Building of Informatics, Technology and Science (BITS)
Volume 4, No 1, Juni 2022 Page: 241−246
ISSN 2684-8910 (media cetak)
ISSN 2685-3310 (media online)
DOI 10.47065/bits.v4i1.1758

Supervised Learning Approaches for Nested People Entity Extraction in Indonesian Translated Quran

Dimitri Irfan Dzidny[1], Moch. Arif Bijaksana[2], Kemas Muslim Lhaksmana[3]

Informatics, School of Computing, Telkom University, Bandung, Indonesia
Email: [1]dimitriirfan@student.telkomuniversity.ac.id, [2]arifbijaksana@telkomuniversity.ac.id,
[3]kemasmuslim@telkomuniversity.ac.id
Correspondence Author Email: dimitriirfan@student.telkomuniversity.ac.id
Submitted: **24/06/2022**; Accepted: **30/06/2022**; Published**: 30/06/2022**

Abstract−Since the Quran is the primary holy book for Muslims, information extraction research on Quranic texts, especially in a form of People Entity Extraction, is an important task for further Quran and Tafseer understanding. The challenges in extracting people entities from the Quranic text is that many verses have a complex structure, such as nested entities, making it crucial to build a system that can extract the entity automatically, accurately, and quickly. People Entity Extraction on Quran itself is a task that aims to extract people entities in a sentence or verse, such as the name of a person, the name of a group, etc. on the Quranic texts. Example of input taken from snippet Surah Al-Baqarah verse 46 which reads "Those who believe that they will meet their Lord and that they will return to him" from that input the people entity extraction system is expected can identify people entities i.e. "Those who believe that they will meet their Lord". Currently, People Entity Extraction research for the Quran has not been widely carried out, only a few algorithms with scattered results have been conducted. In this research, we will use several supervised models which are Conditional Random Field (CRF), BiLSTM-CRF, and a pre-trained deep learning model based on IndoBERT transformers. We apply and perform a comparative analysis for the performance of those several models. We found out that deep learning based model, namely BiLSTM-CRF perform best at extracting people entities, whilst probabilistic based model, namely CRF, had difficulty in extracting people entities, specifically nested people entities.

Keywords: Quran; People Entity Extraction; Supervised Learning; Comparative Analysis; Nested Entities

1. INTRODUCTION

Quran is the Muslim holy book consisting of 114 surahs and around 6200 verses [1]. Because of the large number of surahs and verses, as well as various phrases in the Quran that have a complicated structure, such as nested entities, it is crucial to develop a system that can accurately, rapidly, and automatically extract information, particularly in the form of people entities. The application of the Named Entity Recognition system, which is dedicated to extract people entities, can aid in the automatic extraction of people entities and can also be used for future related research, making it valuable for a better comprehension of the Quran and its Tafseer.

Named Entity Recognition, or NER is a sub-task in information extraction that aims to identify a specific entity in a text, such as a person's name, organization, or geographic location [2]. In this research, what will be emphasized is the use of NER in extracting people entities in the Quran. People entity extraction is a NER task that extracts only people entities rather than the complete named entity, such as a person's name, a group's name, and so on. The text of the verse from the Quran serves as the input to the people entity extraction system, and the people entity extracted by the system serves as the output. Figure 1, shows an example of input from Surah Al-Anfal verses 46-47 the system is expected to identify people entities in the input sentences or ayat. Extracted sentences three and four are examples of people entity extraction from nested people entities, where green sentences are level one entities and blue sentences are level two entities. Only nested entities up to two levels is used in this research. The development of a NER system that can automatically identify and extract people entities in the Quran is important for furthering our understanding of the Quran; moreover, the extracted people entity may be useful for future research that requires people entities to obtain specific information such as a question answering system for the Quran.

