JURNAL MEDIA INFORMATIKA BUDIDARMA
Volume 4, Nomor 4, Oktober 2020, Page 1181-1186
ISSN 2614-5278 (media cetak), ISSN 2548-8368 (media online)
Available Online at https://ejurnal.stmik-budidarma.ac.id/index.php/mib
DOI 10.30865/mib.v4i4.2276

Implementation of TF-IDF Method and Support Vector Machine Algorithm for Job Applicants Text Classification

Muhammad Faris Luthfi*, Kemas Muslim Lhaksamana

Fakultas Informatika, Universitas Telkom, Bandung, Indonesia
Email: [1]frsluthfi@student.telkomuniversity.ac.id, [2]kemasmuslim@telkomuniversity.ac.id
Email Penulis Korespondensi: frsluthfi@student.telkomuniversity.ac.id

Abstrak−Puluhan ribu orang melamar pekerjaan di PT. Telkom setiap tahun. Tujuan dari proses rekrutmen adalah untuk mendapatkan karyawan baru yang dapat memenuhi PT. Budaya kerja Telkom. Karena banyaknya pelamar, proses rekrutmen memakan banyak waktu dan mempengaruhi biaya yang lebih tinggi untuk dibelanjakan. Untuk mengatasinya, digunakan metode populer Term Frequency-Inverse Document Frequency (TF-IDF) sebagai metode ekstraksi fitur dan Support Vector Machine (SVM) untuk melakukan klasifikasi untuk menyaring teks wawancara pelamar. SVM umumnya menghasilkan akurasi yang lebih baik dalam klasifikasi teks dibandingkan dengan algoritma Random Forest atau K-Nearest Neighbors (KNN). Namun, TF-IDF memiliki beberapa pengembangan untuk memperbaiki kekurangannya, salah satunya adalah Term Frequency-Relevance Frequency (TF-RF). Sebagai perbandingan, dalam penelitian ini digunakan tiga metode ekstraksi fitur: TF saja (tanpa IDF), TF-IDF, dan TF-RF. Penelitian ini menggunakan teks wawancara dari PT. Telkom sebagai sumber data. Hasil kombinasi SVM dengan TF-IDF dapat menghasilkan akurasi 86,31 %, dengan TF hanya dapat menghasilkan 85,06 %, dan dengan TF-RF dapat menghasilkan akurasi 83,61 %. Hasilnya menunjukkan metode ekstraksi TF-IDF masih dapat mengungguli TF-RF dalam hal akurasi.

Kata Kunci: Proses Rekrutmen, Ekstraksi Fitur, Support Vector Machine, Term Frequency-Inverse Document Frequency, Term Frequency-Relevance Frequency

Abstract−Tens of thousands of people are applying for job in PT. Telkom each year. The goal of the recruitment process is to get new employees which can fit PT. Telkom's working culture. Due to the high number of applicants, the recruitment process takes a lot of time and affecting higher cost to spend. We're proposing a popular combination of Term Frequency-Inverse Document Frequency (TF-IDF) as the extraction method and Support Vector Machine (SVM) as the classifier to filter the applicants' interview text. SVM generally produces better accuracy in text classification compared to Random Forest or K-Nearest Neighbors (KNN) algorithm. However, TF-IDF has several developments to improve its flaws, one of them is Term Frequency-Relevance Frequency (TF-RF). As a comparison, in this study we use three extraction methods: TF only (without IDF), TF-IDF, and TF-RF. We use interview texts from PT. Telkom as the data source. The results of combination SVM with TF-IDF can produce 86.31\% of accuracy, with TF only can produce 85.06\%, and with TF-RF can produce 83.61\% of accuracy. The results show extracting method TF-IDF can still outperform TF-RF in term of accuracy.

Keywords: Recruitment Process, Feature Extraction, Support Vector Machine, Term Frequency-Inverse Document Frequency, Term Frequency-Relevance Frequency

1. INTRODUCTION

Statistics Indonesia or locally known as Badan Pusat Statistik (BPS) stated that in 2019 there were 5 out of every 100 workers in Indonesia who were still unemployed [1]. This figure is relatively high. The impact can be seen in tens of thousands of people applying for employment in PT. Telkom Indonesia every year as an example. The large number of applicants makes the recruiting process more complicated. The higher time that increases costs should be reduced by using the text classification method. However, not much recent research has focused on the classification of job interview texts.

