Cross Lingual NER using Multilingual Word Embeddings

Aashish Singh[1], Simmi Mourya[1] John Zhang[1], and Sanjeevini Ganni[1]

University of Pennsylvania

Abstract. In this project we implement and evaluate various cross lingual NER models using bilingual and multilingual word embeddings. Our current method explores the use of a bi-LSTM deep neural network model in the NER task. Our reimplementation of the published baseline in [15] achieves an F1 score of 54.02 on the test set. With some extensions, we were able to boost the F1 score to 63.70.

1 Introduction

The task of NER- labelling words as Person, Organization, Location, Miscellanious and Others is a long standing task in NLP. Using bilingual embeddings to perform cross lingual task has enabled NLP scientists to make much progress in old/low resource language revival. Cross lingual NER tagging using bilingual embeddings is also very useful for information retrieval from untapped information hidden in low resource languages. Such cross lingual information transfer can be done through deep learning models making it a computational linguistics task. Our aim in this project is to be able to effectively transfer NER knowledge from a source language to a target language. Eventually this can then be adapted to transfer of NER knowledge from high resource source language to a low resource target language.

Problem definition: As described earlier, NER is the task in which the goal is to tag a word or sequences of words with a predefined category of labels which are most commonly- Person, Organization, Location and Other. The problem we seek to solve is- to generate a model that can identify the named entities in a target language L_t given labelled NER training data in a source language L_s, monolingual corpora of L_s, L_t (which we will convert to bilingual word embeddings) and a bilingual dictionary from L_s to L_t. This results in an *unsupervised transfer* of resources from L_2 to L_1 since no labelled training data is given for the target language. Motivation for solving this kind of problem comes from the fact that it can potentially help develop NER models for a low resource language L_t given high resource language L_s. In some instances, potentially we will have multiple source languages $L_{s_1}, ..., L_{s_n}$ from which we can perform transfer onto the target language. There are also some instances where we use resources shared between $L_{s_1}..L_{s_n}$ and L_t like m-BERT word embeddings to perform better on the L_t NER tagging model results

Illustrative Example: Given input training data (note that our training set also included Part-of-speech tag and syntactic chunk tag but for clarity here we do not show them) shown as left 2 columns in table.

English Word	NER tag	Spanish Word	NER tag
EU	B-ORG	El	O
rejects	O	Abogado	B-PER
German	B-MISC	General	I-PER
call	O	del	I-PER
to	O	Estado	I-PER
boycott	O	,	O
British	B-MISC	Darryl	B-PER
lamb	O	Williams	I-PER

The output of the model will be our model's NER tagging result on unseen Spanish text like as shown in right 2 columns of the table.

Motivation and Interest: We are interested in this cross lingual NER problem since it makes computational linguistics stronger to enable preservation/revival of old/low resource language. Additionally it allows knowledge gain from untapped sources in low resource language, helps understand the demographics, economics, political scenario of low resource language regions. Most importantly it also allows Pandemic spreading in small regions in local languages to be picked up if cross learning models like our NER model is in place. Additionally prevention methods in high resource language can reach low resource languages.

2 Literature Review

The shared tasks we will be adapting from are the CoNLL 2002 and 2003 language-independent named entity recognition tasks. In these shared tasks combined, 4 languages are represented: Spanish, Dutch, English, and German, and in each language we have four types of named entities as described above: persons, locations, organizations and names of miscellaneous entities that are not in the former three groups.

Participants are offered training and validation data for both languages, with a held-out test set used to rank the predictions of the final models. When these tasks were offered, most participants trained models for the datasets of the two languages separately. However, we will attempting to use the data in one language to predict the tags for another. The overview papers can be found in: [12] and [13], and links to the shared task homepages can be found here: *CONLL2002* and *CONLL 2003*.

One of the very first papers to discuss cross-lingual transfer in the NER setting is described in [14]. This paper was also one of the first to combine semi-supervised learning with cross-lingual transfer. In place of word embeddings, they describe a method of word clustering, an unsupervised probabilistic method which clusters words into a set number of clusters based on local distributional information.This paper was one of the first to show that a cross-lingual transfer of distributional information could provide a significant boost for transfer models.