In its application, NER system can be divided into several approaches. Earlier NER systems were built with handcrafted rules, lexicons, orthographic features, and ontologies. The system is then followed up with NER based on feature engineering and machine learning [3]. Then, in recent years, NER systems based on neural networks with minimal feature engineering are becoming more popular [4]. The purpose of this study is to extract people entities with several algorithms, then compare their performance. The systems are built based on machine learning and neural networks. For machine learning, a probabilistic model, Conditional Random Field (CRF) algorithm is used [5], and for neural networks, BiLSTM-CRF and pre-trained deep learning model IndoBERT is used. BiLSTM-CRF which was first introduced by Zhiheng Huang *et al*, is a combination of Bidirectional LSTM and CRF [6]. IndoBERT is a pre-trained BERT model for Indonesian language which are trained on indo4B dataset that consists of social media text, blog, news, and website [7], BERT itself is a transformer based based model that uses attention mechanism [8]. The three models were chosen because they were often used at the time for NER and sequence labeling problems.

The main contribution of this research is the extraction of people entities using the aforementioned algorithms and providing a full comparative analysis of the performance benchmarks of the algorithms used. The performance of each algorithm will be measured by the evaluation metric, namely F1-score. This research of several supervised learning algorithms will provide insight for researchers who will use these approaches in the future.

Building of Informatics, Technology and Science (BITS)
Volume 4, No 1, Juni 2022 Page: 241−246
ISSN 2684-8910 (media cetak)
ISSN 2685-3310 (media online)
DOI 10.47065/bits.v4i1.1758

Input ayat

> "Dan taatilah Allah dan **Rasul-Nya** dan janganlah kamu berselisih, yang menyebabkan kamu menjadi gentar dan kekuatanmu hilang dan bersabarlah. Sungguh, Allah beserta **orang-orang sabar.**"

> "Dan janganlah kamu seperti **{{{orang-orang yang keluar dari kampung halamannya} dengan rasa angkuh dan ingin dipuji orang (ria)}** serta menghalang-halangi (orang) dari jalan Allah.**}** Allah meliputi segala yang mereka kerjakan."

Output entity

1. Rasul-Nya (His Apostle)
2. orang-orang sabar. (those who endure)
3. orang-orang yang keluar dari kampung halamannya (those who went out of their homes)
4. orang-orang yang keluar dari kampung halamannya dengan rasa angkuh dan ingin dipuji orang (those who went out of their homes full of their own importance)

Figure 1. An example of the input paragraph with the intended output of people entities; note that in the second paragraph there are examples of nested entities wrapped in red brackets; entities extracted at the output are only nested entities up to two levels, namely those colored green and blue.

2. RESEARCH METHODOLOGY

2.1 Experimentation Flow

In this research, there are three models that are used to extract nested people entities in Indonesian translated Quran, that is; CRF, BiLSTM-CRF and IndoBERT. The three models are trained with a dataset based on the Indonesian Quran corpus for the training and evaluation process. Before the three models are trained, the dataset is processed first in pre-processing phase, afterwards the data is then divided into two sets, namely train data to train the model and test data to evaluate the model. After carrying out the training phase, the three models are carried out in the testing phase, where the three models predict labels on data that have not been seen in the training phase. The results of the testing phase are then evaluated for the model's ability to extract entities at level-1 and level-2. Figure 2 shows the process of the experiment carried out from start to finish.

