SAB at SemEval-2023 Task 2: Does Linguistic Information Aid in Named Entity Recognition?

Siena Biales
University of Tübingen
sienab1@gmail.com

Abstract

This paper describes the submission to SemEval-2023 Task 2: Multilingual Complex Named Entity Recognition (MultiCoNER II) by team SAB. This task aims to encourage growth in the field of Named Entity Recognition (NER) by focusing on complex and difficult categories of entities, in 12 different language tracks. The task of NER has historically shown the best results when a model incorporates an external knowledge base or gazetteer, however, less research has been applied to examining the effects of incorporating linguistic information into the model. In this task, we explored combining NER, part-of-speech (POS), and dependency relation labels into a multi-task model and report on the findings. We determine that the addition of POS and dependency relation information in this manner does not improve results.

1 Introduction

Named Entity Recognition (NER, Grishman and Sundheim, 1996) is a well established Natural Language Processing (NLP) task which entails extracting named entities (NEs), or specific types of proper nouns, from a text and classifying the type of entity. In the original task from 1996, entities were considered either a person, organization, or geographic location, but since then the classes have expanded to include more complex entity types such as creative works (book or film titles, etc.), groups, and products. While it can be considered a task on its own, NER has many applications including Information Extraction (IE) and Question Answering (QA) (Sun et al., 2018).

Complex named entities (such as titles of creative works, products, or groups) are especially difficult for NER systems, because unlike traditional NEs which are generally noun phrases, complex named entities may take various other forms (Ashwini and Choi, 2014). In addition, the domain of complex named entities is constantly expanding as new movies, books, and products are released. As a result, NER models must be adaptable in order to successfully identify and classify such entities.

SemEval-2023 Task 2: Multilingual Complex Named Entity Recognition II (MultiCoNER II, Fetahu et al., 2023b) is a shared task with the aim of investigating methods of identifying complex named entities in 11 individual language tracks, as well as a multilingual track. This task is a sequel to SemEval-2022 Task 11 (Malmasi et al., 2022b), but with different language tracks and a much more in-depth set of entity classes. Apart from having 33 fine-grained entity classes, the task also included simulated errors added to some of the test set language tracks to make the task more realistic and challenging.

A large portion of NLP research focuses on English and neglects lower-resource languages (Klemen et al., 2022). The language tracks defined in the task are English (EN), Spanish (ES), Hindi (HI), Bangla (BN), Chinese (ZH), Swedish (SV), Farsi (FA), French (FR), Italian (IT), Portuguese (PT), Ukrainian (UK), and German (DE), as well as a multilingual track (MULTI), to include a mix of high- and low-resource languages. It is interesting to note that each language either has approximately 9,700 training samples, or approximately 16,500 training samples. The 33 fine-grain entity classes can be generalized to 6 coarse parent classes: location, creative work, group, person, product, and medical.

Results from last year's MultiCoNER challenge indicated that transformers alone could not achieve high scores on complex named entities without the use of external knowledge bases or ensembles (Malmasi et al., 2022b). For this task, rather than attempting to achieve the top score, we explore how much can be gained by incorporating linguistic data (part-of-speech and dependency relation information) into a multi-task setup built on XLM-RoBERTa (XLM-R, Conneau et al., 2020).

Proceedings of the The 17th International Workshop on Semantic Evaluation (SemEval-2023), pages 1131–1137

2 Related Work

The notion of training NER and POS tasks in a multi-task setup was first proposed by Collobert and Weston in 2008. They utilized a Time-Delay Neural Network (TDNN, Waibel et al., 1989) to jointly train a network on six NLP tasks including POS and NER, and demonstrated that multi-task learning improved the generalization of the tasks.

Transformer architectures have shown promising results for NER on the common benchmark datasets CoNLL03 and OntoNotes (Devlin et al., 2019), however, BiLSTM architectures with a CRF layer have also achieved high performance (Huang et al., 2015). Most NER systems use some form of additional information, from POS tags to gazetteers or knowledge bases (Ratinov and Roth, 2009).

Klemen et al. (2022) analyze the effect of adding morphological features to long short-term memory (LSTM) and BERT models for NER, among other tasks. These morphological features include POS tags and universal feature embeddings. As with the experiments described in this paper, the additional linguistic information was obtained using Stanza. The paper experiments on 11 different languages, although there is minimal overlap with the languages examined in this task. They use BIO tagging for the NER task, but only include location, person, and organization as named entity classes. Unlike the XLM-R model used for the submitted system described in this paper, they use the cased multilingual BERT base model for their experiments.