This study focuses on the classification of the PT interview texts. The Telkom recruitment process, which consists of nine parameters of categories of work culture: Solid, Speed, Intelligent, Imagine, Focus, Action, Integrity, Enthusiasm, Totality. Each label has two classes that represent a greater number of classes, a greater probability of being chosen. The goal is to rank each given interview text entry, whether it corresponds to a lower or higher class.

Most text classification systems are integrated into four sentences: feature extraction, dimension reduction, classifier selection, and evaluation [2]. In this study, the dimension reduction method is not used because it is not really necessary due to the limited data sources available. There are several approaches to text classification, which are vector space models, probabilistic models, and inference network models [3]. The most common techniques for extraction characteristics are the Term Frequency-Inverse Document Frequency (TF-IDF) and the standard Term Frequency (TF) [2]. In some cases, the TF-IDF has several drawbacks that reduce its ability to determine the value of different conditions normally. Therefore, the research [4] introduced a developed frequency-based model called the Term Frequency-Relevance Frequency.

As there are not many studies that focus on classification interview texts, we tried to propose a basic method for further comparison in the classification of interview texts specifically. The survey studied in the study [2] states that the Support Vector Machine (SVM) generally offers better results in text classification compared to other traditional classification algorithms. Therefore, this study uses SVM as the classification method. For best results,

Submitted: **06/08/2020**; Accepted: **21/07/2020**; Published: **20/10/2020**

we compared the three characteristics of the feature extraction process: TF, TF-IDF, and TF-RF to determine which combination is best suited to classify text interviews with limited data.

The purpose of this study is to build a system to implement job classifier text classifications into two classes using TF-IDF and SVM and to analyze the classification performance results obtained from applying TF-IDF compared to TF methods. RF and TF. In the end result investigation process, various process scenarios are applied to obtain the best end results.

2. RESEARCH METHODOLOGY

2.1 Term Wighting Methods

There are several approaches that can be made to classify text, including vector space models, probabilistic models, and inference network models, where the vector space model is most frequently used among others [3]. The term weighting method is a method often used in text classification. Term weighting can be interpreted as a method of producing the value of a term that can be a word, phrase, or other unit in a text [5]. The term weighting method has several variations, each of which aims to optimize its performance, of which two of them will be explained as follows.

TF-IDF (Term Frequency-Inverse Dcoument Frequency) is a feature extraction method commonly used in vector space-based text classifications. TF-IDF weighs each word that appears and calculates the inverse value in the sentence. The word represents each feature in the document. TF-IDF is represented in an array, where each row of the array contains data and the columns of the array contain words or features [6]. The weight obtained is used as input for the classification.

In the TF-IDF weighting method, there are several variations that have been developed to improve classification results, including TF-IDF-CF (Term Frequency-Inverse Dcoument Frequency -Class Frequency) [3], TF-IGM (Term Frequency-Inverse Gravity Moment) [7], and TF-RF (Frequency-Relevance Frequency) [4]. TF-IDF is formulated as follows:

$$a_{ij} = tf_{ij} * \log\left(\frac{N}{n_j}\right) \tag{1}$$

Where tf_{ij} represents the term frequency *j* in document *i*, N represents the total document in the data set and n_j represents the number of documents where I am in it. However, when the values of N and n_j are equal, the results of the TF-IDF calculation will be zero. To avoid zero TF-IDF results, smoothing is performed in TF-IDF calculations [3].

$$a_{ij} = \log(tf_{ij} + 1.0) * \log\left(\frac{N + 1.0}{n_j}\right) \tag{2}$$

To overcome the deficiencies in TF-IDF, a calculation of the parameters representing the characteristics of the class frequency called in class is performed, where the method becomes TF-IDF-CF, with the formula [3]:

$$a_{ij} = \log(tf_{ij} + 1.0) * \log\left(\frac{N + 1.0}{n_j}\right) + \frac{n_{cij}}{Nci} \tag{3}$$

Dimana n_{cij} merepresentasikan jumlah dokumen yang di dalamnya terdapat *term j* berada pada kelas *c* yang sama dengan dokumen *i*, dan Nci merepresentasikan jumlah dokumen pada kelas *c* pada dokumen *i*.

