Kernel Analysis of Support Vector Machine for Multi-Label Text Classification of Hadith Indonesia Translation

Mohamad Irwan Afandi[1]
Dept. of Informatics
Telkom University
Bandung, Indonesia
irwanafandi@student.telkomuniversity.ac.id

Widi Astuti[3]
Dept. of Informatics
Telkom University
Bandung, Indonesia
astutiwidi@ telkomuniversity.ac.id

Adiwijaya[2]
Dept. of Informatics
Telkom University
Bandung, Indonesia
adiwijaya@telkomuniversity.ac.id

***Abstract*— Hadith is the second source of law and guidance for Muslims after the Holy Quran. Many Hadiths have been narrated, but according to Islamic scholars, the Bukhari Hadith has the highest level of validity. However, learning Hadith is not easy. The number of Hadiths that exist and which have not been categorized make learning Hadith under specific categories challenging. Therefore, the author conducted research on the classification of Indonesian language translation of the Bukhari Hadith into three classes: advice, prohibition, and information. This method is expected to facilitate the public to learn Hadith quickly. The Support Vector Machine (SVM) algorithm was chosen because, based on previous research, it can work well with multidimensional data. Classification accuracy can also be improved through proper the right kernel selection, and the linear kernel has produced the highest accuracy. This is because the linear kernel can work well in separating two different data, and mostly text classification problems are linearly separable, so it's very suitable to use. The better hamming loss value for this classification is 0.0686. It means 93.14% of the multi-label Hadith data were classified correctly.**

Keywords—Bukhari Hadith, Linear kernel, Multi-label, Support vector machine, text classification

I. INTRODUCTION

The Hadith is all forms of speech, actions, and provisions of the Prophet Muhammad that is used as a guide and a second source of law for Muslims after the Holy Quran [1][2]. The Hadith was passed down from generation to generation by Hadith experts so that the next generation could emulate the actions and habits of the Prophet Muhammad during his lifetime. Many Hadith experts have narrated various Hadiths, but most of the scholars agree that the Hadith narrated by Imam Bukhari has the highest level of validity [3]. Therefore, many Muslims study the Hadith of Bukhari to get closer to *Allah S.W.T*.

Based on the fact, studying the science of Hadith is not easy. Even though technology has made the Hadith easier to obtain, the number of Hadiths that exist and which have not been categorized makes learning Hadith under specific categories very difficult. Basically the Prophetic traditions contain three main classes: advice, prohibitions, and information. The most exciting thing about Hadith is that it can be classified into one class, or it could also be a combination of all three classes. The purpose of this research is to organize the Hadith according to class, which is expected to facilitate the reader in learning and finding the Hadith according to his or her desired category. Although The classification of Bukhari Hadith into the three classes was done in previous research [3],[4], but the study focused on single-label classification. [5]–[7] The studies, as mentioned earlier, had carried out a multi-label classification, but they used different data for the system built. [8] These studies conducted a multi-label classification of Hadith using the Backpropagation Neural Network, but there were still numerous misclassifications in the study.

The Classification of Bukhari Hadith falls into the category of text classification. Text classification is a topic that has often been discussed in recent years. Text classification is used to group related texts into several predetermined classes, for example, grouping on Twitter, news, books, e-mail, sentiment, and so on [9]–[12]. In classifying text, the most common problems are usually related to the number of dimensions in the data, which can be too high. It happens because of the large number of vocabularies produced by the dataset, which can slow down system performance and reduce the accuracy of the classification itself. After feature selection proses are complete, the next step is feature extraction and ends with the classification process to build a model of the Hadith.

II. Related Work

Feature selection is a method that is often used to overcome the high number of dimensions in text classification [11]. In the feature selection process, unique features (words) are selected and assigned a significant influence on the class determination of data. Meanwhile, for non-selected features, the same words are ignored and will no longer be used in the next stage. This method can certainly reduce the number of features available in the dataset, so the dimensions of the dataset used will not be too high.

There are several methods for selecting features that are often used in cases of text classification in the previous research [13]–[16]. One method of feature selection is MI [17], [18]. In one study, MI was used to select the essential terms for identifying relevant e-mails and spam. MI was also used to reduce the number of feature dimensions in e-mail, so that the resulting features can contribute significantly towards identifying the class of each e-mail. The best classification accuracy in determining critical e-mail and spam in the research was 97.3%, using regression as the classification method. In the research conducted by Fahmi [9], MI was used to select features in a news dataset. The features with high MI values had a large effect on the class determination of news data. The results of the research indicated that, with the same method, the classification using MI yielded an F1-score accuracy of 75.34% and a 45.95% accuracy without using MI.

Feature weighting is a process that must be applied after the features in the dataset are obtained. During this process, the weight was calculated, then weighting results were input into the classification system that was built. Wu used TF-IDF to weight words in the corpus of a Bengali document [19]. The result of the weighting was a corpus representation vector, which was then used during the text classification process. [20] Another study used TF-IDF to weight documents in Bengali before the documents were classified using the SVM method. The classification results show that weighting using TF-IDF with a unigram model resulted in an accuracy of 88.13%, while Chi-Square Distribution outputted an accuracy of 82.82%. The output of this process was presented in the form of a matrix, where the columns are feature's weight, and the rows are the document number.

