File Training Generator For Indonesian Language In Named Entity Recognition Using Anago Library

Irfan Fadil[1], Dwi Yuniarto[2], Esa Firmansyah[3], Dody Herdiana[4], Fidi Supriadi[5], Ali Rahman[6]
{fadilirfan@stmik-sumedang.ac.id[1], dwi@stmik-sumedang.ac.id[2], esa@stmik-sumedang.ac.id[3], dody@stmik-sumedang.ac.id[4], fsupriadi@stmik-sumedang.ac.id[5], ali@uinsgd.ac.id[6]}

STMIK Sumedang, UIN Sunan Gunung Djati Bandung

Abstract. Named Entity Recognition (NER) or Named Entity Recognition and Classification (NERC) is one of the main components of an information extraction task that aims to detect and categorize named entities in a text. NER is generally used to detect people's names, place names, and organization of a document, but can also be extended to identify genes, proteins, and others as needed. NER is useful in many NLP (Natural Language Processing) applications such as question-answering, summaries, and dialog systems because it can reduce ambiguity. NER also deals with other information extraction tasks such as relation detection, event detection, and temporal analysis. To avoid this need to train data source. The data train can be taken from various sources of news/articles crawled on the internet. The news will then be annotated by users with various labels. The news/article sources are in the thousands, while to make this training by using file is manual. And sometimes there is an error because this manual was made when it will form the NER model as needed. This research will be made so that training files can be assisted by using applications so that the error rate can be smaller or there will be no errors.

Keywords: Machine Learning, Named Entity Recognition, Anago Library, Data Train.

1 Introduction

Named Entity Recognition (NER) or Named Entity Recognition and Classification (NERC) is one of the main components of an information extraction task that aims to detect and categorize named entities in a text. NER is generally used to detect people's names, place names, and organizations of a document, but can also be extended to identify genes, proteins, and others as needed.

NER is useful in many NLP (Natural Language Processing) applications such as question-answering, summaries, and dialog systems because it can reduce ambiguity. NER also deals with other information extraction tasks such as relation detection, event detection, and temporal analysis.

Dictionary-based approaches cannot be used because new names always appear at any time. Another problem is ambiguity. For example the word "Istana" in Indonesian phrase like "... di depan istana" and "Pihak Istana Menyetujui .. " has a different class of entities.

Currently, the most common approach for the NER task is to use machine learning, especially classification. Machine learning is an approach that looks for patterns in pre-existing data to predict new data. The technique that currently has the best performance is a deep learning

ICONISTECH 2019, July 11-12, Bandung, Indonesia

DOI 10.4108/eai.11-7-2019.2297618

technique. To use machine learning, a training file is needed. This training file is created manually using the TSV format (Tab Separated Value) before it is entered into the machine learning technique. Because the making of training files is done manually. So the error level of the training file is very high. Besides that, it takes a long time to make one training file. Therefore a generator application is needed to produce the training file so that it can be immediately recognized by the machine learning technique used.

2 Literature Review

In this section will explain some of the theories that support this research and several related journals related to the same topic about Named Entity Recognition (NER).

2.1 Anago Library

Anago is a Python library for sequence labeling (NER, PoS Tagging, and many more) implemented in Keras who uses technique bi-LSTM + CRF. This technique is one of the highest performing techniques (F1 90.94 for the CoNNL 2003 dataset). Anago can solve sequence labeling tasks such as named entity recognition (NER), part-of-speech tagging (POS tagging), semantic role labeling (SRL) and so on. Unlike traditional sequence labeling solver, anaGo don't need to define any language dependent features. Thus, we can easily use anaGo for any languages[1].

Anago supports following features such as : Model Training, Model Evaluation, Tagging Text, Custom Model Support, Downloading pre-trained model, GPU Support, Character feature, CRF Support, Custom Callback Support. anaGo officially supports Python 3.4–3.6 and using library Tensorflow dan Keras.

