**Multi-Label Topic Classification for Hadith of Bukhari in Indonesian Translation using Information Gain and Backpropagation Neural Network**

Muhammad Yuslan Abu Bakar, Adiwijaya, Said Al Faraby

Dept. of Informatics, Telkom University, Indonesia

**ABSTRACT**

Hadith is the second source of law and guidance for Moslems after the Qur'an and there are so many hadiths that have been narrated by the experts of hadith so far. This research builds a system that can classify shahih hadith of Bukhari in Indonesian language. This topic is raised to meet the needs of importance for Moslems so they know about the suggestions and restrictions contained in a hadith. Backpropagation neural network is chosen because this method can do classification with a huge number of varied features. Assisted with information gain as a feature selection in order to select influential features for each class label on multi-label data and single label which has never been done before. The results show that 88.42% of multi-label hadith data can be correctly classified. Meanwhile, 65.275% of the single label hadith data can be classified correctly using information gain as a feature selection.

**INTRODUCTION**

Hadith is the second source of law after the Qur'an as a guide for the life of Moslems (Aldhlan & Zeki, 2012). There are so many hadiths narrated by experts of hadith and have been grouped into several categories to make it easier for others to understand. One of the hadith experts who has narrated many hadiths of the Prophet Muhammad is Bukhari.

Hadith is used to help Muslims especially in understanding the sciences of Islam. There are so many hadiths that encourage us to do good. There is also a prohibition in the matter of sinning to Allah SWT. In order to know that, a system is needed to identify whether the hadith is recommended, prohibited or informed. From that point, the question arises as to how a translated hadith in Indonesian language can be classified into a more specific form of hadith of suggestion, hadith of prohibition and hadith of information. A hadith may be one of the three types of classes but it does not rule out that it is a combination of all three.

But there is currently no research that has classified the hadith into several classification groups or commonly referred to as multi-label classification. This is necessary given that a hadith contains not only information but a special message to be conveyed as a form of advice and prohibitions on a particular scientific topic.

On that basis, the authors will focus the research in making the classification model based on the hadith of Bukhari as much as 1064 data hadith into the form of multi-label classification. These hadiths have been labeled previously for the purpose of training the system. Information Gain is selected as a feature selection because it works by assigning a value to each feature and only selected features correlated to the appropriate class. The method that will be used is Backpropagation Neural Network to solve the problems that have been described previously. The Backpropagation Neural Network method was chosen based on research previously conducted by Min-Ling Zhang and Zhi-Hua Zhou on multi-label Neural Networks with Applications to Functional Genomics and Text Categorization (Zhang & Zhou, 2006). The results obtained from this research indicate that Backpropagation Neural Network method can do better multi-label text classification than some other methods. Features that have been obtained in the selection process and feature extraction, will be used as the input on the Backpropagation Neural Network classification system. Meanwhile, the results obtained from the classification process is the class label of each data.

Some of the problems restricted to this research are the use of a dataset of 1064 Indonesian language translations of hadiths with three labels of advice, prohibitions, and information. The proportion of data for each class is also uneven, as evidenced by the presence of a class that has a total of about 75% of the data. The difficulty to find data with 3 types of labels is also one of the obstacles in this research.

An evaluation method is needed to measure how well is the system built and applied to this research. Hamming Loss is chosen as an evaluation metric to measure the performance of the multi-label classification results that have been obtained. Hamming Loss is one of the commonly used evaluation metrics for multi-label data testing by calculating the error rate of the classification results.

The next sections in this journal are as follows. In the second part, there are related studies that describe previous research related to the current research and some literatures related research. In the third section, it will explain about the system to be built on this research, which is the classification of Bukhari hadith translation in Indonesian language using the method of Information Gain and Backpropagation Neural Network. Then, the test results and the analysis of results that have been obtained will be presented in the fourth section. Furthermore, in the fifth section, it will explain about the conclusions obtained from this research.

**RELATED RESEARCHS**

Hadith can be classified into many special groups. As explained by (Aldhlan & Zeki, 2012), hadith can be categorized as saheeh, maudo ', dhaif and hasan. Hadiths can also be categorized as hadith of suggestions, hadith of restrictions and hadith of information as described by (Jasin & Faraby, 2017) and (Kusumaningrum & Faraby, 2017). The classification of the hadith into several groups is necessary to obtain more specific information from the hadith. By knowing the group of a hadith, it is hoped that the search for information related to the hadith can be easily done.

