TELKOMNIKA, Vol.14, No.4, December 2016, pp. 1462~1471
ISSN: 1693-6930, accredited A by DIKTI, Decree No: 58/DIKTI/Kep/2013
DOI: 10.12928/TELKOMNIKA.v14i4.3876

Supervised Entity Tagger for Indonesian Labor Strike Tweets using Oversampling Technique and Low Resource Features

Ayu Purwarianti*[1], Lisa Madlberger[2], Muhammad Ibrahim[3]
[1]School of Electrical Engineering and Informatics, Bandung Institute of Technology,
Jl. Ganesa No. 10, Bandung, Indonesia, telp/fax: +62-22-2508135/+62-22-2500940
[2]Institute of Software Technology and Interactive Systems, Vienna University of Technology,
Favoritenstrasse 9-11/188, A-1040 Vienna, Austria, telp: +43-1-58801-188642
[3]Faculty of Computer Science, University of Indonesia,
Kampus UI Depok 16424 Indonesia, telp: +62-21-7863419, fax: +62-21-7863415
*Corresponding author, e-mail: ayu@stei.itb.ac.id[1], lisa.madlberger@tuwien.ac.at[2],
mochamad.ibrahim@ui.ac.id[3]

Abstract

We propose an entity tagger for Indonesian tweets sent during labor strike events using supervised learning methods. The aim of the tagger is to extract the date, location and the person/organization involved in the strike. We use SMOTE (Synthetic Minority Oversampling Technique) as an oversampling technique and conducted several experiments using Twitter data to evaluate different settings with varying machine learning algorithms and training data sizes. In order to test the low resource features, we also conducted experiments for the system without employing the word list feature and the word normalization. Our results indicated that different treatment of different types of machine learning algorithms with low resource features can lead to a good accuracy score. Here, we tried Naïve Bayes, C4.5, Random Forest and SMO (Sequential Minimal Optimization) algorithms using Weka as the machine learning tools. For the Naïve Bayes, due to the data distribution based of the class probability, the best accuracy was achieved by removing data duplication. For C4.5 and Random Forest, SMOTE gave higher accuracy result compared to the original data and the data with data duplication removal. For SMO, there is no significant difference among various sizes of training data.

***Keywords**: Indonesian Entity Tagger, SMOTE, supervised learning, word level feature, word window feature, labor strike tweets*

Copyright © 2016 Universitas Ahmad Dahlan. All rights reserved.

1. Introduction

A strike can be defined as a planned action of employees or workers unions which is performed collectively to stop or to slow down work. Labor strikes have severe consequences for all involved parties, foremost corporations, employees and customers. Supply chain disruptions, blocked transportation routes, delay of delivery, loss of productivity and reputational damage are just some of the consequences companies and customers face as result of labor strikes. Damages could be reduced through timely and efficient responses, however valueable time is often lost as parties are informed too late about a strike at their supplier or transportation partners. The problem is that there is a lack of structured information on labor strike events that is provided in a timely manner. At the same time, more and more people use social media to report what is happening around them in real-time. We want to use Twitter data in order to extract structured event information on labor strikes.

In this paper we use strike-related Tweets posted by local users such as citizens, activists, local news media or labor unions in order to extract the date, the location and organizations involved in strike events. We focus our experiments on Indonesia and Indonesian Twitter data, as it counts as an important supplier country of raw materials and manufactured goods in international supply chains and at the same time has a high number of Social Media Users.

The goal of Entity Recognition is to identify and classify entities in a given text, which is an information extraction task. Different to Named Entity Recognition (NER), an entity in our

Received April 24, 2016; Revised October 25, 2016; Accepted November 13, 2016

research is not only a *named* entity, but can be also a common entity,having certain roles in our domain of labor strike, like for example ``taxi drivers''. Applying entity tagging to user-generated texts originating in social networks imposes additional difficulty compared to formal texts. User-generated texts typically involve informal words, abbreviations and affixes, as well as the use of informal word order and grammar. Furthermore, Indonesian social media data particularly exhibits the mixed use of languages including the official language Bahasa Indonesia, English and several Indonesian regional languages.