Anago is a library based on journals that have been made by [2][3]. Input from Anago neural net is word embedding, which can be pretrained. Furthermore, Anago can generate form word embedding itself based on training data. Bi-LSTM uses two LSTM networks, one reads sequences from left to right and one from right to left. The output from the network is processed using the CRF (Conditional Random Field) technique. Anago's architecture can be described as follows:

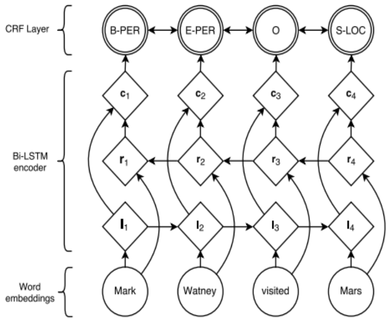


Fig. 1 Anago Architecture [2][3]

2.2 Annotation File Format

The annotation format used is BIO (Begin, In and Other). For example for the sentence in Indonesian language "Jusuf Kalla going to Bandung", then the BIO format annotation is (assuming the person's name tag label is the PER and LOC location):

Words	**Tag**
Jusuf	B-PER
Kalla	I-PER
pergi	O
ke	O
Bandung	B-LOC

Data is stored in text files with delimited tab format (words and tags separated by tabs). Between sentences separated by blank lines.

The Anago Library requires that the format of the tag is B-[whatever], I-[whatever] and O. So after B- and I- can be filled in anything tag, but the prefix B-, I- and the word O cannot be changed [4].

Entity tags that can be used in this modeling are as follows :

Table. 1 Entity Result Of NER

No	Entity	Definition	Example	Tag
1	PERSON	Names of people, including fictional characters	Jokowi, Anies Baswedan, etc	PER
2	NORP	Nationalities or religious or political groups.	PBNU, PDIP, Muhammadiyah, Governor, Minister,MPR, DPR etc	NOR
3	FACILITY	Buildings, airports, highways, bridges, etc.	Soetta Airport, Cipali Toll Road, Suramadu Bridge, etc	FAC
4	ORG	Companies, agencies, institutions, etc.	CNN, HSBC, Indonesian Employers Association, Apindo, etc	ORG
5	GPE	Countries, cities, states.	Jakarta, Indonesia, West Java, Java, Bali, Kalimantan etc	GPE
6	LOC	Non-GPE locations, mountain ranges, bodies of water.	Harmony Area, BI Thamrin Building, White House, State Palace, etc	LOC
7	PRODUCT	Objects, vehicles, foods, etc. (Not services.)	Recommendations, aircraft, Bitcoin, Survey Results, etc.	PRD
8	EVENT	Named hurricanes, battles, wars, sports events, etc.	Montana Just Fun Hiking 2017, Elections, Campaigns, etc.	EVT
9	WORK_OF_ART	Titles of books, songs, etc.	Indonesia Raya, Habis Gelap Terbitlah Terang, etc	WOA
10	LAW	Named documents made into laws.	STNK, BPKB, SHGB, SHM, APBN, KUHP etc	LAW
11	LANGUAGE	Any named language.	English, Indonesian Language, etc	LAN
12	DATE	Absolute or relative dates or periods.	Tuesday, November 21, 2017, 2018, January, etc.	DAT
13	TIME	Times smaller than a day.	11:30 PM, etc	TIM

14	PERCENT	Percentage, including "%".	5,4%, 0,3 percent, etc	PRC
15	MONEY	Monetary values, including unit.	Rp.424.92 billion, US dollars, etc.	MON
16	QUANTITY	Measurements, as of weight or distance.	6.113, 59,05, etc	QTY
17	ORDINAL	"first", "second", etc.	I, first, etc	ORD
18	CARDINAL	Numerals that do not fall under another type.	three in the sentence (“At least three people”),	CRD
19	RELIGION	Everything about religion	Allah SWT	REG

2.3 Related Work

In the topic regarding named entity recognition, several journals or references have been written by several authors including: In research journals [5] Produce journals on Designing Named Entity Recognition in Accounting for Transaction Identification based on Indonesian Text. This journal describes labeling design in accounting transactions to identify accounts. The labeling process of accounting transactions in this study is grouped into date label, company name label, quantity label, transaction type label, and money amount label. Labeling This accounting transaction is based on natural language processing in Indonesian language texts.