Another paper that deals with the issue of cross-lingual NER is given in [9]. This paper achieved the state of the art, with around 65.95 F1 score on the Spanish dataset, and it did so while utilizing less resources than many other methods at the time. These other methods utilized resources such as large spans of parallel corpora (not just lexicons), and/or features learned using Wikipedia pages in both languages. Their method combines well with orthogonal features, and thus can also use Wikipedia features when needed. Other models at the time, such as [10] use two separate models trained using annotation projection, and projection of word embeddings, and combine their output to perform transfer of NER prediction.

Utilizing both annotation projection (from pure translation) as well as information obtained from projection of distributional information, [15] explores the unsupervised transfer of NER from resource-rich language to languages with no annotated resources. They propose to use self-attention to improve robustness in word order differences across languages, and to improve lexical item mapping, they propose a method to find translations based on bilingual word embeddings. The overall model is broken coarsely into four steps: 1) By using monolingual corpora, train separate word embedding matrices X and Y in the source and target languages respectively. 2) Projection of the embeddings from these two languages into a shared embedding space by optimizing embedding alignment via a provided dictionary. 3) Finding a nearest neighbour translation for each word in the source language using the shared embedding space. 4) Training an NER model, using these word translations along with the named entity tags from the English corpus (source training data).

The experiments are compared with the dictionary-based baseline proposed by [9]. Evaluation on extremely low-resource language 'Uyghur' has also been performed. Overall, the model achieves state-of-the-art results on commonly tested languages under cross-lingual setting, with lower resource requirements as compared to other past approaches. For this project, we plan to closely follow the methods and findings explained in [15] and possibly try to improve upon it.

[1] and [3] make use of subword features and use transfer learning to apply the Named Entity recognition task across language barriers. In [1], the authors use universal phonological character representations to make an NER model, that can be adapted to a new language with minimal data. They use attention neural models(LSTM-CRF) to do supervised training on one language.

More research on the use of multilingual approaches to cross-lingual tasks is warranted, given the generalizability of such methods and the seemingly better results. One such method, in the task of NER, is described in [4].

The proposed model uses multinomial adversarial networks to learn language-independent features that are shared by all the source languages, as well as a mixture-of-experts model to learn 'private' features that are specific to each source language. The final MAN-MoE (Mixture-of-Experts) model can utilize these language-invariant features, as well as weighted proportions of the language specific features.

When applied to the CoNLL NER dataset, the MAN-MoE model was able to achieve an F1 score of 73.5 for Spanish, even without using architecture specialized for the NER task such as a CRF layer at the end for sequence labeling. Their model also benefited greatly from using the unsupervised multilingual word embeddings [5] introduced in their previous work. Thus, it may be of interest to explore using these word embeddings in our model.

3 Experimental Design

3.1 Data

English data of CoNLL-2003 Shared task with POS tags: https://www.clips.uantwerpen.be/conll2003/ner/

The English data comes from the CoNLL 2003 shared task [13]. The CoNLL-2003 shared task data files contain four columns, the word, the part-of-speech (POS) tag, a syntactic chunk tag and the named entity tag, respectively. We focus on the word and the named entity tag, and as explained above O denotes non-entity, and the entities are PER, LOC, ORG, and MISC, with a B to denote the beginning of a phrase and I to denote inside. Here is an example:

U.N.	NNP	I-NP	I-ORG
official	NN	I-NP	O
Ekeus	NNP	I-NP	I-PER
heads	VBZ	I-VP	O
for	IN	I-PP	O
Baghdad	NNP	I-NP	I-LOC
.	.	O	O

Spanish Data of the CoNLL 2002 Shared Task with POS tags [12]: https://www.cs.upc.edu/ nlp/tools/nerc/nerc.html

The data mentioned we used is the POS enhanced version of Language-Independent Named Entity Recognition (I) (CoNLL-2002) shared task. The original data consists of only word and Entity tag for each entry. https://www.clips.uantwerpen.be/conll2002/ner/

The data consists of three columns: the word, POS tag, and named entity tag, respectively. As explained above in the literature review, the only difference here is that there is no column for the syntatic chunk tag. Here is an example:

