International Journal of Mathematics and Computation,
ISSN 0974-570X (Online), ISSN 0974-5718 (Print),
Vol. 30; Issue No. 2; Year 2019,

Comparative Study of Combination of Preprocessing, N-Gram Feature Extraction, Feature Selection, and Classification Method in Indonesian Sentiment Analysis with Imbalanced Data

Rezkya Putri Septiani[1] and **Margaretha Ari Anggorowati**[2]

[1] Computational Statistics, Institute of Statistics (STIS), Jakarta, Indonesia;
Email : rezkya.putri@bps.go.id

[2]Institute of Statistics (STIS) Jakarta, Indonesia;
Email : m.ari@stis.ac.id

ABSTRACT

Social media makes a shift in lifestyles of people. People tend to use microblogging such as Twitter to criticize the controversial issues. The most controversial Indonesian economics policy in recent year is tax amnesty. Predicting positive and negative sentiments on tax amnesty policy could be developed by supervised machine learning. The performance of classification results can be improved by using the right combination of preprocessing technique, feature extraction, and feature selection. We aim to compare the performance and find the best combination of preprocessing technique, N-Gram feature extraction, feature selection, and classification method by conducting experiments. Data collection was developed by crawling using Twitter API. Imbalanced data is one of challenge in machine learning which can produce unsatisfactory classifiers and normalize the Indonesian slang can also be more challenging. This research uses an imbalanced dataset to know the performance of the combination algorithms handling the imbalanced data which measured by nested cross-validation. The experimental results show that the best combination of algorithms in this research performs well in handling imbalanced data and the performance of models can be improved and really depend on the combination of preprocessing, N-Gram feature extraction, feature selection, and classification method.

Keywords: Text mining, sentiment analysis, preprocessing, feature selection, classification.
Mathematics Subject Classification: 68T05, 62H30, 68Q25
Computing Classification System: I.2.6, I.5.1

1. INTRODUCTION

Technology affects a shift in the lifestyle of many people, social media has become a necessity that can not be separated from everyday life. People tend to use microblogging such as Twitter to criticise the controversial issues and hot topics which the sentiment can be predicted using supervised machine learning algorithm. Preprocessing transforms the retrieved Twitter data into structured textual data and comparing the different approach of preprocessing techniques is also required to determine the best combination of preprocessing techniques which can optimize the performance. Preprocessing techniques to be compared related to word conversion into normal form (normalization) and stemming process. Preprocessing also has a challenge i.e. The existing Indonesian stemming algorithm can only eliminate the prefixes and suffixes and not for infixes. Besides that, Indonesian

slang called ‘alay’ used in Indonesian tweet also gives bad effect and ambiguity in the result of preprocessing when using the stemming method.

Furthermore, We can extract the features from preprocessing result using N-Gram. There are various types of N-gram, such as unigram, bigram, and trigram. The use of N-Gram feature extraction in text mining may achieve better result although there’s a probability that simple unigram still outperforms others in the experiment (Basile et al., 2017). We may retrieve large tweet dataset that consists of many features. We must choose informative features using feature selection which can also reduce the feature dimension and training time. In addition, large and overlapping features can degrade the performance of models (Padmanaban, Baker, & Greger, 2018). Features reduction is the way to achieve better performance in text classification (Aliwy & Ameer, 2017). Most widely used feature selection are Document Frequency Thresholding(DFT), Information Gain (IG) and Chi-square(CHI).

Naïve Bayes is widely used supervised machine learning in sentiment analysis because the algorithm is simple and effective which has low computational complexity (Abellán & Castellano, 2017). Naive Bayes method uses Bayes theorem and naïve independence assumption (Ogada & Mwangi, 2015). Naïve Bayes only performs well if the features are independent (Purohit, Atre, Jaswani, & Asawara, 2015). On the contrary, Maximum Entropy is good for features that are not independent and for large features. Maximum Entropy can be used if we do not know about the distribution of data. Maximum Entropy is feature-based that calculate the weight of the features (Gupte, Joshi, Gadgul, & Kadam, 2014) and can handle large data and features (Wang & Manning, 2012). One of the newest ensemble methods by (Breiman, 2001) is Random Forests which consists of many decision trees to improve accuracy in predicting classes and more robust in handling noise. Random Forests is an ensemble algorithm that developed from CART that using Bagging (Bootstrap Aggregating) (Breiman, 1996) and random feature selection which the result of classification decided by voting of all decision trees (Ren, Cheng, & Han, 2017). Random Forests is a robust algorithm that can improve the prediction and not too affected by outliers (Rubén & Chuvieco, 2017).

One of the controversial government policy in Indonesia is tax amnesty. We aim to predict the sentiment of Indonesian tax amnesty policy which divided into positive and negative. Moreover, we conducted experiments to compare the performance of the combination of preprocessing techniques, N-Gram feature extractions, feature selections, and classification methods. We also choose the best performance of combination among generated models.

2. GENERATION OF THE DATA

The data are obtained from Twitter crawling process. This research uses the Twitter API (Application Programming Interface) for the crawling process and R programming language. Data collected from Twitter that related to "Tax Amnesty". The amount of tweets data which finally selected for analysis are 1100 that divided into 888 positive tweets and 212 negative tweets.