For Indonesian social media, there are several researchers on NER ([1-5]). Existing approaches can generally be divided into rule-based ([1, 2]) and statistical approaches ([3-5]). In rule based systems, researchers define rules in from of string patterns used to identify and classify named entities. In statistical systems, the named entity extraction rules are learned automatically based on previously labeled data by machine learning algorithms. Since the rules are not easily defined manually by a human, recent researches tend to apply and enhance the statistical methods for named entity recognition. In line with these argumentations, we chose to employ the statistical approach in our study.

Existing studies applied Indonesian NER for different applications scenarios in multiple specific domains including e-commerce transactions ([3]), citizen complaints ([4]) and traffic conditions ([1-2], [5]); and generic domains ([6, 7]). We developed the first entity tagger for tweets sent during civic strike and protest events in Indonesia.

Statistical entity tagger requires the definition and extraction of features from the original text. Features for entity tagger can be divided into word level features[8], word window features, word list features [8] and document features[8]. Word level features are characteristics of a particular word, e.g. the length of a word. Word window features are characteristics relating to a defined number of previous or succeeding words, e.g. whether the previous word has been identified as an entity. Word list features indicate whether a word occurs in a predefined list of entities e.g. geographic gazetteers. Document features relate to other documents, in our case other tweets, e.g. the occurrence number of a word.

Most NER systems developed for Indonesian texts use word level features, word window features and word list features. Khodra & Purwarianti [3] employed word level and word window features. They reported an accuracy of 81.49% by including two preceding words in the word window. The best algorithm employed was IBk, compared to Naïve Bayes and C4.5. Anggareska & Purwarianti [4] employed word level features, word window features and word list features. They reported the best accuracy of 85.6% achieved by applying the SMO algorithm (compared to Naïve Bayes and IBk). The features applied included word window features, the current word with its orthographical information as the word level features and several word lists of location gazetteer, cluelist and stopword as the word list features. Here, in our research, similar with Anggareska & Purwarianti [4], we will use the word window features, the word level features and word list features. But, in the experiment, we will show that without using the word list features, the system can still achieve a good accuracy by using oversampling technique on the training data. Another difference with Anggareska & Purwarianti [4] is that we do not use cluelist as the word list feature since it is not easily built for new entity class such as ours in labor strike information.

2. Indonesian Entity Tagger on Strike Information for Twitter Text using Supervised Learning

In this research, our goal is to tag important entities automatically for strike information from Indonesian tweets. For the strike information, there are several alternatives of the important entity type such as people who do the strike, the strike target (which can be people or organization), the location of the strike, and the date or time of the strike. In our research, we decided to have three types of entities: 1) people-organization (involved in the strike); 2) location of the strike; 3) date or time of the strike event. The examples of tweet and their important entities are shown in Table 1.

The complete process applied in our entity tagger is depicted in Figure 1. The entity tagger consists of three parts, namely preprocessing, feature extraction and classification. Each part has an important role in order to achieve a high accuracy score of entity tagger.

Table 1. Example of Tweet and the Important Entity on Strike Information

Tweet Text	People/Org	Location	Date
Mahasiswa mulai menggulirkan rencana aksi **10 September** melalui #IndonesiaDarurat, ayo kita dukung. @ypaonganan {English: **Students** start rolling **September 10th** strike plan through #IndonesiaDarurat, let's support @ypaonganan}	Mahasiswa (English: Students)		10 September
Angkot di **bogor** pada mogok kerja,jalanan tuh serasa milik sendiri bebas dari macet :D {English: **Bus driver** in **bogor** doing strike, feels like having our own road without traffic jam :D}	Angkot (English: Bus driver)	Bogor	

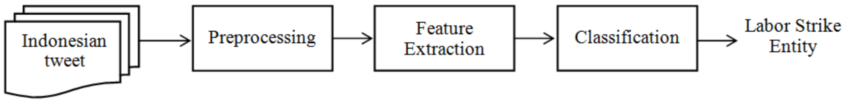

Figure 1. Flow of Labor Strike Entity Tagger System

2.1. Preprocessing

The preprocessing module first split up the tweet text into tokens (tokenization) and consequently transforms informal words into formal words (normalization). Another step which could be applied here in the future would be a Part-of-Speech Tagger. As up to now, no available POS-Tagger for Indonesian Social Media data can be used in our system, the available one is POS-Tagger for Indonesian common sentences such as in articles [9]. Thus, we only employed the tokenization and word normalization. The word normalization and POS-tagging are language-dependent modules which would have to be replaced or removed when the system is applied to another language. In our experiments we compare the system's accuracy in settings including and excluding the word normalization module.