They conclude that the additional information does not make a practical difference in the BERT-based model, although many of the languages do display a slight improvement when the additional information is applied to the LSTM model. This suggests that BERT already captures this information. It is unclear, however, if this should also hold for the multi-task setup on XLM-R at the center of the experiments in this paper.

Nguyen and Nguyen (2021) introduce PhoNLP, a multi-task learning model for joint part-of-speech tagging, named entity recognition and dependency parsing in Vietnamese. The model has a BERT-based encoding layer followed by three decoding layers of POS tagging, NER and dependency parsing. They employ $\text{PhoBERT}_{\text{base}}$ (Nguyen and Tuan Nguyen, 2020), a pre-trained Vietnamese language model, to encode the sentences. Then, the three tasks' decoding layers are applied.

They conclude from their experiments that PhoNLP's joint multi-task learning model performs better than single-task training on PhoBERT in Vietnamese, demonstrating that multi-task learning with POS and dependency tags applied with a transformers encoder can potentially be beneficial to this task.

3 Data

The dataset used for this exploration is Multi-CoNER v2 (Fetahu et al., 2023a). It is the second iteration of the original MultiCoNER dataset (Malmasi et al., 2022a). This dataset was assembled with the specific goal of addressing the contemporary challenges in NER with a focus on complex named entity recognition. The data is uncased, and often the text is very short, providing minimal context to the model. Additionally, it includes a large number of syntactically complex entities like movie titles, and long-tail entity distributions.

The test set that systems were evaluated on is far larger than the training and dev splits. The reasoning for this is to assess the generalizability of the models.

Named entities are categorized into a total of 33 classes, within 6 coarse parent classes: location (LOC), creative work (CW), group (GRP), person (PER), product (PROD), and medical (MED).

The label distributions of the training data are not even across classes, and only vaguely even across languages. The PER coarse-grain class on average represents about a third of the entities. LOC, GRP, and CW all represent about 15-17% of entity labels each. The least represented coarse classes are MED and PROD, which at worst represent under 5% of entities (FR), and at best only represent 11% (HI).

Hindi has the most even class distribution, with the greatest disparity being 17 percentage points between PER and CW. Italian has the worst disparity of the language tracks, with 34 percentage points between the PER and MED classes.

Noise was added to 30% of the instances on the test sets of (EN, ES, SV, PT, FR, ZH, and IT). The noise was added to either the context or entity tokens and is meant to represent typing errors based on the keyboard layout of the respective languages. The goal of this is to help determine whether the noise in entity tokens has an impact on NER prediction.

4 Methodology

In this section, we discuss the setup and methods used in the final model.

4.1 Preprocessing

The original dataset was in CoNLL format, with samples separated by blank lines, and each word of the sample in the first column, with the NER class label in the fourth column in BIO format. In order to facilitate easier dataset loading within the scripts, the dataset was first preprocessed into JSON format, which could then be loaded into the training scripts using the HuggingFace `datasets` library.

We wanted to utilize part-of-speech (POS) information as well as dependency relation information as part of our model. To obtain this on the dataset provided to us, we utilized Stanza (Qi et al., 2020) to tag all languages in the dataset with universal POS (UPOS) tags except for Bangla. For this, we were able to obtain POS tags using the `bnlp`[1] Python library, which were then manually mapped to UPOS format. Unfortunately, dependency relation labels were unavailable for Bangla. All of the POS and dependency relation tags were then included in the JSON dataset file as part of the preprocessing script.

4.2 Hyperparameter Tuning

Due to the sheer number of models to be trained, tuning hyperparameters for each model individually within the timeframe was not feasible for this task. The hyperparameters selected for tuning were learning rate, weight decay, and warm-up ratio. Primarily, hyperparameter tuning was performed on the `German` and `English` datasets, as those were the languages whose results could be manually analyzed, and they represented the two sizes of training samples present in the dataset, with `German` having 9,785 training samples and `English` having 16,778. The best set of hyperparameters found were then applied to the other languages.

The hyperparameter search was a Bayes search done using the Weights & Biases (`wandb`)[2] Python library. The batch size used for all multi-task models was 8. A more thorough explanation of the hyperparameter search and final hyperparameter selection is included in Appendix A.

[1] https://github.com/sagorbrur/bnlp
[2] https://wandb.ai/

4.3 Model Setup

The base model used in this task is XLM-RoBERTa$_{\text{base}}$ (henceforth referred to as XLM-R) (Conneau et al., 2020). The larger models could not be used due to limited computational resources, however, since our goal was primarily to observe relative changes rather than obtain the top score, this was sufficient for the task.