The research related to the classification of hadiths in earlier Indonesian translations is to classify whether they are strong (saheeh) or weak hadiths. So far, the classification of hadiths with the class of hadith of information, hadith of restrictions and hadith of suggestions have been made on research (Jasin, 2017; Kusumaningrum, 2017). The research classified only into one class of three existing classes or in other words the classification model that was built is a multiclass classification. As for most similar studies that have previously been done are still in Arabic form.

(Aldhlan & Zeki, 2012) has undertaken a mechanism to improve the performance of the classification of hadiths. The method used is the Decision Tree because according to him, it is the right approach for the classification of hadith due to the ease of induction rules and interpretation results. This method is also used because it has the ability to overcome the lost value but the effort required to achieve that is considered as its weakness. Ignoring this lost value can lead to critical decisions such as the misclassification of hadith. To overcome this, they use a mechanism called Missing Data Detector (MDD).

While (Al-Kabi, 2005) used the TF-IDF method with the help of Microsoft Visual Basic Programming Language. The reason for using this programming language is because it supports Arabic writing / text and data in the form of documents / files. The classification is done into 8 classes form, which are: "Knowledge", "Praying", "Eclipse", "Call to Prayer", "Faith", "Good Manners", "Fasting" and "Almsgiving". The hadith used in this classification is Bukhari hadith with an average accuracy of 83.2%. The result of this accuracy depends entirely on the process of stopwords and stemming.

(Jasin, 2017) classified the hadith of suggestion, hadith prohibition and hadith information on Hadiths shahih Bukhari based on Unigram model using Artificial Neural Network (ANN). It also uses the help of the TF-IDF feature extraction method to get the values on every word. The Performance results obtained with F1-Score of 85%. Similar research has also been done by (Kusumaningrum, 2017) using the help of Support Vector Machine method. K-Fold is used to divide data into training data and test data. The average performance obtained using this method is 88%.

For multi-label cases in text, there have been several studies related to it. (Zhang & Zhou, 2006) have conducted multi-label related research on text categorization. The method used is a neural network called BP-MLL (Backpropagation for Multi-Label Learning). This method derives from popular backpropagation algorithms using error functions that capture multi-label learning characteristics, i.e. the label of a feature must be ranked higher than that not included in the instance. They used the method because the result of accuracy is higher than other multi-label learning methods such as BoosTexter, ADTBoost.MH, Rank-SVM and BasicBP.

(Huang, 2018) using JSFC method utilizes Fisher Discriminant-Based on feature selection to minimize the distance between the same classes and maximize the distance between different classes so that the data can be further separated. Given these methods, the mutually influential features for each class label can be easily selected and the learning for the classification model can be built more effectively. But the lack of this method is the difficulty of configuring the parameters in the feature selection so that it will require a lot of time for data with a large number of features. (Liu & Zhang, 2017) constructed the method that can reduce the overfitting and noisy problems in the data by adopting PLS and l1-norm regression to model the relationships between data. The difficulty faced is the difficulty of finding a way to determine the value of the optimal variable that will give a big influence on the performance.

(Soleimani, 2017) modeled several classes together so that there’s a possibility of interdependence between different classes and using Markov Chain Monte Carlo method can overcome the association between batch sets and fluctuating features. Given this, the built model able to study the topic class only with a small number of labeled training documents.

(Tsoumakas & Katakis, 2007) grouped the multi-label classification method into two main categories, namely the problem transformation method which is the methods that convert the multi-label classification problem into one or more single label or regression classifications, since both have a large bibliography in learning algorithms and algorithmic adaptation methods which are methods that extend specifically to algorithm learning to handle multi-label data directly. They implemented PT3, PT4 and PT6 methods on "genbase", "yeast" and "scene" data with classifier *k*NN, C4.5, Naïve Bayes and SMO.

**METHODS**

This research builds 2 (two) system that can classify multi-label data and single label. Both systems use a dataset of Bukhari shahih hadith translations in Indonesian which have been labeled previously. The first system consists of 1064 multi-label data with 289 of them having more than one label. The second system consists of 455 single label data. The class label on multi-label data consists of 3 (three) classes, the recommended class, the prohibition class and the information class whereas in the single label data consists of 5 (five) classes, namely class 1 is about "faith", class 2 is about "science", class 3 is about "wudhu", class 4 is about "salat", and class 5 is about "prayer times". As for each multi-label hadith data can be either one of the three classes or a combination of several classes. The data representation will be shown as in Table 1 and Table 2.