Abogado	NC	B-PER
General	AQ	I-PER
tiene	VMI	O
lugar	NC	O
después	RG	O
de	SP	O
que	CS	O
un	DI	O
juez	NC	O
del	SP	O
Tribunal	NC	B-ORG
Supremo	AQ	I-ORG
del	SP	O
estado	NC	O
de	SP	O
Victoria	NC	B-LOC
(	Fpa	O
Australia	NP	B-LOC

The data consists of three files per language: one training file and two test files testa and testb, the sizes of the datasets are below:

Data File	Number of Sentences	Number of Words
eng.train (train)	14041	217661
eng.testa (train)	3250	54611
eng.testb (train)	3453	49887
esp.train (test)	8323	273037
esp.testa (dev)	1914	54837
esp.testb (test)	1517	53049

For our task, since we will be using two languages, we will combine all the English data sets together for training, use the Spanish development set for validation, and use the Spanish training and test set for the final evaluation. Thus, we will be using 20744 sentences in total for training, 1914 for development, and 9840 for testing. If we use the Dutch training set as well, we will be including an extra 23419 sentences. With all three datasets combined, labels overwhelmingly favor O, with non entities make up around 87% of all tokens. The data comes from respective news articles in the three languages, and were manually annotated by experts at the University of Catalonia and the University of Barcelona for the Spanish dataset and the University of Antwerp for the Dutch and English datasets. [12] [13]

3.2 Evaluation Metric

The metric we will be using for our task is the F1 score. It is the harmonic mean of the precision and recall, thus weighing the affects of both scores equally, but having a value closer to the minimum of the two. The equation is given by:

$$F_1 = \frac{2 \cdot \text{Precision} \cdot \text{Recall}}{\text{Precision} + \text{Recall}}$$

where Precision = tp/(tp+fp) and Recall = tp/(tp+fn). The F1 score is commonly used as an evaluation metric for categorization tasks with skewed class labels, and was first introduced in [11]. Most past papers in the NER domain use this metric, for example: [14], [9], [10], [8], [4], [1], [3], [15], etc.

3.3 Simple Baseline

Since the majority of labels are O in the task of NER, and average F1 score is taken over all the named entities, we could not use a majority class baseline for this problem. We also wanted a simple baseline where we could follow the setting of our task: training on one language and predicting on another. Thus, our simple baseline consisted of training a simple logistic regression model on the cross-lingual word embeddings of the two languages present. We included the embeddings of the previous, current, and next word as inputs to our model. The results of our baseline can be seen in the table below:

Processed 51533 tokens Gold: 3558 phrases Found: 3937 Phrases Correct: 1410	
Metric	Percentage
Overall F1:	37.63
LOC F1:	49.93
MISC F1:	4.34
ORG F1:	33.21
PER F1:	47.29

The low score for the MISC class can be explained by the fact that embeddings for miscellaneous entities are more varied. Our simple baseline gets an overall F1 score of about 38%, which is quite alright. This suggests that with sources of cross-lingual knowledge such as word embeddings, this cross-lingual task of NER is quite feasible.

4 Experimental Results

4.1 Published Baseline

Given an input of sequence of words (w1, w2, w3, ...) and each word's character sequence, we generate a kind of "hybrid word embedding" by concatenating the character embedding with the word embedding. The embeddings mentioned here are generated by a char and a word Bi-LSTM respectively. For OOV words we initialize vectors with random values from $-\sqrt{3/embeddingsize}$ to $\sqrt{3/embeddingsize}$. The word Bi-LSTM models the contextual dependency within each sentence. Hence the output of this word Bi-LSTM is context aware hidden representations (h1, h2, h3, ...) Then, a Self-Attention module is added on top of the word Bi-LSTM. It provides each word with a context feature vector based on all the words of a sentence, hence disregarding the ordering of words. As the context vectors are obtained irrespective of the wordsâ positions/order in a sentence, at test time, the model is more likely to see vectors similar to those seen at training time, which makes the model robust with respect to word order and hence offers better generalization. CRFs tend to encode the sequential information well, that is why a lot of applications of NER and POS tagging use CRF as a decoder in the end. The output of word level LSTM (h1, h2, h3, âŠ) are concatenated with the Attention output (h1a, h2a, h3a, ...) to make the final input for the Chain CRF layer, ([h1, h1a], [h2, h2a], [h3, h3a], ...) Finally, a CRF layer is applied on top of self-Attention word Bi-LSTM outputs ([h1, h1a], [h2, h2a], [h3, h3a], ...). The CRF defines the joint distribution of all possible output label sequences. For e.g. an example output label sequence could be the following: O , O, I-PER, B-PER, I-ORG, B-ORG, O, O. We use Viterbi algorithm for decoding the transition matrix weights to final output labels. To compute the loss associated with CRF, we also send in a mask input to the model which looks like this, it ensures to penalize only for the input positions for which there exists a label.