While in the Research Journal [6], one of the journals outside Indonesia produced a journal about Named Entity Recognition with Extremely Limited Data. In this journal, it is explained about indexing corpus annotations and used them in searches. Giving this entity is labeled by humans in the number of thousands or tens of thousands. In this search, an exploration of the introduction of named entities is proposed, where there are two components, namely entity classes, and relevant documents.

In research journals [7] Produce journals about the Indonesian Named Entity Recognition for 15 Classes Using Ensemble Supervised Learning. This journal describes the development of NERs for newspaper articles in Indonesia with 15 Classes. In addition, this research also conducted experiments to find the best combination of the best attributes and algorithms with high accuracy. This study compares word level attributes, sentence level, and document level. To achieve the best accuracy this study used 457 news articles using ensemble techniques in which the results of several machine learning algorithms were used as features for a machine learning algorithm.

In the journal [8] explains how to extract events in an article to obtain structured information 5W1H. This modeling is carried out with a corpus 610 paragraph text taken from 57 articles that were manually annotated.

3 Research Methods

In this study, an Entity Recognition Named Model Using Indonesian Language will be built. The steps that will be taken include: create tokenize, Formatting BIO With TSV, Clustering File Training, Generate Model see figure 2. The input used is the source of the article taken from several websites that have the same theme example about general news, sport, political and many more. Whereas for the output from some of these methods produced is a model of the Indonesian Named Entity Recognition.

The articles needed for this model require a very large amount of data. Because it will affect the accuracy of the model that will be builts. The accuracy is good for this model ranges from 70% to 90%. However, this study only focuses on two steps of the method, namely the create tokenize process and the formulation annotation of BIO With TSV. Both of these methods were initially done manually using the help of notepad ++ software. So that when the model formation process, it is often found some errors from training files due to annotation tag, punctuation, space, and word format errors. To deal with the problem, the solution is to create software to reduce the error.

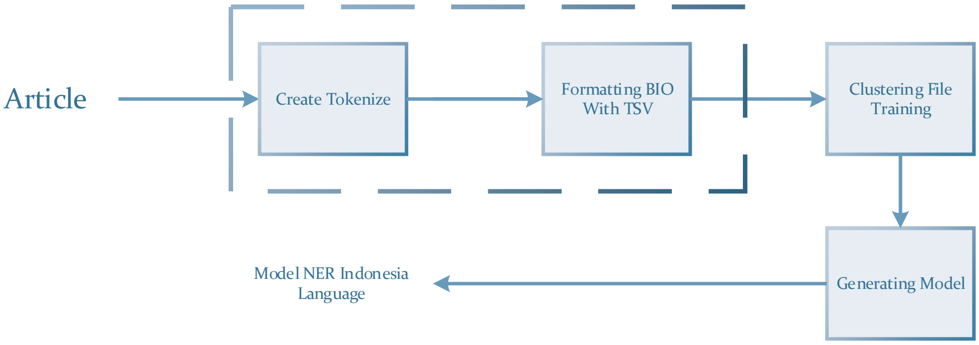


Fig. 2 Proposed Block Diagram

Articles taken from the website will create a tokenize process to perform word tokens. In this process, each sentence or punctuation in the article will be separated by a line. So that each line will have one word or one punctuation. Then each of these lines will be annotated according to the BIO format described in section 2.2. One article can consist of hundreds or thousands of lines and annotation tags. Articles that have been performed tokenize and formatting BIO are stored in one file with txt format. In other words, one article will produce one file with the txt format. If the training file from the article reaches hundreds, the next step is clustering training files.

In the Clustering File Training Method, the training file consists of 3 clusters, namely training data, valid data, and testing data. The article files with TSV format that has been obtained will be divided into 3 clusterings with the provision of filling 60% for training data, 20% for valid data, and 20% for test data. Each of these clusters must not have the same article. Suppose if there are 100,000 Articles, the training data will get 60,000 articles, valid data get 20,000 articles and data testing also get 20,000 articles. If the clustering process has been done, the next step is to perform the NER model using the anago library.