The essence of the text classification process is the classification of text into several classes. SVM is one method that is often used in various classification cases. The SVM method was used to classify the sound recordings of a person speaking in the Mandarin language [21]. The classification of the built system must be able to recognize the category of emotions from the spoken voice. Then, the system determined whether the sound was spoken when the person was angry, happy, sad, bored, or neutral. The best accuracy value generated from the research was 84.2%. This value is better than the Neural Network method, which was only 80.8% accurate. Past research [22], also used the SVM, KNN, and (Naïve-Bayes) NB methods to group documents into four categories: environment, sports, politics, and the arts. The classification results show that the average value of the F1-Score using the SVM method reached 86.26%. The accuracy value was better than the accuracy value produced using the KNN or NB method. The multi-label classification using the SVM method was also carried out in another study [23]. The study used the Binary Relevance (BR) approach to transform multi-label classification problems into several binary classification problems. Then, each binary classification formed was classified as a single label using the SVM classification algorithm. After obtaining the result of the predictions, the result was then corrected by considering the probability of relations between labels to improve classification accuracy. The results of this classification indicate that the use of the SVM method for multi-label cases was better than that of the NB and C4.5 methods.

III. Method

This research designed a system that can classify multi-label data. The number of data used was 1064 Bukhari Hadith, where the class of each Hadith had been labeled previously. The class used in the dataset consists of three types: advice, prohibitions, and information. In multi-label data, a Hadith can belong to one class, but this does not rule out Hadith with a combination of all three classes. In the dataset used consists of four columns. The first column is the Hadith content, and the remaining three are classes. The value 1 in class x means that the Hadith belongs to class x. Meanwhile, 0 means that the Hadith is not classified into class x. The representations of Hadith data used in this study are shown in Table I.

TABLE I. Multi-Label Data Representation

Hadith Data	**Hadith Class**		
	Advice	***Prohibition***	***Information***
Barang siapa berwudu hendaklah mengeluarkan (air dari hidung), dan barang siapa beristinja' dengan batu hendaklah dengan bilangan ganjil.	1	0	1
Tinggalkanlah apa yang tidak kalian sanggupi, demi Allah, Allah tidak akan bosan hingga kalian sendiri yang menjadi bosan, dan agama yang paling dicintai-Nya adalah apa yang senantiasa dikerjakan secara rutin dan kontinyu.	1	1	1

Based on the example above, the first example of a Hadith belongs to the advice and information class. It is because the value of both of class are 1 while the prohibition class is 0, it means the Hadith does not classify into prohibition class. For the second example, all class values are one, mean the Hadith classified to all classes.

In this research, the researcher developed a system that was able to classify the Bukhari Hadith text into three classes automatically. The system was divided into several stages; preprocessing, feature selection, feature extraction, classification of data, and classification model evaluation. The flowchart of the proposed system is presented in Fig.1.

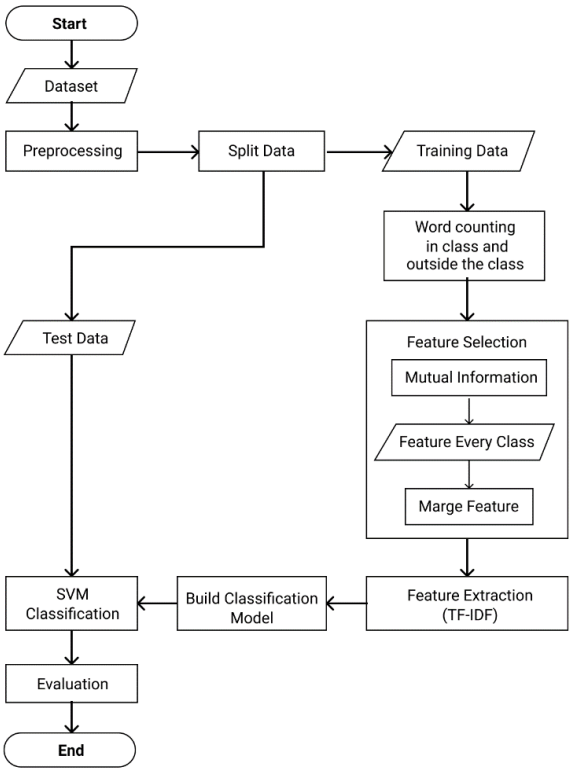

Fig. 1. Flowchart System.

Based on the flowchart system, this research will propose two methods to make automatically classify the Bukhari Hadith text into three classes. The first method is Mutual Information (MI) that use as feature selection. This method can select the features of the Hadith which has a significant impact on determining the class. The second algorithm is Support Vector Machine (SVM) that use as a classification method. This method will separate the data into two sides or more using a line called a hyperplane. The different sides will be classified as a different class too.

A. Preprocessing

Preprocessing was carried out to eliminate the noise, redundancy, imperfections, and inconsistencies contained in the text data [24], so that the data becomes ideal and optimal for the next mining process [25]. The preprocessing step for this classification process is illustrated in Fig.2.

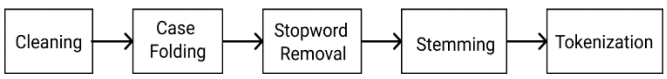

Fig. 2. Preprocessing step

- Cleaning, the process of removing punctuation, numbers, and symbols in Hadith sentences.
- Case folding, the process of changing all Hadith letters in a dataset to lowercase letters so that each word in the sentence will be uniform.
- Stopword Removal, the process of removing words that are not considered necessary in a Hadith sentence. Deleted words are words that have no meaning or words that appear too often.
- Stemming, the process of returning a word in Hadith sentence to its basic word form.
- Tokenization, the process of cutting Hadith sentences into words from the sentence.