We achieved a F1 score of 0.58 on validation set by re-implementing the published baseline architecture. The paper achieves an F1 score of 72.37 ± 0.65. The difference in F1 score is majorly because of different training and testing dataset used by our experiment and the of algorithm used for cheap translations. We use the esp.train and esp.testb for testing (so that we have a bigger testing dataset) which contributes towards the low F1 score. The reason for using a standard dataset is multi-fold, 1.)availability, 2.) comparison with other benchmarks 3.) making results reproducible.

4.2 Extensions

Extension 1 Using m-BERT/BETO
Training data: As our first extension we employed contextualized word embeddings from multilingual BERT as our first sub experiment and BETO as our second sub experiment. For our first sub experiment using m-BERT we used the pre-trained model Bert-Base Multilingual Cased with 12 heads, 768 hidden, 110M parameters developed using [6]. This resulted in 768 dimension word embedding. We used this to get word embeddings for our cheap translation. This was then concatenated and passed along with normal 100 dimension word embeddings and char embeddings from reimplemented published baseline model to the word bidirectional LSTM. The final input to Attention Network is of the following dimension: *Batch_size * Max_length_of_statement * 968*. The 968 channels are a result of the aforementioned concatenation: (char embedding: 100 + word embedding: 100 + m-BERT embedding: 768).

Training: Unless explicitly stated, we train all our experiments for 30 epochs (for approx 120 minutes) with a learning rate of 0.015 using SGD optimizer. We use the BiLSTM-Attention-CRF architecture which was implemented before in the published baseline experiment. We modify the training and data pre-processing pipelines to accommodate the m-BERT embeddings. The final score was achieved using various parameter tuning approaches. We experimented with various learning rates, weight decays and optimizers. After careful observation of training and validation metrics over 30 epochs, we noticed that the model is over-fitting since training F1 reaches almost 0.90 while validation F1 saturates at 0.64. To overcome this, we use a weight-decay factor of 1e-4 for training and observe a bump of 0.011 in validation in F1 score. We also experiment with Adam optimizer with same learning rate (0.015) and a weight-decay factor of 1e-4. The results of this experiment were slightly worse than the SGD counterpart. The best F1 score for this experiment was 0.5907.

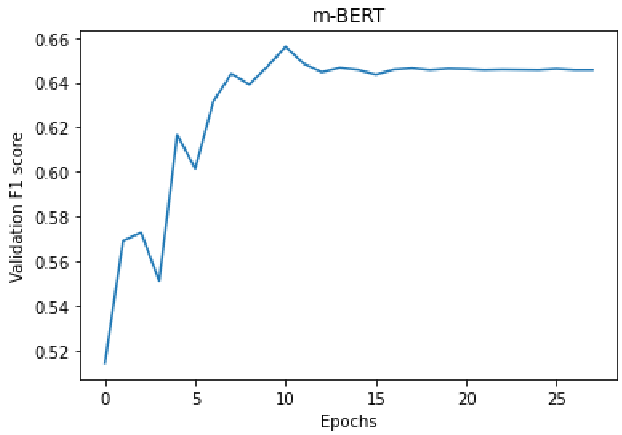

Fig. 1: Validation F-1 score trend for m-BERT best model

Best model (m-BERT experiment) configurations: Our best model from the m-BERT experiment produced an F1 score of 0.6560. We use the same BiLSTM-Attention-CRF architecture that was used in published baseline experiment. The model parameters are:
1. Number of epochs = 30
2. Optimizer = SGD (momentum=0.9)
3. Learning rate = 0.015
4. Weight-decay = 1e-4

Code Optimizations: For faster prototyping we vectorize the word lookup from m-BERT embeddings as well as minimize unnecessary interaction between CPU and GPU for faster computation.