Where n_{cij} represents the number of documents in which the term *j* is in class *c*, which is the same as document *i*, and Nci represents the number of documents in class *c* in document *i*.

There is a problem when term *j* number *a* is in class *c* and term *j* number *d* is not in class *c*, which results in the same IDF value even though it has different comparisons of values *a* and *d* as in the following table [4].

Table 1. Example of three terms having different a : d proportion but the same IDF value.

Term	Total(a,d)	a : d	IDF
j_1	100	10 : 1	$\log\left(\frac{N}{100}\right) = 3.322$
j_2	100	1 : 1	$\log\left(\frac{N}{100}\right) = 3.322$
j_3	100	1 : 10	$\log\left(\frac{N}{100}\right) = 3.322$

To overcome these deficiencies, we need a solution that can separate the values in the document where the development focuses on increasing the influence of discrimination on the values of *a* and *d* called the rf (relevance frequency) factor, which is formulated as follows [4]:

$$rf = \log\left(2 + \frac{a}{d}\right) \tag{4}$$

With the variety of TF-IDF methods that have been developed, this research will focus on combining the use of various methods developed from TF-IDF to extract features with text classification of job applicants, thus can be seen which method is most accurate in the case of job applicant text classification.

2.2 Support Vector Machine (SVM)

SVM is a supervised classification method with a concept that is basically similar to ANN, where they both need to find the correct hyperplane to separate the classes in the data. A hyperplane can be said to be optimal when it is located exactly between two classes where it has the greatest distance from the outer edge of the two classes [8]. There are several types of SVM, where the SVM binary class is used in this study. SVM pseudocode as follows [8]:

Algorithm 1 Support Vector Machine (SVM)

```
1) Initiation $\alpha_i = 0$, calculate matrix $D_{ij} = y_i y_j (K(x_i x_j) + \lambda^2)$
2) Do the steps below for $i = 1,2,\dots,l$
      a. $E_i = \sum_{j=1}^{l} a_j D_{ij}$
      b. $\delta a_i = \min\{\max[\gamma(1 - E_1), -a_i], C - a_i\}$
      c. $a_i = a_i + \delta a_i$
3) Repeat step 2 untul $a$ is convergence
```

2.3 The System Built

This study focuses on comparing Term Frequency (TF) weighting, Term Frequency-Inverse Document Frequency (TF-IDF), and Term Frequency-Relevance Frequency (TF-RF) methods for feature extraction and Support Vector Machine (SVM) to classify the text of the job applicants represented in the text of the interview obtained from PT. Telkom. This research is divided into several steps from preprocessing, feature extraction using TF, TF-IDF, and TF-RF, sharing data into training data and test data using K-Fold cross-validation, classification using SVM, and evaluation uses precision calculation and F1 score.