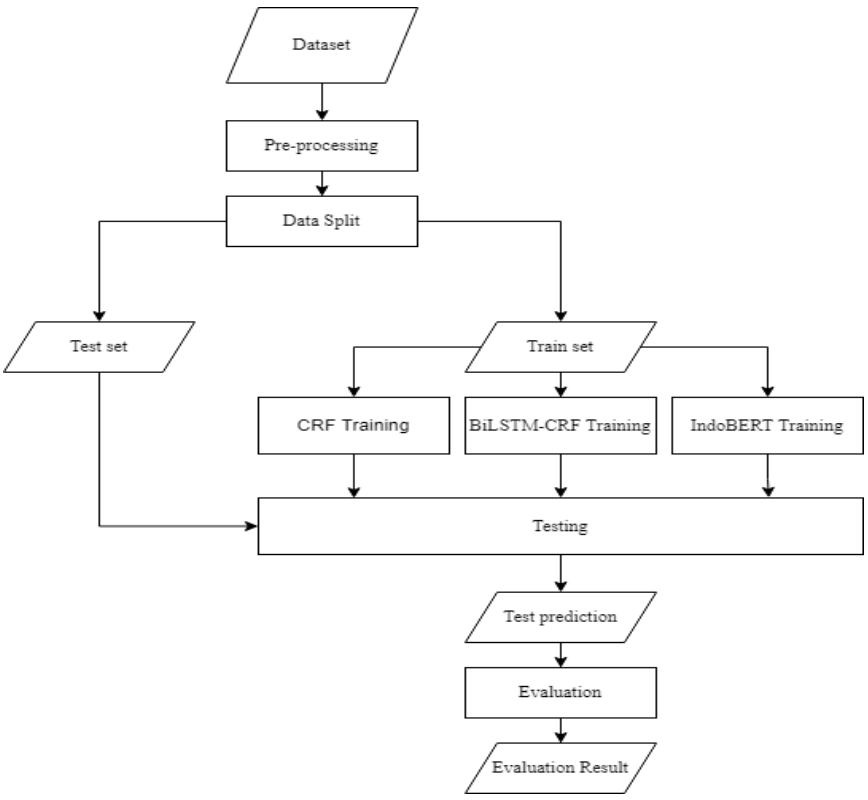

Figure 2. Flow of the experiment that are carried out

Building of Informatics, Technology and Science (BITS)
Volume 4, No 1, Juni 2022 Page: 241−246
ISSN 2684-8910 (media cetak)
ISSN 2685-3310 (media online)
DOI 10.47065/bits.v4i1.1758

2.2 Dataset

The dataset is taken from the Tanzil Quran corpus which includes Juz 1 through Juz 6. The entity tag used in this research is PER (person), which represents people entities, and O for entities outside people entities. The format used to label people entities is the IOB format, which was first introduced by Ramshaw and Marcus [9], with the dataset format in accordance with the dataset in the NER CoNLL-2003 [10]. There are two types of labeling based on the level of nested entities, the type of labels is utilized based on the label format used in Pham Quang Nhat Minh's research in the NER VLSP-2018 work [11], level-1 for nested entities up to one level, level-2 for nested entities up to two levels, and joint-tag, which is a concatenation of label level-1 and level-2. In the dataset there is also a POS feature for every word obtained at the pre-processing stage. Table 1 shows the structure of the dataset used, taken from a snippet of a Quranic verse. The dataset used can be found on https://github.com/dimitriirfan/nested-people-entities-quran.

Table 1. Example of the dataset structure used

Surat	Ayat	Kata	POS	Level-1	Level-2	Joint-tag
2	154	orang-orang (those)	NN	B-PER	B-PER	B-PER+ B-PER
		yang (who)	SC	I-PER	I-PER	I-PER+ I-PER
		gugur (died)	VB	I-PER	I-PER	I-PER+ I-PER
		di (in)	IN	O	I-PER	O+I-PER
		jalan (the way of)	NN	O	I-PER	O+I-PER
		Allah (Allah)	NNP	O	I-PER	O+I-PER
		bahwa (that)	SC	O	O	O+O

2.3 Data Pre-processing

Before the data enters the training and testing stage, the data will be pre-processed first. The pre-processing techniques that will be carried out on the data are as follows:

a. **Case folding**, i.e. the process by which all words are converted into lower case format
b. **Punctuation**, removing all punctuation marks, such as ".", ",", "!", etc. on a sentence.
c. **Tokenization**, breaking sentences into small units of words (tokens)
d. **POS Tagging**, marks the word of a sentence as a function of its sentence structure or its Part-of-speech

2.4 Data Split

To carry out training and testing the model, two separate and non-overlapping datasets are needed, with that dataset is divided into two, namely train data and test data. Train data is 80% of the total data and test data is 20% of the total data.