The primarily investigated model used was a multi-task setup utilizing the POS and dependency relation labels, in addition to the NER labels provided in the dataset. The setup consisted of a separate base XLM-R model per task, but a single shared encoder. This could be configured to train using any combination of the NER, POS, and dependency relation tags. Each individual model in the multi-task setup was fed the same complete list of possible class labels across tasks. The basic structure of the multi-task model is shown in Figure 1.

For all submissions apart from the `Bangla` and `MULTI` tracks, the full multi-task setup using all three tag sets was used. `Bangla` was trained with only NER and POS data due to the lack of dependency relation data available, and due to this gap in the dependency relation tags, the multilingual track could not be trained on all three tag sets without completely excluding Bangla from the training. Thus, the multilingual track was also only trained using NER and POS data.

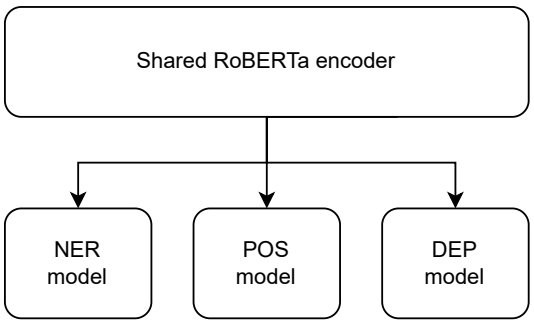

Figure 1: The multi-task model consists of a shared encoder and separate heads for each of the three tasks.

5 Results

Systems in this task were evaluated by F1 score on all 33 fine-grained classes as well as the 6 coarse-grained classes. The official ranking metric is the overall fine-grained macro-averaged F1 score. The test set included simulated errors in 7 of the language tracks (`EN`, `ES`, `SV`, `PT`, `FR`, `ZH`, and `IT`) in

Scores	BN	DE	EN	ES	FA	FR	HI	IT	PT	SV	UK	ZH	MULTI
Baseline F1	0.65	0.61	0.55	0.59	0.56	0.59	0.68	0.61	0.58	0.61	0.62	0.53	0.64
Overall F1	0.56	0.55	0.51	0.48	0.52	0.55	0.62	0.57	0.54	0.58	0.59	0.44	0.59
Clean scores			0.54	0.51		0.58		0.60	0.58	0.62		0.48	
Noisy scores			0.45	0.43		0.49		0.52	0.48	0.51		0.33	
Dev scores	0.68	0.64	0.52	0.52	0.59	0.59	0.74	0.64	0.64	0.63	0.65	0.58	0.63

Table 1: Macro-averaged F1 scores on the clean and noisy test sets individually, as well as base NER model results on the test set and development set F1 scores from the submitted models for comparison.

Category	BN	DE	EN	ES	FA	FR	HI	IT	PT	SV	UK	ZH	MULTI
LOC	0.83	0.81	0.78	0.75	0.72	0.75	0.84	0.78	0.79	0.88	0.82	0.67	0.81
MED	0.75	0.75	0.63	0.64	0.58	0.61	0.78	0.64	0.65	0.70	0.71	0.52	0.69
PROD	0.57	0.60	0.49	0.54	0.55	0.55	0.65	0.57	0.60	0.64	0.63	0.43	0.61
PER	0.81	0.87	0.87	0.87	0.79	0.89	0.83	0.91	0.87	0.89	0.88	0.76	0.89
GRP	0.78	0.72	0.64	0.67	0.66	0.67	0.82	0.72	0.71	0.71	0.76	0.62	0.73
CW	0.65	0.69	0.63	0.65	0.63	0.73	0.66	0.79	0.68	0.69	0.69	0.51	0.72

Table 2: Coarse-grained F1 scores for each language track evaluated on the test set.

order to make the task more realistic and difficult.

We submitted across all language tracks, achieving our best results on Hindi. Table 1 shows the results for each language track on the test set, and also displays the baseline and development set evaluation for comparison. In our experiments, we observed that the multi-task models could not outperform our baseline model. The most likely factor is the shared class labels across models, which was done in order to ensure the encoder would never be confused about duplicate IDs corresponding to different class labels. Although theoretically, the models should easily be able to learn that some labels are never used, this seems to not be the case. This is likely a result of using cross entropy loss in the models. With such a large label space, the theoretical worst case cross entropy is much higher, resulting in worse scores.