2.2. Feature Extraction

The feature extraction module extracts information from single tokens (word level features) and sequences of tokens (word window features). Particularly, word level features include the normalized form of a token lexical and the token orthographical information. The word window features take the token sequence, include the preceding and succeeding token lexical along with the entity class of preceding token. Another additional feature is a word list feature. Here, we employ an easily gathered word list, which is the gazetteer of locations provided by Geonames (http://www.geonames.org/). Another word list that we used is a stop word list. As we mentioned earlier, we compare the usage of these word lists in the experiments. Table 2 shows the features and its examples for word "bogor" of second tweet in Table 1.

2.3. Classification

In the classification, the features taken from the tweet is classified using a classification model into seven classes that cover three information mentioned before. The seven classes are as follows: 1) Token is the beginning of a location (LOC-B); 2) Token is part of a location (LOC-I); 3) Token marks the beginning of people or organization involved in the strike (PEORG-B); 4) Token defines a word as part of a people or organization identifier (PEORG-I); 5) Token marks the beginning of a date (DATE-B); 6) Token defines a word as part of a date (DATE-I); 7) the token is not a named entity of interest (OTHER).

Table 2. Features on the Entity Tagger on Strike Information and its Examples

Feature Name	Description	Example
Lexical (n)	Lexical of the current token	bogor
Lexical (n-1)	Lexical of one preceding token	di (eng: in)
Lexical (n+1)	Lexical of one succeeding token	pada (eng: doing)
NE class (n-1)	NE tag of one preceding token	Other
Orthography (n)	Orthography information of the current token	normal alphabet
TokenKind (n)	Type of current token	word
IsMention (n)	True if the current token is a mention	False
IsLink (n)	True if the current token is a link	False
IsTime (n)	True if the current token format is a date or time	False
IsGazetteer (n)	True if the current token is a member of Gazetteer	True
IsStopWord (n)	True if the current token is a member of stop word list	False

2.3.1. Using SMOTE to Handle Imbalanced Dataset

Similar with NER, in each input, the number token labeled with defined entity (LOC, PEORG, DATE) is smaller than the number of non entity label. This condition of imbalanced dataset may lead to low accuracy of labor strike entity tagger. Since only few tokens carry information about the location, organization or date; naturally most of the tokens will fall into the last class “OTHER”.

To handle the inbalanced data, we employed SMOTE (Synthetic Minority Oversampling Technique) [10] as the oversampling technique. SMOTE oversampling the minority by adding new data predicted from several nearest neighbor data of each minority class. For the basic step, we first removed the duplicate data and resampled the dataset several times by applying SMOTE.

2.3.2. Machine Learning Algorithm

We compared several machine learning algorithms to build the classification model including Naive Bayes, C4.5, SMO and Random Forest. We select these algorithms since these algorithms are widely used in text mining studies.

Naïve Bayes makes use Bayesian theorem to calculate a class probability of a given feature input as the representation of a token with assumption that the features are independent [11]. C4.5 (or known as J48 in Weka [11] – this term is used later in the experiment) use divide-and-conquer algorithm on training data to form a decision tree that represent classification rules [12]. SMO makes use sequential minimal optimization algorithm for training a support vector classifier [13]. Random forest employs voting over several tree constructed from random training data [14].

3. Experiments

We have conducted several experiments applying the proposed strike entity tagger of Indonesian tweets to compare different machine learning algorithms, different features, and training data sizes. To evaluate the performance on each class, we used F-Measure score calculation such as below.

$$F - measure = 2 \frac{Precision\,.\, Recall}{Precision + Recall} \tag{1}$$

Where the precision and recall scores are calculated as follows:

$$Precision = \frac{\#correctly\ classified\ token\ for\ certain\ entity}{\#token\ classified\ as\ certain\ entity} \tag{2}$$

$$Recall = \frac{\#correctly\ classified\ token\ for\ certain\ entity}{\#reference\ token\ for\ certain\ entity\ (manually\ tagged)} \tag{3}$$