**Table 1: Multi-Label Data Representation**

|  |  |  |  |
| --- | --- | --- | --- |
| **Data** | **Advice** | **Prohobition** | **Information** |
| Janganlah kalian berdusta terhadapku (atas namaku), karena barangsiapa berdusta atas namaku dia akan masuk neraka. | 0 | 1 | 1 |
| Kami pernah shalat maghrib Bersama Nabi ketika matahari sudah tenggelam tidak terlihat. | 0 | 0 | 1 |
| Apabila seorang dari kalian memperbaiki keIslamannya maka dari setiap kebaikan akan ditulis baginya sepuluh (kebaikan) yang serupa hingga tujuh ratus tingkatan, dan setiap satu kejelekan yang dikerjakan akan ditulis satu kejelekan saja yang serupa dengannya. | 1 | 0 | 1 |

**Table 2: Single Label Data Representation**

|  |  |
| --- | --- |
| **Data** | **Book Class** |
| Janganlah kalian berdusta terhadapku (atas namaku), karena barangsiapa berdusta atas namaku dia akan masuk neraka. | 0 1 0 0 0 |
| Kami pernah shalat maghrib Bersama Nabi ketika matahari sudah tenggelam tidak terlihat. | 0 0 0 0 1 |
| Apabila seorang dari kalian memperbaiki keIslamannya maka dari setiap kebaikan akan ditulis baginya sepuluh (kebaikan) yang serupa hingga tujuh ratus tingkatan, dan setiap satu kejelekan yang dikerjakan akan ditulis satu kejelekan saja yang serupa dengannya. | 1 0 0 0 0 |

In the course of this research, the system will be able to classify translated hadith translations into 3 (three) multi-label data classes and 5 (five) single-label data classes as defined previously. The system will be divided into several main processes, namely preprocessing process that will generate clean data, followed by feature selection process which will sort the features, feature extraction process that will give value for each feature and the last is the classification process that will determine the class of each hadith. In general, the system will be shown in Figure 1.

A close up of a piece of paper

Description generated with high confidence

**Figure 1: Flowchart system**

A screenshot of a social media post

Description generated with very high confidenceA preprocessing stage is done to remove the noise contained in the dataset. The complete process of preprocessing is illustrated in Figure 2.

**Figure 2: Preprocessing process**

1. Case folding, is the process of turning each word into lowercase and to uniform every word. Case folding in this research uses the help of function str.lower () in the Python programming language.
2. Cleaning, is a process to remove punctuation and or symbols that exist in the data. This process requires the help of existing libraries in the Python programming language that is Regular Expression.
3. Tokenization, is a process that will cut off any word contained in each input data. The tokenization process in this research uses the function str.split () that exists in the Python programming language.
4. Stopword removal, is a process to eliminate words that are considered less influential. Usually the characteristic of these words is the frequency of its occurrence that is quite frequent compared with other words.

After obtaining clean data through the preprocessing process, the next step is to select and determine the features needed for the classification process. This stage is also called the feature selection. This step will determine what features / words that are relevant to each class. The feature selection used is information gain where the selection of this feature will calculate the value of each word in a set of words. Words with high information gain value will be the main feature for each data in the classification process. The following equation describes the formula to calculate the value of the information gain.

where is the entropy of the class and is the conditional entrophy of each class of attribute / word (Pratiwi, 2018).

The greater the value, the more it signifies that the word affects the existing classes. After that, words with a value above the standard (threshold) will be selected as the main feature in the classification. The threshold can be determined by looking at the plot of each word obtained.

Next, it will be performed the feature extraction stage to get the value of each feature. Term Frequency-Inverse Document Frequency or commonly referred to as TF-IDF is one of the popular feature extraction methods used especially in text classification. TF-IDF represents a matrix in which each row of the matrix is the data and the existing column as a word or feature (term) (Zareapoor, 2015). This matrix will be filled by a value or weight of each word against a document. The weights that have been obtained will be the input for the neural network classification system. The TF (Term Frequency) is the number of occurrences of each word in each document and the IDF (Inverse Document Frequency) is the number of related documents containing a particular word (Kusumaningrum, 2017). The weight calculation on TF-IDF is found as in the following equation.

