A Multilingual Evaluation of NER Robustness to Adversarial Inputs

Akshay Srinivasan
University of Ottawa, Canada *
asrin033@uottawa.ca

Sowmya Vajjala
National Research Council, Canada
sowmya.vajjala@nrc-cnrc.gc.ca

Abstract

Adversarial evaluations of language models typically focus on English alone. In this paper, we performed a multilingual evaluation of Named Entity Recognition (NER) in terms of its robustness to small perturbations in the input. Our results showed the NER models we explored across three languages (English, German and Hindi) are not very robust to such changes, as indicated by the fluctuations in the overall F1 score as well as in a more fine-grained evaluation. With that knowledge, we further explored whether it is possible to improve the existing NER models using a part of the generated adversarial data sets as augmented training data to train a new NER model or as fine-tuning data to adapt an existing NER model. Our results showed that both these approaches improve performance on the original as well as adversarial test sets. While there is no significant difference between the two approaches for English, re-training is significantly better than fine-tuning for German and Hindi.

1 Introduction

NLP systems are traditionally evaluated and compared against a gold standard, which is generally immutable. Recent research has shown that even the NLP systems that perform well on the standard test set show a significant drop in performance even for small perturbations in the input test data, across a range of NLP tasks (Gardner et al., 2020). Although this strand of research covered many tasks, it has been exclusively focused on English, with a few exceptions (e.g., Shmidman et al. (2020), for Hebrew). Further, to our knowledge, the primary usage of such adversarial test sets have been in either evaluating NLP models or in usage as additional, augmented data to improve model robustness, without much focus on using the new data for fine-tuning, instead of re-training. A better understanding of fine-tuning with adversarial test sets is important, and useful in most real-world scenarios, where we may not have access to the original training data while having access to the trained model itself.

Named Entity Recognition (NER) is among the most common NLP tasks both in research and in industry applications (Lorica and Nathan, 2021). Although much progress has been made on NER over the past decades, existing NER systems were also shown to be sensitive to small changes in input data in the past (Lin et al., 2021; Vajjala and Balasubramaniam, 2022). Table 1 shows an example of how predictions can change with minor changes in input, for one of the state of the art NER models[1]. Going by the original sentence, all three sentences should carry the `LOC` tag for the entity in the sentence. However, that is not the case, as the outputs shows. Clearly, small, and seemingly harmless changes are changing model predictions.

Original: It was the second costly blunder by **Syria_LOC** in four minutes .
Altered: It was the second costly blunder by **Hyderabad_ORG** in four minutes .
Altered: It was the second costly blunder by **Hyderabad_LOC** in four hours .

Table 1: Illustration of an NER model's predictions with minor changes to an original test set sentence

Even recent large language models such as ChatGPT struggle with sequence tagging tasks such as NER, across multiple languages(Qin et al., 2023; Lai et al., 2023; Wu et al., 2023), which clearly illustrates that NER is far from being considered solved. In this backdrop, considering the significance of NER in research and practical scenarios, a better understanding of how a model's predictions

* Work done during an internship at National Research Council, Canada

[1] https://huggingface.co/flair/ner-english

Proceedings of the 8th Workshop on Representation Learning for NLP (RepL4NLP 2023), pages 40–53

change with slight changes in input becomes an important issue to address. Hence, we explore the following questions in this paper:

1. How does the performance of NER models across three languages change with small changes to the original input?

2. How does retraining an NER model with adversarial data augmentation compare with adversarial fine-tuning of NER across languages?

Our contributions are summarized as follows:

- We conducted the first comparative study of the robustness of NER models beyond English, covering three languages, in a space where all previous work focused on English alone.

- We report first results on the comparison between data augmentation and adversarial fine-tuning for NER, for all the three languages.

- We show how existing methods for data augmentation can be repurposed to develop language-agnostic methods to generate adversarial test sets for NER.

Starting with a conceptual background (Section 2), we describe our methods for adversarial dataset creation (Section 3) and the general experimental setup (Section 4) followed by a detailed discussion of our results (Section 5) and a summary (Section 6), focusing on the ethical impacts, limitations and broader impact towards the end.

2 Related Work

Evaluating using multiple datasets is one of the ways to assess the robustness and generalization capabilities of NLP models. Developing challenge sets, and generating adversarial datasets that can potentially cause a model to fail, are some possibilities in this direction (Isabelle et al., 2017; Ettinger et al., 2017; Glockner et al., 2018; Gardner et al., 2020). Adversarial data generation in NLP focuses on surface-level perturbations to the input text, proposing various means of insertion/deletion/swapping of words/characters/sentences (Jia and Liang, 2017; Gao et al., 2018; Ribeiro et al., 2018). Other approaches such as paraphrasing (Iyyer et al., 2018), generating semantically similar text using other deep learning models (Zhao et al., 2018; Michel et al., 2019), using a human-in-the-loop (Wallace et al., 2019b) were also explored in the past. While many of the proposed methods are black-box approaches, assuming no knowledge about the NLP models themselves, some of the approaches are white box, with more access to the inner workings of a model (Liang et al., 2018; Blohm et al., 2018; Wallace et al., 2019a), and some models implement both (Li et al., 2019). We focus on one specific NLP task - NER, and only work on black-box methods in this paper.

In terms of the strategies to protect models against adversarial attacks, the most common approach followed by past NLP research has been to incorporate adversarial data into the training process, through data augmentation, or using adversarial training as a regularization method (See Goyal et al. (2022) and Zhang et al. (2020) for a detailed overview). Using adversarial data for full re-training of a model (as is the case with data augmentation) assumes access to the original data and the model, which is not practical in many real-world scenarios. One possibility to explore in such cases is to test whether using adversarial data to fine-tune a trained NER model improves its robustness. We compare using the adversarial data (through data augmentation) for re-training an NER model versus using it only for fine-tuning a previously trained NER model in this paper.

Adversarial Testing and Data Augmentation in NER: Adversarial testing approaches for NER in the previous work was entirely done for English datasets, and primarily focused on methods replace entities in the original test set with new ones using gazetteers or other means (Agarwal et al., 2020; Vajjala and Balasubramaniam, 2022). Lin et al. (2021) used entity linking and masked language models coupled with an existing NER model to generate adversarial test sets for NER. More recently, Das and Paik (2022) used grammatical case information to generate adversarial test sets for NER. Simoncini and Spanakis (2021) proposed other ways to make small changes to the context around entities in a sentence, to generate adversarial test sets. Other related work (Mathew et al., 2019; Ding et al., 2020; Zhu et al., 2021) focused on data augmentation for NER, that require training of new, additional models. While we use our adversarial datasets for data augmentation too later in the paper, the novelty of the current research, compared to this existing body

of work focusing on NER, comes in two forms:

1. While all previous work exclusively focused on English NER so far, we perform experiments with three languages - English, German and Hindi.