As for the overall evaluation, we employed accuracy score with equation such as below. For example, if there are 100 correctly classified tokens among 1000 tokens in the testing data, then the accuracy is 100/1000 = 10%.

$$Accuracy = \frac{\#correcly\ classified\ token}{\#token\ in\ testing\ data} \quad (4)$$

3.1. Experimental Data and Baseline Experiment of Indonesian Strike Entity Tagger

We used the DMI-Twitter Capture and Analysis Toolkit to collect Tweets from the public Twitter Streaming API that matched a list of defined strike-related keywords: “aksi demo”, “aksi dukung”, “aksi kerja”, “aksi mogok”, “mogok kerja”. We collected Tweets for a period of two months (01.05.2015- 30.06.2015). For our experiments, we retrieved a random sample of this dataset consisting of 18,999 token (from 1,046 tweets). 10 Indonesians were asked to annotate words in the Twitter Data manually by using a mobile annotation tool [15]. The final annotation label is chosen by the majority label. We selected 20% as the testing data and 80% as the training data. The data size for each class is shown in Table below.

Table 3. Data Size for Each Entity Class

NE Class	Data Size (token)	Training	Testing
OTHER	14046	11195	2799
LOC-B	525	420	105
LOC-I	584	467	117
PEORG-B	1740	1391	348
PEORG-O	1513	1211	303
DATE-B	255	204	51
DATE-I	336	268	68
Total	18999	15156	3791

For the baseline, we conducted an experiment using features of current word lexical surface and previous entity label. We compared several algorithms in the baseline experiments of Naive Bayes (NB), J48, Random Forest (RF) and SMO using Weka [11] for the entity tagger. The experimental result is shown below. Here, SMO algorithm achieved the best accuracy result of 94.54%.

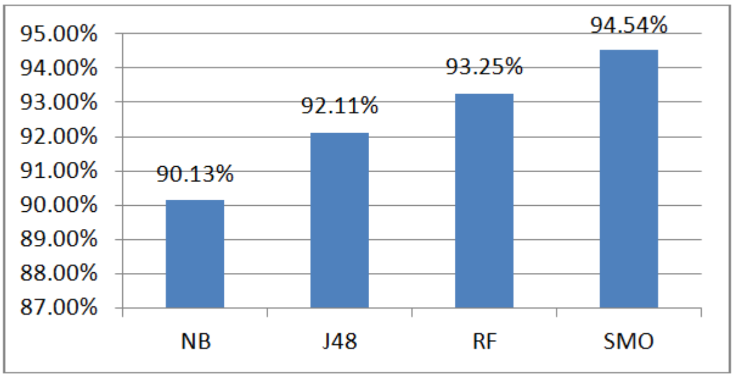

Figure 2. Accuracy of Baseline Experiment on Indonesian Strike Entity Tagger

The accuracy of each algorithm above couldn’t show the exact performance for each entity. Thus, table below shows the F-Measure score for each entity type of the baseline experiment. Here, even though the accuracy of J48 and Random Forest outperformed Naïve Bayes, but for “I-LOC” entity type, the best f-measure was achieved by Naïve Bayes algorithm. On the other hand, Naïve Bayes algorithm couldn’t able to extract the B-DATE since the out-of-vocabularies couldn’t be handled by only using the current lexical entry and the previous entity label.

.

Table 4. F-Measure for Each Strike Entity Class for Baseline Experiment

	NB	J48	RF	SMO
OTHER	0.94	0.961	0.963	0.967
LOC-B	0.391	0.729	0.734	0.773
LOC-I	0.835	0.657	0.711	0.877
PEORG-B	0.797	0.88	0.9	0.914
PEORG-I	0.856	0.798	0.86	0.916
DATE-B	0	0.623	0.68	0.66
DATE-I	0.868	0.85	0.899	0.919

3.2. Experiment on Using Low Resource Feature for Indonesian Strike Entity Tagger

In this experiment, we propose the usage of several easily prepared features which aim is to handle the out of vocabulary problem. The complete features are shown in Table 2. The experimental result for each algorithm is shown in figure below. For algorithms of Naïve Bayes, Random Forest and SMO, the accuracy result is higher, but for J48 algorithm, the accuracy is lower. The higher number of feature for the J48 doesn't give higher accuracy result. Using complete feature for J48 algorithm yielded the root node of the tree as previous entity label, while using only two features yielded the root node of the tree as current lexical surface. There is a rule in the complete feature J48 algorithm saying that if the previous entity label is OTHER then the class result is OTHER. This rule is supported by 10,420 correct data and 1,771 incorrect data. This rule caused the low accuracy of J48 algorithm.