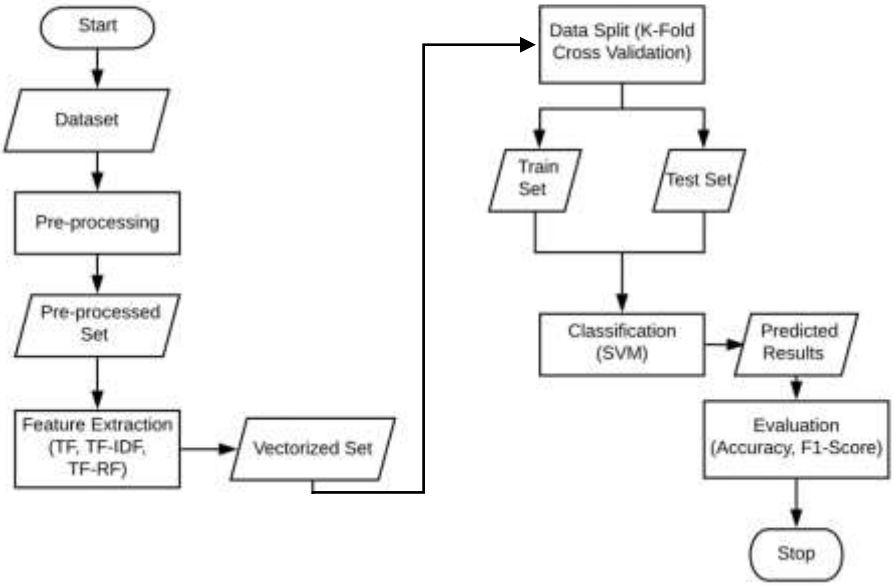

Figure 1. System's Flowchart

There are 482 single-label data used as data sets in this study obtained from the PT interview process. Telkom Data falls into nine categories: Integrity, Enthusiasm, Wholeness, Solid, Speed, Smart, Imagination, Focus, and Action. All the data has been labeled by experts in their fields and each category has two classes: 1 and 2. Class 2 has a higher value than class 1, which means a greater probability of being accepted.

The preprocessing stage aims to erase the data set of terms that are not needed. In text sorting, there are generally four main stages in preprocessing: lowercase, tokenization, empty word removal, and derivation [9]. The preprocessing stages in this study are illustrated in the flowchart in Figure 2.

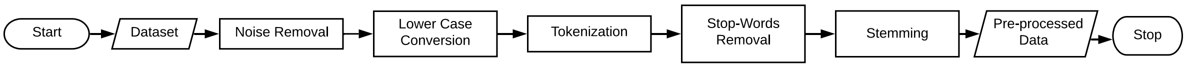

Figure 2. Pre-processing Flowchart

The noise removal stage is carried out to eliminate components that are not necessary in the classification process, such as punctuation marks, numbers and other non-alphabetic characters contained in the text. An example of the noise removal stage is in Table 2 below.

Table 2. Noise Removal Example

Input	Output
Bagi saya organisasi adalah segalanya. Dalam organisasi jiwa kepemimpinan saya terbentuk. Maka dari itu, semasa kuliah saya mengikuti banyak sekali organisasi baik dalam maupun luar kampus. Dengan total 121 organisasi yang terdapat di kampus, setidaknya saya telah mengikuti 30 diantaranya.	Bagi saya organisasi adalah segalanya Dalam organisasi jiwa kepemimpinan saya terbentuk Maka dari itu semasa kuliah saya mengikuti banyak sekali organisasi baik dalam maupun luar kampus Dengan total organisasi yang terdapat di kampus setidaknya saya telah mengikuti diantaranya

At the lowercase conversion stage, all characters in the text are lowercase, because the requester's text is case sensitive and does not affect the content of the text. Standardization of upper and lower case letters is generally done before entering the classification stage [9]. Table 3 below shows an example of lowercase conversion.

Table 3. Lowercase Conversion Example

Input	Output
Bagi saya organisasi adalah segalanya Dalam organisasi jiwa kepemimpinan saya terbentuk Maka dari itu semasa kuliah saya mengikuti banyak sekali organisasi baik dalam maupun luar kampus Dengan total organisasi yang terdapat di kampus setidaknya saya telah mengikuti diantaranya	bagi saya organisasi adalah segalanya dalam organisasi jiwa kepemimpinan saya terbentuk maka dari itu semasa kuliah saya mengikuti banyak sekali organisasi baik dalam maupun luar kampus dengan total organisasi yang terdapat di kampus setidaknya saya telah mengikuti diantaranya

Tokenization is done to separate the text into words, phrases, or other forms that have meaning. The results of the text separation are called tokens. The following is an example of the tokenization process.

Table 4. Tokenization Example

Input	Output
bagi saya organisasi adalah segalanya dalam organisasi jiwa kepemimpinan saya terbentuk maka dari itu semasa kuliah saya mengikuti banyak sekali organisasi baik dalam maupun luar kampus dengan total organisasi yang terdapat di kampus setidaknya saya telah mengikuti diantaranya	‘bagi’,’saya ’,’organisasi ’, ’adalah ’,’segalanya’,’dalam’, ’organisasi ’,’jiwa’, ‘kepemimpinan’,’saya ’,’terbentuk’, ‘maka’,’dari ’,’itu ’,’semasa ’,’kuliah’, ’saya ’,’mengikuti ’,’banyak ’,’sekali’, ‘organisasi’,’baik ’,’dalam ’,’maupun’, ’luar ’,’kampus ’,’dengan ’,’total’, ’organisasi’,’yang ’,’terdapat ’,’di’, ’kampus ’,’setidaknya ’,’saya ’,’telah’, ’mengikuti ’,’diantaranya’

Stop-word removal is performed to remove words that are not considered important to the content of the text, such as conjunctions, prepositions, and others. With the loss of these words, the dimensions of the text are reduced so that the classification process can be executed more effectively.