2.5 Model Training Phase

At this stage, the three models used were trained on the same train data. The model is trained on joint-label which is a concatenation of entity labels at level one (level-1) and level two (level-2). The three models have different parameter settings for each model used, the source code of the implementation can be found on https://github.com/dimitriirfan/nested-people-entities-quran. The following is a breakdown of each model used:

a. **CRF**, input for CRF is in the form of features of words in the data train. The features that will be used are word-level features (word cases, suffixes, prefixes, and neighboring words), and POS feature.
b. **BiLSTM-CRF** , the input for BiLSTM-CRF is the word from the data train which is represented in the word embedding. This model uses the Flair framework [12] where there are several word embedding that can be used. In this experiment the word embedding that will be used include Word2Vec [13], FastText [14], [15], and Glove [16]. Word2Vec and FastText uses two training methods namely CBOW and skip-gram. The three word embeddings were trained using the Indonesian Wikipedia corpus with embedding dimensions of 300 dimensions.
c. **IndoBERT** , this model is a BERT model that has been pre-trained on the Indonesian language corpus Indo4B [7], which is then used for the task of sequence labeling, in this case the extraction of people entities. The model will have an additional layer on the output which is a dropout layer, and one dense neuron. The optimizer that will be used is Adam. The input of the model is an embedding vector consisting of ID tokens for each word obtained from the pre-trained tokenizer IndoBERT.

2.6 Model Testing Phase

The three models that have been trained are then carried out in the testing phase using test data consisting of data that has never been seen by the model. The data goes through the same pre-processing stages as the train data. The three models that have been trained will then predict joint-tag labels for each data in the test data (B-PER+B-PER, I-PER+I-PER, O+I-PER, O+O). To evaluate the model's ability to predict nested entities at each level, the prediction results are then separated based on the entity level.

2.7 Evaluation Metrics

Building of Informatics, Technology and Science (BITS)
Volume 4, No 1, Juni 2022 Page: 241−246
ISSN 2684-8910 (media cetak)
ISSN 2685-3310 (media online)
DOI 10.47065/bits.v4i1.1758

In this research, there are three main measurable metrics that are used to measure the performance of each model to be built. Precision is the percentage of correctly named entities found by the system, recall is the percentage of named entities found by the system [10], then F1-score which is the harmonic average between precision and recall. The prediction of the named entity is said to be true if the entity extracted by the system matches the entity in the actual data [10]. The evaluation was carried out using the Seqeval software which was built based on the evaluation of the NER CoNLL task [17].

3. RESULT AND DISCUSSION

In this research we use three different algorithms to extract nested people entities in Indonesian translated Quran, and compare those three algorithms performance based on our primary metrics F1 score. We found that BiLSTM-CRF perfomed best compared to the other two model that is CRF, and IndoBERT. Based on the results of the experiments that have been carried out, each model and the variation of features used have different results, table 3 shows the detail

Table 2. Best performing model

Model	Fitur	Level-1 F1	Level-2 F1	Average F1
IndoBERT	IndoBERT embedding	0.78	0.73	0.755
CRF	Prefix + POS	0.763	0.719	0.741
BiLSTM-CRF	**FastText (skip-gram)**	**0.81**	**0.76**	**0.785**

of evaluation results for all variants, and table 2 shows the best performed variants of each model. For CRF, the variant is the model that uses the prefix + POS feature, variant that does not use suffix feature, which produces an F1 score in extracting level-1 entities of 76% and an F1 score in extracting level-2 entities by 72% on average producing a 74% score. It can be seen in table 2 that the removal of prefix feature for the CRF model decreases the model's performance by 7% on level-1 entities and level-2, this shows that prefix features are important in achieving high CRF performance. The removal of POS or suffix feature slightly improves the model performance, the removal of POS feature increases performance by 0.7% on level-1 and 1.1% on level-2, as for the removal of suffix feature increases performance by 0.6% on level-1 and 1.9% on level-2, removal of both suffix and POS feature increases performance by 0.1% on level-1 and 1.9% on level-2. The removal of suffix is the best variant for CRF, since the model performance increases by