2.1. Preprocessing

This research outline using the preprocessing steps i.e. case folding, tokenizing, filtering, stemming. In detail the preprocessing steps used in this study as follows:

1. Case folding dan tokenizing

At this step, it will remove the delimiter in tweets and tokenize the word. The processes undertaken at this step are deleting mentions, links, hashtags, retweets, punctuation, numbers, and changing the letters into lower case.

2. Filtering

In the filtering step, there are two processes, i.e. normalization and removing the stop words. There are many synonyms of a word, Twitter users often use different important terms that have the same meaning. The important terms related to tax amnesty topic, such as the same meaning of "*amnesti*", "*pengampunan*" and "*amnesty*". So, we should use normalization technique to convert different important terms that have the same meaning. The using normalization or not will give different result in preprocessing. This research will test two conditions, i.e. using normalization technique or not. removing stop words in tweets uses Indonesian stop words in (Tala, 2003). Finally, important to remove the unneeded whitespace from the tweets due to the process of uniformity and removal of stop words.

3. Stemming

The Results of previous processes are tokens that will be used for stemming. Stemming will transform the words into standardized grammatical form (Rajeswari, Juliet, & Aradhana, 2017). We use Indonesian stemmer (Adriani, Asian, Nazief, & Williams, 2007) algorithm which remove prefixes, suffixes, and combined prefixes-suffixes.

2.2. Methods

In figure 1, it can be seen that this research uses twitter data which are obtained from crawling data using keywords that related to "Tax Amnesty" sentiment. The twitter data should use preprocessing techniques to remove delimiters and all uninformative information from the data (Htet & Myint, 2018). Preprocessing techniques that used are case folding, filtering, and stemming. In the stemming step, this research uses two conditions, i.e the condition when using stemming step or not. We also use N-gram in the form of unigram, combined unigram and bigram, and combined unigram, bigram, and trigram. In addition, we will also compare the preprocessing step using word conversion (normalization) or not. The various combinations are used for observing the effect of preprocessing technique to improve the performance. Document-Term Matrix (DTM) is one of text transformation that often used in text mining (Kobayashi, Mol, Berkers, Kismihok, & Hartog, 2018). The result of the DTM matrix will correspond to the combination of N-Gram, normalization, and stemming that used.

The preprocessing data will be divided into training and testing data, the performance of each model is measured by 5X5 Folds Nested Cross-validation. Cross-validation used for decreasing the sensitivity of machine learning to training (Raczko & Zagajewski, 2017). The dimension of features matrix will be large and sparse. Fitting using a very sparse matrix will decrease the performance of text classification (Ong, Goh, & Xu, 2015). Reducing matrix dimension is an important step to obtain more accurate classification results, the process is known as feature selection. If the features more often appear than others, it indicates that the terms more important (Azam & Yao, 2012). In addition, if the features have a very small number of frequencies, it can be eliminated (Tripathi, 2015). Comparison of several feature selection methods is also important to decide the best one. The feature selection methods used are Document Frequency Threshold, Information Gain, and Chi-Square. We choose the features that have IG value more than zero (Roobaert, Karakoulas, & Chawla, 2006). In Chi-Square approach, If the feature and class are independent, the feature will be selected for text classification and Chi-squared considered as a good performing algorithm in handling feature selection (Tang, Kay, & He, 2016). We prefer to select features that have non zero and high value of Chi-Squared (Bai & Manimegalai, 2017). Therefore, the combination of preprocessing, N-Gram feature extraction, feature selection and classification method will be used for selecting the best combination and how the combination affects the performance.

Training data will be modeled using Maximum Entropy, Random Forests, and Naïve Bayes with the combination of preprocessing technique, N-gram feature extraction, and feature selection. Thus, this research will obtain 108 combination model. The performance of all models will be compared using 5X5 Folds Nested Cross-validation which the accuracy, F1-Scores, precision, recall, AUC, ROC curve, and execution time will be considered. Combination of all methods will obtain the best model of sentiment analysis in this research.