We can also see from the scores on the noisy test data in Table 1 that our system may not be very robust against such errors. The F1 scores on the corrupt data drop approximately 10 percentage points compared to the clean data across all affected language tracks.

From the coarse-grained F1 scores on the test set shown in Table 2, we can observe several trends. The PROD class consistently has the worst performance across all tracks. CW or MED are the second worst performing classes across all language tracks. The best performing classes are LOC and PER. These results are in line with expectations based on the known challenges of identifying and classifying open-class entity types such as products and creative works.

The full table of fine-grained F1 scores on the test set for the submitted models can be found in Appendix B. Identifying artwork seems to be particularly challenging for the models across all languages. Even the best performing model, Hindi, obtained an F1 score of 0.043 on artwork. Software seems like a comparatively easier form of creative work for the models, although still somewhat difficult. As expected, labels in the LOC and PER categories are the easiest for the models to identify.

As previously mentioned, our best performing language track was Hindi. It was interesting that English did not perform particularly well, considering that English is a high-resource language and Hindi is less so. Upon investigating why Hindi might have performed so well, one hypothesis is that it may be due to the proportion of unseen entities in the test data. While English has approximately 72% unseen entities in the development set and 62% in the test set, the test set for Hindi has a mere 28.7% unseen entities. This drastic difference may be why Hindi achieved such higher scores than English.

The goal of building a multi-task setup was to hopefully show improvement on the task of NER when including additional information from POS and dependency relation tags. However, as previously discussed, this unfortunately was not the case. Worse than adding nothing useful to the

model, the additional information actually showed a markedly detrimental impact on the task. Klemen et al. (2022) showed that BERT-based models did not benefit from extra POS information, but it did not harm the models.

There are many speculations which might explain these results. It may be that with the multi-task setup, the NER task was not seen frequently enough within the 10 epoch training span. This could be explored in future work either by extending the number of training epochs, or altering the NER task's learning rate, hopefully making the model favor the NER task. Another speculation is that the additional information may have hurt results because the POS and dependency relation labels were generated using Stanza, and are not necessarily all correct. Perhaps they are adding more noise than useful information.

6 Conclusion

This paper describes the multi-task system built on XLM-R submitted to the MultiCoNER II shared task at SemEval 2023 by SAB. Our system explores utilizing part-of-speech and dependency relation information, in addition to the named entity recognition labels provided in the dataset.

We obtain the strongest results on the `Hindi` language track, which we attribute to the data's far lower percentage of unseen entities in the test set than other languages. The simulated noise in the test set also consistently lowers scores by approximately 10 percentage points, indicating that these models do not generalize very well. We also show that our multi-task model using POS and dependency relation information does more harm than good for this NER task.

In future experiments we will explore separating out the labels for each model in the multi-task setup, to observe if this had an effect on the loss function. We would also potentially explore other means of obtaining the part-of-speech and dependency relation information besides Stanza, as every tagger produces different results of varying quality. Additionally it would be interesting to see if a different type of tokenizer (i.e. character-based instead of sub-word) would perform better or worse when tested on the simulated errors in the test dataset.

References

Sandeep Ashwini and Jinho D Choi. 2014. Targetable named entity recognition in social media. *arXiv preprint arXiv:1408.0782*.

Ronan Collobert and Jason Weston. 2008. A unified architecture for natural language processing: Deep neural networks with multitask learning. In *Proceedings of the 25th international conference on Machine learning*, pages 160–167.

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*, pages 8440–8451, Online. 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, Minneapolis, Minnesota. Association for Computational Linguistics.

Besnik Fetahu, Zhiyu Chen, Sudipta Kar, Oleg Rokhlenko, and Shervin Malmasi. 2023a. MultiCoNER v2: a Large Multilingual dataset for Fine-grained and Noisy Named Entity Recognition.

Besnik Fetahu, Sudipta Kar, Zhiyu Chen, Oleg Rokhlenko, and Shervin Malmasi. 2023b. SemEval-2023 Task 2: Fine-grained Multilingual Named Entity Recognition (MultiCoNER 2). In *Proceedings of the 17th International Workshop on Semantic Evaluation (SemEval-2023)*. Association for Computational Linguistics.

Ralph Grishman and Beth M Sundheim. 1996. Message understanding conference-6: A brief history. In *COLING 1996 Volume 1: The 16th International Conference on Computational Linguistics*.

Zhiheng Huang, Wei Xu, and Kai Yu. 2015. Bidirectional lstm-crf models for sequence tagging. *ArXiv*, abs/1508.01991.

