# Analysing the effects of transfer learning on low resourced named entity recognition performance

**Michael Beukman**
University of the Witwatersrand
School of Computer Science and Applied Mathematics
1825748@students.wits.ac.za

## Abstract

Transfer learning has led to large gains in performance for nearly all NLP tasks, while making downstream models easier and faster to train. This has also been extended to low resourced languages, with some success. We investigate the properties of transfer learning between 9 low resourced languages, from the perspective of a named entity recognition task, specifically how much pre-training helps, the efficacy of zero-shot transfer as well as the effect of learning on the contextual embeddings computed from the model. Our results give some insight into zero shot performance as well as the impact of different pre-trained models and data overlap. We publicly release our code[1] and models[2].

## 1 Introduction

The technique of using a pre-trained Natural Language Processing (NLP) model and fine-tuning it on task specific data has recently taken the NLP world by storm, achieving state of the art scores in many different tasks (Jiang et al., 2020; Hendrycks et al., 2021; Raffel et al., 2020). Although much of the focus of pre-trained models is on english (Devlin et al., 2019; Radford et al., 2018, 2019), there are also monolingual models for different languages (de Vries et al., 2019; Masala et al., 2020; Canete et al., 2020) and multilingual models that were trained on a massive corpus of data from different languages (Conneau et al., 2020; Xue et al., 2021). This idea of pre-training on a large, unlabelled corpus and then fine-tuning on task specific data can significantly reduce labelled data requirements (Howard and Ruder, 2018).

In an effort to explore NLP beyond English (Ruder, 2020), we specifically investigate cross lingual transfer between 9 low resourced African languages in a named entity recognition (NER) task (Adelani et al., 2021) by determining which features transfer, as well as how different training schemes affect performance.

## 2 Background

### 2.1 Named Entity Recognition (NER)

Named Entity Recognition is a token classification task in which the objective is to classify each token as one of a few classes, e.g. person, location, date, organisation, or no entity. NER is an impactful field (Sang and Meulder, 2003; Lample et al., 2016) with many applications (Marrero et al., 2013).

### 2.2 Transfer Learning

Transfer learning is a technique that is often used in NLP to improve performance while requiring less task specific data (Ruder et al., 2019). This usually functions by training a large language model on a massive corpus of unlabelled data, and using these learned weights as the starting point for a specific problem, and fine-tuning further on task specific labelled data (Ruder, 2021). The idea is that the pre-training process instilled knowledge into the model about how language functions on a general level, which then does not need to be learned from scratch using the smaller amount of task specific data. The entire network is usually fine-tuned in an end-to-end fashion, although there are techniques that only learns a small fraction of the weights by using adapters, leading to more efficient transfer (Houlsby et al., 2019).

## 3 Methodology

This report builds upon the work of Adelani et al. (2021) and investigates the following questions:

1. How much does the pre-trained model affect

[1] https://github.com/Michael-Beukman/NERTransfer
[2] https://huggingface.co/mbeukman

downstream performance after fine-tuning on task specific data?

2. Which languages are the best for doing zero shot transfer from?

3. What features or aspects get transferred between the languages we examine?

To answer the above questions, we fundamentally fine-tune different pre-trained models on the MasakhaNER dataset (Adelani et al., 2021) and compare their performance. We consider all languages (Hausa, Igbo Kinyarwanda, Luganda, Luo, Nigerian Pidgin - pcm, Swahili, Wolof, Yorùbá) except Amharic due to the different script and computational restrictions. In this report, for consistency, we refer to pre-training as any approach that trains a language model on a large, unlabelled corpus, whereas fine-tuning means taking a pre-trained model, and training that end-to-end on a smaller, labelled dataset. In particular, we also refer to domain adaptive fine tuning as pre-training, where (in our case) an existing pre-trained model is further fine-tuned on monolingual, unlabelled data using a language modelling objective.

For point (1) we largely follow what Adelani et al. (2021) did, and use this as a basis for the subsequent parts. For point (2) above, we consider how good zero shot transfer is when fine-tuning on NER data from other languages. To answer (3) we examine the statistical properties of the datasets as well as the contextual word embeddings obtained after various pre-training and fine-tuning steps.

## 4 Experiments & Results

### 4.1 Experimental Setup

We fine-tune each model 5 times with 5 different seeds (to account for variability), and report the mean and standard deviation here. We use the MasakhaNER implementation[3] and use the same language codes as Adelani et al. (2021). All metrics reported are overall F1 scores, using the 'begin' repair strategy as specified by Palen-Michel et al. (2021).

### 4.2 Pre-trained Models

In this section we determine the effect of using different pre-trained models. Each of the models we consider are based on xlm-roberta (Conneau et al., 2020). The first model we consider is called 'base', and it is simply xlm-roberta-base, downloaded from Huggingface[4]. The other models[5] we consider used xlm-roberta-base as their starting point, but additionally performed domain adaptive fine-tuning on a monolingual corpus and were shown to perform better on NER tasks (Adelani et al., 2021).