The results of preprocessing are pieces of words from each Hadith stored in an array. Then, the dataset was split into two parts, that is training data and test data using k-fold. The next step involved selecting each word in the training data to obtain the most important features for determining each data class. This step is called the feature selection process.

B. Feature Selection

The method used for the feature selection process in this study was Mutual Information (MI). MI is based on the concept of calculating how much information is contained in word terms, which contribute to making appropriate classification decisions in a class [26]. Words with a high MI value will be selected as the main feature for the data classification process [17]. The calculation of the MI value is given by Equation (1).

$$I(U, C) = \sum_{et\epsilon\{1,0\}} \sum_{ec\in\{1,0\}} A \quad (1)$$

where,

$$A = P(U = et, C = ec) \log_2 \frac{P(U=et,C=ec)}{P(U=et)P(C=ec)} \quad (2)$$

In Equation (1), $I(U, C)$ is the value of mutual information of term U in C class. Then, et or ec has the value of 1 or 0, et = 1 (contain term U), et = 0 (does not contain term U), ec = 1 (in C class), ec = 0 (not in C class). This equation can be described in more detail as in Equation 3.

$$I(U, C) = X + Y \quad (3)$$

where,

$$X = \frac{N_{11}}{N} log_2 \frac{N.N_{11}}{N_1.N_1} + \frac{N_{01}}{N} log_2 \frac{N.N_{01}}{N_0.N_1} \quad (4)$$

$$Y = \frac{N_{10}}{N} log_2 \frac{N.N_{10}}{N_1.N_0} + \frac{N_{00}}{N} log_2 \frac{N.N_{00}}{N_0.N_0} \quad (5)$$

Based on Equation (4 and 5), the variable *Ns* is the number of sentences that contain and do not contain term x with the value of ec and et. Suppose that N_{10} is a sentence which contain term x (et = 1) but not in class c (ec = 0). $N_1 = N_{10} + N_{11}$ is the number sentence contain term x (et = 1) and the number of independent sentence members of the class ($ec \in \{1,0\}$). $N = N_{00}+N_{01}+N_{10}+N_{11}$, is the number of all terms in the dataset.

Based on these two equations, the number of terms that appear cannot influence the high value of MI. The number of terms in class and not in a class can also affect the values. The higher value of MI in a word, the more affected the word will be to the existing classes. Furthermore, the features that have been obtained in the three classes were merged into one unit that was then sorted according to the highest MI value. Based on the ordering of MI values, as many as 30-300 top features were used in the next stage.

C. Feature Extraction

After the feature was obtained, the next step was to extract each dataset based on the feature. The Term Frequency-Inverse Document Frequency (TF-IDF) is one of the feature extraction methods, which is often used in cases of text classification [27]. The result of this process is a matrix where each row of the matrix represents the data, and the column indicates the feature. This matrix was filled with the weight of the value of each feature against the document or the values from the multiplication of the TF value by the IDF value [28]. The TF (Term Frequency) is the number of occurrences of a word in a document while IDF (Inverse Document Frequency) is the number of related documents containing a particular word. The weight that was obtained was input into the classification system used. The weight for the TF-IDF was calculated using Equation (6).

$$W_{i,j} = tf_{i,j} x \log(\frac{N}{df_j}) \quad (6)$$

Where $W_{i,j}$ is a word weighted t_j to document d_i, $tf_{i,j}$ is the number of occurrences of the word t_j to document d_i, N is the total number of documents, and df_j is the total number of documents that contain the word$j$.

D. Classification

In this study, the Hadith data were classified using the Support Vector Machine (SVM) method. SVM works by finding the optimal separator hyperplane value between positive and negative samples [29]. The hyperplane is considered optimal if the margin formed between positive and negative samples has the closest distance. Once the location of the hyperplane is found, the class of the new sample that enters can automatically be determined. If the sample is on the positive side, it means the class of the data is positive, and vice versa. The test data $T = \{(x_1, y_1), \dots (x_n, y_n)\}$, where $x_i \in R^n$ are input patterns, and $y_i \in \{-1, 1\}$ is a class label for two classes (positive and negative). The main problem is that SVM tries to find a classifier $f(X)$, which minimizes the error rate from the classification. The equation for the linear kernel $f(X)$ in SVM is given by Equation (7).

$$\min(\frac{1}{2}||W||^2 c(\sum_{i=1}^{t} \xi_i)) \quad (7)$$

Based on equation (3), $y_i\big((W, X_i) + b\big) \geq 1 - \xi_i$ and $\xi_i \geq 0$ where C is a regularization parameter that is used to balance the complexity and accuracy of classification on T-test data. By using a linear kernel, data will be separated into two parts in a straight line (positive and negative). This kernel will work better if the dataset is linearly separable. The linear kernel also still working well although there are many features in a dataset**.** That is because mapping the data to a higher-dimensional space *does not really improve* the performance [30]. The determination of the hyperplane is not only done with linear kernels, but also with a non-linear kernel function, which converts linear SVM to a more flexible non-linear SVM. The kernels are also often used in classifying data such us Gaussian RBF, sigmoid, and polynomial [31], [32].

Gaussian RBF (Radial Basis Function) is a function which creates hyperplane depends on the distance between some point. Usually, the hyperplane formed is curved or surround a group of a class which a smaller amount. The area outside the hyperplane will be classified into another class. This kernel works well in a separate group of class, practice, and they are relatively easy to calibrate because it only has one parameter (gamma).. The equation for the Gaussian RBF kernel in SVM is given by Equation (8).

$$K(\bar{x}_i, \bar{x}_j) = \exp(-\frac{||\bar{x}_i - \bar{x}_j||^2}{2\sigma^2}) \quad (8)$$

Where $\| \vec{x}_i - \vec{x}_j \|^2$ is squared euclidean distance between the two-feature vector. σ is a free parameter. An equivalent definition involves a parameter $\gamma = 1/2\sigma^2$. If the value of γ increase the model will gets outfits, if decrease the model underfits.