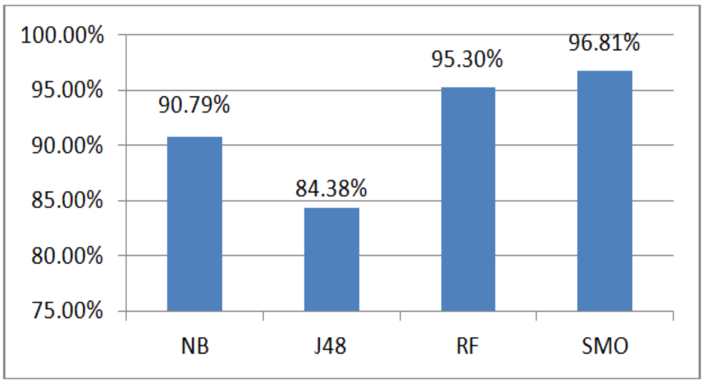

Figure 3. Accuracy of Complete Features Experiment on Indonesian Strike Entity Tagger for Various Machine Learning Algorithms with Original Data Collection (15156 Training Data Size)

Table 5. F-Measure for Each Strike Entity Class for Original Data Collection

	NB	J48	RF	SMO
OTHER	0.944	0.905	0.974	**0.981**
LOC-B	0.734	0.188	0.888	**0.893**
LOC-I	0.829	0.915	0.713	**0.935**
PEORG-B	0.815	0.006	**0.936**	0.934
PEORG-I	0.848	0.893	0.908	**0.95**
DATE-B	0.435	0.109	0.699	**0.782**
DATE-I	0.919	0.788	0.939	**0.97**

In the F-Measure score for each class such as shown in Table 5, it is shown that using only 2 features such as in the baseline gave better result than using complete features for the J48 algorithm. The low results are shown for all "B" label such as LOC-B, PEORG-B and DATE-

B. Most of "B" label were classified as OTHER. Other finding is that -similar with the baseline conclusion- the SMO algorithm outperformed other machine learning algorithms.

3.3. Experiment on Using SMOTE for Indonesian Strike Entity Tagger

The dataset size shown in Table 3 indicates that the dataset at hand is highly unbalanced, which means that the number of entities in the training data vary between classes. This is a challenge because low size of one class can be insufficient to represent the training data, thus it will be harder for the learning algorithm to learn the characteristics of this group. To overcome this problem, we conducted experiments on using SMOTE to add the dataset. Here, we tried 2 strategies. The difference is that in the second strategy, we employed the data duplication removal before applying SMOTE. The experimental result for each strategy is explained below.

3.3.1. Only Using SMOTE to Enhance Training Dataset

We applied SMOTE several times to the original dataset and yielded several new training dataset as shown in Table 6. There are 4 types of SMOTE: (1) double the data of each class (except OTHER) by SMOTE; (2) multiply the data of each class (except OTHER) 4 times by SMOTE; (3) conduct SMOTE until the data in each class reach about 1:2 compared to the OTHER class; (4) conduct SMOTE until the data in each class has about the same size as the OTHER class.

Table 6. Various Training Dataset Size for Experiments on Indonesian Strike Entity Tagger using SMOTE only (1st Strategy)

	Original Data	SMOTE (x2) (1)	SMOTE (x4) (2)	SMOTE (1:2) (3)	SMOTE (1:1) (4)
OTHER	11195	11195	11195	11195	11195
LOC-B	420	840	1680	5040	10080
LOC-I	467	934	1868	5604	11208
PEORG-B	1391	2782	5564	5564	11128
PEORG-I	1211	2422	4844	4844	9688
DATE-B	204	408	816	5304	10608
DATE-I	268	536	1072	5360	10720
Total	15156	19117	27039	42911	74627