,

where:

the weight of the word against the document

number of occurrences of the word in document

number of documents

number of occurrences of words in document

This research used k-fold cross validation for testing. In k-fold cross validation, the data set is randomly partitioned into k-parts of approximately the same size, . Each k on one loop will be used as test data and the rest will be used as training data for the built classifier. The estimated error rate is calculated from the proportion of errors from all parts of the sample (Yang, 2011). In this research we use k = 5.

In this research, the built classifier is Backpropagation Neural Network. This classifier works by performing two calculation phases which is a forward calculation that will calculate the error value between the system output value with the value it should and the countdown to fix the weights based on the error value (Suyanto, 2011). Train data is used in the learning process to obtain parameter values that produce the best results on the backpropagation neural network system. The best results are obtained when the value of error meets the standard or in other words the error value is very small. Figure 3 is a backpropagation neural network architecture built on this research.

|  |  |
| --- | --- |
| A close up of a map  Description generated with very high confidence  **(a). Multi-Label** | A close up of a map  Description generated with very high confidence  **(b). Single Label** |

**Figure 3: Backpropagation Neural Network Architecture**

Figure 3(a) is a backpropagation neural network architecture for multi-label data while Figure 3(b) is a backpropagation neural network architecture for single label data. Each data will be trained by both systems and then it will be determined which data goes into which class.

One way to know whether the system built is good or not is to measure the performance of the resulting classification. In multi-label cases, hamming loss is one of the most commonly used performance metrics. Hamming loss will calculate the error value of the classification result. The resulting performance will be perfect if the value of . The smaller the value of generated, the better the performance of the system. The following equation will show the calculation of .

where is a lot of data classification and is the number of class labels. While is the number of misclassification that occurred.

**EVALUATION**

To see the results of the evaluation on Backpropagation Neural Network that has been built, this research conducted several test scenarios. The first test was conducted to determine the effect of stemming process on Bukhari hadith data of Indonesian translation of hamming loss and F1-Score value as well as computation time. To perform the stemming process, we use the help of library Sastrawi.Stemmer.StemmerFactory (). The next test is done to see the effect of threshold value on the information gain to the value of hamming loss and F1-Score obtained. The last test is done on changes in the value of learning rate found in the learning system to the results of hamming loss and F1-Score obtained in multi-label and single label data. Here are some test scenarios for multi-label and single label data.

1. Testing Scenario with Stemming Process

|  |  |
| --- | --- |
| A screenshot of a cell phone  Description generated with very high confidence | A screenshot of a cell phone  Description generated with very high confidence |

**Figure 4: Testing scenario with stemming process on multi-label data**

|  |  |
| --- | --- |
| A screenshot of a cell phone  Description generated with very high confidence | A screenshot of a cell phone  Description generated with very high confidence |

**Figure 5: Testing scenario with stemming process on single label data**

In Figure 4 a test scenario is conducted to see if there is an effect of stemming and non-stemming processes on the results of hamming loss and computation time obtained in multi-label data. The result of hamming loss shows that testing without stemming process get better result than using stemming process that is equal to 0.119298246 or it can be said that about 88.07% data are classified correctly. As for the computation time, testing without stemming process also get better results compared to using the stemming process that is 1493.53 seconds. This is because in multi-label data, every word cannot be generalized by using the stemming process because it can eliminate the special characteristics of Bukhari's hadith-labeled multi-label data as indicated on the label, the word "-lah" indicates a suggested command on the word "tinggalkan-lah". So, this word will have different meaning if it is generalized by using stemming process.

As for the test on the single label data, it obtained F1-Score results of 61.75% of data classified correctly and it has better computing time that is 4149.24 seconds using the stemming process, compared to the test that didn’t use the stemming process as shown in Figure 5. It is because affixes to a word can be generalized to one word using the stemming process as in the data with the label of salat, the word "shalat-lah" and "menyalatkan" can be generalized to the word "shalat".

1. Testing Scenario with Information Gain Threshold

|  |  |
| --- | --- |
| A screenshot of a cell phone  Description generated with very high confidence  **(a). Multi-Label** | A screenshot of a cell phone  Description generated with very high confidence  **(b). Single Label** |

**Figure 6: Testing Scenario with Information Gain Threshold**

Figure 6 shows the effect of threshold information gain value against multi-label and single label data. Figure 6 (a) shows that the test scenario in multi-label data using threshold starts from 0.6 to 0.90 with a change in value of 0.05. The result shows that the threshold value of 0.75 get better hamming loss value compared to the others, which is equal to 0.1158 or 88.42% data can be classified correctly compared with other threshold values. This is because the words or features that affect each label are on the information gain value above 0.75. Meanwhile, for words that have an information gain value below 0.75, it does not really affect the class determination of each data.