To perform generating a NER model, prepare a directory where data train, test data and valid data. Also prepare a directory to store the model that will be produced by this library. Then enter the following source code:

```
import anago
from anago.reader import load_data_and_labels
dir = "/home/ner/training/"
x_train, y_train = load_data_and_labels(dir+'train.txt')
x_valid, y_valid = load_data_and_labels(dir+'valid.txt')
```

```
x_test, y_test = load_data_and_labels(dir+'test.txt')
model = anago.Sequence()
model.train(x_train, y_train, x_valid, y_valid)
model.eval(x_test, y_test)
dirOutput = "/home/ner/model/"
model.save(dirOutput)
words = 'Presiden Jokowi Mengunjungi Bali dalam Rangka
Konferensi PBB.'.split()
result = model.analyze(words)
print("Result Of Ner = "+str(result))
```

Run the process of the source code, it will show the training process, the time required is approximately 30 minutes to 1 hour depending on the computer specification, configuration and the amount of training data. If the program is successfully executed then the model directory there will be a model containing 3 files. To load the model from the results of previous training so there is no need to training anymore in a long time can use the following source code:

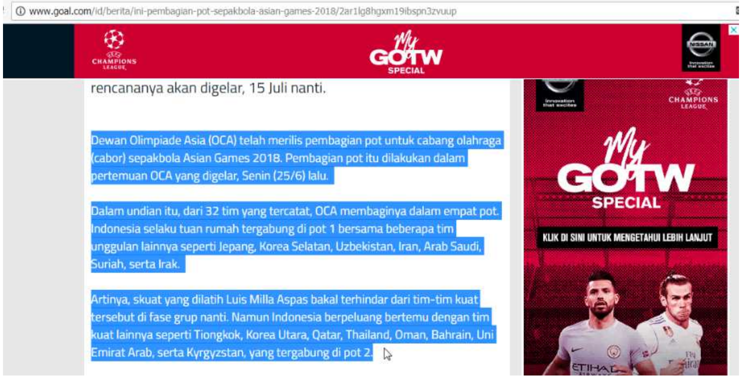
```
import anago
import pprint
dirOutput = "/home/ner/model"
model = anago.Sequence().load(dirOutput)
words = ' Presiden Jokowi Mengunjungi Bali dalam Rangka
Konferensi PBB'.split()
result = model.analyze(words)
pprint.pprint(result)
```

Repeat the steps in figure 2 until the accuracy of the model obtained in the training process reaches 70 -90%. To speed up the process of performing a new model. The old models that have been trained can be used to create new training files so that the process of tokenizing and formating BIO can be guessed directly by the model. The user only corrects if guesses annotation and the tag are incorrect.

4 Research Result

In this section will explain the results of this study which consists of creating tokenize using applications and formating BIO using applications with the help of a previously built NER model.

In Figure 3, the article on a website is used for the training process by copying it. The article copied then stored in the application as shown in Figure 4.



www.goal.com/id/berita/ini-pembagian-pot-sepakbola-asian-games-2018/2ar1lg8hgxm19ibspn3zvuup

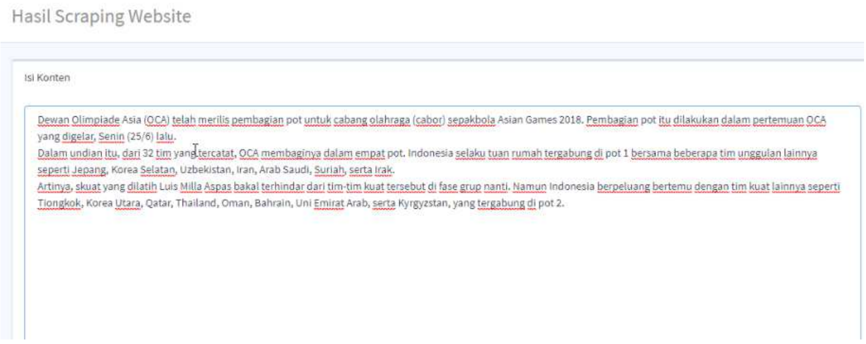
rencananya akan digelar, 15 Juli nanti.

Dewan Olimpiade Asia (OCA) telah merilis pembagian pot untuk cabang olahraga (cabor) sepakbola Asian Games 2018. Pembagian pot itu dilakukan dalam pertemuan OCA yang digelar, Senin (25/6) lalu.