Table 3. Experiment results for every model variants. "[all]" feature consist of feature: prefix + suffix + POS

Surat	Ayat	Level-1			Level-2		
Model	**Fitur**	**Precision**	**Recall**	**F1 Score**	**Precision**	**Recall**	**F1 Score**
IndoBERT	**IndoBERT embedding**	0.76	0.80	**0.78**	0.71	0.74	**0.73**
Conditional	[all]	0.803	0.716	0.757	0.742	0.662	0.700
Random	[all] -POS	0.812	0.721	**0.764**	0.756	0.671	0.711
Field (CRF)	**[all] -suffix**	0.815	0.716	0.763	0.769	0.676	**0.719**
	[all] -prefix	0.741	0.631	0.681	0.693	0.590	0.637
	[all] -POS -suffix	0.810	0.712	0.758	0.769	0.676	**0.719**
	[all] -POS -prefix	0.737	0.631	0.680	0.684	0.586	0.631
	baseline	0.680	0.604	0.640	0.650	0.577	0.611
BiLSTM-CRF	Word2Vec (CBOW)	0.77	0.65	0.70	0.73	0.62	0.67
	Word2Vec (skip-gram)	0.78	0.59	0.67	0.71	0.53	0.61
	FastText (CBOW)	0.79	0.74	0.76	0.76	0.71	0.74
	FastText (skip-gram)	0.83	0.79	**0.81**	0.78	0.74	**0.76**
	GloVe	0.84	0.72	0.77	0.75	0.65	0.70

1.25% on average, the highest average increase compared to the other variants. For the best BiLSTM- CRF model is the model which uses the FastText (skip-gram) feature as word embedding which results in an F1 score in extracting level-1 entities of 81% and an F1 score in extracting level-2 entities of 76%, a significant increase compared to using another word embedding. BiLSTM- CRF with Word2Vec (skip-gram) feature produces the worst performance among other BiLSTM-CRF models. Finally, the IndoBERT model produces an F1 score in extracting level-1 entities of 78% and an F1 score in extracting level-2 entities of 73%. Based on the evaluation results in table 3, the BiLSTM-CRF model with the FastText (skip-gram) feature is the best model that produces the highest F1 score both in predicting entities at level-1 or level- 2, then followed by IndoBERT.

Furthermore, to more deeply compare the performance of the model, a model evaluation will be carried out in classifying the labels on the joint-tag for the three models taken from each algorithm using the feature that produces the best F1 score, according to the model in table 2.

Building of Informatics, Technology and Science (BITS)
Volume 4, No 1, Juni 2022 Page: 241−246
ISSN 2684-8910 (media cetak)
ISSN 2685-3310 (media online)
DOI 10.47065/bits.v4i1.1758

Table 4. Experiment results for every model variants. For CRF, "[all]" feature consist of feature: prefix + suffix + POS

Label	IndoBERT			BiLSTM-CRF			CRF			Support
	Precision	Recall	F1 Score	Precision	Recall	F1 Score	Precision	Recall	F1 Score	
B-PER+B-PER	0.94	0.91	0.92	0.95	0.90	0.92	0.97	0.85	0.91	222
I-PER+I-PER	0.96	0.85	0.91	0.94	0.82	0.87	0.96	0.70	0.81	792
O+I-PER	0.54	0.38	**0.44**	0.80	0.32	**0.46**	0.81	0.15	**0.25**	114
O+O	0.98	0.99	0.98	0.95	0.90	0.92	0.96	1.00	0.98	7174