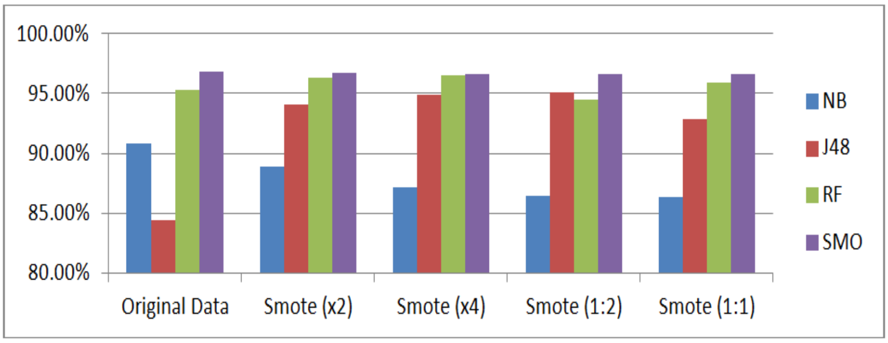

Figure 4. Accuracy Scores for Various Dataset Size of Indonesian Strike Entity Tagger using SMOTE only (1st Strategy)

The experimental result shown in Figure 4 indicates the difference effects of SMOTE for each algorithm. Applying SMOTE for Naïve Bayes unfortunately lower the accuracy since Naïve Bayes algorithm is highly depend on the data distribution. This is different with the decision tree algorithm such as J48 or Random Forest (RF). In the decision tree algorithm, SMOTE technique increases the accuracy since the information gain of each tree branch depends on the data distribution of classes. The significant improvement is in J48 algorithm where there is only one tree calculation. In both algorithms, there is a point of data size where it will reach maximum accuracy and after that the training data will overfit and the accuracy will be lower. As for SMO algorithm, SMOTE technique doesn't give any effect since SMO algorithm doesn't depend on data distribution which means that 10% of rare case has the same result with 2% of it. Here, we also show the highest F-Measure score for each algorithm in Table 7. Almost similar with the conclusion for Table 5, the highest F-Measure score was achieved by SMO except for LOC-B which is lower than the Random Forest algorithm.

Table 7. F-Measure Score for each Algorithm using SMOTE (1st Strategy)

	Naïve Bayes	J48	Random Forest	SMO
OTHER	0.944	0.97	0.98	**0.981**
LOC-B	0.734	0.828	**0.924**	0.893
LOC-I	0.829	0.918	0.874	**0.935**
PEORG-B	0.815	0.917	**0.934**	**0.934**
PEORG-I	0.848	0.926	0.948	**0.95**
DATE-B	0.435	0.72	0.69	**0.782**
DATE-I	0.919	0.886	0.933	**0.97**
Schema	Original	SMOTE (1:2)	SMOTE (4x)	Original

3.3.2. Using Data Duplication Removal before SMOTE to Enhance Training Dataset

In the second strategy, before applying SMOTE, we conducted data duplication removal first. The training datasets are shown in Table 8. There are 6 treatments: (1) only using the data duplication removal; (2) applying SMOTE for Loc and Date class after the duplication removal; (3) applying SMOTE for all classes except OTHER class after the duplication removal; (4) applying SMOTE until each class data reaches 1:4 compared to OTHER class, after the duplication removal; (5) applying SMOTE until each class data reaches 1:2 compared to OTHER class, after the duplication removal; (6) and arrange the data in each class to be almost the same size as the OTHER class, after the duplication removal.

Table 8. Various Training Dataset Size for Experiments on Indonesian Strike Entity Tagger using SMOTE and Data Duplication Removal (2nd Strategy)

	Original Data	Data Duplication Removal (1)	Removal + SMOTE (2x for Loc & Date) (2)	Removal + SMOTE (x2) (3)	Removal + SMOTE (1:4) (4)	Removal + SMOTE (1:2) (5)	Removal + SMOTE (1:1) (6)
OTHER	11195	6442	6442	6442	6442	6442	6442
LOC-B	420	277	554	554	1662	3324	6315
LOC-I	467	290	580	580	1740	3393	6446
PEORG-B	1391	913	913	1826	1826	3469	6591
PEORG-I	1211	779	779	1558	1558	3116	6232
DATE-B	204	118	236	236	1652	3304	6277
DATE-I	268	177	354	354	1770	3451	6556
Total	15156	8996	9858	11550	16650	26499	44859

Figure 5 shows the accuracy score for each algorithm and each SMOTE implementation for our second strategy. The conclusion is similar with the previous section where SMOTE gave effect mostly on the decision tree algorithm, while it lowered the accuracy of Naïve Bayes algorithm and had no effect on the SMO algorithm. Figure 5 also shows that since Naïve Bayes algorithm is a probabilistic algorithm which depend a lot on the data distribution of each class, the data duplication removal gives the best training data for Naïve Bayes where it reached highest accuracy among all scenarios on the Naïve Bayes algorithm.