Table 5. Stop-word Removal Example

Input	Output
‘bagi’,’saya ’,’organisasi ’, ’adalah ’,’segalanya’,’dalam’, ’organisasi ’,’jiwa’, ‘kepemimpinan’,’saya ’,’terbentuk’,	‘bagi’,’saya ’,’organisasi’, ’segalanya’,’organisasi’, ’jiwa’, ‘kepemimpinan’,’saya ’,’terbentuk’, ’semasa

Submitted: 06/08/2020; Accepted: 21/07/2020; Published: 20/10/2020

Input	Output
‘maka’,’dari ’,’itu ’,’semasa ’,’kuliah’, ’saya ’,’mengikuti ’,’banyak ’,’sekali’, ‘organisasi’,’baik ’,’dalam ’,’maupun’, ’luar ’,’kampus ’,’dengan ’,’total’, ’organisasi’,’yang ’,’terdapat ’,’di’, ’kampus ’,’setidaknya ’,’saya ’,’telah’, ’mengikuti ’,’diantaranya’	’,’kuliah’, ’saya’, ’mengikuti’, ’banyak ’, ‘organisasi’,’dalam’,’luar ’,’kampus’, ’total’, ’organisasi’,’terdapat’, ’kampus ’,’setidaknya ’,’saya ’,’telah’, ’mengikuti ’,’diantaranya’

Each word in the text can have the same meaning but in a different way, like a fixed word (example: "membaca" with "baca"). The derivation stages are carried out so that the two word forms are uniform, so that the dimensions of the features in the text become even lower. However, the derivation stage does not always provide optimal results in the classification, where in some cases it can reduce the precision of the classification process [10]. Especially when discovering the characteristics of job applicants, adjectives and conjunctions can be important characteristics for the classification process. However, research [11] has shown that the derivation process produces efficient results for the classification of texts in the system. Therefore, this study also made comparisons between texts that experienced derivations and those that did not.

The next stage is to extract features. The preprocessed data is transformed into a vector space model using the three methods TF, TF-IDF and TF-RF. The formula that was written in the previous section is used at this stage. The result of this stage is the single vector (depending on the method used) used in the classification.

The data division stage uses K-Fold cross validation. Although the amount of data established in this study can be classified as a small amount and may give rise to the possibility of overlapping training data, research [12] shows that training data that can even be classified as highly overlapping produces a relatively real model for precision. In this study using 10 as the K value in the K-Fold cross validation.

After the training data is formed, the next step is to train the training data and classify the test data using SVM. One of the challenges is finding the correct kernel in the classification process, therefore, in this study, optimization methods using Grid Search [13] with ranges of hyperparameters are listed in Table 6.

Table 6. Hyper Parameter Settings

Parameter	Kernel	Value	Type
C	RBF, Linear, Polynomial, Sigmoid	0, 10, 100, 1000	*Integer*
gamma	RBF	0.001, 0.0001	*Real*

After there are classification results, an analysis is performed based on the calculation of the precision and the F1 score obtained from a confusion matrix consisting of True positive (TP), True negative (TN), False positive (FP), False negative (FN). The formula used for the value-based evaluation in the confusion matrix is calculated using the following equation [2]:

$$accuracy = \frac{(TP + TN)}{(TP + FP + FN + TN)} \tag{5}$$

$$precision = \frac{\sum_{I=1}^{L} TP_I}{\sum_{I=1}^{L} TP_I + FP_I} \tag{6}$$

$$recall = \frac{\sum_{I=1}^{L} TP_I}{\sum_{I=1}^{L} TP_I + FN_I} \tag{7}$$

$$F1 - Score = \frac{\sum_{I=1}^{L} 2TP_I}{\sum_{I=1}^{L} 2TP_I + FP_I + FN_I} \tag{8}$$

3. RESULTS AND DISCUSSION

3.1 Results and Analysis on The Usage of Stemming Procedure Scenario

The first scenario is to determine whether the bypass procedure will be used in preprocessing or not. The results in Table 7 below show that the derivation provides greater precision and F1-Score results than the average of the three feature extraction methods.