Dalam undian itu, dari 32 tim yang tercatat, OCA membaginya dalam empat pot. Indonesia selaku tuan rumah tergabung di pot 1 bersama beberapa tim unggulan lainnya seperti Jepang, Korea Selatan, Uzbekistan, Iran, Arab Saudi, Suriah, serta Irak.

Artinya, skuat yang dilatih Luis Milla Aspas bakal terhindar dari tim-tim kuat tersebut di fase grup nanti. Namun Indonesia berpeluang bertemu dengan tim kuat lainnya seperti Tiongkok, Korea Utara, Qatar, Thailand, Oman, Bahrain, Uni Emirat Arab, serta Kyrgyzstan, yang tergabung di pot 2.

Fig. 3 Source Article Web



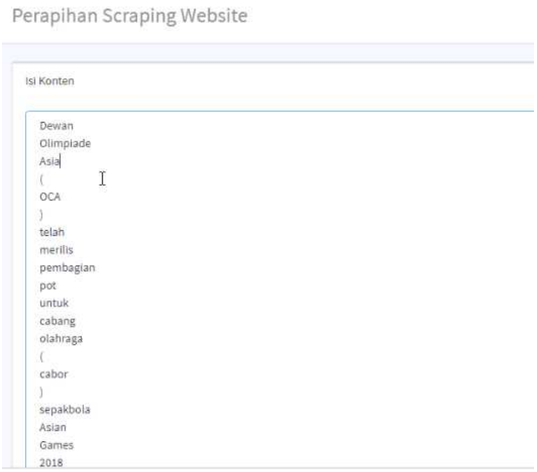
Hasil Scraping Website

Isi Konten

Dewan Olimpiade Asia (OCA) telah merilis pembagian pot untuk cabang olahraga (cabor) sepakbola Asian Games 2018. Pembagian pot itu dilakukan dalam pertemuan OCA yang digelar, Senin (25/6) lalu.
Dalam undian itu, dari 32 tim yang tercatat, OCA membaginya dalam empat pot. Indonesia selaku tuan rumah tergabung di pot 1 bersama beberapa tim unggulan lainnya seperti Jepang, Korea Selatan, Uzbekistan, Iran, Arab Saudi, Suriah, serta Irak.
Artinya, skuat yang dilatih Luis Milla Aspas bakal terhindar dari tim-tim kuat tersebut di fase grup nanti. Namun Indonesia berpeluang bertemu dengan tim kuat lainnya seperti Tiongkok, Korea Utara, Qatar, Thailand, Oman, Bahrain, Uni Emirat Arab, serta Kyrgyzstan, yang tergabung di pot 2.

Fig. 4 Input Article

After the article is entered in the application, the next step is to create tokens for the words in the article. The results of making word tokens can be seen in Figure 5.



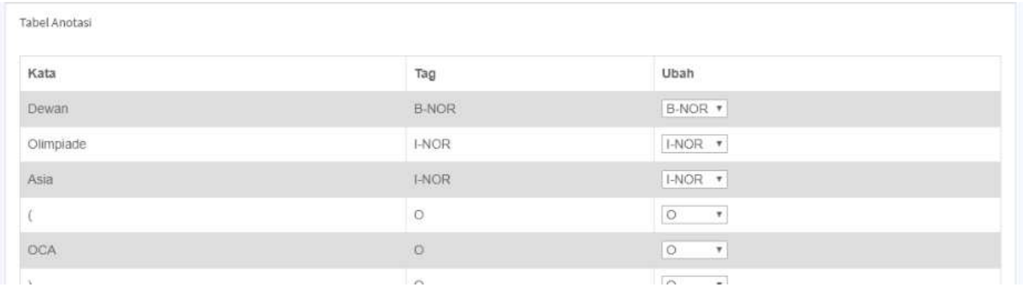
Perapihan Scraping Website

Isi Konten

Dewan
Olimpiade
Asia
(
OCA
)
telah
merilis
pembagian
pot
untuk
cabang
olahraga
(
cabor
)
sepakbola
Asian
Games
2018