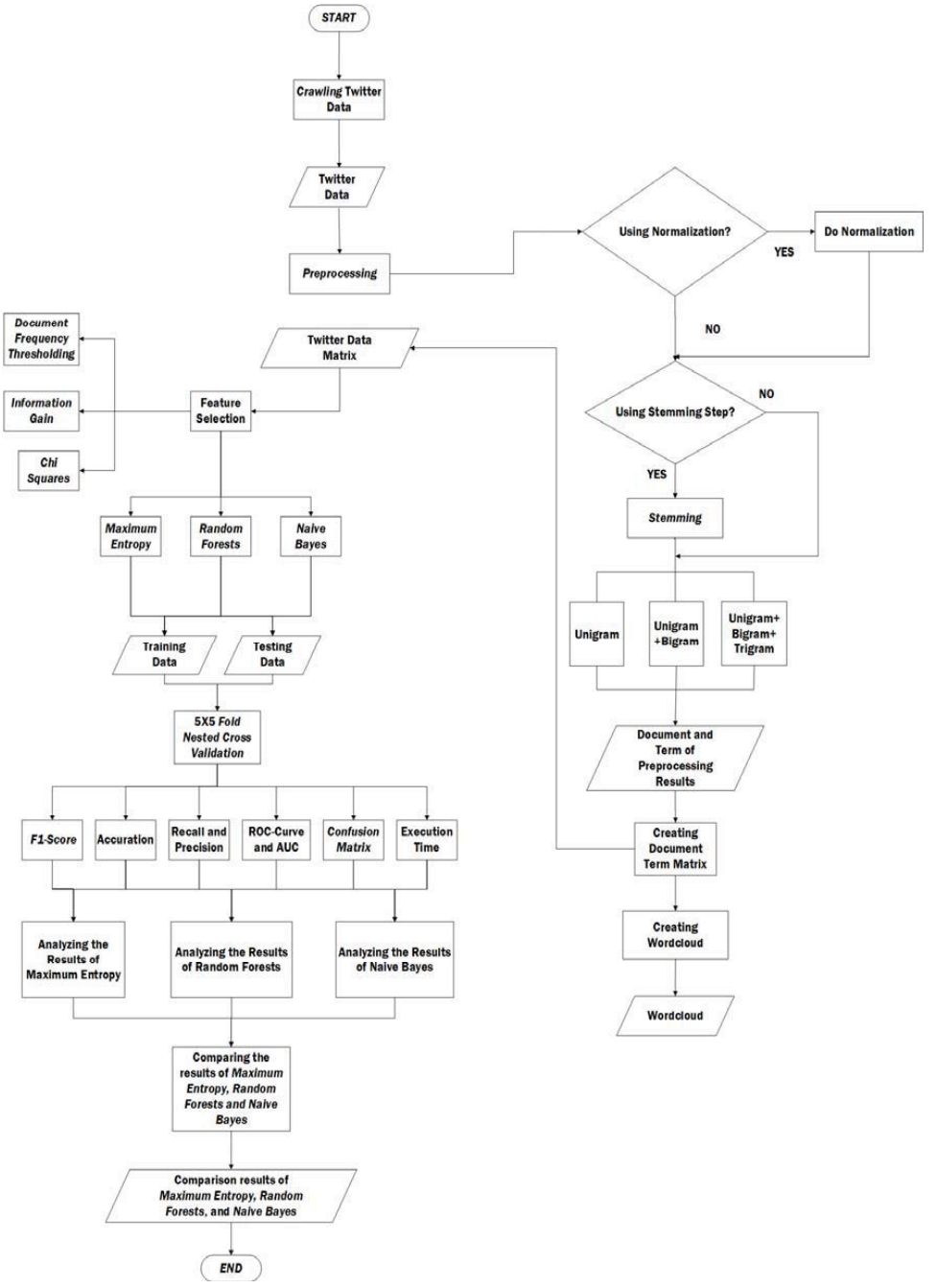

Figure 1. Methodology

3. RESULTS

3.1. Performance of Classification Methods using Normalization

In this section, it analyzes the performance of classification methods with normalization technique based on F1-Score and AUC. The scores in the table below show the successful events when comparing the classification methods (preprocessing, N-Gram and feature selection are constant). Based on Table 1, we know that Random Forests outperforms Maximum Entropy and Naïve Bayes. Random Forests has 12 out of 18 successful events, it means Random Forests has highest F1-Score than Maximum Entropy and Naïve Bayes in 12 times. The main difference between Maximum Entropy and Random Forests is that Random Forests more performs well when using IG and CHI instead of DFT. It shows the capability of ME performs well although it uses simple feature selection which reduces the sparsity of document-term matrix using a cut-off point. Furthermore, Naïve Bayes just has zero successful event which indicates Naïve Bayes never has the highest F1-Scores in comparison. Generally, Random Forests takes much execution time than others (79.86 seconds until 1434.5 seconds) as shown in Table 2 because of the complexity of its algorithm. But, Random Forests yielded an extremely good result in handling imbalanced dataset and robust. In the other hand, Naïve Bayes has short execution time but it has poor performance with lowest AUC values among others.

Table 1: Comparison the performance of classification methods based on F1-Scores using normalization

Classification Method	FS	S+N+U	S+N+B	S+N+T	NS+N+U	NS+N+B	NS+N+T	Total
ME	DFT	1	1	0	1	1	1	5
	IG	0	0	0	0	1	0	1
	CHI	0	0	0	0	0	0	0
	Total							6
RF	DFT	0	0	1	0	0	0	1
	IG	1	1	1	1	0	1	5
	CHI	1	1	1	1	1	1	6
	Total							12
NB	DFT	0	0	0	0	0	0	0
	IG	0	0	0	0	0	0	0
	CHI	0	0	0	0	0	0	0
	Total							0

FS= Feature Selection
RF= Random Forests
DFT= *Document Frequency Threshold*
CHI= *Chi-Square*
NS= Without *Stemming*
NN= Without Normalization
B=Combined Unigram+Bigram
ME= Maximum Entropy
NB= Naïve Bayes
IG= *Information Gain*
S= With *Stemming*
N=Normalization
U= Unigram
T= Combined Unigram+Bigram+Trigram