For each language X, we use 3 different models, base, base-Swahili and base-X, where the latter 2 were further fine-tuned with domain adaptive fine-tuning.

The results are shown in Table 1, and the language adaptive pre-trained models usually perform much better than the base model, with the Swahili model being in between. The standard deviations between the different seeds are quite large however, so not all results are statistically significant (using a Mann-Whitney U test). In most cases we replicate Adelani et al. (2021) and Palen-Michel et al. (2021), with the single exception of Nigerian Pidgin that was fine-tuned from a language-adaptive model, possibly because of different model versions.

### 4.3 Cross Lingual Transfer

This experiment investigates fine-tuning one of the above models on one specific language (e.g. Yorùbá) and evaluating on another (e.g. Hausa). These results are shown in Figure 1. Specifically, in Figure 1a, as expected, the diagonal is brighter than the off-diagonal elements, as fine-tuning on the same language one evaluates on improves scores significantly. Figure 1b shows a mixed result, as for some language pairs, using a pre-trained model that has been pre-trained on the same language as one fine-tunes on helps, but for others this effect is minor. We also notice that for other languages, notably Swahili and Hausa, using those pre-trained models (and fine-tuning on that NER data) diminishes the transfer capabilities from these languages (see Figure 1g), possibly indicating overfitting, similar to what Pfeiffer et al. (2020) found.

In Figure 1c we use different pre-trained models, but fine-tune on the same Swahili NER data, and again evaluate on each language. On first glance, horisontal lines can be seen, indicating that the pre-trained model does not affect the final score that much in this case, although the diagonal is usually slightly brighter. Again, we see a similar overfitting

[3] https://github.com/masakhane-io/masakhane-ner/

[4] https://huggingface.co/xlm-roberta-base

[5] https://huggingface.co/Davlan

| | wol | pcm | yor | hau | ibo | luo | lug | kin | swa |
|---|---|---|---|---|---|---|---|---|---|
| Pre-trained (same-language) | 66.9 (1.7) | 87.1 (0.8) | **83.3 (0.3)**$^*$ | **91.6 (0.4)**$^*$ | **87.9 (0.5)**$^*$ | 76.2 (1.2) | **84.5 (0.5)**$^*$ | **78.3 (1.0)**$^*$ | **89.6 (0.6)**$^*$ |
| Pre-trained (swa) | **67.3 (1.3)**$^*$ | 88.0 (0.8) | 78.3 (1.0) | 88.8 (0.2)$^*$ | 84.3 (0.8) | 77.2 (1.4) | 82.0 (0.5)$^*$ | 75.2 (1.0) | **89.6 (0.6)**$^*$ |
| Pre-trained (base) | 64.2 (1.3) | 87.3 (0.9) | 77.9 (0.3) | 89.5 (0.4) | 84.9 (0.7) | 74.5 (1.3) | 80.2 (0.7) | 73.7 (0.7) | 87.8 (0.5) |

Table 1: Comparing the performance of different pre-trained models after fine-tuning and evaluating on NER data. We use a Mann-Whitney U test (Mann and Whitney, 1947) for consistency as some data failed a Shapiro Wilks normality test (Shapiro and Wilk, 1965). $^*$ indicates a statistical significant difference ($p < 0.05$) between the base model and the one under consideration, **bold** indicates a statistical significant difference, as well as being the maximum for this language.

problem to the above when using a Swahili pre-trained model and fine-tuning on Swahili NER.

In general, the diagonal elements perform worse than using the base model and training on the same language we evaluate on (naturally), but the off-diagonal elements (i.e. transfer) do increase dramatically, particularly when starting from Luo, Nigerian Pidgin and Wolof. While the increase may be large, the initial F1 scores were quite small, so the final transfer performance is still not very high. We also consider the above in slightly more detail by looking at each NER category individually, to see if any perform much better or worse than the others. Figure 2 indicates that dates transfer quite poorly, particularly for Luo.