2. The approaches we used are lightweight, language agnostic means to generated adversarial test sets for NER, which do not rely on the availability of additional tools like entity linkers, and do not also need any additional training to generate the datasets.

3 Adversarial Test Set Creation

Our adversarial dataset creation methods can be broadly classified into two approaches - replacing entities and changing contexts. All except one method work for all the three languages we tested, and can be easily expanded to add other languages. Relevant code and generated datasets are provided as supplementary material[2].

3.1 Replacing Entities

We implemented two methods that replace the entity occurrences in the test set with another entity of the same category, keeping the rest of the sentence unchanged. Thus, they don't change the grammatical structure of the sentence and tell us how much the NER systems learn beyond memorizing the entities.

Random Sampling (*RS*) All the entity occurrences of the same type are shuffled throughout the test set in this approach. This is a simple and easily portable method across languages, which could serve as a strong baseline.

Gazetteers (*Faker*) This approach replaces existing entities with new entities of the same type using an existing gazetteer. Faker[3] is a python library that generates fake data for various application purposes, which supports multiple languages, and regions. We used it to replace the *Person* and *Location* entities in all the datasets, in all three languages we experimented with. Vajjala and Balasubramaniam (2022) used Faker for adversarial NER test sets, but they used only English (on OntoNotes dataset). We randomly choose among three locale settings for English (USA, Canada, India) and German (Germany, Austria, and Switzerland) respectively for replacement. For Hindi, there was only one locale setting provided (HI-IN).

3.2 Changing the Context

These approaches deal with changing the context in which the entities occur in a sentence by making small changes to the tokens around it.

Masking (*Mask*) We leveraged transformer based pre-trained language models trained with a masked language modeling objective to change the context in the original test datasets. We masked up to three randomly chosen non-entity tokens per sentence, and used the language model to generate those tokens, thereby creating new sentences with the same entities, but slightly altered contexts. Since there are multilingual pre-trained language models available, this approach is applicable to multiple languages.

Paraphrasing (*Para*) The objective behind this approach is to alter the structure of the input sentences, while keeping the named entities intact. About 500 sentences were randomly chosen from the test set and fed to an online, subscription based english paraphraser, Quillbot[4], which was shown to generate better paraphrases than other approaches such as back-translation or using GPT-3 in recent research (Shiri et al., 2022) and in our initial evaluations. The paraphrased output obtained for each sentence is then taken and the entity tokens from the original test set are mapped to the entity tokens of the paraphrased sentences with the respective entity tags, leaving the rest of the tokens as `O`. Limiting to 500 sentences is primarily due to the fact that Quillbot did not provide an API, and there is a limit to the amount of text one can paraphrase per request, even with subscription.

There are some challenges with this approach, though. While Quillbot's paraphrases when we choose the "Fluency" setting are of good quality, and are always grammatically correct, they sometimes alter the entities themselves (e.g., United States can become U.S. in the paraphrased version) or change the original tokenization of the dataset. In such cases where an automatic mapping between the entity tags from the original sentence and tokens of the paraphrased sentence failed, we discarded the sentence from our test set. Note that this

[2] https://dx.doi.org/10.6084/m9.figshare.22674079

[3] https://faker.readthedocs.io/

[4] https://quillbot.com/

Test set	Sentence
Orig	"We suspect that these killings are linked to politics," spokesman Bala Naidoo told Reuters.
RS	"We suspect that these killings are linked to politics," spokesman Deborah Compagnoni told Watford.
Faker	"We suspect that these killings are linked to politics," spokesman Jeremy Shukla told Reuters.
Mask	"We suspect the these killings are linked to politics," spokesman Bala Naidoo tells Reuters,
Para	"We assume that these killings are political in nature", spokesman Bala Naidoo told Reuters.
M+R	now we suspect that these killings are connected in politics, now spokesman Deborah Compagnoni told Watford.

Table 2: Adversarial variations generated for a test sentence from conll03-En

approach is compatible only with English and we aren't aware of any reliable paraphrasers for other languages. To our knowledge, full paraphrasing wasn't used for adversarial test sets in NER before. A total of 399 sentences from conll03-en, 373 sentences from mconer21-en, and 378 sentences from wnut17 were finally used as test sets.

3.3 Changing Entity + Context

Masking + Random Sampling (*M+R*) All the previous approaches focused on either trying to alter the context alone or replace the entities alone. This approach, does both by combining masking with random sampling. Since both these approaches are straightforward and can work across languages, they can be combined to create a new adversarial test set in all three languages. Table 2 shows one example of how the various approaches alter a single sentence from one of the English datasets. As with most data augmentation or adversarial generation approaches in NLP, some of the generated text may contain minor grammatical errors, as seen in in Mask and M+R settings. However, Compared to other common approaches such as those that involve insertion/deletion/swapping of words/characters or back-translation, the potential errors introduced by masking alone are minor. Further even the original datasets themselves contain sentences with such minor errors. So, we don't foresee this affecting the main findings of the paper. A publicly available NER model [5] predicts correctly for all these sentences.

4 Experimental Setup

We experimented with three English, two German, and one Hindi NER datasets [6]. The first half of our experiments focused on testing the NER models on adversarial test sets, following which we compared the effects of adversarial fine-tuning and data augmentation on the performance of the NER models. All the experiments were carried out on a high performing computing resource that ran an Nvidia A100 GPU with 32 GB RAM. Further details on the experimental setup are as follows.

4.1 Datasets

NER datasets from three popular shared tasks - conll03 (Tjong Kim Sang and De Meulder, 2003), multiconer21 (Malmasi et al., 2022) and wnut17 (Derczynski et al., 2017) were considered and their corresponding language subsets for English (conll03-en, mconer21-en, wnut17), German (conll03-de, mconer21-de) and Hindi (mconer21-hi) were used. Conll03 datasets have a tag set with four entity types (PER, LOC, ORG and MISC), and multiconer21 and wnut17 share the tag set consisting of six entity types (PER, LOC, CORP, GRP, CREATIVE-WORK and PROD). While the sentences in conll03 came from news articles, multiconer21 was collected from three domains (wikipedia sentences, questions, search queries), and wnut17 consisted of sentences from social media sources such as twitter, youtube and reddit.[7] We created five adversarial test sets for each of the three English datasets, four adversarial sets for each of the two German datasets and the Hindi dataset respectively, resulting in a total of 27 adversarial test sets covering three languages and six datasets.