Table 2: AUC and execution time using normalization

Classification Method	FS	Performance	S+N+U	S+N+B	S+N+T	NS+N+U	NS+N+B	NS+N+T
ME	DFT	AUC	0.65	0.606	0.6148	0.637	0.6102	0.614
		Time	59.74	86.57	104.16	50.998	71.703	85.486
	IG	AUC	0.62	0.573	0.5645	0.592	0.6425	0.5615
		Time	28.84	72.56	184,76	29.37	83.62	202.04
	CHI	AUC	0.62	0.596	0.606	0.601	0.6265	0.628
		Time	57.14	89.42	115.19	54.996	84.979	114.29
RF	DFT	AUC	0.57	0.567	0.567	0.5563	0.5675	0.5594
		Time	116.34	173.6	214.8	98.565	155.67	180.57
	IG	AUC	0.62	0.607	0.608	0.5896	0.6	0.5889
		Time	79.86	184.58	360.68	84.614	223.61	388.14
	CHI	AUC	0.61	0.609	0.595	0.5918	0.5889	0.5872
		Time	603.54	998.82	1431.7	524.06	986.26	1434.5
NB	DFT	AUC	0.548	0.5423	0.538	0.5452	0.5445	0.5461
		Time	4.833	4.57	4.853	4.114	0.897	4.461
	IG	AUC	0.5540	0.5212	0.5141	0.5303	0.5189	0.5117
		Time	18.515	71.07	193.91	20.991	84.24	211.22
	CHI	AUC	0.635	0.6044	0.562	0.6015	0.5886	0.5612
		Time	50.875	98.467	144.63	52.355	96.714	143.6

3.2. Performance of Classification Methods Without using Normalization

Based on Table 3, Maximum Entropy outperforms Random Forests and Naïve Bayes when we don't use the normalization process which contrary to the preceding result above when we use normalization. It shows the superiority of Maximum Entropy which is not too vulnerable to randomness or unnormalized of features. If the distribution of data unknown and the matrix is sparse enough which has numerous features, it is suitable to consider Maximum Entropy as the primary option which more flexible in assumptions and yielded better results. Moreover, Maximum Entropy performs well although the features are dependent. Actually, Random Forests depends on the dimension of the matrix and word conversion. If the word conversion just enlarges the matrix which obtains more features, it will make training process takes so long due to the complexity of ensemble algorithm. As the preceding result when using normalization, Naïve Bayes still performs worst in all experiments.

In Table 4, As we know that Random Forest algorithm in the training process performs feature selection that makes the training time longer than Maximum Entropy and Naïve Bayes. However, Random Forests outperforms Naïve Bayes that has shorter execution time. Imbalanced data is one of the problems in machine learning that affects the accuracy of classifiers. AUC is one of important measure in imbalanced data. Naïve Bayes has AUC value below 0.5 that means it tends to predict the sentiments on one class. Moreover, Maximum Entropy and Random Forests can more handle the imbalanced data than Naïve Bayes because they have higher AUC value than Naïve Bayes.

Table 3: Comparison the performance of classification methods based on F1-Scores without using normalization

Methods	FS	S+NN +U	S+NN +B	S+NN +T	NS+NN +U	NS+NN +B	NS+NN +T	Total
ME	DFT	1	1	1	1	1	1	6
	IG	1	1	1	1	1	1	6
	CHI	0	0	0	0	0	0	0
	Total							12
RF	DFT	0	0	0	0	0	0	
	IG	0	0	0	0	0	0	
	CHI	1	1	1	1	1	1	6
	Total							6
NB	DFT	0	0	0	0	0	0	0
	IG	0	0	0	0	0	0	0
	CHI	0	0	0	0	0	0	0
	Total							0

Table 4: AUC and execution time without using normalization

Methods	FS	Performance	S+NN+ U	S+NN+ B	S+NN+ T	NS+NN+ U	NS+NN +B	NS+NN +T
ME	DFT	AUC	0.6613	0.6866	0.694	0.6537	0.6538	0.6568
		Time	51.004	71.763	86.003	49.196	68.178	78.181
	IG	AUC	0.7271	0.7449	0.7403	0.6875	0.7222	0.7218
		Time	30.96	77.894	189.70	30.434	89.797	222.94
	CHI	AUC	0.6485	0.6543	0.6510	0.6125	0.6350	0.6520
		Time	48.85	74.275	91.559	48.013	72.467	93.479
RF	DFT	AUC	0.593	0.6023	0.6035	0.5717	0.5871	0.5901
		Time	115.69	142.55	158.24	91.446	124.17	144.06
	IG	AUC	0.6607	0.6341	0.6287	0.6076	0.5934	0.6003
		Time	84.289	182.87	341.71	74.566	175.05	358.05
	CHI	AUC	0.6242	0.6251	0.6288	0.5909	0.5903	0.6063
		Time	468.56	921.98	1275.4	443.51	930.57	1153.3
NB	DFT	AUC	0.5	0.4889	0.4936	0.4968	0.4953	0.4947
		Time	3.941	3.911	4.586	5.316	3.935	3.814
	IG	AUC	0.5455	0.5260	0.5212	0.5308	0.5234	0.5164
		Time	17.156	71.689	190.34	21.350	82.201	221.11
	CHI	AUC	0.6386	0.6012	0.6288	0.5926	0.5965	0.5770
		Time	48.303	91.771	1275.4	46.558	86.387	122.06

3.3. Performance of Stemming in Preprocessing Techniques

Based on the experimental result in Table 5, stemming technique yielded better performance which has 41 out of 54 successful events. Stemming can increase the frequency of individual feature and decrease the sparsity of the matrix. Therefore, the use of stemming must be considered in increasing the accuracy of predictions on sentiment analysis.