Matej Klemen, Luka Krsnik, and Marko Robnik-Šikonja. 2022. Enhancing deep neural networks with morphological information. *Natural Language Engineering*, pages 1–26.

Shervin Malmasi, Anjie Fang, Besnik Fetahu, Sudipta Kar, and Oleg Rokhlenko. 2022a. MultiCoNER: A large-scale multilingual dataset for complex named entity recognition. In *Proceedings of the 29th International Conference on Computational Linguistics*, pages 3798–3809, Gyeongju, Republic of Korea. International Committee on Computational Linguistics.

Shervin Malmasi, Anjie Fang, Besnik Fetahu, Sudipta Kar, and Oleg Rokhlenko. 2022b. SemEval-2022

Task 11: Multilingual Complex Named Entity Recognition (MultiCoNER). In *Proceedings of the 16th International Workshop on Semantic Evaluation (SemEval-2022)*. Association for Computational Linguistics.

Dat Quoc Nguyen and Anh Tuan Nguyen. 2020. PhoBERT: Pre-trained language models for Vietnamese. In *Findings of the Association for Computational Linguistics: EMNLP 2020*, pages 1037–1042, Online. Association for Computational Linguistics.

Linh The Nguyen and Dat Quoc Nguyen. 2021. PhoNLP: A joint multi-task learning model for Vietnamese part-of-speech tagging, named entity recognition and dependency parsing. In *North American Chapter of the Association for Computational Linguistics*.

Peng Qi, Yuhao Zhang, Yuhui Zhang, Jason Bolton, and Christopher D Manning. 2020. Stanza: A Python natural language processing toolkit for many human languages. *arXiv preprint arXiv:2003.07082*.

Lev Ratinov and Dan Roth. 2009. Design challenges and misconceptions in named entity recognition. In *Proceedings of the Thirteenth Conference on Computational Natural Language Learning (CoNLL-2009)*, pages 147–155, Boulder, Colorado. Association for Computational Linguistics.

Peng Sun, Xuezhen Yang, Xiaobing Zhao, and Zhijuan Wang. 2018. An overview of named entity recognition. In *2018 International Conference on Asian Language Processing (IALP)*, pages 273–278. IEEE.

Alexander Waibel, Toshiyuki Hanazawa, Geoffrey Hinton, Kiyohiro Shikano, and Kevin J Lang. 1989. Phoneme recognition using time-delay neural networks. *IEEE transactions on acoustics, speech, and signal processing*, 37(3):328–339.

A Hyperparameter tuning

Table 3 shows the search criteria of the hyperparameters for the models. Batch size was held consistent at 8 and the searches were performed over 7 epochs. The learning rate was selected from a log uniform distribution within the range of 1e-3 and 1e-6. Weight decay and warm-up ratio were selected from an array of values ranging from 0 to 0.1. These searches were primarily performed on `English` and `German`, but the `MULTI` model required its own hyperparameter search, as well as `Bangla`, since those models could not include the dependency relation task.

The final hyperparameters used depended on whether the language's dataset was of comparable size to `English` or `German`, with the exception of `Bangla` and `MULTI`. The hyperparameters used in the final models are shown in Table 4. All models were trained for 10 epochs, and then the top model in the final three epochs was selected. All models were also trained with batch size 8.

Hyperparameter	Values
Learning rate	1e-3 to 1e-6
Weight decay	0, 0.05, 0.1
Warm-up ratio	0, 0.001, 0.01, 0.05, 0.1

Table 3: The hyperparameter values searched over for each model.

B Fine-grained results

Table 5 shows the full fine-grained results of the multi-task model on all 33 classes in the dataset.

Model	Learning rate	Weight decay	Warm-up ratio
DE, HI, ZH	3.5e-5	0.1	0.06
EN, ES FA, FR, IT, PT, SV, UK	5e-5	0.1	0.1
BN	1.5e-5	0.1	0
MULTI	4e-6	0.1	0.05

Table 4: The hyperparameter values used in the final models.