4.2 NER Models

We used a combination of existing state of the art NER models (if available) and fine-tuning

[5] https://huggingface.co/flair/ner-english

[6] While it is theoretically possible to add more languages to this paper, the choice of these languages is motivated by the authors' familiarity with the languages, which is essential for qualitative analysis

[7] Some statistics about the datasets are shown in Appendix A.1.

a pre-trained language model for all the languages/datasets. They are explained below.

TNER and Fine-tuned BERT: TNER[8] is a Python library to train transformer based language models for NER tasks (Ushio and Camacho-Collados, 2021), which has pre-trained for mconer21-en, wnut17-en and conll03-en datasets. We use these models to perform our experiments for English NER. We will refer to this approach as *tner*. While TNER's model hub had publicly available models for other languages as well, the performance of the models was lower compared to the state of the art, and hence, we fine-tuned the multilingual BERT (Devlin et al., 2019) model hosted on Huggingface[9] for NER on German (conll03-de, mconer21-de) and Hindi (mconer21-hi) datasets. We will refer to this approach as *mbertft*.

We followed the same approach for the later re-training (i.e., training the NER model again using the original training data augmented with adversarial data) and fine-tuning (using adversarial data is used only to fine-tune the existing NER model) experiments in Section 5.4, and report the results with *tner* for English and *mbertft* for German and Hindi, as this setup gave the best results even in those experiments. For adversarial fine-tuning, *tner* and *mbertft* were fine-tuned for 4 epochs, learning rate was set to 0.0001 and the batch size was set to 16. For the Adversarial re-training however, the models were trained for 6 epochs since it is being trained from scratch, while the other hyper parameters were kept the same. The hyperparameter settings were from the original BERT paper (Devlin et al., 2019). We used 60% of the adversarial test set for data augmentation+training/fine-tuning and used the remaining 40% to test the approaches. For both adversarial fine-tuning and re-training, we report the average F1 score over 10 runs, with different random seeds, and compare them in terms of statistical significance using a paired t-test.

Stanza: Stanza (Qi et al., 2020) is a Python based NLP toolkit that hosts a few pre-trained NER models trained with a BiLSTM+CRF architecture. We evaluated Stanza's pre-trained conll-en and conll-de models using our generated adversarial test sets.

Flair: Flair is a popular NLP library that is widely used for performing NLP tasks (Akbik et al., 2019). We evaluated the pre-trained NER models provided by Flair for conll-en and conll-de.

There are other NER models that offer slightly better performance than Flair/Stanza/BERT-fine tuning, and there are other large language models to explore, but we focused on publicly available/downloadable models for NER and and easily re-implementatble benchmarks (e.g., BERT fine-tuning) in this paper. It would be interesting to extend this to cover additional methods in future, but we limit to a smaller set of models to maintain a manageable number of experiments and do a meaningful analysis later.

4.3 Evaluation

Micro-F1 score from seqeval (Nakayama, 2018) was used as the evaluation metric to test the robustness of the NER models, as it is the most commonly reported measure for this task.

Nervaluate: Nervaluate[10] is a Python library for performing a more fine-grained evaluation of NER, and is based on the metrics from a SemEval 2013 task (Segura-Bedmar et al., 2013). Apart from giving a single F1 score, it calculates the efficiency of the model using five error categories: correct, incorrect, partially correct, missing labels (an entity tagged as non-entity) and spurious labels(a non-entity tagged as entity). The error metrics are reported in four formats: strict (both entity span and entity type match), exact (entity span matches, irrespective of the type), partial (partial span match, irrespective of the type), and type (some overlap between gold annotation and system prediction). We used nervaluate to compare the performance of NER on the original and adversarial test sets, to understand what kind of errors affect their performance.

5 Results

Our experiments aimed at understanding the robustness of NER models to adversarial test sets and exploring whether adversarial data augmentation and fine tuning will help boost the performance on adversarial test sets. The results of these experiments are discussed below.

5.1 Adversarial Testing

Adversarial test sets were created by implementing the approaches mentioned in Section 3 and tested

[8] https://github.com/asahi417/tner

[9] https://huggingface.co/bert-base-multilingual-uncased

[10] https://github.com/MantisAI/nervaluate

on pre-trained NER models for all the three English datasets, and for conll03-de. As described earlier, we also fine-tuned a multilingual bert model for German and Hindi datasets. Tables 3, 4 and 5 summarize the performance of all the NER models we tested on, for English, German and Hindi respectively, in terms of the micro-F1 score.

	conll03-en			**wnut17**	**mconer21**
test set	tner	stanza	flair	tner	tner
Orig	0.91	0.92	0.92	0.60	0.81
RS	0.87	0.89	0.89	0.63	0.76
Faker	0.84	0.85	0.86	0.64	0.79
Mask	0.84	0.85	0.85	0.58	0.75
Para	0.80	0.72	0.78	0.65	0.64
M+R	0.82	0.85	0.83	0.53	0.77

Table 3: English NER performance (Micro-F1)

	conll03-de			**mconer21**
test set	mbertft	stanza	flair	mbertft
Orig	0.83	0.85	0.83	0.70
RS	0.80	0.82	0.78	0.58
Faker	0.81	0.85	0.81	0.55
Mask	0.78	0.81	0.81	0.53
M+R	0.75	0.80	0.76	0.50

Table 4: German NER performance (Micro-F1)

For English NER, results from Table 3 show a clear drop in NER model performance for two out of the three datasets (conll03 and mconer21), with the largest drop seen in the test set obtained by paraphrasing using Quillbot. However, it is important to note that the size of the test set for paraphrasing is far smaller (399, 373 and 378 sentences respectively for conll-03, mconer21 and wnut17) than the original test set, as explained earlier in Section 4. So, the drop is not directly comparable with other test sets which are larger in size.

Apart from this, there are also differences among individual methods for all the datasets. For example, Faker dataset appears to have had a stronger effect on all the three models trained on conll03-en dataset compared to multiconer-en dataset. Considering that there is generally similar drop across all three models of conll03-en, we would speculate that the difference in performance is due to the differences in the dataset composition and entity categories.

Test set	**mconer21-hi**
Orig	0.62
RS	0.55
Faker	0.61
Mask	0.52
M+R	0.48

Table 5: Hindi NER performance (Micro-F1)

For wnut17, the drop is largest for M+R, followed by Mask. Considering that only those test sets involving masking resulted in a drop for this dataset, we could safely attribute this drop to the difference in the nature of the data that the masked language model was exposed to, compared to the very noisy social media data in wnut17. An interesting aspect of testing with wnut17 model is that the model's performance was better on 3 out of 5 adversarial test sets, compared to the original test set. We believe it is important to note in this context that the NER model for wnut17 also has the lowest performance on the original test set among the three datasets and wnut17 is the noisiest data of them all (social media content). Considering that a model with a much lower overall F1 score, and trained on the most noisy dataset among the three, was still relatively more robust to three of the adversarial test sets, future research should perhaps also take a closer look at other, additional means of evaluating NER models instead of using the standard F1 score as the sole criterion to choose the best model, as was also suggested by other recent research on the topic. (Vajjala and Balasubramaniam, 2022).