Table 5: Successful events in using stemming technique

Stemming	Normalization			Without Normalization			Total
	ME	RF	NB	ME	RF	NB	
(1)	(2)	(3)	(4)	(5)	(6)	(7)	(8)
YES	3	8	6	8	9	7	41
NO	6	1	3	1	0	2	13

3.4. Performance of using N-Gram

Combined (unigram and bigram) outperforms others and yielded good performance although the difference of successful events of unigram and combined (unigram and bigram) is not too much as shown in Table 6. On the contrary, combined (unigram, bigram, and trigram) only has 7 out of 36 successful events, it because the combined (unigram, bigram, and trigram) produces more features than other types of N-Gram which may lead to overfitting. We can see that RF has 4 out of 7 successful events when using combined (unigram, bigram, and trigram) which means it's the highest score that can handle overfitting.

Table 6: Successful events in using N-Gram

N-Gram	**Normalization**			**Without Normalization**			**Total**
	ME	**RF**	**NB**	**ME**	**RF**	**NB**	
U	2	4	2	1	2	3	14
U+B	3	2	4	3	0	3	15
U+B+T	1	0	0	2	4	0	7

3.5. Performance of Normalization Techniques

In general, using normalization (word conversion) yielded the better result but it depends on the classification method that used. Based on Table 7, when using Maximum Entropy and Random Forests, the normalization technique doesn't give the better result. On the contrary, normalization technique performs better when using Naïve Bayes and Document Frequency Thresholding which has more than 0.5 AUC value that slightly better than without using normalization technique.

Table 7: Combination result of preprocessing, N-Gram feature extraction, feature selection, and classification method

Methods	FS		Using Normalization					
			Stemming			Without Stemming		
			U	U+B	U+B+T	U	U+B	U+B+T
ME	DFT	F1	0.9032	0.8923	0.8879	0.8947	0.8951	0.8911
		AUC	0.6522	0.6056	0.6148	0.6365	0.6102	0.6140
	IG	F1	0.7759	0.9040	0.6766	0.8052	0.9128	0.6721
		AUC	0.6210	0.5734	0.5645	0.5917	0.6425	0.5615
	CHI	F1	0.8987	0.8866	0.8948	0.8895	0.8903	0.8953
		AUC	0.6231	0.5960	0.6061	0.6010	0.6265	0.6275
RF	DFT	F1	0.8947	0.8917	0.8917	0.8887	0.8922	0.8878
		AUC	0.5679	0.5670	0.5670	0.5563	0.5675	0.5594
	IG	F1	0.9108	0.9087	0.9081	0.9031	0.9079	0.9061
		AUC	0.6237	0.6065	0.6076	0.5896	0.6000	0.5888
	CHI	F1	0.9090	0.9056	0.9009	0.9035	0.9025	0.9008
		AUC	0.6123	0.6090	0.5950	0.5918	0.5889	0.5872
NB	DFT	F1	0.8761	0.8721	0.8718	0.8732	0.8750	0.8782
		AUC	0.5480	0.5423	0.5381	0.5452	0.5445	0.5461
	IG	F1	0.9015	0.8974	0.8961	0.8969	0.8970	0.8956
		AUC	0.5540	0.5212	0.5141	0.5303	0.5189	0.5117
	CHI	F1	0.8788	0.9046	0.8986	0.8857	0.9001	0.8975
		AUC	0.6352	0.6044	0.5621	0.6015	0.5886	0.5612
Methods	**FS**		**Without Using Normalization**					
			Stemming			**Without Stemming**		
			U	**U+B**	**U+B+T**	**U**	**U+B**	**U+B+T**
ME	DFT	F1	0.8989	0.9055	0.9013	0.8987	0.8867	0.8941
		AUC	0.6613	0.6866	0.694	0.6537	0.6538	0.6568
	IG	F1	0.9164	0.9248	0.9238	0.9142	0.9228	0.9222
		AUC	0.7271	0.7449	0.7403	0.6875	0.7222	0.7218
	CHI	F1	0.8811	0.8974	0.9020	0.8875	0.8910	0.8969
		AUC	0.6485	0.6543	0.6510	0.6125	0.6350	0.6520
RF	DFT	F1	0.8912	0.8907	0.8918	0.8270	0.8829	0.8839
		AUC	0.5932	0.6023	0.6035	0.5717	0.5871	0.5901
	IG	F1	0.9119	0.9097	0.9099	0.9020	0.9010	0.9061
		AUC	0.6607	0.6341	0.6287	0.6076	0.5934	0.6003
	CHI	F1	0.9114	0.9083	0.9063	0.9006	0.9000	0.9027
		AUC	0.6242	0.6251	0.6288	0.5909	0.5903	0.6063
NB	DFT	F1	0.8934	0.8803	0.8812	0.8864	0.8829	0.8823
		AUC	0.5	0.4889	0.4936	0.4968	0.4953	0.4947
	IG	F1	0.9008	0.8983	0.8974	0.8975	0.8980	0.8965
		AUC	0.5455	0.5260	0.5212	0.5308	0.5234	0.5164
	CHI	F1	0.8844	0.9054	0.9035	0.8844	0.9045	0.9020
		AUC	0.6386	0.6012	0.5823	0.5926	0.5965	0.5770