Fig. 5 Input Article

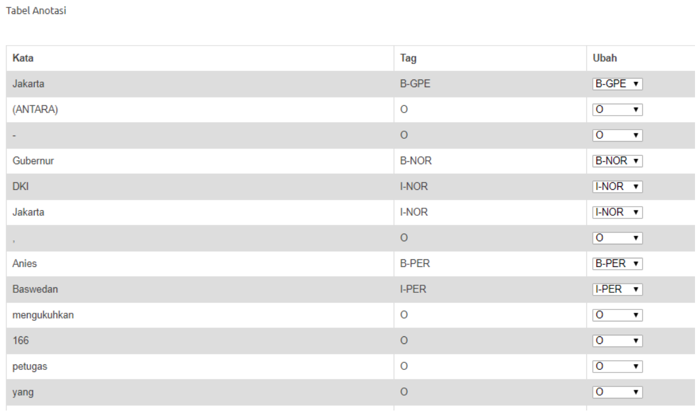
The token will be annotated with an annotation tag in the BIO format. The annotation tag on this token it can be done in two ways. The first way is to do an annotation tag from the beginning to the end of the token. In this way, the user is required to fill all the annotation tags that are in the article token, see in Figure 6. While the second way, the annotation tag is done by the NER model that was previously builts. By using this method, users only make corrections to some of the wrong annotation tags. So that it does not fill the full entire annotation tag that is on the token. See Figure 7



Tabel Anotasi

Kata	Tag	Ubah
Dewan	B-NOR	B-NOR ▾
Olimpiade	I-NOR	I-NOR ▾
Asia	I-NOR	I-NOR ▾
(	O	O ▾
OCA	O	O ▾
)	O	O ▾

Fig. 6 Annotation TAG Stand Alone



Tabel Anotasi

Kata	Tag	Ubah
Jakarta	B-GPE	B-GPE ▾
(ANTARA)	O	O ▾
-	O	O ▾
Gubernur	B-NOR	B-NOR ▾
DKI	I-NOR	I-NOR ▾
Jakarta	I-NOR	I-NOR ▾
,	O	O ▾
Anies	B-PER	B-PER ▾
Baswedan	I-PER	I-PER ▾
mengukuhkan	O	O ▾
166	O	O ▾
petugas	O	O ▾
yang	O	O ▾

Fig. 7 Annotation Tag Using Model NER

5 Conclusion

The results of this study produce a scheme that can correct errors in manually making training files needed by anago library. Users can easily and quickly create training files from articles taken. Besides that, the results of this study have formed an Indonesian language NER model. So that new articles that will be used as training files can be recognized by the model. The user then verifies whether the results of the training files formed by the model are appropriate or need to be corrected in some results. The user only corrects the training file errors predicted by the model. The existence of this scheme is expected to accelerate the need for accuracy of the NER model for Indonesian than to more than 90%.

References

[1] Nakayama H 2019 No Title

[2] Lample G, Ballesteros M, Subramanian S, Kawakami K and Dyer C 2016 Neural architectures for named entity recognition *2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, NAACL HLT 2016 - Proceedings of the Conference*

[3] Peters M, Neumann M, Iyyer M, Gardner M, Clark C, Lee K and Zettlemoyer L 2018 Deep Contextualized Word Representations

[4] Wibisono Y 2018 NER (Named Entity Recognition) Bahasa Indonesia dengan Stanford NER

[5] Iswandi; I and Supriana; I 2015 Perancangan Named Entity Recognition dalam Akuntansi untuk Identifikasi Transaksi berdasarkan Teks Indonesia Perancangan Named Entity Recognition dalam Akuntansi *Semin. Nas. Pengemb. Aktual Teknol. Inf.*

[6] Foley J, Sarwar S M and Allan J 2018 Named Entity Recognition with Extremely Limited Data 2–7

[7] Wibawa A S and Purwarianti A 2016 Indonesian Named-entity Recognition for 15 Classes Using Ensemble Supervised Learning *Procedia Computer Science* vol 81 (Elsevier B.V.) pp 221–8

[8] Anon Pengenalan Entitas Bernama Otomatis untuk Bahasa Indonesia dengan Pendekatan Pembelajaran Mesin | Request PDF