The polynomial kernel is a kernel function commonly used with support vector machine (SVM) and other kernelized models that represent the similarity of vectors (training samples) in a feature space over polynomials of the original variables. This kernel usually works well in non-linearly separable data. Intuitively, the polynomial kernel looks not only at the given features of input samples to determine their similarity, but also combinations of these. This similarity will be used to separate one class from another class. This kernel so popular in image processing. The equation for the polynomial kernel in SVM is given by Equation (9).

$$K(\bar{x}_i, \bar{x}_j) = (\bar{x}_i{}^T \bar{x}_j + c)^d \quad (9)$$

Where $\vec{x}_i$ and $\vec{x}_j$ are vector in the input space, i.e. vectors of features computed from training or test samples. Then c is constant that allows to trade off the influence of the higher order and lower order terms, d is the "order" of the kernel.

The sigmoid kernel comes from the neural network field, where the bipolar sigmoid function is often used as an activation function for artificial neurons. This kernel can work well in practice. The equation for the sigmoid kernel in SVM is given by Equation (10).

$$K(\bar{x}_i, \bar{x}_j) = tanh\left(a\bar{x}_i{}^T \bar{x}_j + c\right) \quad (10)$$

Where α is a scaling parameter of the input data. For $\alpha < 0$, the dot-product of the input data is not only scaled but reversed. Then c is a shifting parameter that controls the threshold of mapping.

In multi-label data, how to determine hyperplane is done repeatedly depending on the number of the label in the data. In this experiment, the data has three labels: advice, prohibitions,

and information, where we have to find the hyperplane for each label. The first iteration is used to find the hyperplane that can separate the advice Hadith and non-advice Hadith, then continue to find the hyperplane on prohibitions and information Hadith. Based on the process, we will be obtained three hyperplanes where there will be an area, which is the intersection of two hyperplanes or even all three. The intersection area is what makes the Hadith data can have more than one class.

The following simulation to determine how the system gets the hyperplane in multi-label data. This simulation uses ten dummy data with three features (words) as dimensions to make the data can be visualized, because the number of dimensions depends on the feature that used. The representations of the dummy data used in this simulation shown in Table II.

TABLE II. DUMMY DATA REPRESENTATION

Hadith Data	TF-IDF			Hadis Class		
	Janganlah	*Tunaikan*	*Kerjakan*	*A*	*P*	*I*
Hadith 1	0	0.424	0.528	1	0	1
Hadith 2	0.495	0.621	0.765	1	1	1
Hadith 3	0.878	0	0	0	1	0
Hadith 4	0	0	0.892	1	0	0
Hadith 5	0.376	0.538	0	0	1	1
Hadith 6	0	0.827	0	0	0	1
Hadith 7	0.775	0	0.836	1	1	0
Hadith 8	0.657	0.121	0	0	1	0
Hadith 9	0.461	0.247	0.649	1	1	0
Hadith 10	0.473	0.826	0	0	1	1

Based on that data, we can get the data visualization with the hyperplane in each class. The data visualization result can be seen in Fig.2, Fig.3, and Fig.4. On the figure below, the red circle symbol mean the Hadith data doesn't in that class and the green rectangle symbol mean the Hadith data is in that class.