: without using normalization outperforms normalization based on F1-Scores

: without using normalization outperforms normalization based on AUC

: normalization outperforms without using normalization when using Naïve Bayes and Document Frequency Threshold

3.6. Performance of Feature Selection

Information Gain has the highest successful events which perform better than Document Frequency Threshold and Chi-Square as shown in Table 8. DFT often used in text mining because it's easy to implement which has a simple algorithm. Looking at the successful events in Table 8, it can be seen that DFT has only 2 out of 36 successful events in experiments. When we compare the performance of each machine learning towards feature selection used as shown in Figure 2, 3, and 4, it can be seen that each machine learning performs better when it combined with certain feature selection. The result shows that Maximum Entropy outperforms Naïve Bayes and Random Forests when using DFT. Maximum Entropy is more flexible than Naïve Bayes and Random Forests in select the best threshold for the feature selection in DFT. Because of the independence assumption in Naïve Bayes, selecting the best threshold in DFT is the difficult part and highly considered because the slight change of cut-off point can extremely decrease the performance that's why the Naïve Bayes has the lowest F1-Scores when using DFT. But, Naïve Bayes can improve the performance better using Chi-Square.
In Figure 3, The F1-Scores of Random Forests more stable than Maximum Entropy in all experiments which has the higher value of F1-Score than Naïve Bayes. Furthermore, Random Forests outperforms Maximum Entropy and Naïve Bayes when using Chi-Square feature selection shown in Figure 4.

Table 8: Successful events in using feature selection

FS	Normalization			Without using Normalization			Total
	ME	RF	NB	ME	RF	NB	
DFT	2	0	0	0	0	0	2
IG	2	5	2	6	6	2	23
CHI	2	1	4	0	0	4	11

