# Studio Ousia at the NTCIR-15 SHINRA2020-ML Task

Sosuke Nishikawa
Studio Ousia, Japan
The University of Tokyo, Japan
sosuke-nishikawa@nii.ac.jp

Ikuya Yamada
Studio Ousia, Japan
RIKEN AIP, Japan
ikuya@ousia.jp

## ABSTRACT

SHINRA2020-ML task aims to classify Wikipedia entities in 30 languages into Extended Named Entity[1] based on 920K Japanese Wikipedia entities with gold-standard entity types. To address this task, we propose a novel method to extract effective features from the Wikipedia descriptions. In particular, we use the two types of features, i.e., *text-based* and *entity-based* features, based on state-of-the-art neural embedding models. As a result, we achieve the highest micro F1 score in two languages (i.e., French and German) on the final submission, and competitive results on the other 7 languages.

## TEAM NAME

Studio Ousia

## SUBTASKS

SHINRA2020-ML (German, French, Arabic, Spanish, Hindi, Italian, Portuguese, Thai, Chinese)

## 1 INTRODUCTION

SHINRA2020-ML is a shared task that aims to assign fine-grained entity types to Wikipedia entities. Entity typing has shown to be useful for many downstream natural language processing (NLP) applications such as relation extraction [5] and question answering [2]. In this shared task, participants are required to assign entity types to Wikipedia entities in 30 *target* languages given 920K entities with gold-standard entity labels in the *source* language (i.e., Japanese). Extended Named Entity (ENE), consisting of 219 entity types, is adopted as the target entity types. For the detailed description of the task, please refer to Sekine et al..

In recent years, various studies have been conducted to assign fine-grained entity types to Wikipedia entities using several kinds of information sources such as the textual descriptions of entities [13], the contextual words of entity hyperlinks [10], and information stored in a knowledge graph [14]. Although these information sources are proven to be useful for this task, the effectiveness of combining these sources has not yet been well-explored.

In this paper, we propose a novel method that assigns fine-grained types to entities based on two kinds of effective features extracted from the Wikipedia descriptions. In particular, the proposed method is based on one *text-based* feature and two *entity-based* features using the state-of-the-art neural embedding models. These three features are simply concatenated, and used to address the task (see Figure 1).

[1]https://ene-project.info

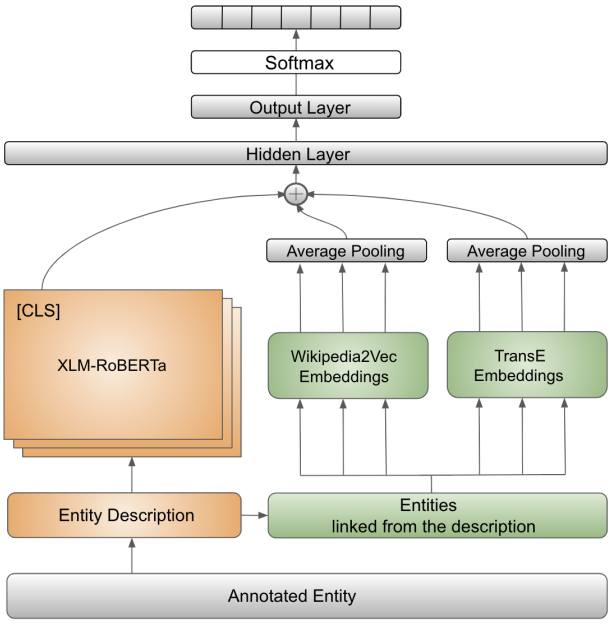


**Figure 1: The model architecture used in our experiments: Text-based features (orange) and entity-based features (green) are concatenated and fed into the hidden and output layer**

As a result, we obtain the first place in two languages, i.e., German and French, and competitive results in other languages. Furthermore, we find that combining the text-based and the entity-based features achieves enhanced results than the baseline model based only on the text-based feature.

## 2 OUR APPROACH

Our model uses *text-based* and *entity-based* features to assign fine-grained entity types to entities.

### 2.1 Text-based Features

To compute text-based feature, we use entity descriptions obtained from the corresponding Wikipedia pages, and simply feed them into the XLM-RoBERTa model [8], which is a multi-lingual contextualized representations that supports 100 languages, and achieve state-of-the-art results in various cross-language tasks. We use the output text representation (i.e., representation corresponding to the [CLS] input token) as a text-based feature.

| Model name | Precision | Recall | Micro F1 |
|---|---|---|---|
| XLM-RoBERTa$_{base}$ | 0.713 | 0.713 | 0.713 |
| XLM-RoBERTa$_{base}$ + Wikipedia2Vec | 0.724 | 0.724 | 0.724 |
| XLM-RoBERTa$_{base}$ + Wikipedia2Vec + TransE | 0.725 | 0.725 | 0.725 |
| XLM-RoBERTa$_{base}$ + Wikipedia2Vec + TransE (+Japanese) | 0.739 | 0.741 | 0.739 |

**Table 1: Micro F1 scores in German on the leaderboard**

## 2.2 Entity-based Features

Knowledge Base (KB) entities (e.g., Wikipedia), unlike words, provide unambiguous semantics and can be effectively used as features of text classification tasks [3, 12].