For the second sub experiment in this extension, we used pre-trained BERT (found at [2]) which is equivalent of BERT just for Spanish text. We extracted the 768 dimension word embeddings from BETO. We used this to get word embeddings for our cheap translation. These embeddings were then concatenated and passed along with normal 100 dimension word embeddings and char embeddings from published baseline model to the word bidirectional LSTM. Tweaking for optimizer and using some regularization through weight decay we achieved our best F1 Validation score of 0.6270. While we hoped that BETO would perform better since it is mapping spanish words to its exclusive space as compared to m-BERT that maps 104 language word embeddings to a common space, we ended up getting better validation F1 score on m-BERT. This we suspect is due to the fact that m-BERT is much bigger than BETO in terms of encompassing number of spanish words. So while for BETO embeddings we were marking OOV words with a unique word embedding <UNK> for m-BERT we received word embeddings for all the words required by our model. Table below shows our best experiments results for each configuration.

Extension 1 results		
Type of Word embedding used	Training F1 score	Validation F1 score
m-BERT and Glove	0.89	0.656
Beto and GLOVE	0.94	0.627
Only Glove	0.92	0.580

Noting that m-BERT gave us the best score on the validation set, we continue with this model and note that it achieves an F1 score of 62.69 on the test set.

Extension 2 For our second extension, we tried a combination of two ideas. One is an idea following the paper described in [9], where the authors found that, as Dutch was more similar in some syntactic features to Spanish, using Dutch and English to transfer to Spanish gave better results. To help with this, we use the

UMWE multilingual vectors described in [5]. We develop vectors using English and Dutch as source languages and Spanish as the target, and we use these vectors to create a Dutch to Spanish translation as well as a new English to Spanish translation. As well, we use the 300 dimensional Spanish vectors created in this method to train our model. As such, our first idea consists of training on both the English and Dutch training set, and using the newly created UMWE vectors for training, which we will denote by +NED/UMWE.

For our second idea, we follow an idea proposed by [7], who use previous Language-Independent Entity Type Distributions. We introduce two methods for trying to include information about previous tags into our data. One is that for the English (and Dutch, if used) data, we use the ratio of times each word appears as an O, PER, LOC, ORG, or MISC entity as an extra 5 float input into our model. For our Spanish data, we find the closest English word, and use the ratios of that word as input. We denote this by +RATIO. Another tactic we tried was finding the average embedding of all the words in the training data for each of the five categories. Then, as an extra 5 float input, we pass in the distances to each of these averages. Note that both these methods use no extra data, and the second method takes into account some of the natural benefits of using word embeddings, where vector values are meaningful and similarity can be found using cosine distance. We denote this by +DIST.

Our results for the combinations of the above methods are found below, with scores being indicated on the test set:

Model	Test F1 Score
LSTM	54.02
LSTM +DIST	54.51
LSTM +DIST +NED/UMWE	**56.31**
LSTM +RATIO	50.03
LSTM +RATIO +NED/UMWE	54.34

Final Integrated Model After developing these two extensions, we tried to integrate the best performing combinations from both experiments into one model, with the hope that the improvements would be orthogonal. Thus, we used the m-BERT embeddings, as well as using distance to the average category embeddings and bringing in the Dutch dataset. As m-BERT handles finding multilingual embeddings, we decided to ditch the UMWE vectors and find the distance using the m-BERT values instead. The results are presented in the table below:

Model	Test F1 Score
LSTM	54.02
LSTM +DIST +NED/UMWE	56.31
LSTM +m-BERT and Glove	62.69
LSTM +m-BERT +DIST +NED	**63.70**

A graph of the CRF loss over the batches, as well as the training and validation F1 score, can be seen below:

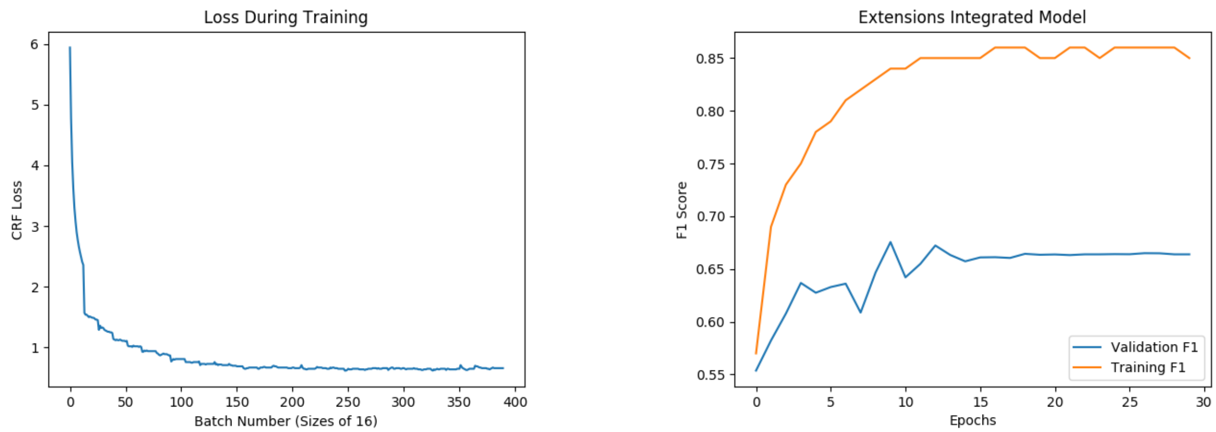

Fig. 2: CRF Loss and F1 Scores as the Model Trains

As we can see from the plateauing of the training and validation F1 curves, our model does not experience any overfitting on the data.

4.3 Error Analysis

As we can see in the confusion matrix below, the model best identifies the entities PER and ORG. It has a hard time trying to identify MISC entities. It usually misidentifies MISC as ORG or LOC based on the context.

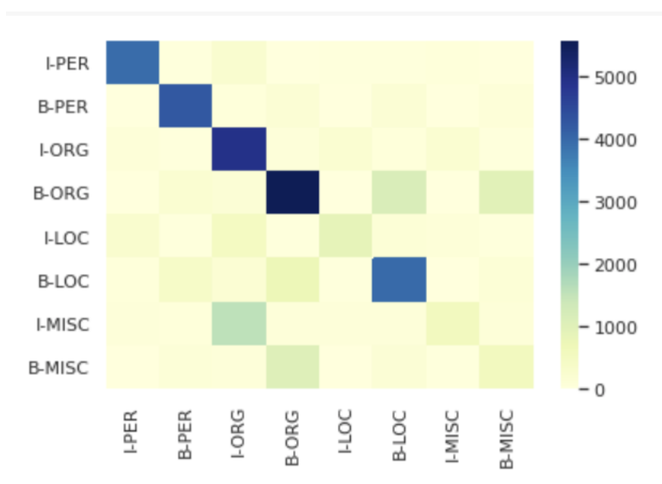

Fig. 3: Confusion Matrix for Entities

For example, it misidentifies "CrimeNet" as ORG instead of MISC. This could be due to that CrimeNet is a website and shares similar context to an organisation. "Ciudad" is the name of a city and also a publication. Even though it appears in the context of the publication(MISC) it is misidentified as LOC, this could be due the entity distribution information."Santander" is both Location and Organisation. It is misidentified most of the time as "ORG".

Some words like "de la" which mean "of the" are usually not considered entities when they appear in entities.

For example: "Ernesto Gomez de la Hera" (Ernesto Gomez of the Hera) all the words are considered of type PER. But, the model predicts "Ernesto Gomez" as PER as "Hera" as LOC.
When a LOC entity such as "A Coruna", the first word is labelled wrongly as non-entity and subsequently "Coruna" was labelled "B-LOC" instead of "I-LOC". The model confuses when two entities are next one another. It considers the second entity as part of the first entity. For example:

Entity	True Label	Predicted Label
Tribunal	B-ORG	B-ORG
Supremo	I-ORG	I-ORG
del	O	I-ORG
estado	O	I-ORG
de	O	I-ORG
Victoria	B-LOC	I-ORG

5 Conclusion

In this project we tried to use bilingual and multilingual word embeddings to solve the cross-lingual Named Entity Recognition problem. We started with a logistic regression model for our simple baseline. With the help of the work done in [15] we were able to implement a baseline that uses an LSTM-self attention-CRF based model. This resulted in an increase in the test F1 score from 37.63% to 54.02%. By using mutlilingual embeddings BERT, Dutch dataset and Entity distribution information we were successful in further improving the published baseline F1 score by 9%.