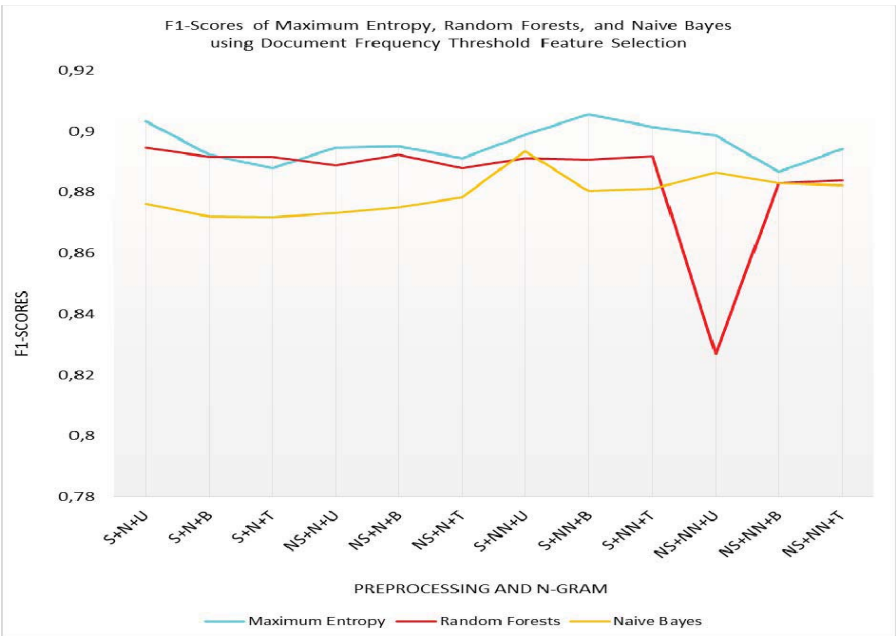

Figure 2. F1-Scores of ME, RF, NB using DFT feature selection

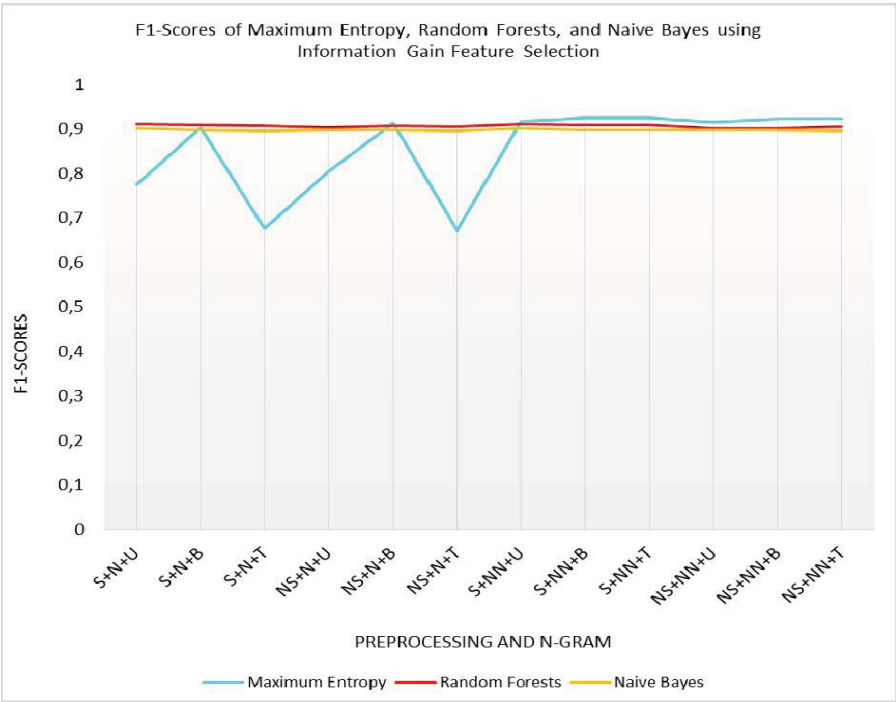

Figure 3. F1-Scores of ME, RF, NB using IG feature selection

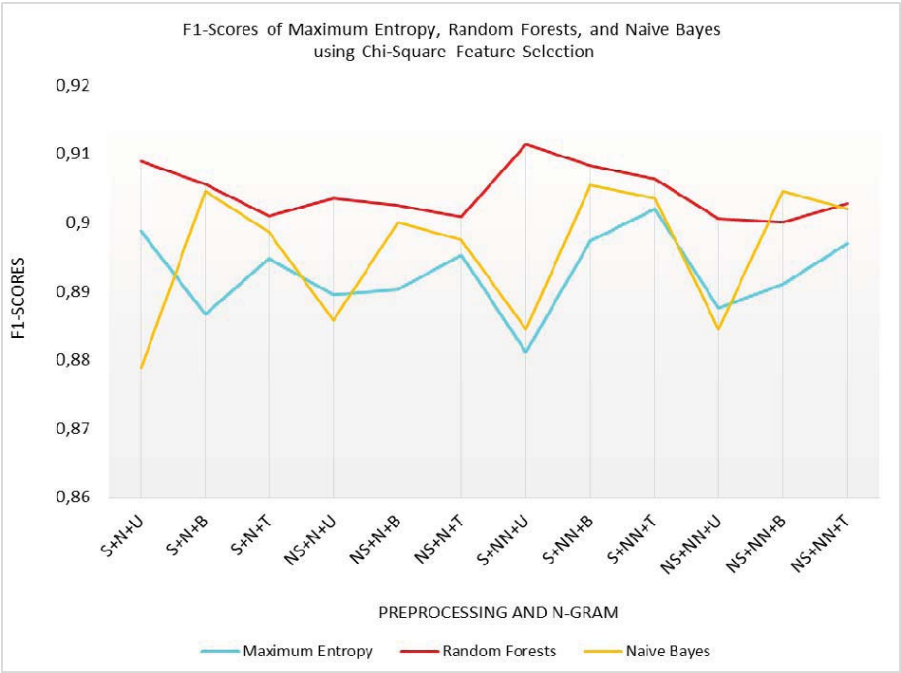

Figure 4. F1-Scores of ME, RF, NB with CHI feature selection

3.7. Best Combination of Preprocessing, N-Gram Feature Extraction, Feature Selection, and Classification Method based on research

Based on all 108 combinations used in this research, we obtain the best performance of the combination of preprocessing, N-Gram feature extraction, feature selection, and classification method. As can be seen in Table 7, the combination of stemming, without using normalization, combined (unigram and bigram), Information Gain, and Maximum Entropy yielded the best performance which has 0.9248 of F1-Score, 0.8746 of accuracy, 0.7449 of AUC, 0.8956 of precision, and 0.9561 of recall. This research uses the imbalanced dataset which has 888 positives and 212 negatives. Although the data are imbalanced enough, the performance of the best combination still yielded a good result which has 0.7449 of AUC value that belongs to “Acceptable” category.

4. DISCUSSION AND CONCLUSION

Experimental results show that although we skip the normalization step in preprocessing, Maximum Entropy still outperforms Random Forests and Naive Bayes which also can more handle the large size of features with shorter execution time than Random Forests. Unlike Naive Bayes, Maximum Entropy is not too vulnerable to sparsity changing in DFT feature selection and better in handling imbalanced data which has higher values of AUC. ME also predicts the sentiment better without any assumption of data and without knowing the data distribution. In the other hand, RF performs better when we use normalization in preprocessing combined with statistical feature selection like IG and CHI instead of DFT. Unfortunately, NB performs worst in handling imbalanced data which has zero successful events and lowest AUC value among others.

In order to improve the performance, the use of stemming process in preprocessing can increase the frequency of features appearance, it’s useful to indicate the important features and decrease the sparsity of matrix. Both stemming and normalization techniques can improve the performance of each model despite the complexity of Indonesian slang and the limitation of stemmer algorithm. The goals of the stemming process can be reached depends on the normalization technique which can decrease the ambiguous words in the dataset. Normalization challenge must able to distinguish the slang, synonyms, acronyms, and words that related to specific study case such as tax amnesty.

Actually, N-Gram types affect the size of extracted features which may lead to overfittings like the combined unigram, bigram, and trigram in this experiment. Deciding the best types of N-Gram must consider the applied algorithms to select the features, the complexity of training algorithm regarding training time consumption and size of extracted features which may lead overfitting or underfitting. Combined unigram and bigram can improve the performance of prediction which outperforms other types of N-Grams that used in this research.