To compute the entity-based features, we first extract the entities linked from the Wikipedia descriptions of the target entity. [2] Then, we obtain the features by computing element-wise average of the embeddings corresponding to the extracted entities. Regarding the entity embeddings, we use Wikipedia2Vec [11] and TransE [1]. Wikipedia2Vec is an open-source tool[3] for learning the embeddings of entities using contextual words of hyperlinks and the internal hyperlink structure in Wikipedia. TransE is a conventional method for encoding information in a knowledge graph into embeddings.

| label pairs | | num |
|---|---|---|
| Ship | Weapon | 6503 |
| Archaeological_Place_Othe | Castle | 1428 |
| Company | Channel | 1200 |
| Line_Other | Car | 1123 |
| Shopping_Complex | Car_Stop | 1080 |
| Aircraft | Weapon | 1034 |
| Vehicle_Other | Weapon | 586 |
| Water_Route | Ship | 410 |
| Organization_Other | Channel | 399 |
| Company | Product_Other | 353 |

**Table 2: The top 10 frequent label pairs**

## 2.3 Model Architecture

The architecture of our proposed model is shown in Figure 1. We address this task based on a multi-class text classification model. As explained above, given an entity and its Wikipedia description, we first compute the three types of its representations using XLM-RoBERTa, Wikipedia2Vec, and TransE. We pass the concatenated representation to a hidden layer and tanh activation, and an output layer with softmax activation.

Additionally, since our model is based on the classification predicting a single label, it is impossible to assign multiple entity types to an entity. This is problematic because entities rarely have multiple entity types in this shared task. To address this issue, we introduce a simple heuristic to assign multiple entity types to an entity based on statistics obtained from the gold-standard labels. In particular, we first extract entity type pairs that frequently co-occur in the gold-standard labels. Then, if our model predicts an entity type contained in one of the extracted pairs, and the logit of the other entity type in the pair is above a threshold, we also add the other type to the prediction. We extract the top 10 frequent entity type pairs shown in Table 2 from the gold-standard labels, and tune the threshold based on the micro F1 score on the validation set.

# 3 EXPERIMENTAL SETUP

In this section, we describe our setup adopted in our experiments.

## 3.1 Data

To train our proposed model, we use the entity descriptions and their gold-standard entity type annotations provided by the organizers. To augment the training data, we adopt a simple strategy to concatenate the annotated entity descriptions of the target language and those of the source language (i.e., Japanese).

We create the text-based features based on the textual descriptions of entities in source and target languages. Note that because the XLM-RoBERTa model is capable of performing *cross-lingual* text classification, the performance of the model in the target language can be improved by training the model using the textual descriptions in the source language.

As mentioned in Section 2.2, we use Wikipedia2Vec and TransE embeddings to create entity-based features. To train the Wikipedia2Vec embeddings, we use the Wikipedia dump in the target language provided by the organizers. Furthermore, to compute entity-based features from an entity description of the source language, we convert the entities in the source language linked from the description to entities in the target language using the language links obtained from Wikidata.[4]

Regarding TransE embeddings, we use pre-trained embedding based on PyTorch BigGraph [4]. This embedding is trained on Wikidata.[5] We compute entity-based feature based on embeddings of entities linked from the description.

Due to the limitation of computational resources, we conduct preliminary experiments to investigate the effectiveness of combining features only in German. The dataset contains 274,732 German entities correspond to annotated Japanese entities. We create the validation set by randomly selecting 3000 from these entities.

[2] We use Wikipedia2Vec to extract entities linked from an entity description.
[3] https://github.com/wikipedia2vec/wikipedia2vec
[4] https://dumps.wikimedia.org/wikidatawiki/entities/
[5] https://github.com/facebookresearch/PyTorch-BigGraph

### 3.2 Setup

For Wikipedia2Vec embeddings, we set dim size to 300, and default values are used for other hyper-parameters. For the XLM-RoBERTa$_{base}$ model, we set the maximum word length to 512, batch size to 64, and dropout rate to 0.1. Also, the number of units in the hidden layer is 768

We train the model using AdamW optimizer [6] with the learning rate of 2e-5 and a gradient clipping of 1.0. We also employ a learning rate scheduler that linearly warms up the learning rate from zero to 500 steps. To reduce the training time and the GPU memory, we train the model using mixed precision [7]. Our implementation is built on PyTorch[6] and the Transformers library.[7] For the final submission to the shared task, we replace the text-based model with the XLM-RoBERTa$_{large}$ model and change the learning rate to 1e-05.

## 4 RESULTS

Table 1 shows the results of our preliminary experiments. The primary evaluation metric is the micro F1 score in this task. All scores reported in this paper are obtained using models trained on the training set and evaluated by submitting our model to the public leaderboard provided by the organizers.