For both German and Hindi NER (Table 4 and Table 5), we observe that the drop is the highest for masking+random sampling datasets in both the languages. Between the two German datasets, the drop in f1 scores for adversarial datasets appear to be much larger for mconer21 than conll03. A possible reason could be the poorer performance of the original mconer21 model itself.

While there are several other interesting results to compare and discuss across languages and datasets, overall, these results indicate that the NER models are not fully robust when tested against new test sets created with easily replicable, generally language-agnostic approaches. Their performance drop is larger when context altering approaches are employed. One question we asked ourselves at this point is: what exactly are the adversarial test sets changing in the model performance?

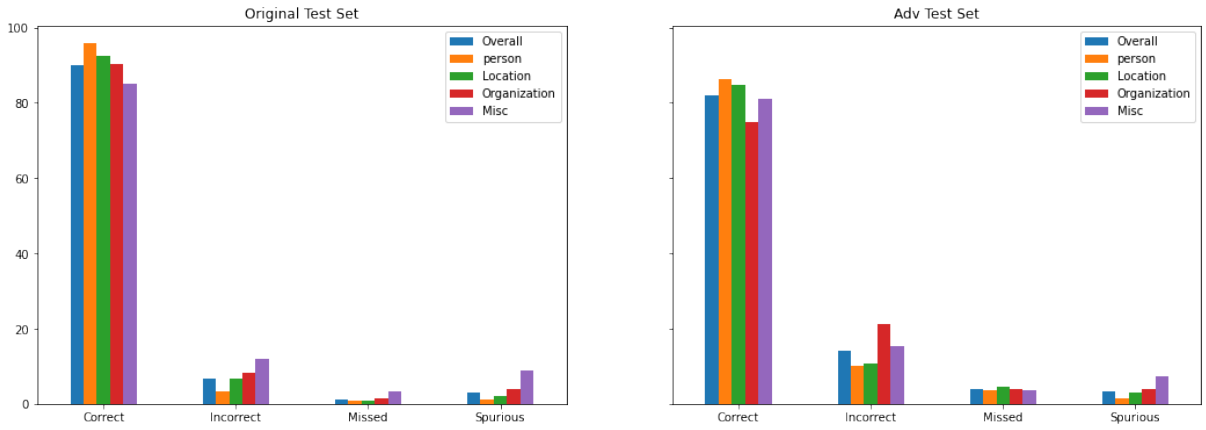

Figure 1: The performance (micro-F1) of an English NER model on original and M+R adversarial test sets

5.2 Fine-Grained Evaluation

We used nervaluate to understand what aspect of NER performance is mainly affected by the adversarial data. Since there are many models, train and test sets, we choose one train/test set and model combination for this analysis. Figure 1 shows the analysis for TNER's pre-trained NER model trained on conll03-en dataset, as a comparison between the original test set and the M+R adversarial test set (Figures 2 and 3 in the appendix show the same analysis for German and Hindi respectively).

Apart from the overall decline in performance, a closer look at the 'correct' and 'incorrect' categories indicate that this NER model's performance resulted in a larger drop for 'organization' entity type in English. This could be because of the ambiguity involved in the entity type itself, as not every 'organization' entity suits every context in which that entity type appears. Surprisingly, although the 'misc' category suffers from the same problem, we don't see a large dip for that category.

Figure 1 also indicates that there are more missed entities in the adversarial test set compared to spurious labels (non-entities tagged as one of the entity types). We compared the "strict" versus "exact" evaluation schema in nervaluate, to understand whether this increase in missed entities is a result of getting the span right, but identifying the entity type wrong. This comparison showed that while there is a 9% drop in the overall F1 score between the original and the adversarial test sets with 'strict' evaluation, there is only a 3% drop in terms of identifying the entity spans correctly (More details in Table 9 in the Appendix. Tables 10 and 11 in the appendix show this comparison for German and Hindi datasets).

5.3 Qualitative Error Analysis

Apart from quantitative analysis, we also did some manual analysis to understand what kind of transformations led NER models to predict erroneous tags. Table 6 shows some of the correctly tagged examples taken from conll03-en test set and their adversarial counterparts with some tagging errors, using Stanza's NER model.

When the entity 'Nicole' was replaced with the entity 'Major' (Examples **a** and **b**), the first one resulted in the model missing the entity (Major not recognised as person), and the second example saw the replaced entity 'Timor-Leste' (another name for the country East Timor) being mis-identified as an ORG instead of a LOC. While the transformed sentence in **c** appears very different from its source, there is only one entity, and a human reader would not find it difficult to identify the entity. However, the model missed identifying it altogether. Finally, both the masked examples (**d** and **e**) made very minor changes to the original sentence. But the model predictions changed from ORG $->$ LOC for the entity 'Victoria' in a sentence, and LOC $->$ ORG for the entity 'Indianapolis' in the other. In the last two cases, it can be argued that the original labels are ambiguous themselves. While that is, indeed, the case, the issue we particularly highlight is the way model predictions changed because of textual changes that should not really cause label changes. Similar trends can be observed in German and Hindi as well (Examples for German and Hindi are in the appendix in Table 12 and Figure 4). While it is definitely possible to do further qualitative analysis, we would speculate that combining this kind of analysis with explainable NLP approaches may give a more complete picture in

(a) `Orig`: Nicol$_{PER}$ was full of praise for his opponent who has battled testicular cancer to return to the circuit .
`RS`: Major$_{notidentified}$ was full of praise for his opponent who has battled testicular cancer to return to the circuit .
(b) `Orig`: [Nader Jokhadar]$_{PER}$ had given Syria$_{LOC}$ the lead with a well-struck header in the seventh minute . `Faker`: [Roger Turner]$_{PER}$ had given [Timor-Leste]$_{ORG}$ the lead with a well-struck header in the seventh minute .
(c) `Orig`: The richest parts of the property to the north and south of the central region have been estimated by **Bre-X**$_{ORG}$ to contain 57 million ounces of gold . `Para`: **Bre-X**$_{notidentified}$ estimates that the richest areas of the property to the north and south of the centre region contain 57 million ounces of gold .
(d) `Orig`: Tasmania$_{LOC}$ 352 by three ([David Boon]$_{PER}$ 106 not out , [Shaun Young]$_{PER}$ 86 not out , [Michael DiVenuto]$_{PER}$ 119) v **Victoria**$_{ORG}$. `Mask`: Tasmania$_{LOC}$ 352 by three ([David Boon]$_{PER}$ 106 not out , [Shaun Young]$_{PER}$ 86 not out , [Michael DiVenuto]$_{PER}$ 119) v **Victoria**$_{LOC}$:
(e) `Orig`:**Indianapolis**$_{LOC}$ closes with games at [Kansas City]$_{LOC}$ and Cincinnati$_{LOC}$. `Mask`: **Indianapolis**$_{ORG}$ begins with games at [Kansas City]$_{LOC}$ and Cincinnati$_{LOC}$.