Furthermore, We must consider the types and criteria of machine learning that used before choosing the feature selection algorithm. Actually, we should notice the Naive Bayes independence assumption, feature-based algorithm of Maximum Entropy, and complex ensemble method of Random Forest. Eliminating features using DFT based on N-Gram features result which contains the unpredictable size of features yielding bad results if we fail to choose the best cut-off point in DFT which related to the sparsity of document-term matrix. The cut-off point is very relative and we must attempt several cut-off points to choose the best one. IG feature selection yielding the best performance results based on the successful events and it produces the smaller size of features than DFT and CHI feature selection performs well when it combined with Random Forests.

From 108 generated combination models in this research, we obtain the best performance of model which use the combination of stemming and without normalization technique, combined (unigram and bigram) feature extraction, Information Gain feature selection, and Maximum Entropy classification method. This best combination still performs well which has acceptable category based on AUC value and has the highest performance among others although the data are imbalanced enough.

5. REFERENCES

Abellán, J., & Castellano, J. G. 2017. Improving the Naive Bayes Classifier via a quick variable selection method using Maximum of Entropy. *Entropy*, *19*, 247.

Adriani, M., Asian, J., Nazief, B., & Williams, H. E. 2007. Stemming Indonesian□: A confix-stripping approach. *Association for Computing Machinery*, *6*, 1–33.

Aliwy, A. H., & Ameer, E. H. A. 2017. Comparative Study of Five Text Classification Algorithms with their Improvements. *International Journal of Applied Engineering Research*, *12*, 4309–4319.

Azam, N., & Yao, J. 2012. Comparison of term frequency and document frequency based feature selection metrics in text categorization. *Expert Systems With Applications*, *39*, 4760–4768.

Bai, V. M. A., & Manimegalai, D. 2017. Analysis of feature selection measures for text categorization. *International Journal of Enterprise Network Management*, *8*, 45–60.

Basile, A., Dwyer, G., Medvedeva, M., Rawee, J., Haagsma, H., & Nissim, M. 2017. N-gram: New Groningen author-profiling model in *CLEF 2017 Evaluation Labs*.

Breiman, L. 1996. Bagging Predictors. *Machine Learning*, *24*, 123–140.

Breiman, L. 2001. Random Forests. *Machine Learning*, *45*, 5–32.

Gupte, A., Joshi, S., Gadgul, P., & Kadam, A. 2014. Comparative study of classification algorithms used in sentiment analysis. *International Journal of Computer Science and Information Technologies*, *5*, 6261–6264.

Htet, H., & Myint, Y. Y. 2018. Social media (twitter) data analysis using maximum entropy classifier on big data processing framework (case study□: Analysis of health condition, education status, states of business). *Journal of Pharmacognosy and Phytochemistry*, 7, 695–700.

Kobayashi, V. B., Mol, S. T., Berkers, H. A., Kismihok, G., & Hartog, D. N. Den. 2018. Text Mining in Organizational Research. *Organizational Research Methods*, *21*, 733–765.

Ogada, K., & Mwangi, W. 2015. N-gram based text categorization method for improved data mining. *Journal of Information Engineering and Applications*, *5*, 35–44.

Ong, B. Y., Goh, S. W., & Xu, C. 2015. Sparsity adjusted Information Gain for feature selection in sentiment analysis. *IEEE International Conference on Big Data*, 2122–2128.

Padmanaban, S., Baker, J., & Greger, B. 2018. Feature Selection Methods for Robust Decoding of Finger Movements in a Non-human Primate. *Frontiers in Neuroscience*, *12*, 1-15.

Purohit, A., Atre, D., Jaswani, P., & Asawara, P. 2015) Text Classification in Data Mining. *International Journal of Scientific and Research Publications*, *5*, 1–6.

Raczko, E., & Zagajewski, B. 2017. Comparison of support vector machine, random forest and neural network classifiers for tree species classification on airborne hyperspectral APEX images. *European Journal of Remote Sensing*, *50*, 144–154.

Rajeswari, R. P., Juliet, K., & Aradhana. 2017. Text Classification for Student Data Set using Naive Bayes Classifier and KNN Classifier. *International Journal of Computer Trends and Technology*, *43*, 8–12.

Ren, Q., Cheng, H., & Han, H. 2017. Research on machine learning framework based on Random Forest algorithm. *Advances in Materials Machinery Electronics I AIP Conf.Proc*, *1820*, 1–7.

Roobaert, D., Karakoulas, G., & Chawla, N. V. 2006. Information Gain, correlation and Support Vector Machines. *Springer Berlin Heidelberg*, *207*, 463–470.

Rubén, R., & Chuvieco, E. 2017. Developing a Random Forest Algorithm for MODIS Global Burned Area Classification. *Remote Sens*, *9*, 1193.

Tala, F. Z. 2003. *A study of stemming effects on information retrieval in Bahasa Indonesia*. University of Amsterdam.

Tang, B., Kay, S., & He, H. 2016. Toward optimal feature selection in Naive Bayes for text categorization. *IEEE Transactions on Knowledge and Data Engineering*, *28*, 2508–2521.

Tripathi, S. 2015. Bigram extraction and sentiment classification on unstructured movie data. *International Journal of Electrical and Electronics Research*, *3*, 3–7.

Wang, S., & Manning, C. D. 2012. Baselines and bigrams□: Simple, good sentiment and topic classification. In *Proceedings of the 50th Annual Meeting of the Association for Computational Linguistics*.