We observe a significant improvement by combining XLM-RoBERTa and Wikipedia2Vec embeddings, and achieve a micro F1 score of 0.724. Furthermore, adding TransE embeddings improve the model slightly, with a micro F1 score of 0.725. We also observe that augmenting training data using entities in the source language lead to further improvement of the model, achieving a score of 0.739.

The final results for the shared task are shown in Table 3. Our model achieves the highest scores of 81.86 in German and 81.01 in French, and also achieves competitive results in other 7 languages.

| Language | Micro F1 | Rank |
|---|---|---|
| Arabic | 70.52 | 3 |
| German | **81.86** | **1** |
| Spanish, Castilian | 80.94 | 2 |
| French | **81.01** | **1** |
| Hindi | 69.75 | 3 |
| Italian | 81.21 | 4 |
| Portuguese | 81.40 | 3 |
| Thai | 76.36 | 3 |
| Chinese | 79.76 | 2 |

**Table 3: Results of final submissions**

## 5 CONCLUSION

In this paper, we propose a method to effectively assign types to entities based on the text-based features and entity-based features computed based on the entity descriptions of Wikipedia. We show that incorporating all of these features results in improved performance than a baseline model based only on text-based features. For the final submission, our model achieves the highest rank in German and French with 81.9 and 81.0 micro F1 score and competitive results in other 7 languages.

[6]https://pytorch.org/

[7]https://github.com/huggingface/transformers

## REFERENCES


[1] Antoine Bordes, Nicolas Usunier, Alberto Garcia-Duran, Jason Weston, and Oksana Yakhnenko. 2013. Translating Embeddings for Modeling Multi-relational Data. In *Advances in Neural Information Processing Systems 26*. 2787–2795.

[2] Hady Elsahar, Christophe Gravier, and Frederique Laforest. 2018. Zero-Shot Question Generation from Knowledge Graphs for Unseen Predicates and Entity Types. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*. 218–228.

[3] Evgeniy Gabrilovich and Shaul Markovitch. 2006. Overcoming the Brittleness Bottleneck Using Wikipedia: Enhancing Text Categorization with Encyclopedic Knowledge. In *Proceedings of the 21st National Conference on Artificial Intelligence - Volume 2 (AAAI'06)*. 1301–1306.

[4] Adam Lerer, Ledell Wu, Jiajun Shen, Timothee Lacroix, Luca Wehrstedt, Abhijit Bose, and Alex Peysakhovich. 2019. PyTorch-BigGraph: A Large-scale Graph Embedding System. In *Proceedings of the 2nd SysML Conference*.

[5] Yang Liu, Kang Liu, Liheng Xu, and Jun Zhao. 2014. Exploring Fine-grained Entity Type Constraints for Distantly Supervised Relation Extraction. In *Proceedings of COLING 2014, the 25th International Conference on Computational Linguistics: Technical Papers*. 2107–2116.

[6] Ilya Loshchilov and Frank Hutter. 2019. Decoupled Weight Decay Regularization. In *International Conference on Learning Representations*.

[7] Paulius Micikevicius, Sharan Narang, Jonah Alben, Gregory Diamos, Erich Elsen, David Garcia, Boris Ginsburg, Michael Houston, Oleksii Kuchaiev, Ganesh Venkatesh, and Hao Wu. 2018. Mixed Precision Training. In *International Conference on Learning Representations*.

[8] Sebastian Ruder, Anders Søgaard, and Ivan Vulić. 2019. Unsupervised Cross-Lingual Representation Learning. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics: Tutorial Abstracts*. 31–38.

[9] Satoshi Sekine, Masako Nomoto, Kouta Nakayama, Asuka Sumida, Koji Matsuda, and Maya Ando. 2020. Overview of SHINRA2020-ML Task. In *Proceedings of the NTCIR-15 Conference*.

[10] Yadollah Yaghoobzadeh and Hinrich Schütze. 2015. Corpus-level Fine-grained Entity Typing Using Contextual Information. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*. 715–725.

[11] Ikuya Yamada, Akari Asai, Jin Sakuma, Hiroyuki Shindo, Hideaki Takeda, Yoshiyasu Takefuji, and Yuji Matsumoto. 2020. Wikipedia2Vec: An Efficient Toolkit for Learning and Visualizing the Embeddings of Words and Entities from Wikipedia. *arXiv preprint 1812.06280v3* (2020).

[12] Ikuya Yamada and Hiroyuki Shindo. 2019. Neural Attentive Bag-of-Entities Model for Text Classification. In *Proceedings of the 23rd Conference on Computational Natural Language Learning (CoNLL)*. 563–573.

[13] Ikuya Yamada, Hiroyuki Shindo, and Yoshiyasu Takefuji. 2018. Representation Learning of Entities and Documents from Knowledge Base Descriptions. In *Proceedings of the 27th International Conference on Computational Linguistics*. 190–201.

[14] Yu Zhao, Anxiang Zhang, Ruobing Xie, Kang Liu, and Xiaojie Wang. 2020. Connecting Embeddings for Knowledge Graph Entity Typing. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*. 6419–6428.