Table 4 shows the performance of the best performing IndoBERT, BiLSTM-CRF, and CRF models in classifying labels on joint-tags; the three models have good performance in predicting labels on joint-tags except for O+I-PER labels. The low performance of the model in classifying O+I-PER labels indicates that the model is having a difficulty in extracting level-2 nested entities, however in this experiment the BiLSTM-CRF model remains the best model to be used in extracting people entities since it has the highest F1 score in classifying O+I-PER labels, that is 46%, followed by IndoBERT with a 44% F1 score, and finally the CRF model with an F1 score of 22%.

Table 5. Predictions of the three models for nested entities at level-1

Kata	True label	Model Prediction		
		CRF	BiLSTM-CRF	IndoBERT
Adapun (as for)	O	O	O	O
orang-orang (those)	B-PER	B-PER	B-PER	B-PER
yang (who)	I-PER	I-PER	I-PER	I-PER
beriman (believe)	I-PER	I-PER	I-PER	I-PER
dan (and)	O	O	O	O
mengerjakan (do)	O	O	O	O
amalan-amalan (deeds)	O	O	O	O
yang (that are)	O	O	O	O
saleh (righteous)	O	O	O	O
maka (, then)	O	O	O	O

Table 5 shows examples of good classifications of the three models for entities at level-1, all three models perform well in extracting people entities at level-1

Table 6. Predictions of the three models for nested entities at level-1

Kata	True label	Model Prediction		
		CRF	BiLSTM-CRF	IndoBERT
Adapun (as for)	O	O	O	O
orang-orang (those)	O	O	O	O
yang (who)	B-PER	B-PER	B-PER	B-PER
beriman (believe)	I-PER	I-PER	I-PER	I-PER
dan (and)	I-PER	I-PER	I-PER	I-PER
mengerjakan (do)	I-PER	O	I-PER	I-PER
amalan-amalan (deeds)	I-PER	O	I-PER	I-PER
yang (that are)	I-PER	O	I-PER	I-PER
saleh (righteous)	I-PER	O	I-PER	I-PER
maka (, then)	O	O	O	O

Table 6 shows examples of good classifications of the models except for model CRF, it shows the difficulty of CRF model in predicting people entities at level-2.

Table 7. An example of the three models label predictions at level-1

Kata	True label	Model Prediction		
		CRF	BiLSTM-CRF	IndoBERT
orang-orang (those)	B-PER	B-PER	B-PER	B-PER
yang (who)	I-PER	I-PER	I-PER	I-PER
menepati (keep their)	I-PER	I-PER	I-PER	I-PER
janjinya (promises)	I-PER	O	I-PER	I-PER
apabila (when)	I-PER	O	O	O
ia (they)	I-PER	O	O	O
berjanji (promise)	I-PER	O	O	O

Building of Informatics, Technology and Science (BITS)
Volume 4, No 1, Juni 2022 Page: 241−246
ISSN 2684-8910 (media cetak)
ISSN 2685-3310 (media online)
DOI 10.47065/bits.v4i1.1758

Table 7 shows examples of cases where the three models incorrectly predict level-2 entities. This shows that the three models are not yet able to consistently extract people entities, especially people entities at level-

