning to predict popular fanfiction,” *Soc. Netw. Anal. Min.*, vol. 14, no. 1, p. 58, Mar. 2024, doi: 10.1007/s13278-024-01224-x.

[7] V. Veríssimo, “A narratological study of fanfiction writing,” 2022.

[8] M. Yoder *et al.*, “FanfictionNLP: A Text Processing Pipeline for Fanfiction,” in *Proceedings of the Third Workshop on Narrative Understanding*, Virtual: Association for Computational Linguistics, 2021, pp. 13–23. doi: 10.18653/v1/2021.nuse-1.2.

[9] F. Ferris, “The Myth of Fanfiction: An Examination of Two Deeply Connected Traditions of Storytelling,” *Undergrad. Stud. Res. Internsh. Conf.*, Aug. 2022, [Online]. Available:

https://ir.lib.uwo.ca/usri/usri2022/ReOS/277

[10] K. Yin, C. Aragon, S. Evans, and K. Davis, “Where No One Has Gone Before: A Meta-Dataset of the World’s Largest Fanfiction Repository,” 2017, *Social Science Research Network, Rochester, NY*: 2982568. Accessed: Jan. 12, 2025. [Online]. Available: https://papers.ssrn.com/abstract=2982568

[11] A. Vaswani *et al.*, “Attention Is All You Need,” Aug. 02, 2023, *arXiv*: arXiv:1706.03762. doi: 10.48550/arXiv.1706.03762.

[12] A. Zubiaga, “Natural language processing in the era of large language models,” *Front. Artif. Intell.*, vol. 6, p. 1350306, Jan. 2024, doi: 10.3389/frai.2023.1350306.

[13] M. Ren, “Advancements and Applications of Large Language Models in Natural Language Processing: A Comprehensive Review,” *Appl. Comput. Eng.*, vol. 97, no. 1, pp. 55–63, Nov. 2024, doi: 10.54254/2755-2721/97/20241406.

[14] L. Qin *et al.*, “Large Language Models Meet NLP: A Survey,” May 21, 2024, *arXiv*: arXiv:2405.12819. doi: 10.48550/arXiv.2405.12819.

[15] B. Min *et al.*, “Recent Advances in Natural Language Processing via Large Pre-trained Language Models: A Survey,” *ACM Comput Surv*, vol. 56, no. 2, p. 30:1-30:40, Sep. 2023, doi: 10.1145/3605943.

[16] M. Mars, “From Word Embeddings to Pre-Trained Language Models: A State-of-the-Art Walkthrough,” *Appl. Sci.*, vol. 12, no. 17, Art. no. 17, Jan. 2022, doi: 10.3390/app12178805.

[17] M. U. Hadi *et al.*, *Large Language Models: A Comprehensive Survey of its Applications, Challenges, Limitations, and Future Prospects*. 2023. doi: 10.36227/techrxiv.23589741.

[18] I. Chalkidis, M. Fergadiotis, P. Malakasiotis, N. Aletras, and I. Androutsopoulos, “LEGAL-BERT: The Muppets straight out of Law School,” in *Findings of the Association for Computational Linguistics: EMNLP 2020*, T. Cohn, Y. He, and Y. Liu, Eds., Online: Association for Computational Linguistics, Nov. 2020, pp. 2898–2904. doi: 10.18653/v1/2020.findings-emnlp.261.

[19] J. Lee *et al.*, “BioBERT: a pre-trained biomedical language representation model for biomedical text mining,” *Bioinformatics*, vol. 36, no. 4, pp. 1234–1240, Feb. 2020, doi: 10.1093/bioinformatics/btz682.

[20] C. Shi, Y. Su, C. Yang, Y. Yang, and D. Cai, “Specialist or Generalist? Instruction Tuning for Specific NLP Tasks,” Oct. 23, 2023, *arXiv*: arXiv:2310.15326. doi: 10.48550/arXiv.2310.15326.

[21] Z. Nasar, S. W. Jaffry, and M. K. Malik, “Named Entity Recognition and Relation Extraction: State-of-the-Art,” *ACM Comput. Surv.*, vol. 54, no. 1, pp. 1–39, Jan. 2022, doi: 10.1145/3445965.

[22] N. Kaur, A. Saha, M. Swami, M. Singh, and R. Dalal, “Bert-Ner: A Transformer-Based Approach For Named Entity Recognition,” in *2024 15th International Conference on Computing Communication and Networking Technologies (ICCCNT)*, Jun. 2024, pp. 1–7. doi: 10.1109/ICCCNT61001.2024.10724703.

[23] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova, “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)*, J. Burstein, C. Doran, and T. Solorio, Eds., Minneapolis, Minnesota: Association for Computational Linguistics, Jun. 2019, pp. 4171–4186. doi: 10.18653/v1/N19-1423.

[24] A. Wang, A. Singh, J. Michael, F. Hill, O. Levy, and S. R. Bowman, “GLUE: A Multi-Task Benchmark and Analysis Platform for Natural Language Understanding,” Feb. 22, 2019, *arXiv*: arXiv:1804.07461. doi: 10.48550/arXiv.1804.07461.

[25] A. Radford, K. Narasimhan, T. Salimans, and I. Sutskever, “Improving Language Understanding by Generative Pre-Training,” 2018. Accessed: Jan. 11, 2025. [Online]. Available: https://www.semanticscholar.org/paper/Improving-Language-Understanding-by-Generative-Radford-Narasimhan/cd18800a0fe0b668a1cc19f2ec95b5003d0a5035

[26] E. F. T. K. Sang and F. D. Meulder, “Introduction to the CoNLL-2003 Shared Task: Language-Independent Named Entity Recognition,” Jun. 12, 2003, *arXiv*: arXiv:cs/0306050. doi: 10.48550/arXiv.cs/0306050.

[27] K. K. Chang, M. Cramer, S. Soni, and D. Bamman, “Speak, Memory: An Archaeology of Books Known to ChatGPT/GPT-4,” Oct. 20, 2023, *arXiv*: arXiv:2305.00118. doi: 10.48550/arXiv.2305.00118.

[28] M. Joshi, D. Chen, Y. Liu, D. S. Weld, L. Zettlemoyer, and O. Levy, “SpanBERT: Improving Pre-training by Representing and Predicting Spans,” Jan. 18, 2020, *arXiv*: arXiv:1907.10529. doi: 10.48550/arXiv.1907.10529.

[29] S. Gururangan *et al.*, “Don`t Stop Pretraining: Adapt Language Models to Domains and Tasks,” in *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, D. Jurafsky, J. Chai, N. Schluter, and J. Tetreault, Eds., Online: Association for Computational Linguistics, Jul. 2020, pp. 8342–8360. doi: 10.18653/v1/2020.acl-main.740.

[30] P. K. R, B. M. G, P. Srinivasan, and V. R, “Transformer-Based Models for Named Entity Recognition: A Comparative Study,” in *2023 14th International Conference on Computing Communication and Networking Technologies (ICCCNT)*, Delhi, India: IEEE, Jul. 2023, pp. 1–5. doi: 10.1109/ICCCNT56998.2023.10308039.

[31] “Towards Carbon-efficient LLM Life Cycle,” 2024.