Fine-grained class	BN	DE	EN	ES	FA	FR	HI	IT	PT	SV	UK	ZH	MULTI
Station	0.82	0.63	0.70	0.60	0.76	0.69	0.80	0.64	0.68	0.71	0.71	0.70	0.72
HumanSettlement	0.83	0.84	0.82	0.79	0.75	0.77	0.85	0.82	0.82	0.90	0.85	0.67	0.83
Facility	0.61	0.57	0.57	0.52	0.52	0.59	0.59	0.64	0.56	0.66	0.60	0.51	0.61
OtherLOC	0.63	0.43	0.46	0.24	0.32	0.43	0.65	0.42	0.66	0.89	0.56	0.40	0.64
Symptom	0.62	0.39	0.41	0.15	0.49	0.51	0.67	0.47	0.36	0.46	0.46	0.22	0.46
AnatomicalStructure	0.68	0.69	0.58	0.60	0.47	0.48	0.79	0.58	0.59	0.69	0.72	0.51	0.65
Disease	0.75	0.70	0.58	0.59	0.54	0.59	0.77	0.56	0.63	0.67	0.65	0.51	0.65
Medication/Vaccine	0.71	0.73	0.65	0.64	0.65	0.62	0.74	0.66	0.67	0.71	0.76	0.47	0.70
MedicalProcedure	0.71	0.67	0.53	0.54	0.54	0.51	0.71	0.56	0.58	0.56	0.54	0.42	0.61
Drink	0.68	0.52	0.44	0.49	0.49	0.49	0.74	0.55	0.56	0.63	0.59	0.26	0.56
OtherPROD	0.46	0.50	0.38	0.42	0.51	0.47	0.56	0.48	0.57	0.58	0.54	0.33	0.53
Food	0.52	0.54	0.44	0.47	0.53	0.44	0.66	0.46	0.54	0.60	0.58	0.45	0.54
Vehicle	0.60	0.52	0.40	0.38	0.47	0.42	0.69	0.46	0.45	0.54	0.55	0.46	0.51
Clothing	0.27	0.42	0.46	0.37	0.26	0.47	0.70	0.43	0.37	0.47	0.46	0.29	0.47
Cleric	0.57	0.36	0.42	0.47	0.47	0.53	0.70	0.65	0.57	0.52	0.54	0.29	0.54
SportsManager	0.37	0.42	0.50	0.47	0.52	0.50	0.26	0.64	0.48	0.45	0.58	0.40	0.53
Athlete	0.59	0.68	0.72	0.70	0.55	0.72	0.72	0.82	0.65	0.69	0.78	0.63	0.74
Politician	0.51	0.46	0.49	0.51	0.54	0.54	0.59	0.50	0.55	0.60	0.51	0.38	0.55
Artist	0.60	0.67	0.71	0.70	0.71	0.75	0.64	0.81	0.72	0.71	0.69	0.59	0.75
Scientist	0.27	0.32	0.36	0.11	0.29	0.38	0.37	0.39	0.30	0.33	0.42	0.25	0.40
OtherPER	0.38	0.41	0.39	0.45	0.38	0.42	0.45	0.43	0.46	0.46	0.48	0.35	0.46
ORG	0.79	0.60	0.51	0.51	0.53	0.50	0.79	0.50	0.58	0.59	0.63	0.51	0.61
SportsGRP	0.81	0.80	0.72	0.68	0.81	0.72	0.90	0.76	0.75	0.78	0.84	0.71	0.77
MusicalGRP	0.52	0.58	0.52	0.61	0.59	0.64	0.62	0.73	0.64	0.67	0.75	0.50	0.68
CarManufacturer	0.58	0.52	0.48	0.50	0.64	0.59	0.73	0.63	0.58	0.54	0.62	0.43	0.60
PrivateCorp	0.51	0.18	0.26	0.18	0.32	0.39	0.65	0.21	0.00	0.23	0.20	0.30	0.34
AerospaceManufacturer	0.13	0.60	0.40	0.16	0.74	0.43	0.08	0.31	0.25	0.18	0.32	0.54	0.51
PublicCorp	0.59	0.51	0.46	0.56	0.58	0.53	0.69	0.61	0.71	0.53	0.70	0.38	0.60
VisualWork	0.58	0.59	0.57	0.58	0.71	0.77	0.58	0.86	0.63	0.68	0.69	0.46	0.73
Software	0.74	0.65	0.56	0.68	0.56	0.62	0.76	0.65	0.68	0.67	0.75	0.39	0.71
MusicalWork	0.40	0.64	0.60	0.56	0.49	0.58	0.40	0.75	0.65	0.65	0.59	0.38	0.67
WrittenWork	0.65	0.67	0.55	0.56	0.47	0.69	0.66	0.53	0.54	0.64	0.64	0.54	0.63
ArtWork	0.04	0.51	0.32	0.12	0.08	0.38	0.04	0.49	0.07	0.19	0.31	0.30	0.34

Table 5: Fine-grained F1 scores for each language track on the test set, grouped by coarse-grained category.