Table 7. Comparison of Performance to Stemming Procedure Usage

Stemming Procedure	F1-Score (%)	Accuracy (%)	Time
Used	66.35	84.99	17 minutes 50 seconds
Unused	58.41	82.71	7 minutes 39 seconds

Submitted: 06/08/2020; Accepted: 21/07/2020; Published: 20/10/2020

Based on the results of the first scenario listed in Table 7, the stemming procedure provides greater precision and an F1 score value to the system, but the process takes longer time. In terms of the prediction accuracy of the system, stemming produces better results but a longer time, which is more than double the time without stemming. This research still applies the stemming procedure because in the recruitment process, of course, high precision is needed, so that the results are in accordance with the required criteria. The trend of system time that tends to be long with the implementation of referral procedures is still much less than the time of interviews and expert-based evaluation processes. A higher derived result means that the job applicant text has a variety of words with the same meaning and provides better input to the system because the vectors become less classified.

3.2 Results and Analysis in Determination of K Value

The second scenario relates to the determination of the K value to be used in the K-Fold cross validation validation method. The K values used are 5, 7 and 10.

Table 8. Results on Each K Value

K Value	Precision (%)	Recall (%)	F1-Score (%)	Accuracy (%)
5	31.39	86.68	44.89	78.49
7	32.61	92.53	46.55	79.81
10	52.07	91.44	66.35	84.99

In this scenario, the Accuracy, Recovery, F1 Score, and Accuracy values are the average values for using the three types of feature extraction methods: TF-IDF, TF, and TF-RF. Based on the results of the scenarios in Table 8, it can be seen that the value of K = 10 produces an F1 score and an accuracy that is higher than the values of K = 5 and 7. The precision results in K = 10 have a relatively higher value than the others, which is 52.07%. This shows the number of documents that the system can predict well in the text documents of the job applicant for positive classes (Class 2). While as can be seen in the table, the recall value at the value of K = 7 is slightly higher than the recall value at the other K which is equal to 92.53%. This is related to the precision value at K = 7, which is less than that of K = 10, which is why many documents are classified as positive (Class 2) slightly more than when the value of K = 10. F1 score and accuracy of the K = 10 value has the best results compared to the K = 5 and 7 values. According to the results of the F1 score and accuracy obtained, in this study the K value used is K = 10.

3.3 Analysis on System's Final Result

The next scenario is to observe the final results of the system by comparing the results of the TF-IDF feature extraction method classification combination with the TF and TF-RF and SVM classification algorithm. The results are shown in Table 9 below.

Table 9. System's performance result

Feature Extraction Method	Performance				
	Precision (%)	Recall (%)	F1-Score (%)	Akurasi (%)	Time
TF-IDF	54.74	94.94	69.44	86.31	17 minutes 55 seconds
TF	51.82	92.21	66.36	85.06	17 minutes 42 seconds
TF-RF	49.64	87.18	63.26	83.61	17 minutes 54 seconds

The final result of the system shows that the precision value in each feature extraction method is less than the recall. Low precision results show that the system predicts that the job applicant's text documents will be accepted more accurately (with a higher grade 2) with less precision, but on the other hand, a high withdrawal value indicates that many documents they are positively classified (many documents are classified with class 2) correctly. Imbalance data is believed to be the cause of the low precision value.

From Table 9 it can be seen that the TF-IDF feature extraction method has an accuracy value of 86.31% and an F1 score of 69.44%, where the value is greater than the TF feature extraction method that has a accuracy of 85.06% and an F1 score of 66.26% and even the method developed from TF-IDF weakness which is TF-RF with an accuracy of 83.61% and F1-Score 63.26%. High precision values indicate that the system can correctly classify the value of True Positive (TP) and True Negative (TN). The system can predict documents with correct label 1 in class 1 and correctly label 2 with class 2 with fairly good accuracy. However, F1-Score values that are lower than precision indicate False Positive (FP) and False Negative (FN) values that still have low precision. F1 scores below this precision indicate that the system can predict Class 2 documents correctly, but Class 1 documents are less accurate. This indicates an imbalance in the existing data.