6 Acknowledgements

The researchers are grateful to Dr. Chris Callison-Burch[1]and Tatiana Tsygankova[1] for their help and guidance throughout the research. We would also like to thank Xie et al. [15] for providing us with their GloVe embeddings and since their work in the field inspired our base architecture. At last but not the least, we are thankful for our wonderful team for being utterly helpful, cooperative and mindful of each other's time and efforts given unfortunate COVID-19 circumstances.

[1] Department of Computer and Information Science, University of Pennsylvania

References

1. Bharadwaj, A., Mortensen, D.R., Dyer, C., Carbonell, J.G.: Phonologically aware neural model for named entity recognition in low resource transfer settings. In: Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing. pp. 1462–1472 (2016)
2. Canete, J., Chaperon, G., Fuentes, R., Perez, J.: Spanish pre-trained bert model and evaluation data. In: to appear in PML4DC at ICLR 2020 (2020)
3. Chaudhary, A., Zhou, C., Levin, L., Neubig, G., Mortensen, D.R., Carbonell, J.G.: Adapting word embeddings to new languages with morphological and phonological subword representations. arXiv preprint arXiv:1808.09500 (2018)
4. Chen, X., Awadallah, A.H., Hassan, H., Wang, W., Cardie, C.: Multi-source cross-lingual model transfer: Learning what to share. arXiv preprint arXiv:1810.03552 (2018)
5. Chen, X., Cardie, C.: Unsupervised multilingual word embeddings. arXiv preprint arXiv:1808.08933 (2018)
6. Devlin, J., Chang, M.W., Lee, K., Toutanova, K.: Bert: Pre-training of deep bidirectional transformers for language understanding. arXiv preprint arXiv:1810.04805 (2018)
7. Feng, X., Feng, X., Qin, B., Feng, Z., Liu, T.: Improving low resource named entity recognition using cross-lingual knowledge transfer. In: Proceedings of the Twenty-Seventh International Joint Conference on Artificial Intelligence, IJCAI-18. pp. 4071–4077 (2018)
8. Lample, G., Ballesteros, M., Subramanian, S., Kawakami, K., Dyer, C.: Neural architectures for named entity recognition. arXiv preprint arXiv:1603.01360 (2016)
9. Mayhew, S., Tsai, C.T., Roth, D.: Cheap translation for cross-lingual named entity recognition. In: Proceedings of the 2017 conference on empirical methods in natural language processing. pp. 2536–2545 (2017)
10. Ni, J., Dinu, G., Florian, R.: Weakly supervised cross-lingual named entity recognition via effective annotation and representation projection. arXiv preprint arXiv:1707.02483 (2017)
11. Rijsbergen, C.J.V.: Information Retrieval. Butterworth-Heinemann, USA, 2nd edn. (1979)
12. Sang, T.K., F., E.: Introduction to the CoNLL-2002 shared task: Language-independent named entity recognition. In: COLING-02: The 6th Conference on Natural Language Learning 2002 (CoNLL-2002) (2002)
13. Sang, T.K., F., E., De Meulder, F.: Introduction to the CoNLL-2003 shared task: Language-independent named entity recognition. In: Proceedings of CoNLL-2003. pp. 142–147 (2003)
14. Täckström, O., McDonald, R., Uszkoreit, J.: Cross-lingual word clusters for direct transfer of linguistic structure. In: Proceedings of the 2012 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies. pp. 477–487 (2012)
15. Xie, J., Yang, Z., Neubig, G., Smith, N.A., Carbonell, J.: Neural cross-lingual named entity recognition with minimal resources. arXiv preprint arXiv:1808.09861 (2018)