Table 6: Examples of cases where a NER model fails on adversarial instances

the future on why NER model predictions fluctuate even for minimal perturbations in input text.

5.4 Adversarial Fine-Tuning

One approach to make models more robust to adversarial inputs is to include such data in building the model itself. We explored two methods in that direction: a) augmenting the training data with part of the adversarial dataset and re-training the NER model from scratch b) using adversarial data to fine-tune an existing NER model. The second method is especially useful in real-world scenarios where we have access to a trained model, but not to the original training data itself. We compared both the methods for all three languages, training one NER model per language (with conll-en, conll-de, and mconer-hi datasets respectively), and compared the performance with the `M+R` test set and the original test set in each case. As mentioned in Section 4, 60% of the adversarial test set was used for augmented re-training/fine-tuning and the remaining 40% was used as test data. Table 7 summarizes the results of this experiment.

While there are no significant differences between adversarial fine-tuning and re-training for English, and both give a 5% performance boost on adversarial test set without compromising on the original test set performance, for both German and Hindi, re-training was significantly better than fine-tuning with both test sets ($p<0.001$). A possible reason for this difference could lie in the relatively superior performance of the original model itself. Re-training could be more useful when the performance of the original model is poor, as was the case for German and Hindi. Interestingly, just fine-tuning still improved performance on adversarial test sets by over 5% for all languages.

	orig.	**adv. fine-tuning**	**aug. re-training**
conll-en			
`Original`	0.91	0.90	0.90
`Adv Test`	0.82	0.87	0.87
conll-de			
`Original`	0.83	0.84	0.89*
`Adv Test`	0.75	0.81	0.85*
mconer-hi			
`Original`	0.62	0.64	0.70*
`Adv Test`	0.48	0.55	0.58*

Table 7: Micro-F1 score for Adversarial fine-tuning versus re-training
(* indicates a statistically significant difference)

To connect these results back to our second research question (Section 1), considering that fine-tuning still resulted in better adversarial test set performance in all cases, and since that would not require access to the original training data itself, it could be a feasible, easily implementable approach

to improve the robustness of NER models without compromising on the original model performance. Re-training can be preferred when we have access to the original data and the model. Note that in both the cases, we are assuming no means to procure additional manually labeled training data, and the focus is on improving an NER model's robustness to adversarial input *without* compromising on its performance on normal test data.

6 Conclusions and Discussion

We explored simple, language agnostic approaches to generate adversarial test sets for NER and demonstrated their generalizability by testing on six datasets covering three languages - English, German and Hindi. While exact results differ depending on language/datasets, our key findings from these experiments can be summarized as follows:

1. NER models for all three languages are sensitive to adversarial input.

2. Adversarial fine-tuning and re-training could improve the performance of NER models both on original and adversarial test sets, without requiring additional manual labeled data.

The proposed approaches and tested languages/models are by no means comprehensive, and extending this work to include more NER models, adding new languages, and developing new adversarial data generation methods for NER is an obvious next step, as the current results provide enough evidence on the sensitivity of state of the art NER models to adversarial inputs. The methods we employed for adversarial fine-tuning/re-training too are just a starting point towards exploring the use of adversarial data in building more robust NER systems. We only explored one paraphraser for this task. The usefulness of the recent generative language models for creating such test data can be an interesting next step in this direction.

7 Limitations

The adversarial test sets based on masked language models can introduce new noise into the sentence context, as there is no way to automatically ensure grammatical correctness. However, there were many cases where such introduction of noise did not affect the predictions, in all three languages. Further, adversarial datasets are expected to introduce such noise, as is seen in other research on the topic for other tasks such as sentiment analysis, and the goal of such research is also to understand model robustness in the presence some noise. It is relevant to mention in this context that the NER datasets we considered already consist of other noise and ungrammatical examples such as score cards of sporting matches (conll03-en), social media content (wnut17) and fully lowercased sentences with weakly supervised annotations (mconer21). Further, masking does not alter the entities themselves, and only changes the non-entity tokens. So, the NER models still see the same entities. While there are no established means of quantifying the quality of adversarial datasets to our knowledge, exploring human-in-the-loop approaches to select appropriate examples to include in the final adversarial test set can be one way to address the issue.

8 Ethics and Impact Statement

The paper described the creation of several adversarial test sets for three languages. We used publicly available datasets for this purpose, and the research did not involve human participants. All the datasets we generated and the code to generate them are shared as supplementary material[11], for replication and to further this line of research. Our goal in this paper was to study the sensitivity of state of the art NER systems to adversarial data, and suggest ways to overcome it. As such, the generated datasets are expected to be used only for that purpose, and the limitations of current approaches are discussed in the previous section. Apart from this, since the paper focuses on more foundational question of evaluating NER systems in general, we do not foresee any other potential risks involved with this research.

Broader Impact Considering the number of practical usecases of NER across industries, and the growth of multilingual NLP, NER evaluation beyond English is more important than ever before. In this paper, we explored a previously unexplored space for Named Entity Recognition, i.e., evaluating NER systems beyond English for their sensitivity to adversarial input, which will hopefully lead into better evaluation strategies when developing NER systems across languages in future.

[11] https://dx.doi.org/10.6084/m9.figshare.22674079

Acknowledgements

We thank Justin Lee and Gabriel Bernier-Colborne for their feedback on an earlier draft, and Edwin Thomas for feedback on the final draft of the paper.

References

Oshin Agarwal, Yinfei Yang, Byron C Wallace, and Ani Nenkova. 2020. Entity-switched datasets: An approach to auditing the in-domain robustness of named entity recognition models. *arXiv preprint arXiv:2004.04123*.

Alan Akbik, Tanja Bergmann, Duncan Blythe, Kashif Rasul, Stefan Schweter, and Roland Vollgraf. 2019. FLAIR: An easy-to-use framework for state-of-the-art NLP. In *NAACL 2019, 2019 Annual Conference of the North American Chapter of the Association for Computational Linguistics (Demonstrations)*, pages 54–59.