#### 4.3.1 Data overlap

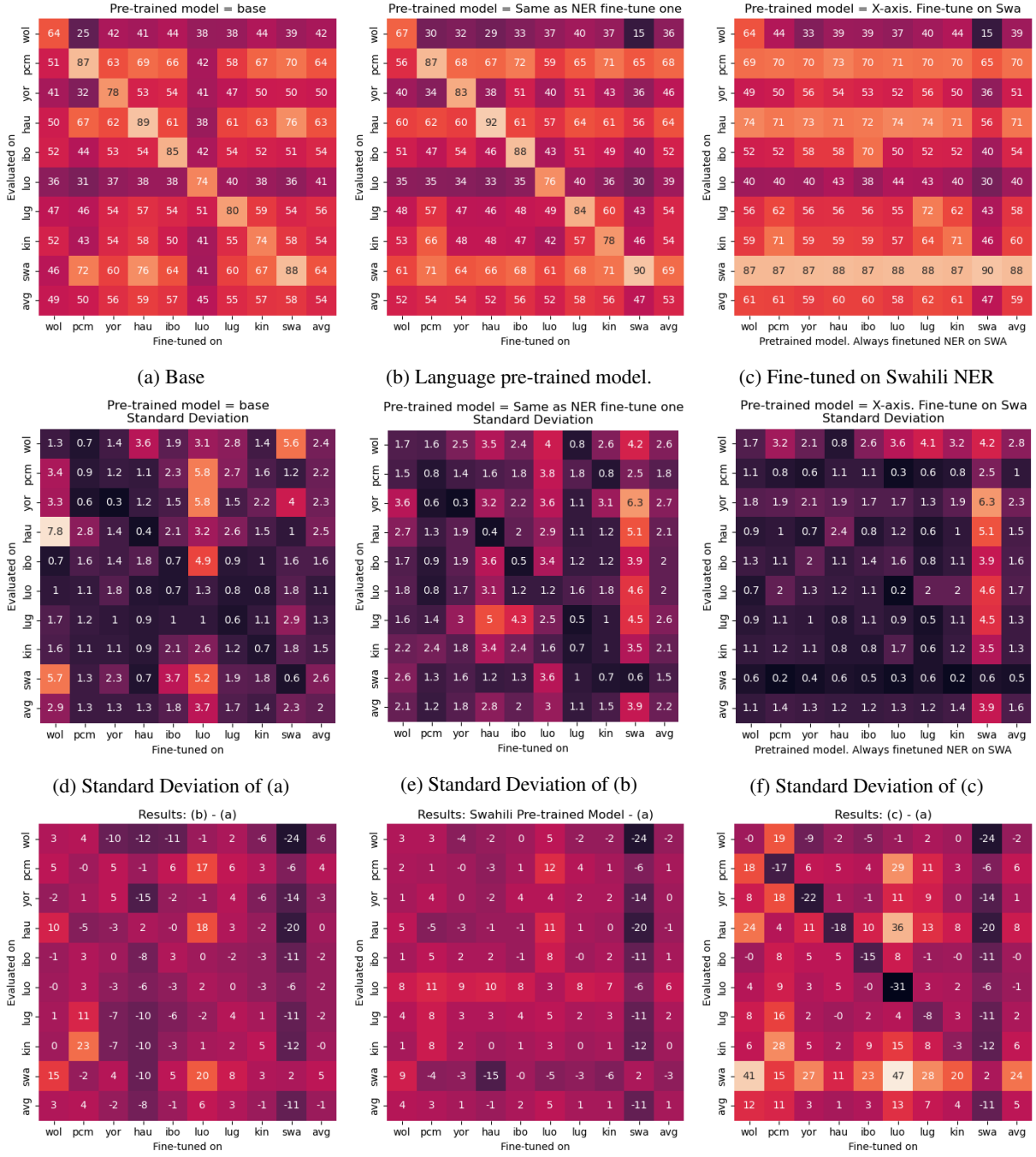
To try and explain some of the results shown in the previous section, here we look at the datasets a bit more carefully, specifically analysing the data overlap between different languages, and whether this has any correlation with the performance when doing transfer. To do so, we investigate the overlap of each entity in the respective datasets. We call a token overlapping when the same token is labelled as the same entity type in two different datasets, and we count the number of overlaps from source language A (x-axis) to target language B (y-axis) as the number of occurrences of this token in A's dataset, as this could measure how much data the model can use from language A that might transfer to language B. We do not distinguish between tokens that are at the beginning of an entity or in the middle thereof (i.e. we consider B-PER and I-PER to be the same for this experiment). We consider the entire dataset, i.e. train + dev + test, to obtain a more representative sample, although this does not calculate overlap between e.g. the train set of A and the test set of B. Other ways of calculating the overlap exist, like only considering unique entities (which we avoid as one entity overlapping multiple times is relevant), or considering the minimum number of tokens that overlap from language A and B. These methods were roughly correlated with and similar to our approach however, so it does not make a large difference. Figure 3a shows the results, and a few things are immediately clear. Firstly, Wolof and Luo have much less data than the other languages, and thus much less overlap, potentially explaining why these two performed badly in previous experiments. Secondly, there seems to be quite a lot of overlap in general. Swahili and Hausa also show many tokens in common, possibly due to Arabic influences on both of these (Versteegh, 2001), but it could also simply be e.g. international names and dates. We clearly see a strong correlation (Pearson's coefficient = 0.72) between how many tokens overlap and the performance in Figure 3b. The procedure here was simply to compute the correlation between the data overlap (as in Figure 3a) and the performance when fine-tuning on one language and evaluating on another, starting from the pre-trained base model (as in Figure 1a). We did not take the diagonal elements into consideration, as that does not count as transfer learning.

### 4.4 Representations

Our final experiment investigates the contextual word embeddings from the different models, specifically looking into how these embeddings change as we perform different fine-tuning operations. The way we approach this is to take the last 4 layers from the language model (i.e. not the dense final layer), and use the sum of these hidden states to obtain a word vector (of size 768). We use the sentences from the dataset, and only extract the 4 different NER categories for computational reasons. We compute the mean vector per category, which we use in the following. To visualise the data, we show the results after performing PCA.