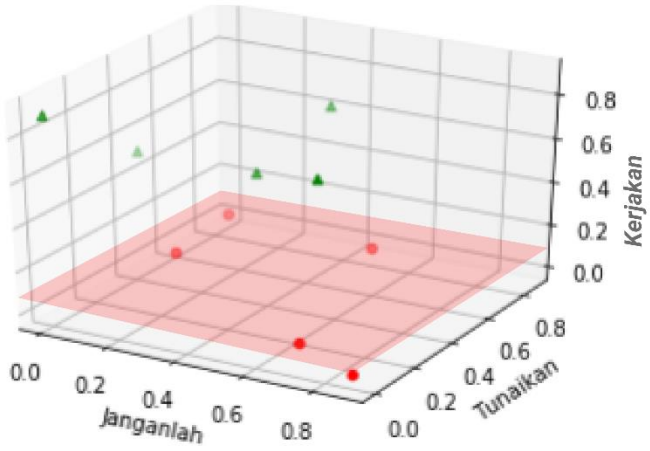

Fig. 2. Visualization of Advice Data

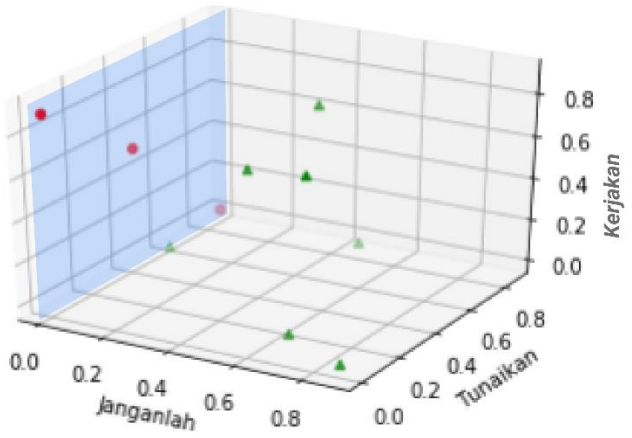

Fig. 3. Visualization of Prohibition Data

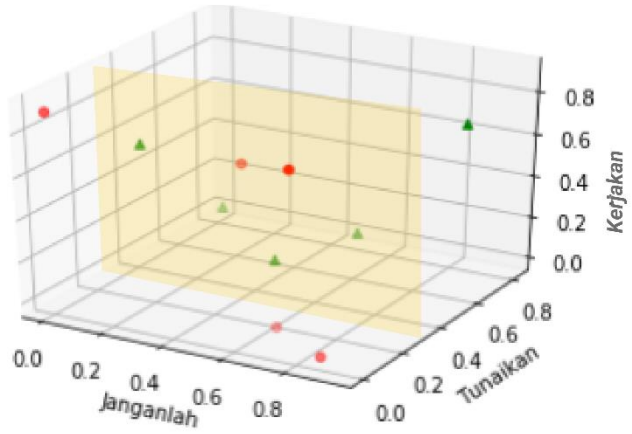

Fig. 4. Visualization of Information Data

The hyperplane that has been obtained will be used to determine the classification model. The model with the highest accuracy will be saved on the system. It will make the work of the system to be lighter because it does not need to find the model when the system starts running. So if there is new data that does not have a class, the data will use that model to find its class.

E. Evaluation

Evaluation is the final stage of the classification process that used to assess the accuracy of the system that is being built. One evaluation method for multi-label classification is Hamming loss, which calculates how much misclassification occurs based on the system built [8], [23]. The smaller the Hamming loss value, the more accurate the system is at classifying data, and vice versa. The method for calculating Hamming loss is given by Equation (8).

$$hloss(h) = \frac{1}{p}\sum_{i=1}^{p}\frac{1}{Q}|h(x_i)\Delta Y_i| \quad (8)$$

Where p is the total number of documents, Q is the number of labels of classes that are used, and $|h(x_i)\Delta Y_i|$ is the amount of misclassification that occurs after document class predictions are obtained.

IV. Experimental Results and Discussions

To observe the results of the tests carried out in this study, the test was conducted several times. There are three focal points in this study that were carried out; preprocessing, feature selection, and classification. The aim of preprocessing is to obtain optimal parameters and prove that the SVM method is a method that is suitable to be applied in the case of text classification. This study used the k-fold cross-validation method for testing. With the k-fold, X dataset is divided into k-sections with the same number, $X = x1, x2, x3, \dots, xk$. For each section in one loop, the data is used as test data, while the rest is used as the training data for building the classification model. This step will repeat until all the data becomes the test, data and the model is built. The accuracy of classification was obtained by calculating the average value of Hamming loss in each fold that was generated. This study used a 10-fold and applied it on 1064 Bukhari Hadith data in Indonesian translation.

The first testing scenario in this research is using Stopword Removal and Stemming in preprocessing process on Hadith dataset. The goal of this test was to observe the effect of using Stopword Removal and Stemming on the value of Hamming loss. To perform Stopword Removal and Stemming on the dataset, the author used Sastrawi's library. In this test, the system built used MI as feature selection and SMV with the linear kernel as the classifier. The results of this first test are shown in Fig. 5.

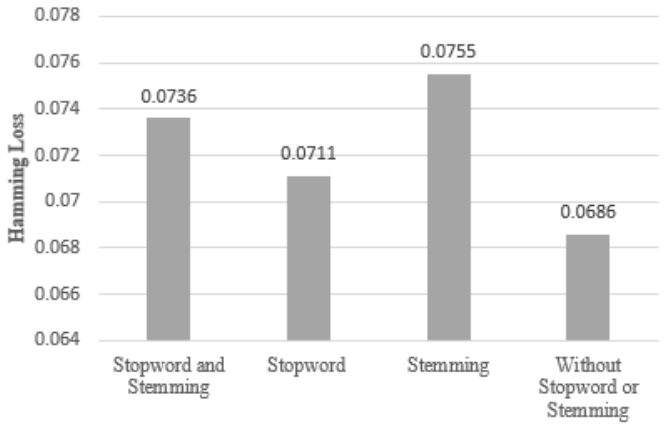

Fig. 5. Testing scenario with stopword removal and stemming process

From the first testing scenario, the best Hamming loss value was generated from testing without Stopword Removal and Stemming. The Hamming loss value was 0.06858, indicating that about 93.14% of the multi-label data were correctly classified. First, Stopword Removal should not be done in the preprocessing step. This is because when Stopword Removal is used, words that are considered to be useless or words that most frequently appear will be reduced. This will make the word that has undergone Stopword Removal unique when entering the feature selection stage. So this word that is actually not meaningful or unique will be selected as a feature and used in the classification process. As a result, the features used will not be relevant to the data class, and this will decrease the accuracy of the classification. Stemming must also not be applied, as it will change all the words in the dataset to their basic word form. In specific, it will change the meaning of a sentence in the Hadith. For example, the word "bangun dan shalatlah" will change to "bangun dan shalat" if the Stemming process was done. Eliminating the word "-lah "in "sholatlah", can change the meaning of the sentence that initially should mean suggestion, to instead turn into information.

The second testing scenario in this research is to test the term model and feature selection used on the Hadith data. The goal of this test was to observe the effect of using Mutual Information as of the feature selection in the classification process based on the term models used. There are two types of term models that were used in this study: unigram and bigram. Unigram models calculate how often a term appears in the body, while bigram is a combination of two consecutive terms. The preprocessing method used was selected based on the best results from the first testing scenario. The results of this second test are presented in Fig. 6.