Matthias Blohm, Glorianna Jagfeld, Ekta Sood, Xiang Yu, and Ngoc Thang Vu. 2018. Comparing attention-based convolutional and recurrent neural networks: Success and limitations in machine reading comprehension. In *Proceedings of the 22nd Conference on Computational Natural Language Learning*, pages 108–118, Brussels, Belgium. Association for Computational Linguistics.

Sudeshna Das and Jiaul Paik. 2022. Resilience of named entity recognition models under adversarial attack. In *Proceedings of the First Workshop on Dynamic Adversarial Data Collection*, pages 1–6, Seattle, WA. Association for Computational Linguistics.

Leon Derczynski, Eric Nichols, Marieke van Erp, and Nut Limsopatham. 2017. Results of the WNUT2017 shared task on novel and emerging entity recognition. In *Proceedings of the 3rd Workshop on Noisy User-generated Text*, pages 140–147, Copenhagen, Denmark. Association for Computational Linguistics.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. Bert: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 4171–4186.

Bosheng Ding, Linlin Liu, Lidong Bing, Canasai Kruengkrai, Thien Hai Nguyen, Shafiq Joty, Luo Si, and Chunyan Miao. 2020. DAGA: Data augmentation with a generation approach for low-resource tagging tasks. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 6045–6057, Online. Association for Computational Linguistics.

Sang Erik F. Tjong Kim and De Meulder Fien. 2003. Introduction to the conll-2003 shared task: Language independent named entity recognition. In *InProceedings of the seventh conference on Natural language learning at HLT-NAACL*, volume 4, pages 142–147.

Allyson Ettinger, Sudha Rao, Hal Daumé III, and Emily M. Bender. 2017. Towards linguistically generalizable NLP systems: A workshop and shared task. In *Proceedings of the First Workshop on Building Linguistically Generalizable NLP Systems*, pages 1–10, Copenhagen, Denmark. Association for Computational Linguistics.

Ji Gao, Jack Lanchantin, Mary Lou Soffa, and Yanjun Qi. 2018. Black-box generation of adversarial text sequences to evade deep learning classifiers. In *2018 IEEE Security and Privacy Workshops (SPW)*, pages 50–56. IEEE.

Matt Gardner, Yoav Artzi, Victoria Basmov, Jonathan Berant, Ben Bogin, Sihao Chen, Pradeep Dasigi, Dheeru Dua, Yanai Elazar, Ananth Gottumukkala, Nitish Gupta, Hannaneh Hajishirzi, Gabriel Ilharco, Daniel Khashabi, Kevin Lin, Jiangming Liu, Nelson F. Liu, Phoebe Mulcaire, Qiang Ning, Sameer Singh, Noah A. Smith, Sanjay Subramanian, Reut Tsarfaty, Eric Wallace, Ally Zhang, and Ben Zhou. 2020. Evaluating models' local decision boundaries via contrast sets. In *Findings of the Association for Computational Linguistics: EMNLP 2020*, pages 1307–1323, Online. Association for Computational Linguistics.

Max Glockner, Vered Shwartz, and Yoav Goldberg. 2018. Breaking NLI systems with sentences that require simple lexical inferences. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 650–655, Melbourne, Australia. Association for Computational Linguistics.

Shreya Goyal, Sumanth Doddapaneni, Mitesh M Khapra, and Balaraman Ravindran. 2022. A survey in adversarial defences and robustness in nlp. *arXiv preprint arXiv:2203.06414*.

Pierre Isabelle, Colin Cherry, and George Foster. 2017. A challenge set approach to evaluating machine translation. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, pages 2486–2496, Copenhagen, Denmark. Association for Computational Linguistics.

Mohit Iyyer, John Wieting, Kevin Gimpel, and Luke Zettlemoyer. 2018. Adversarial example generation with syntactically controlled paraphrase networks. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*, pages 1875–1885, New Orleans, Louisiana. Association for Computational Linguistics.

Robin Jia and Percy Liang. 2017. Adversarial examples for evaluating reading comprehension systems.

In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, pages 2021–2031, Copenhagen, Denmark. Association for Computational Linguistics.

Viet Dac Lai, Nghia Trung Ngo, Amir Pouran Ben Veyseh, Hieu Man, Franck Dernoncourt, Trung Bui, and Thien Huu Nguyen. 2023. Chatgpt beyond english: Towards a comprehensive evaluation of large language models in multilingual learning. *arXiv preprint arXiv:2304.05613*.

J Li, S Ji, T Du, B Li, and T Wang. 2019. Textbugger: Generating adversarial text against real-world applications. In *26th Annual Network and Distributed System Security Symposium*.

Bin Liang, Hongcheng Li, Miaoqiang Su, Pan Bian, Xirong Li, and Wenchang Shi. 2018. Deep text classification can be fooled. In *IJCAI*.

Bill Yuchen Lin, Wenyang Gao, Jun Yan, Ryan Moreno, and Xiang Ren. 2021. RockNER: A simple method to create adversarial examples for evaluating the robustness of named entity recognition models. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 3728–3737, Online and Punta Cana, Dominican Republic. Association for Computational Linguistics.

Ben Lorica and Paco Nathan. 2021. 2021 nlp survey report. Technical report, Gradient Flow.

Shervin Malmasi, Anjie Fang, Besnik Fetahu, Sudipta Kar, and Oleg Rokhlenko. 2022. SemEval-2022 task 11: Multilingual complex named entity recognition (MultiCoNER). In *Proceedings of the 16th International Workshop on Semantic Evaluation (SemEval-2022)*, pages 1412–1437, Seattle, United States. Association for Computational Linguistics.

Joel Mathew, Shobeir Fakhraei, and José Luis Ambite. 2019. Biomedical named entity recognition via reference-set augmented bootstrapping. *arXiv preprint arXiv:1906.00282*.

Paul Michel, Xian Li, Graham Neubig, and Juan Pino. 2019. On evaluation of adversarial perturbations for sequence-to-sequence models. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 3103–3114, Minneapolis, Minnesota. Association for Computational Linguistics.

Hiroki Nakayama. 2018. seqeval: A python framework for sequence labeling evaluation. Software available from https://github.com/chakki-works/seqeval.

Peng Qi, Yuhao Zhang, Yuhui Zhang, Jason Bolton, and Christopher D. Manning. 2020. Stanza: A Python natural language processing toolkit for many human languages. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics: System Demonstrations*.

Chengwei Qin, Aston Zhang, Zhuosheng Zhang, Jiaao Chen, Michihiro Yasunaga, and Diyi Yang. 2023. Is chatgpt a general-purpose natural language processing task solver? *arXiv preprint arXiv:2302.06476*.