#### 4.4.1 Variability

We found a large amount of variability when fine-tuning the models on different random seeds (see




(a) Base

(b) Language pre-trained model.

(c) Fine-tuned on Swahili NER

(d) Standard Deviation of (a)

(e) Standard Deviation of (b)

(f) Standard Deviation of (c)

(g) Language pre-trained model. Swahili and Hausa transfer worse on average, while Luo improves.

(h) Swahili pre-trained model. Hausa fine-tuning performs much worse when evaluated on Swahili.

(i) Fine-tuned on Swahili NER. The Luo pre-trained model in particular has significant increase.

Figure 1: Heatmaps. In general we notice a large standard deviation, indicating that this is unreliable. The bottom row shows the difference between one technique, and base, i.e. how much improvement does this new model give over using the base model. *avg* indicates the average per row or column respectively.

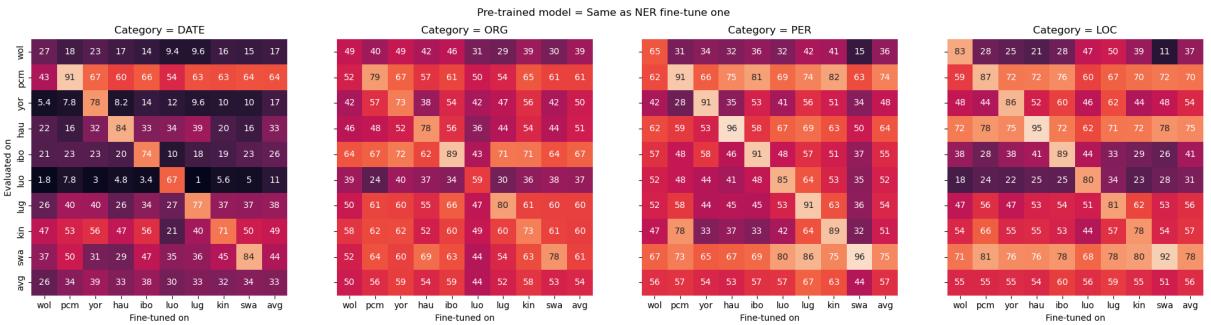



Figure 2: Heatmaps for the language pre-trained model (Figure 1b), broken down by category.

Figures 1d, 1e and 1f) and in this experiment we investigate the effect of using different initialisations on these embeddings. Note, it is not completely rigorous to directly compare word embeddings from different models, as even different embeddings can have the exact same effect if the final linear layer is different. We do still show the embeddings to qualitatively investigate how they change though.

Figure 4 shows the results for a few languages pairs, and immediately we can see that Figure 4a has clusters corresponding to the different categories, even when using different seeds. Figures 4b, 4c and 4d on the other hand cluster more toward seeds, so the categories differ when using different seeds. This could indicate that the Swahili model is more consistent and robust to random initialisations, and learns roughly the same embeddings for each seed. On the other hand, when fine-tuning from Kinyarwanda, Luo or Wolof, there is no clear clustering of categories (despite a relatively large amount data overlap between Kinyarwanda and Hausa), maybe suggesting that these models cannot distinguish Hausa categories very well (possibly substantiated by the poorer results in Figure 1).

#### 4.4.2 Different Languages

Here we consider the same model, and analyse the differences in embeddings from different languages, and how this evolves. For example, in Figure 5a we see that for Nigerian Pidgin (which transferred quite well previously), the predominant clusters are again categories, and the same categories from different languages are grouped together.

#### 4.4.3 Different Models

This experiment examines different models on the same language, specifically looking at what happens to these embeddings when a model is further fine-tuned. Figure 5b shows that performing fine-tuning on models does affect the embeddings quite significantly, although there does still seem to be a similar relative positioning between the categories - almost as if in PCA, one principal component was the model used, and another was the category.

## 5 Analysis, Discussion & Future Work

We touched on a few different topics in this report, all related to transfer learning and how this affects F1 score. We found that the pre-trained model has an effect on transfer performance, and that overfitting is a real risk, so the model that does best on one language often suffers in transfer to other languages, potentially motivating less overspecialised models in favour of more general models which would also hopefully be more robust. As usual, data is king, and having more data overlap is highly correlated with transfer performance, although we tested relatively high quality datasets here, so this might not transfer well to low quality, noisy datasets (Alabi et al., 2019). Future work could include looking at different languages, investigating the geographical (Adelani et al., 2021) or language-family angle, or how combining two (or more) datasets might strike a balance between high single task performance and generalisability.