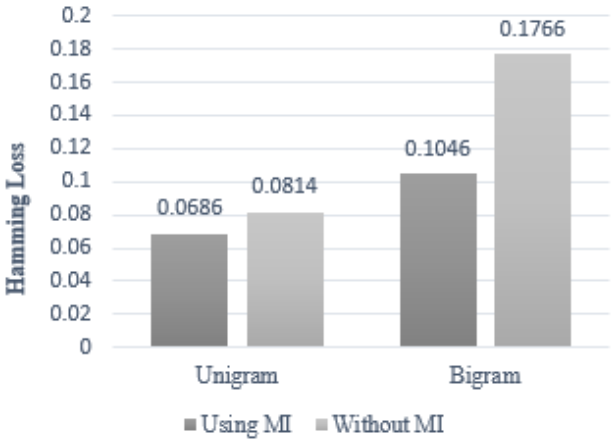

Fig. 6. Testing scenario with term model and mutual information

From the second testing scenario, the best Hamming loss values and the fastest computation time were generated using the classification with MI and the unigram model. In particular, 93.14% of multi-label data was classified correctly, and the system computing time was 228 seconds. Without feature selection, the accuracy was around 91.86%, with a time of 304 seconds. This is because when using feature selection, the feature used is a feature that genuinely describes the class of the sentence, which therefore makes the classification more effective. Meanwhile, without feature selection, all features will be used in the classification process. Consequently, the data will be increasingly spread out, making it difficult for the classification system to distinguish. This experiment in Fig. 7 can also prove this result, where the more features used, the higher the value of Hamming loss generated.

For the unigram and bigram models, the unigram model proved better because the MI calculations were applied to each word (one word). In addition, if the word has a high MI value, it means that the word is unique and describes a

particular class. On the other hand, bigram models differ because the MI calculation model applied to a combination of two consecutive words. As a result, words that are initially not unique can be mistakenly considered as unique and have a high MI value when combined with the second word although the two words actually do not describe a class. For example, the combination of words "jika ia, ke arah, atau di, di bawah, etc." has no meaning at all. However, it will still be used as a feature selection because of its high MI value.

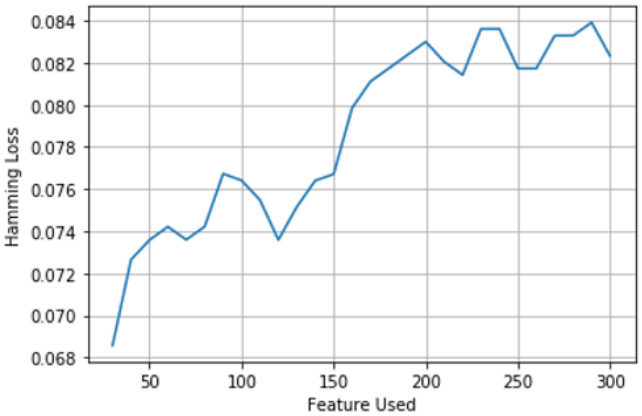

Fig. 7. Testing scenario with term model and mutual information

The third testing scenario in this research is testing the SVM kernel uses to classified the Hadith data. The goal of this test was to know which kernel is suitable for classification in data text. There are four kernels that were used in this study: linear, RBF, polynomial, and sigmoid. The preprocessing method, term model, and feature selection used ware selected based on the best results from the previous testing scenario. The results of this third test are shown in Table III.

TABLE III. HAMMING LOSS RESULT BASED ON SVM KERNEL

	SVM Kernel			
	Linear	***RBF***	***Polynomial***	***Sigmoid***
Hamming Loss	0.0686	0.0717	0,0999	0.0723

From Table III, we know that the best hamming lose value was generated using a linear kernel in SVM classification. Mostly test classification problems are linearly separable; it means that the data will be divided into positive and negative parts. In-text classification, both the numbers of document are used, and the features are large. This problem is very suitable to be solved with linear kernel SVM because the linear kernel works well in linearly separable data, and the higher dimensional space of data does not really improve the performance. So, the Hadith data will be easier to separate with a straight line and provide high classification accuracy.

The fourth testing scenario is used to compare the results of text classification using SVM against the NB, KNN, and NN methods to get the best value of Hamming loss. In SVM classification, the kernel used is the linear kernel. It is because the linear kernel produced better classification accuracy than when using the RBF, polynomial, or sigmoid kernels. Meanwhile, for the NB, KNN, and NN classification methods, the parameters used were the default parameters of the library. In this research, the preprocessing step did not use Stopword Removal or Stemming. The term model used was the unigram, and Mutual Information was used as feature selection. The results of the third test are outlined in Fig. 8.