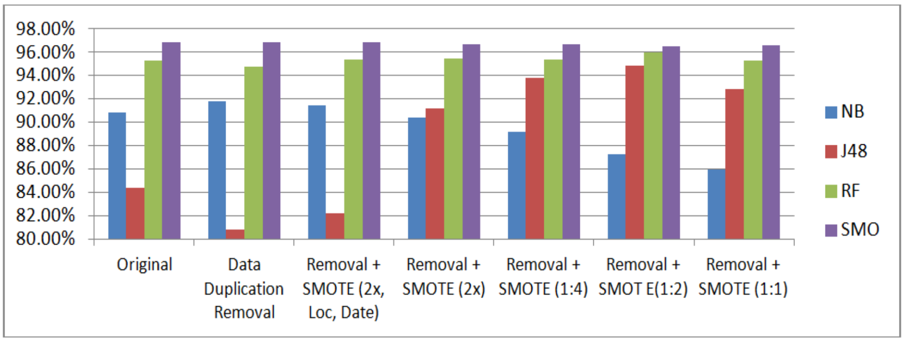

Figure 5. Accuracy Scores for Various Dataset Size of Indonesian Strike Entity Tagger using SMOTE and Data Duplication Removal (2nd Strategy)

Figure 6 shows that the best effect of applying SMOTE is on J48 algorithm where it was able to enhance the accuracy score from 84% to more than 94%. To be more precise, below is the detail of F-Measure score on each training data for J48 algorithm. Here, SMOTE was able to enhance the F-Measure score significantly for B-LOC, B-ENT, B-DATE.

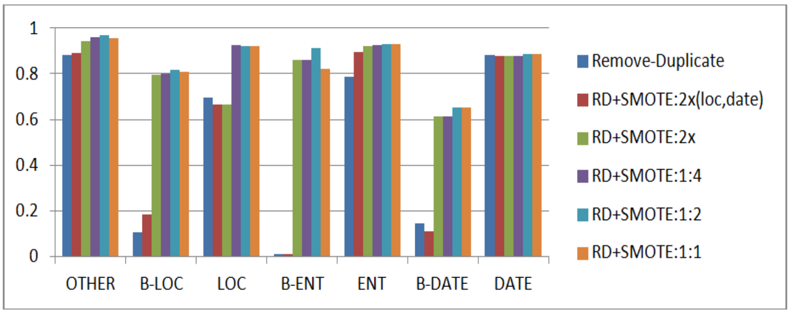

Figure 6. F-Measure Scores for Various Dataset Size of Indonesian Strike Entity Tagger using J48 Algorithm with SMOTE and Data Duplication Removal (2nd Strategy)

3.4. Indonesian Entity Tagger without Word List Feature and without Word Normalization

Our last experiment is to evaluate the usage of only easily prepared features which include word level feature and word window feature. For this purpose we eliminated the word list features and gain only 9 features by removing the features isGazetteer and isStopWord, since these features are manually built. Other than that, we also eliminate the word normalization preprocessing to see the effect on the entity tagger. The experimental result for the original data (15156 tokens) is shown in Figure 7.

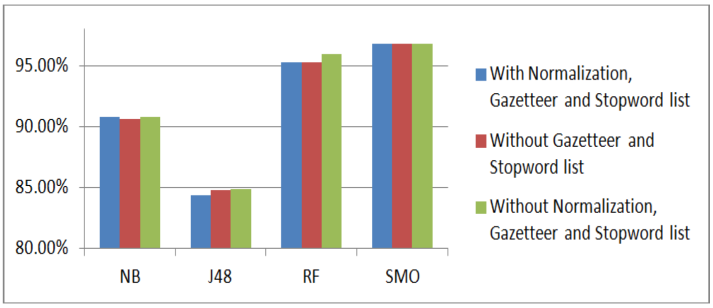

Figure 7. Accuracy of Entity Tagger without Word Normalization and Word List Features

The result shown in Figure 7 indicates that features depend on a word list gave a slight different accuracy score compared to the complete one. It means that in the future, our social media entity tagger can be easily employed for other language in strike domain.