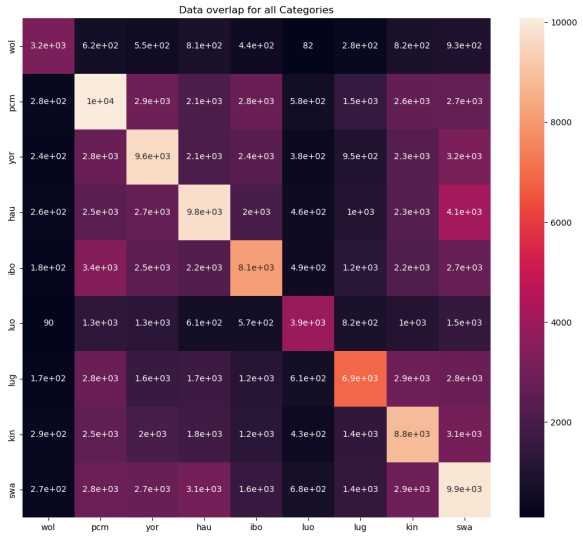
## 6 Conclusion

In summary, we considered many facets of transfer learning, the pre-trained model, the fine-tuning language before doing zero shot transfer, and we investigated some reasons for our findings, notably data overlap and embedding changes. In some cases we found very large variances, making reliably performing transfer learning difficult. We answered our original 3 questions, specifically that (1) using a language-adaptive pre-trained model improves performance on the corresponding language, possibly while reducing transfer performance. Regarding point (2), we found the best language to perform zero shot transfer from does depend on various factors, although the amount of data overlap could help inform this choice, by choosing languages that have large overlap. Finally, for (3) we found that overlapping tokens might be a large part of what is transferred, but some linguistic knowledge could also be transferred, between e.g. Swahili and Hausa, as clear separation of categories was apparent. In other cases (e.g. Luo and Hausa), a less clear clustering was observed.

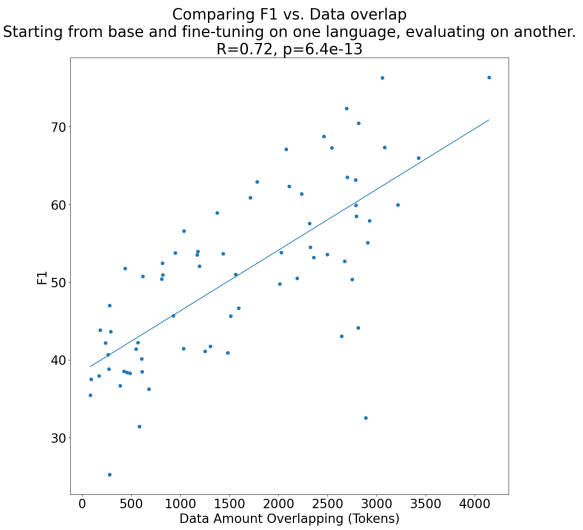
Our main conclusions are that (in our case, with our dataset) using language specific models usually perform better than not after subsequent fine-tuning. We found high levels of data overlap, and found a strong correlation between this and the F1 performance, although this does not imply causality.

## 7 Acknowledgements

Computations were performed using High Performance Computing infrastructure provided by the Mathematical Sciences Support unit at the University of the Witwatersrand.

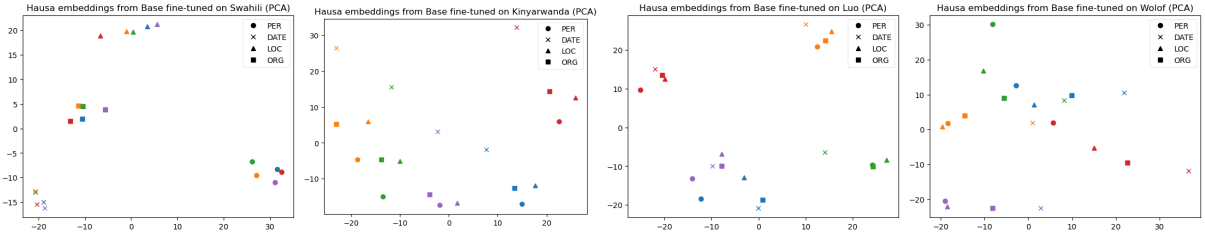



(a) Data Overlap. Row $i$, column $j$ indicates the overlap if language $j$ was the source (i.e. fine-tune language) and language $i$ was the target (i.e. evaluation language).



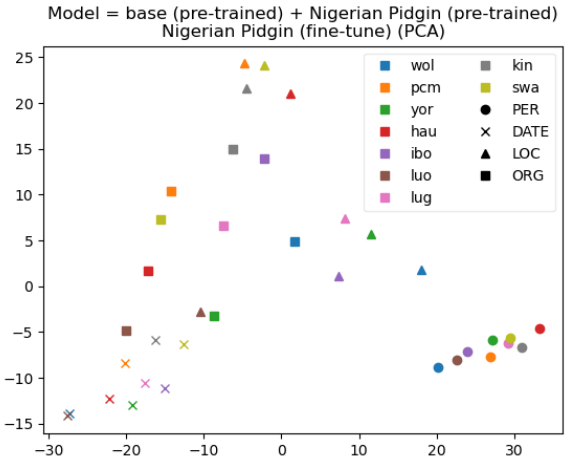

(b) Correlation. 0.72 Pearson's correlation coefficient with $p < 0.05$.

Figure 3: Data Overlap & effect on performance.