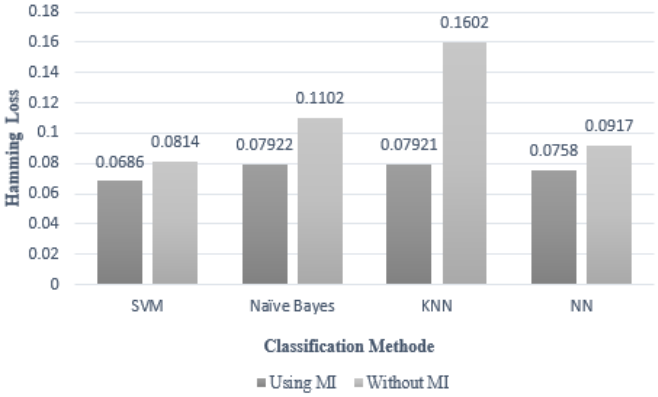

Fig. 7. Comparing SVM with Another Classification Methods

The best test results produced by the classification system using the SVM method yielded 93.14% correctly classified multi-label data. It was followed by NN, with 92.42% correctly classified multi-label data, KNN with 92.079%, and NB with 92.078%. In this research, each Hadith data was multi-dimensional, where the number of data dimensions depends on the number of features used. Although the data was multi-dimensional, the SVM method still performed well. It is because the method looks for the optimal separator between the only positive and the nearest negative data, and is open to all data. Then, for data that is far from the separator, the class will be determined based on where it is located; if it is on the positive side, then, the data has a positive class and vice versa. The SVM method also uses features, which will be used to weight each data. The more features that are in a dataset, the more the weight it will have, so the data will tend to be either on the positive or the negative side. The NN method is a self-learning method where the more training data used, the easier the system will be able to recognize the class from the test data based on the learning process carried out by the system. In this study, the number of training data for each fold was only around 957 or 958 data points, and this likely made the learning process algorithm less than optimal. As a result, the prediction using NN produced an accuracy lower than when using the SVM method. In the KNN method, the large number of data dimensions created a problem. It is because KNN did not study the weight of the data and did not know which features were most influential for class determination. This algorithm only searched for as much of the training data that was the closest to the test data. For example, k = 5, then, from the five closest training data, the number of positive and negative classes was calculated. The class of the test data was determined from the results of the highest number class of five data. If it was the most positive class, then, the value of the test data was positive and vice versa. Meanwhile, the Naïve-Bayes method is a class

determination based on the calculation of the probability of occurrence of the word only. This method did not pay attention to the interactions between features and classes, so the class representation features did not have any effect. If the probability of a positive class is smaller than the probability of a negative class in the data, then the data is automatically categorized into the negative class.

V. Conclusion

Based on several test scenarios in this study, the multi-label classification of Bukhari Hadith data using the SVM method proved to provide more accurate predictive results than other methods such as NB, KNN, and NN, especially for amounts of data that are not too much. Using Mutual Information as a feature selection also proved to increase the number of correctly classified multi-label data from 91.86% to 93.14%. Mutual Information selected the features that genuinely represent the class of a Hadith, while features that do not describe the class of Hadith were ignored. It certainly simplified the system processing and increased its performance. Therefore, reducing the system computing time. Stemming should not be used in the preprocessing stage, especially for the data used in this study, as Stemming can change the meaning of a sentence. The Stopword Removal process and usage of the bigram model must also be avoided when using MI as a feature selection. It is because Stopword Removal will reduce the number of words that are not considered necessary in a sentence. Meanwhile, in the calculation of MI, a word that has a small amount instead of having a high MI value could still be selected because it is deemed unique and is chosen to describe a class. So even words that are considered useless will become a feature, hence, decreasing the accuracy of the classification. Similar to Stopword Removal, the use of the bigram model can initially make non-unique words unique and attribute a high MI value to them. It is because the combination of two words that has no meaning at all can make the word unique, causing it to be used to describe a class, even though the word is not meaningful or unique.

Some suggestions that can be made for further research include the addition of the number of Hadith data, as the number of Bukhari Hadith consists of more than 7,000 Hadith. Hadith data must also be validated by Hadith experts when determining the label of advice, prohibitions, and information. There could be plenty of mistakes in the labeling process caused by the limited knowledge of the labeler. Besides, it tries to develop rule-based feature extraction on the system that was built. The aim is to obtain features that are relevant but not selected as the main features of the feature selection process. The rule to determine the features are made manually within putting in the form of *taggers* of each word in the Hadith utilizing *post tagger* on Natural Language Processing (NLP). Based on this process, the features of each class of traditions will be obtained, which can be used as an additional feature in the system that is built. By adding this rule-based feature selection, I hope the accuracy of the system will increase.

References

[1] M. A. Saloot, N. Idris, R. Mahmud, S. Ja'afar, D. Thorleuchter, and A. Gani, "Hadith data mining and classification: a comparative analysis," *Artif. Intell. Rev.*, vol. 46, no. 1, pp. 113–128, 2016.

[2] A. Mahmood, H. U. Khan, F. K. Alarfaj, M. Ramzan, and M. Ilyas, "A multilingual datasets repository of the Hadith content," *Int. J. Adv. Comput. Sci. Appl.*, vol. 9, no. 2, pp. 165–172, 2018.

[3] S. Al Faraby, E. R. R. Jasin, A. Kusumaningrum, and Adiwijaya, "Classification of Hadith into positive suggestion, negative suggestion, and information," *J. Phys. Conf. Ser.*, vol. 971, no. 1, 2018.

[4] D. T. M. Syair Audi Liri Sacra, Said Al Faraby, "Classification of Suggestion , Prohibition and Information of Shahih Bukhari Hadith Using Naïve Bayes Classifier," vol. 4, no. 3, pp. 4794–4802, 2017.

[5] M. Kang, J. Ahn, and K. Lee, "Opinion mining using ensemble text hidden Markov models for text classification," *Expert Syst. Appl.*, vol. 94, pp. 218–227, 2018.

[6] S. Debaere, K. Coussement, and T. De Ruyck, "Multi-label classification of member participation in online innovation communities," *Eur. J. Oper. Res.*, vol. 270, no. 2, pp. 761–774, 2018.

[7] S. Agrawal, J. Agrawal, S. Kaur, and S. Sharma, "A comparative study of fuzzy PSO and fuzzy SVD-based RBF neural network for multi-label classification," *Neural Comput. Appl.*, vol. 29, no. 1, pp. 245–256, 2018.