4. CONCLUSION

In this study, we compared several supervised learning models in extracting people entities from the Quran. These models include IndoBERT, BiLSTM-CRF, and CRF. Based on experimental results, we found that deep learning-based models have high performance in extracting people entities, these models are BiLSTM-CRF and IndoBERT models, where BiLSTM-CRF with FastText (skip-gram) features is the model that produces the highest performance based on the obtained F1 score. BiLSTM-CRF model is able to produce an F1 score in extracting entities at level-1 of 81% and entities at level-2 of 76%, on average 78.5%. We also found that the three models built had difficulties in classifying the O+I-PER label, where only BiLSTM-CRF and IndoBERT models were able to classify it with an F1 score of more than 44%. This indicates that the performance of the three models in entity extraction at level-2 is not yet optimal. For further research, due to the lack of training data for nested entities, which causes low performance when extracting nested entities, additional training data can be done to improve the performance of the model to be built. In this study, training was only carried out on joint-tag labels; however, the use of alternative model training strategies can also be applied to improve the overall performance of the model to be applied. In this particular study, nested entities are restricted to level-1 only. For further research, the nested entity level can be increased so that it can extract more complex entities.

REFERENCES

[1] S. H. Nasr, C. K. Dagli, M. M. Dakake, J. E. B. Lumbard, and M. Rustom, “The Study Quran,” A new Transl. Comment., vol. 19, 2015.
[2] R. Grishman and B. M. Sundheim, “Message understanding conference-6: A brief history,” 1996.
[3] D. Nadeau and S. Sekine, “A survey of named entity recognition and classification,” Lingvisticae Investig., vol. 30, no. 1, pp. 3–26, 2007.
[4] V. Yadav and S. Bethard, “A survey on recent advances in named entity recognition from deep learning models,” arXiv Prepr. arXiv1910.11470, 2019.
[5] J. Lafferty, A. McCallum, and F. C. N. Pereira, “Conditional random fields: Probabilistic models for segmenting and labeling sequence data,” 2001.
[6] Z. Huang, W. Xu, and K. Yu, “Bidirectional LSTM-CRF models for sequence tagging,” arXiv Prepr. arXiv1508.01991, 2015.
[7] B. Wilie et al., “IndoNLU: Benchmark and resources for evaluating Indonesian natural language understanding,” arXiv Prepr. arXiv2009.05387, 2020.
[8] A. Vaswani et al., “Attention is all you need,” in Advances in neural information processing systems, 2017, pp. 5998–6008.
[9] L. A. Ramshaw and M. P. Marcus, “Text chunking using transformation-based learning,” in Natural language processing using very large corpora, Springer, 1999, pp. 157–176.
[10] E. F. Sang and F. De Meulder, “Introduction to the CoNLL-2003 shared task: Language-independent named entity recognition,” arXiv Prepr. cs/0306050, 2003.
[11] P. Q. N. Minh, “A feature-based model for nested named-entity recognition at VLSP-2018 ner evaluation campaign,” arXiv Prepr. arXiv1803.08463, 2018.
[12] A. Akbik, T. Bergmann, D. Blythe, K. Rasul, S. Schweter, and R. Vollgraf, “FLAIR: An easy-to-use framework for state-of-the-art NLP,” in {NAACL} 2019, 2019 Annual Conference of the North American Chapter of the Association for Computational Linguistics (Demonstrations), 2019, pp. 54–59.
[13] T. Mikolov, I. Sutskever, K. Chen, G. S. Corrado, and J. Dean, “Distributed representations of words and phrases and their compositionality,” Adv. Neural Inf. Process. Syst., vol. 26, 2013.
[14] P. Bojanowski, E. Grave, A. Joulin, and T. Mikolov, “Enriching Word Vectors with Subword Information,” Trans. Assoc. Comput. Linguist., vol. 5, pp. 135–146, 2017.
[15] E. Grave, P. Bojanowski, P. Gupta, A. Joulin, and T. Mikolov, “Learning Word Vectors for 157 Languages,” 2018.
[16] J. Pennington, R. Socher, and C. D. Manning, “GloVe: Global Vectors for Word Representation,” in Empirical Methods in Natural Language Processing (EMNLP), 2014, pp. 1532–1543. [Online]. Available: http://www.aclweb.org/anthology/D14-1162
[17] H. Nakayama, “{seqeval}: A Python framework for sequence labeling evaluation.” 2018. [Online]. Available: https://github.com/chakki-works/seqeval