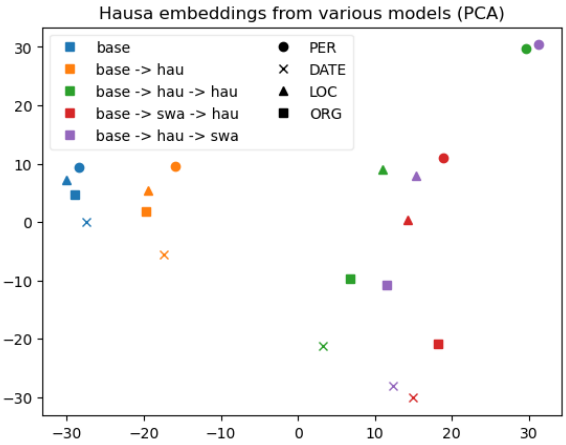



(a) Clear category clusters. (b) Clear colour clusters. (c) Clear colour clusters. (d) Clear colour clusters.

Figure 4: Embeddings. The shapes indicate different categories, whereas the colours indicate different starting points, i.e. seeds.




(a) Embeddings for different languages from a Nigerian Pidgin model, fine-tuned on Nigerian Pidgin.




(b) Hausa embeddings from different models. For example, base -> hau ->swa used a pre-trained Hausa model, fine-tuned on Swahili NER.

Figure 5: Some Embeddings

## 8 Broader Impact

This section discusses some potential broader impact of this work.

### 8.1 Use Cases

The techniques deployed in this report can be used to investigate variability and robustness in other models, as well as to determine whether transfer learning takes advantage of linguistic commonalities, or simply data overlap. The transfer capabilities showcased here, alongside those demonstrated by Adelani et al. (2021) could be used to develop better downstream NER models by leveraging more data, even if this data is not necessarily in the target language. Direct application of these models trained on this dataset is not recommended however, because of limited training data and no real-world verification. To actually use this, more care needs to be taken by collecting a larger variety of data, as well as using a test set that is representative of what is expected to be encountered in production. More analysis also needs to be done on potential biases, concrete failure cases and how to address them, ideally by collaborating with speakers of the specific languages in a participatory fashion (∀ et al., 2020). Furthermore, this dataset only contained 4 entity categories (date, person, location and organisation), so when other entity types are desired, more data does need to be collected and annotated.

### 8.2 Benefits

The benefits of this work could include understanding what multilingual transfer learning models actually learn, and whether / how we could use alternative data sources in different languages to supplement low resourced datasets. The variability demonstrated when using different seeds could also incentivise other researchers to report standard deviation alongside mean, although the computational cost for this is large. Even with the introduction of the MasakhaNER dataset for 10 low resourced languages, we still see a data gap (with Wolof and Luo having much less data than the other languages), which could significantly affect (or at least show a correlation with) performance.

### 8.3 Risks

There are still risks however, namely overgeneralisation - these experiments were only performed on a few languages on one task (NER), with quite specific data (news based), and the results might thus not be applicable to other tasks, languages or even types of data, so care is recommended. We also had a small amount of very high quality data at our disposal, so using much more data or data of lower quality might also have a large effect.

Because all of the models covered here used xlm-roberta-base (Conneau et al., 2020) as their starting point (potentially with domain adaptive fine-tuning on specific languages), this model's limitations can also apply here. These can include being biased towards the hegemonic viewpoint of most of its training data (Bender et al., 2021), being ungrounded (Bender and Koller, 2020) and having subpar results on other languages (possibly due to unbalanced training data).

As Adelani et al. (2021) showed, the models in general struggled with entities that were either longer than 3 words and entities that were not contained in the training data. This could bias the models towards not finding, e.g. names of people that have many words, possibly leading to a misrepresentation of some communities in the results. Similarly, names that are uncommon, and may not have been found in the training data (due to e.g. different languages) would also be predicted less often.

Additionally, if similar NER techniques like this are used to, for example, anonymise private information (Hassan et al., 2018; Kleinberg et al., 2017) by first detecting and then altering personally identifiable information, failures by the model to detect some entities could lead to concrete harm to an individual's privacy. A larger focus on recall could help here, and humans could potentially filter the predictions of the model, as to minimise risk of not flagging something that needs to be removed.

### 8.4 Considerations

The embedding approach used here, while providing an easy way to observe some patterns in the data, should not be considered too literally for a few reasons:

- The actual word embeddings are 768 dimensional vectors, so compressing this into a 2D point potentially removes much of the original structure of the data.
- The embeddings, as well as the final layer is responsible for what the network predicts, and we did not take the differences in the final layer into consideration.

- We also compress all categories into a single point, and perform PCA on these means. This could also have effects on what we observe. Similarly, other dimensionality reduction techniques like T-SNE (Van der Maaten and Hinton, 2008) or Multidimensional Scaling (Borg and Groenen, 2005) might be more desirable and could reveal different insights.

- We additionally do not show the variation that the categories had around the means (for a less cluttered graph), but in general there was quite a lot of dispersion around this mean.

The above, coupled with the fact that contextual word embeddings are prone to pick up (and exhibit) biases from their training data (Tan and Celis, 2019), prompt us to warn against overgeneralising the results, or using these embeddings in downstream tasks, until more in depth analysis / repair is done into these biases.