Marco Tulio Ribeiro, Sameer Singh, and Carlos Guestrin. 2018. Semantically equivalent adversarial rules for debugging NLP models. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 856–865, Melbourne, Australia. Association for Computational Linguistics.

Isabel Segura-Bedmar, Paloma Martínez, and María Herrero-Zazo. 2013. SemEval-2013 task 9 : Extraction of drug-drug interactions from biomedical texts (DDIExtraction 2013). In *Second Joint Conference on Lexical and Computational Semantics (*SEM), Volume 2: Proceedings of the Seventh International Workshop on Semantic Evaluation (SemEval 2013)*, pages 341–350, Atlanta, Georgia, USA. Association for Computational Linguistics.

Fatemeh Shiri, Terry Yue Zhuo, Zhuang Li, Shirui Pan, Weiqing Wang, Reza Haffari, Yuan-Fang Li, and Van Nguyen. 2022. Paraphrasing techniques for maritime qa system. In *2022 25th International Conference on Information Fusion (FUSION)*, pages 1–8.

Avi Shmidman, Joshua Guedalia, Shaltiel Shmidman, Moshe Koppel, and Reut Tsarfaty. 2020. A novel challenge set for Hebrew morphological disambiguation and diacritics restoration. In *Findings of the Association for Computational Linguistics: EMNLP 2020*, pages 3316–3326, Online. Association for Computational Linguistics.

Walter Simoncini and Gerasimos Spanakis. 2021. SeqAttack: On adversarial attacks for named entity recognition. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, pages 308–318, Online and Punta Cana, Dominican Republic. Association for Computational Linguistics.

Erik F. Tjong Kim Sang and Fien De Meulder. 2003. Introduction to the CoNLL-2003 shared task: Language-independent named entity recognition. In *Proceedings of the Seventh Conference on Natural Language Learning at HLT-NAACL 2003*, pages 142–147.

Asahi Ushio and Jose Camacho-Collados. 2021. T-ner: An all-round python library for transformer-based named entity recognition. In *Proceedings of the 16th Conference of the European Chapter of the Association for Computational Linguistics: System Demonstrations*, pages 53–62.

Sowmya Vajjala and Ramya Balasubramaniam. 2022. What do we really know about state of the art ner? In *Proceedings of the Language Resources and Evaluation Conference*, pages 5983–5993, Marseille, France. European Language Resources Association.

Eric Wallace, Shi Feng, Nikhil Kandpal, Matt Gardner, and Sameer Singh. 2019a. Universal adversarial triggers for attacking and analyzing NLP. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 2153–2162, Hong Kong, China. Association for Computational Linguistics.

Eric Wallace, Pedro Rodriguez, Shi Feng, Ikuya Yamada, and Jordan Boyd-Graber. 2019b. Trick me if you can: Human-in-the-loop generation of adversarial examples for question answering. *Transactions of the Association for Computational Linguistics*, 7:387–401.

Shijie Wu, Ozan Irsoy, Steven Lu, Vadim Dabravolski, Mark Dredze, Sebastian Gehrmann, Prabhanjan Kambadur, David Rosenberg, and Gideon Mann. 2023. Bloomberggpt: A large language model for finance. *arXiv preprint arXiv:2303.17564*.

Wei Emma Zhang, Quan Z Sheng, Ahoud Alhazmi, and Chenliang Li. 2020. Adversarial attacks on deep-learning models in natural language processing: A survey. *ACM Transactions on Intelligent Systems and Technology (TIST)*, 11(3):1–41.

Zhengli Zhao, Dheeru Dua, and Sameer Singh. 2018. Generating natural adversarial examples. In *International Conference on Learning Representations*.

Wenjing Zhu, Jian Liu, Jinan Xu, Yufeng Chen, and Yujie Zhang. 2021. Improving low-resource named entity recognition via label-aware data augmentation and curriculum denoising. In *Chinese Computational Linguistics: 20th China National Conference, CCL 2021, Hohhot, China, August 13–15, 2021, Proceedings*, pages 355–370. Springer.

A Appendix

A.1 Dataset Statistics:

Dataset	# train	# dev	# test
(Erik F. Tjong Kim and Fien, 2003)			
conll03-en	14,987	3,466	3,684
conll03-de	12,705	3,068	3,160
wnut17[12]	3394	1009	1287
(Malmasi et al., 2022)			
mconer21-en	15,300	800	217,818
mconer21-de	15,300	800	217,824
mconer21-hi	15,300	800	141,565

Table 8: Dataset statistics in terms of number of sentences per split

We used the dev set from mconer21 to create adversarial test sets for all the three languages, considering the large size of its test set.

A.2 Detailed Evaluation

	orig. test set		adv. test set	
	Strict	Exact	Strict	Exact
Overall	0.91	0.95	0.82	0.92
PER	0.96	0.98	0.87	0.92
LOC	0.92	0.96	0.85	0.95
ORG	0.89	0.94	0.75	0.91
MISC	0.83	0.89	0.79	0.88

Table 9: Strict versus Exact evaluation for English (conll-en)

	orig. test set		adv. test set	
	Strict	Exact	Strict	Exact
Overall	0.73	0.81	0.61	0.73
PER	0.88	0.92	0.8	0.86
LOC	0.78	0.83	0.67	0.76
PROD	0.7	0.79	0.58	0.73
GRP	0.68	0.81	0.55	0.71
CORP	0.66	0.8	0.53	0.75
CW	0.59	0.67	0.44	0.53

Table 10: Strict versus Exact evaluation for German (multiconer21-de)

	orig. test set		adv. test set	
	Strict	Exact	Strict	Exact
Overall	0.62	0.73	0.48	0.61
PER	0.71	0.82	0.56	0.62
LOC	0.77	0.82	0.57	0.72
PROD	0.54	0.63	0.48	0.61
GRP	0.67	0.81	0.52	0.67
CORP	0.56	0.76	0.42	0.65
CW	0.47	0.57	0.35	0.46

Table 11: Strict versus Exact evaluation for Hindi (mconer21-hi)