4. Conclusion

We propose an Indonesian entity tagger to extract entities from tweets sent during strike events using supervised learning. To handle the inbalanced distribution of classes, we employed SMOTE as an oversampling technique. Our experiments on several machine learning algorithms showed that SMOTE has different effects for different algorithms. SMOTE gives significant good effect on decision tree based algorithms such as J48 and random forest, but it lowers the accuracy of Naïve Bayes algorithm and has not effect on SMO algorithm. Using only word level features and word window features, the system can still achieve similar accuracy as when using all features since the word normalization and word list employed here do not give significant effect on the accuracy. In future research we want to apply the system also in other languages.

Acknowledgements

Research reported in this publication was jointly supported by the ASEAN-European Academic University Network (ASEA-UNINET), the Austrian Federal Ministry of Science, Research and Economy and the Austrian Agency for International Cooperation in Education and Research (OeAD-GmbH).

References

[1] Endarnoto SK, Pradipta S, Nugroho AS, Purnama J. *Traffic Condition Information Extraction & Visualization from Social Media Twitter for Android Mobile Application*. International Conference on Electrical Engineering and Informatics. Bandung, Indonesia. 2011.
[2] Hanifah R, Supangkat SH, Purwarianti A. *Twitter Information Extraction for Smart City*. International Conference on ICT for Smart Society (ICISS). Bandung, Indonesia. 2014.
[3] Khodra ML, Purwarianti A. Ekstraksi Informasi Transaksi Online pada Twitter. *Cybermatika*. 2013; 1(1).
[4] Anggareska D, Purwarianti A. *Information Extraction of Public Complaints on Twitter Text for Bandung Government*. International Conference on Data and Software Engineering. Bandung, Indonesia. 2014.
[5] Hasby M, Khodra ML. *Optimal Path Finding based on Traffic Information Extraction from Twitter*. International Conference on ICT for Smart Society (ICISS). Jakarta, Indonesia. 2013.
[6] Liu X, Zhang S, Wei F, Zhou M. *Recognizing Named Entities in Tweets*. Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics. Portland, Oregon. 2011: 359-367.
[7] Ritter A, Clark S, Mausam, Etzioni O. *Named Entity Recognition in Tweets: an Experimental Study*. Proceedings of the 2011 Conference on Empirical Methods in Natural Language Processing. Edinburgh, Scotland, UK. 2011: 1524-1534.
[8] Nadeau D. Semi-Supervised Named Entity Recognition: Learning to Recognize 100 Entity Types with Little Supervision. Disertation. Ottawa, Canada: Ottawa-Carleton Institute for Computer Science, School of Information Technology and Engineering, University of Ottawa; 2007.
[9] Wicaksono A, Purwarianti A. *HMM based Part-of-Speech Tagger for Bahasa Indonesia*. Fourth International MALINDO Workshop. Jakarta, Indonesia. 2010.
[10] Chawla NV, Bowyer KW, Hall LO, Kegelmeyer WP. SMOTE: Synthetic Minority Over-sampling Technique. *Journal of Artificial Intelligence Research*. 2002; 16(1): 321-357.
[11] Witten IH, Frank E. Data Mining: Practical Machine Learning Tools and Technique. 2nd edition. Morgan Kauffman. 2005.
[12] Quinlann JR. C4.5: Programs for machine learning. San Francisco: Morgan Kauffman. 1993.
[13] Platt J. Fast training of support vector machines using sequential minimal optimization. In: B Scholkopf, C Burges, A Smola. *Editors.* Advances in kernel methods: Support vector learning. Cambridge, MA: MIT Press. 1998.
[14] Breiman L. Random Forests. *Machine Learning.* 2001; 45(1): 5-32.
[15] Madlberger L, Romadhony A, Ibrahim M, Purwarianti A. *Gotong Royong in NLP research – a mobile tool for collaborative text annotation in Indonesia.* The 20th International Conference on Asian Language Processing (IALP) 2016. Tainan, Taiwan. 2016.