### 8.5 Societal Implications

The transfer learning capabilities, as well as their limitations demonstrated here might contribute to a better real world usage of these models, although care and thoughtfulness is required before applying these techniques and models.

## References


David Ifeoluwa Adelani, Jade Z. Abbott, Graham Neubig, Daniel D'souza, Julia Kreutzer, Constantine Lignos, Chester Palen-Michel, Happy Buzaaba, Shruti Rijhwani, Sebastian Ruder, Stephen Mayhew, Israel Abebe Azime, Shamsuddeen Hassan Muhammad, Chris Chinenye Emezue, Joyce Nakatumba-Nabende, Perez Ogayo, Aremu Anuoluwapo, Catherine Gitau, Derguene Mbaye, Jesujoba O. Alabi, Seid Muhie Yimam, Tajuddeen Gwadabe, Ignatius Ezeani, Rubungo Andre Niyongabo, Jonathan Mukiibi, Verrah Otiende, Iroro Orife, Davis David, Samba Ngom, Tosin P. Adewumi, Paul Rayson, Mofetoluwa Adeyemi, Gerald Muriuki, Emmanuel Anebi, Chiamaka Chukwuneke, Nkiruka Odu, Eric Peter Wairagala, Samuel Oyerinde, Clemencia Siro, Tobius Saul Bateesa, Temilola Oloyede, Yvonne Wambui, Victor Akinode, Deborah Nabagereka, Maurice Katusiime, Ayodele Awokoya, Mouhamadane Mboup, Dibora Gebreyohannes, Henok Tilaye, Kelechi Nwaike, Degaga Wolde, Abdoulaye Faye, Blessing Sibanda, Orevaoghene Ahia, Bonaventure F. P. Dossou, Kelechi Ogueji, Thierno Ibrahima Diop, Abdoulaye Diallo, Adewale Akinfaderin, Tendai Marengereke, and Salomey Osei. 2021. Masakhaner: Named entity recognition for african languages. *Transactions of the Association for Computational Linguistics*, 9:1116–1131.

Jesujoba O. Alabi, Kwabena Amponsah-Kaakyire, David Ifeoluwa Adelani, and Cristina España-Bonet. 2019. Massive vs. curated word embeddings for low-resourced languages. the case of Yorùbá and Twi. *CoRR*, abs/1912.02481.

Emily M. Bender, Timnit Gebru, Angelina McMillan-Major, and Shmargaret Shmitchell. 2021. On the dangers of stochastic parrots: Can language models be too big? In *Proceedings of the 2021 ACM Conference on Fairness, Accountability, and Transparency*, FAccT '21, page 610–623, New York, NY, USA. Association for Computing Machinery.

Emily M Bender and Alexander Koller. 2020. Climbing towards nlu: On meaning, form, and understanding in the age of data. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 5185–5198.

Ingwer Borg and Patrick JF Groenen. 2005. *Modern multidimensional scaling: Theory and applications*. Springer Science & Business Media.

José Canete, Gabriel Chaperon, Rodrigo Fuentes, Jou-Hui Ho, Hojin Kang, and Jorge Pérez. 2020. Spanish pre-trained bert model and evaluation data. *Pml4dc at iclr*, 2020:2020.

Alexis Conneau, Kartikay Khandelwal, Naman Goyal, Vishrav Chaudhary, Guillaume Wenzek, Francisco Guzmán, Edouard Grave, Myle Ott, Luke Zettlemoyer, and Veselin Stoyanov. 2020. Unsupervised cross-lingual representation learning at scale. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics, ACL 2020, Online, July 5-10, 2020*, pages 8440–8451. 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, NAACL-HLT 2019, Minneapolis, MN, USA, June 2-7, 2019, Volume 1 (Long and Short Papers)*, pages 4171–4186. Association for Computational Linguistics.

∀, Wilhelmina Nekoto, Vukosi Marivate, Tshinondiwa Matsila, Timi Fasubaa, Tajudeen Kolawole, Taiwo Fagbohungbe, Solomon Oluwole Akinola, Shamsuddee Hassan Muhammad, Salomon Kabongo, Salomey Osei, et al. 2020. Participatory research for low-resourced machine translation: A case study in african languages. *Findings of EMNLP*.

Fadi Hassan, Josep Domingo-Ferrer, and Jordi Soria-Comas. 2018. Anonymization of unstructured data via named-entity recognition. In *International conference on modeling decisions for artificial intelligence*, pages 296–305. Springer.

Dan Hendrycks, Collin Burns, Saurav Kadavath, Akul Arora, Steven Basart, Eric Tang, Dawn Song, and Jacob Steinhardt. 2021. Measuring mathematical problem solving with the MATH dataset. *CoRR*, abs/2103.03874.

Neil Houlsby, Andrei Giurgiu, Stanislaw Jastrzebski, Bruna Morrone, Quentin de Laroussilhe, Andrea Gesmundo, Mona Attariyan, and Sylvain Gelly. 2019. Parameter-efficient transfer learning for NLP. In *Proceedings of the 36th International Conference on Machine Learning, ICML 2019, 9-15 June 2019, Long Beach, California, USA*, volume 97 of *Proceedings of Machine Learning Research*, pages 2790–2799. PMLR.