[8] 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.

[9] F. S. Nurfikri, M. S. Mubarok, and Adiwijaya, "News topic classification using mutual information and Bayesian network," *2018 6th Int. Conf. Inf. Commun. Technol. ICoICT 2018*, vol. 0, no. c, pp. 162–166, 2018.

[10] A. I. Pratiwi and Adiwijaya, "On the Feature Selection and Classification Based on Information Gain for Document Sentiment Analysis," *Appl. Comput. Intell. Soft Comput.*, vol. 2018, 2018.

[11] M. Zareapoor and S. K. R, "Feature Extraction or Feature Selection for Text Classification: A Case Study on Phishing Email Detection," *Int. J. Inf. Eng. Electron. Bus.*, vol. 7, no. 2, pp. 60–65, 2015.

[12] M. Rezwanul, A. Ali, and A. Rahman, "Sentiment Analysis on Twitter Data using KNN and SVM," *Int. J. Adv. Comput. Sci. Appl.*, vol. 8, no. 6, pp. 19–25, 2017.

[13] A. K. Uysal, "An improved global feature selection

scheme for text classification," *Expert Syst. Appl.*, vol. 43, pp. 82–92, 2016.

[14] S. Maruf, K. Javed, and H. A. Babri, "Improving Text Classification Performance with Random Forests-Based Feature Selection," *Arab. J. Sci. Eng.*, vol. 41, no. 3, pp. 951–964, 2016.

[15] G. Feng, J. Guo, B. Y. Jing, and T. Sun, "Feature subset selection using naive Bayes for text classification," *Pattern Recognit. Lett.*, vol. 65, pp. 109–115, 2015.

[16] A. K. Uysal, "On Two-Stage Feature Selection Methods for Text Classification," *IEEE Access*, vol. 6, pp. 43233–43251, 2018.

[17] B. Tang, S. Kay, and H. He, "Toward Optimal Feature Selection in Naive Bayes for Text Categorization," *IEEE Trans. Knowl. Data Eng.*, vol. 28, no. 9, pp. 2508–2521, 2016.

[18] R. B. Pereira, A. Plastino, B. Zadrozny, and L. H. C. Merschmann, "Categorizing feature selection methods for multi-label classification," *Artif. Intell. Rev.*, vol. 49, no. 1, pp. 57–78, 2018.

[19] L. Wu, Y. Wang, S. Zhang, and Y. Zhang, "Fusing Gini Index and Term Frequency for Text Feature Selection," *Proc. - 2017 IEEE 3rd Int. Conf. Multimed. Big Data, BigMM 2017*, pp. 280–283, 2017.

[20] M. S. Islam, F. E. Md Jubayer, and S. I. Ahmed, "A support vector machine mixed with TF-IDF algorithm to categorize Bengali document," *ECCE 2017 - Int. Conf. Electr. Comput. Commun. Eng.*, pp. 191–196, 2017.

[21] T. L. Pao, Y. Te Chen, J. H. Yeh, and P. J. Li, "Mandarin emotional speech recognition based on SVM and NN," *Proc. - Int. Conf. Pattern Recognit.*, vol. 1, pp. 1096–1099, 2006.

[22] Z. Liu, X. Lv, K. Liu, and S. Shi, "Study on SVM compared with the other text classification methods,"

Fig. 6. Testing scenario on the number of features used

2nd Int. Work. Educ. Technol. Comput. Sci. ETCS 2010, vol. 1, pp. 219–222, 2010.

[23] D. Fu, B. Zhou, and J. Hu, "Improving SVM based multi-label classification by using label relationship," *Proc. Int. Jt. Conf. Neural Networks*, vol. 2015-Septe, 2015.

[24] M. Nørgaard *et al.*, "Optimization of preprocessing strategies in Positron Emission Tomography (PET) neuroimaging: A [11C]DASB PET study," *Neuroimage*, vol. 199, pp. 466–479, 2019.

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

[26] A. Shiri, "Introduction to Modern Information Retrieval (2nd edition)," *Libr. Rev.*, vol. 53, no. 9, pp. 462–463, 2004.

[27] A. I. Kadhim, "Term Weighting for Feature Extraction on Twitter: A Comparison between BM25 and TF-IDF," *2019 Int. Conf. Adv. Sci. Eng. ICOASE 2019*, pp. 124–128, 2019.

[28] Y. Cong, Y. B. Chan, and M. A. Ragan, "A novel alignment-free method for detection of lateral genetic transfer based on TF-IDF," *Sci. Rep.*, vol. 6, pp. 1–13, 2016.

[29] M. Goudjil, M. Koudil, M. Bedda, and N. Ghoggali, "A Novel Active Learning Method Using SVM for Text Classification," *Int. J. Autom. Comput.*, vol. 15, no. 3, pp. 290–298, 2018.

[30] T. Joachims, "Text Categorization with SVM: Learning with Many Relevant Features," pp. 2–7.

[31] M. N. Saad, Z. Muda, N. Sahari, and H. Abd, "Multiclass classification for chest x-ray images based on lesion location in lung zones," *Adv. Sci. Lett.*, vol. 9, no. 1, pp. 19–23, 2016.

[32] E. A. Roxas, R. R. P. Vicerra, L. A. Gan Lim, E. P. Dadios, and A. A. Bandala, "SVM compound kernel functions for vehicle target classification," *J. Adv. Comput. Intell. Intell. Informatics*, vol. 22, no. 5, pp. 654–659, 2018.