The F1 score and precision of the TF-IDF feature extraction method are higher than TF and TF-RF, indicating that the combination of TF-IDF and SVM methods is in accordance with the text characteristics of the applicants for use, although the time is one or two seconds longer than the feature extraction method the other.

Submitted: 06/08/2020; Accepted: 21/07/2020; Published: 20/10/2020

4. CONCLUSION

From the results of the study, it can be concluded that the system can classify the text of job applicants well and has good potential to be implemented as a substitute for a job applicant recruitment system. In preprocessing, stemming provides higher aspiration results and F1 score. In the feature extraction method, TF-IDF provides better vectors that imply better classification results than TF and TF-RF. This study shows that TF-IDF produces an accuracy value of 86.31% and F1-Score 69.44%, better than TF with 85.06% accuracy and F1-Score 66.26% and even with the results of the TF-IDF development method, it is say TF-RF with an accuracy of 83.61% and F1-Score 63.26%. For future research, we recommend adding more data to obtain a more balanced representation of data, so that the value of the F1 score can be increased to produce more optimal classification results.

ACKNOWLEDGMENT

Thanks to those who have supported the implementation of this research.

REFERENCES

[1] Badan Pusat Statistik, “Keadaan Ketenagakerjaan Indonesia Agustus 2019,” 2019. [Online]. Available: https://www.bps.go.id/pressrelease/2019/11/05/1565/agustus-2019--tingkat-pengangguran-terbuka--tpt--sebesar-5-28-persen.html. [Accessed: 05-Jun-2020].

[2] K. Kowsari, K. J. Meimandi, M. Heidarysafa, S. Mendu, L. Barnes, and D. Brown, “Text classification algorithms: A survey,” *Inf.*, vol. 10, no. 4, pp. 1–68, 2019.

[3] M. Liu and J. Yang, “An improvement of TFIDF weighting in text categorization,” *Int. Conf. Comput. Technol. Sci.*, vol. 47, no. Iccts, pp. 44–47, 2012.

[4] M. Lan, C. L. Tan, and H. B. Low, “Proposing a new term weighting scheme for text categorization,” *Proc. Natl. Conf. Artif. Intell.*, vol. 1, pp. 763–768, 2006.

[5] M. Lan, C. L. Tan, S. Member, J. Su, and Y. Lu, “Supervised and Traditional Term Weighting Methods for Automatic Text Categorization,” vol. 31, no. 4, pp. 721–735, 2009.

[6] M. Y. Abu Bakar, Adiwijaya, and S. Al Faraby, “Multi-Label Topic Classification of Hadith of Bukhari (Indonesian Language Translation)Using Information Gain and Backpropagation Neural Network,” *Proc. 2018 Int. Conf. Asian Lang. Process. IALP 2018*, pp. 344–350, 2019.

[7] K. Chen, Z. Zhang, J. Long, and H. Zhang, “Turning from TF-IDF to TF-IGM for term weighting in text classification,” *Expert Syst. Appl.*, vol. 66, pp. 1339–1351, 2016.

[8] Suyanto, *Machine Learning Tingkat Dasar dan Lanjut*. Bandung: Penerbit Informatika, 2018.

[9] A. K. Uysal and S. Gunal, “The impact of preprocessing on text classification,” *Inf. Process. Manag.*, vol. 50, no. 1, pp. 104–112, 2014.

[10] A. F. Hidayatullah, U. I. Indonesia, C. I. Ratnasari, and U. I. Indonesia, “Analysis of Stemming Influence on Indonesian Tweet Classification,” no. August, 2016.

[11] F. Song, S. Liu, and J. Yang, “A comparative study on text representation schemes in text categorization,” pp. 199–209, 2005.

[12] T. Wong and N. Yang, “Dependency Analysis of Accuracy Estimates in k-fold Cross Validation,” vol. 4347, no. c, pp. 1–12, 2017.

[13] I. Syarif and G. Wills, “SVM Parameter Optimization using Grid Search and Genetic Algorithm to SVM Parameter Optimization Using Grid Search and Genetic Algorithm to Improve Classification Performance,” no. December, 2016.

Submitted: 06/08/2020; Accepted: 21/07/2020; Published: 20/10/2020