Jeremy Howard and Sebastian Ruder. 2018. Universal language model fine-tuning for text classification. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 328–339, Melbourne, Australia. Association for Computational Linguistics.

Haoming Jiang, Pengcheng He, Weizhu Chen, Xiaodong Liu, Jianfeng Gao, and Tuo Zhao. 2020. SMART: robust and efficient fine-tuning for pre-trained natural language models through principled regularized optimization. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics, ACL 2020, Online, July 5-10, 2020*, pages 2177–2190. Association for Computational Linguistics.

Bennett Kleinberg, Maximilian Mozes, Yaloe van der Toolen, et al. 2017. Netanos-named entity-based text anonymization for open science.

Guillaume Lample, Miguel Ballesteros, Sandeep Subramanian, Kazuya Kawakami, and Chris Dyer. 2016. Neural architectures for named entity recognition. In *NAACL HLT 2016, The 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, San Diego California, USA, June 12-17, 2016*, pages 260–270. The Association for Computational Linguistics.

Laurens Van der Maaten and Geoffrey Hinton. 2008. Visualizing data using t-sne. *Journal of machine learning research*, 9(11).

H. B. Mann and D. R. Whitney. 1947. On a Test of Whether one of Two Random Variables is Stochastically Larger than the Other. *The Annals of Mathematical Statistics*, 18(1):50 – 60.

Mónica Marrero, Julián Urbano, Sonia Sánchez-Cuadrado, Jorge Morato, and Juan Miguel Gómez-Berbís. 2013. Named entity recognition: fallacies, challenges and opportunities. *Computer Standards & Interfaces*, 35(5):482–489.

Mihai Masala, Stefan Ruseti, and Mihai Dascalu. 2020. Robert - A romanian BERT model. In *Proceedings of the 28th International Conference on Computational Linguistics, COLING 2020, Barcelona, Spain (Online), December 8-13, 2020*, pages 6626–6637. International Committee on Computational Linguistics.

Chester Palen-Michel, Nolan Holley, and Constantine Lignos. 2021. Seqscore: Addressing barriers to reproducible named entity recognition evaluation. *arXiv preprint arXiv:2107.14154*.

Jonas Pfeiffer, Ivan Vulic, Iryna Gurevych, and Sebastian Ruder. 2020. MAD-X: an adapter-based framework for multi-task cross-lingual transfer. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing, EMNLP 2020, Online, November 16-20, 2020*, pages 7654–7673. Association for Computational Linguistics.

Alec Radford, Karthik Narasimhan, Tim Salimans, and Ilya Sutskever. 2018. Improving language understanding by generative pre-training.

Alec Radford, Jeffrey Wu, Rewon Child, David Luan, Dario Amodei, Ilya Sutskever, et al. 2019. Language models are unsupervised multitask learners. *OpenAI blog*, 1(8):9.

Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, and Peter J. Liu. 2020. Exploring the limits of transfer learning with a unified text-to-text transformer. *J. Mach. Learn. Res.*, 21:140:1–140:67.

Sebastian Ruder. 2020. Why You Should Do NLP Beyond English. `http://ruder.io/nlp-beyond-english`.

Sebastian Ruder. 2021. Recent Advances in Language Model Fine-tuning. `http://ruder.io/recent-advances-lm-fine-tuning`.

Sebastian Ruder, Matthew E Peters, Swabha Swayamdipta, and Thomas Wolf. 2019. Transfer learning in natural language processing. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Tutorials*, pages 15–18.

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, CoNLL 2003, Held in cooperation with HLT-NAACL 2003, Edmonton, Canada, May 31 - June 1, 2003*, pages 142–147. ACL.

Samuel Sanford Shapiro and Martin B Wilk. 1965. An analysis of variance test for normality (complete samples). *Biometrika*, 52(3/4):591–611.

Yi Chern Tan and L. Elisa Celis. 2019. Assessing social and intersectional biases in contextualized word representations. In *Advances in Neural Information Processing Systems 32: Annual Conference on Neural Information Processing Systems 2019, NeurIPS 2019, December 8-14, 2019, Vancouver, BC, Canada*, pages 13209–13220.

Kees Versteegh. 2001. Linguistic contacts between arabic and other languages. *Arabica*, 48:470–508.

Wietse de Vries, Andreas van Cranenburgh, Arianna Bisazza, Tommaso Caselli, Gertjan van Noord, and Malvina Nissim. 2019. BERTje: A Dutch BERT Model. arXiv:1912.09582.

Linting Xue, Noah Constant, Adam Roberts, Mihir Kale, Rami Al-Rfou, Aditya Siddhant, Aditya Barua, and Colin Raffel. 2021. mt5: A massively multilingual pre-trained text-to-text transformer. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, NAACL-HLT 2021, Online, June 6-11, 2021*, pages 483–498. Association for Computational Linguistics.