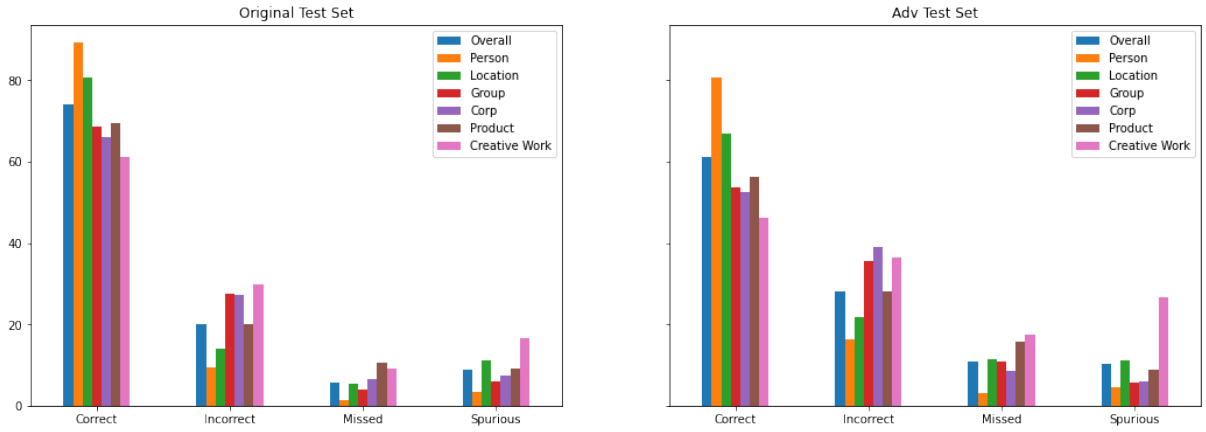

Figure 2: A plot visualising the performance of a German NER model (trained with mconer21 data) on original and M+R adversarial Test sets

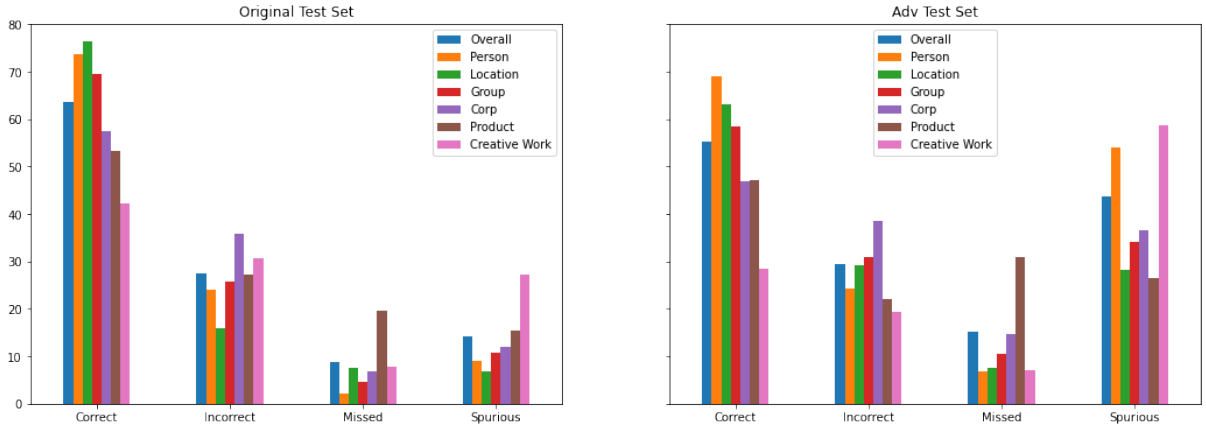

Figure 3: A plot visualising the performance of a Hindi NER model (trained with mconer21 data) on original and M+R adversarial Test sets

`Orig`: Der kleine **Elmir**$_{PER}$ verläßt den Raum .
`RS`: Der kleine **Treutel**$_{notidentified}$ verläßt den Raum .
`Faker`: Der kleine Melek$_{notidentified}$ verläßt den Raum .
`Orig`: Die Verwertungsgesellschaft Gebrauchte Kunststoffverpackungen in [**Bad Homburg**]$_{LOC}$ sei " offenbar mit ihrer Aufgabe überfordert " .
`RS`: Die Verwertungsgesellschaft Gebrauchte Kunststoffverpackungen in [**Stadt**]$_{LOC}$ [**Hanau**]$_{notidentified}$ sei " offenbar mit ihrer Aufgabe überfordert " .
`Orig`: Im [Korea-Krieg]$_{MISC}$ **hatte** China$_{LOC}$ das kommunistische Nordkorea$_{LOC}$ **unterstützt .**
`Mask`: Im Korea-Krieg$_{LOC}$ **hatten** China$_{LOC}$ das kommunistische Nordkorea$_{LOC}$ **gewonnen** |
`Orig`: Ihm stehen 20 000 Mark zur **Verfügung** , um **die Ausstellung** im November 1993 im Stadtmuseum$_{LOC}$ zu realisieren .
`Mask`: Ihm stehen 20 000 Mark zur **wahl** , um **diese skulptur** im November 1993 im Stadtmuseum$_{notidentified}$ zu realisieren .
`Orig`: **Oder die** [Gauck-Behörde]$_{MISC}$?
`Mask`: **in der** Gauck-Behörde]$_{LOC}$?

Table 12: Examples of cases where a German NER model fails on adversarial input, but makes correct predictions on the original text

Orig: १४९२ में एक चार्टर के आधार पर, उसके पिता ने उसे **वाडोविस_LOC** के उत्तराधिकारी के रूप में छोड़ दिया
RS: १४९२ में एक चार्टर के आधार पर, उसके पिता ने उसे **पेरू_CORP** के उत्तराधिकारी के रूप में छोड़ दिया
Masked: पिता का एक चार्टर की आधार पर, उसके पिता ने उसे **वाडोविस_CORP** की उत्तराधिकारी की रूप की छोड़ दिया

Orig: इस तरह का पहला "बेड़ा, **आर12_PROD** " १९४८ में सेवा में लगाया गया था।
M+R: इस तरह का पहला "बेड़ा, **कास्केट_NONE** बड १९४८ में चीन में लगाया गया था।

Orig: रुसब्रिजर २०१६ की फिल्म **स्नोडेन_CW** में एक मीटिंग मॉडरेटर के रूप में एक कैमियो भूमिका के साथ दिखाई देते हैं
Masked: रुसब्रिजर **2007_CW** फिलम फिल्म **स्नोडेन_CW** में एक मीटिंग मॉडरेटर के साथ में एक कैमियो भूमिका के साथ दिखाई देते हैं

Orig: १९३७ में उनका अगला क्लब **[सेंट बर्नार्ड एफ.सी.]_GRP** था।
Masked: 1937 में उनका परथम पातर **[सेंट बर्नार्ड एफ.सी.]_PER** !

Orig: उन्होंने **[फ्रेंकस्टीन अनबाउंड]_CW** के साथ एक बार फिर निर्देशन में वापसी की।
Masked: उन्होंने **[फ्रेंकस्टीन]_CORP [अनबाउंड]_GRP** के साथ एक बार अपन गाव में वापसी की।

Figure 4: Examples comparing how predictions change for a Hindi NER model on original versus adversarial test sets