While for single label data indicates that the obtained F1-Score result is better when using threshold equal to 0.6 compared with using another threshold with F1-Score value equal to 65.275% as shown in Figure 6 (b). This is because the words or features contained on each label have the same value of information gain and are equally influential for each label.

1. Testing Scenario with Learning Rate Value

|  |  |
| --- | --- |
| A screenshot of a cell phone  Description generated with very high confidence  **(a). Multi-Label** | A screenshot of a cell phone  Description generated with very high confidence  **(b). Single Label** |

**Figure 7: Testing Scenario with Learning Rate Value**

Figure 7 shows the effect of learning rate on multi-label data and single label data with learning rate value starting from 0.01 to 0.1 with a change of 0.01. In Figure 7 (a) it can be seen that the test with the value of learning rate 0.02 and 0.1 get the best hamming loss of 0.1193 or 88.07% of data can be classified correctly.

As for the single label data in Figure 7 (b), it shows that the value of learning rate of 0.01 in the classifier training process obtained F1-Score results which are better than the value of other learning rate, with F1-Score result of 61.099% data classified correctly. This is because the value of a large learning rate will result in a learning process that is too fast so it doesn’t get the value of changes in the optimum weight.

**CONCLUSIONS AND FUTURE WORK**

From the results of several test scenarios that have been done in this research, the effect of threshold value on different information gain for multi-label data and single label data can be concluded. In multi-label data, the resulting performance is better by taking some features that actually have high information gain value or above 0.75. While for the single label data, the features that affect each label cannot produce good performance results on the classifier if it only takes features with information gain values above 0.75. The effect of using information gain works effectively on multi-label data.

As for the use of the stemming process, the best performance results are obtained on single label data and it does not work on multi-label data. This is because in multi-label data, the stemming process will remove the characteristic feature for each data label. While for single label data, word generalization with stemming process can influence the result of a better system performance.

Some suggestions that can be applied for the development of further research is by paying more attention to the data label, especially on the single label data which currently, there are many labels that don’t suit the data correctly. Next is to try to analyze the meaning of words in multi-label data so that the results obtained can be better.

**REFERENCES**

Aldhlan, K. A., Zeki, A. M., Zeki, A. M., & Alreshidi, H. A. (2012, November). Novel mechanism to improve hadith classifier performance. In *Advanced Computer Science Applications and Technologies (ACSAT), 2012 International Conference on* (pp. 512-517). IEEE.

Al-Kabi, M. N., Kanaan, G., Al-Shalabi, R., Al-Sinjilawi, S. I., & Al-Mustafa, R. S. (2005). Al-Hadith text classifier. *Journal of Applied Sciences*, *5*(3), 584-587.

Huang, J., Li, G., Huang, Q., & Wu, X. (2018). Joint feature selection and classification for multilabel learning. *IEEE transactions on cybernetics*, *48*(3), 876-889.

Jasin, E., Faraby, S., & Adiwijaya, A. (2017). Klasifikasi Anjuran, Larangan Dan Informasi Pada Hadis Sahih Al-bukhari Berdasarkan Model Unigram Menggunakan Artificial Neural Network (ann). *eProceedings of Engineering*, *4*(3).

Kusumaningrum, A., Faraby, S., & Adiwijaya, A. (2017). Klasifikasi Anjuran, Larangan, Dan Informasi Pada Hadits Shahih Bukhari Menggunakan Metode Support Vector Machine. *eProceedings of Engineering*, *4*(3).

Liu, H., Li, X., & Zhang, S. (2017). Learning instance correlation functions for multilabel classification. *IEEE transactions on cybernetics*, *47*(2), 499-510.

Pratiwi, A. I. (2018). On the Feature Selection and Classification Based on Information Gain for Document Sentiment Analysis. *Applied Computational Intelligence and Soft Computing*, *2018*.

Soleimani, H., & Miller, D. J. (2017). Semisupervised, multilabel, multi-instance learning for structured data. *Neural computation*, *29*(4), 1053-1102.

Suyanto, A. I., & Msc, A. I. (2011). Searching, Reasoning, Planning, Learning. *Informatika Bandung* (pp. 184-186). 2014.

Tsoumakas, G., & Katakis, I. (2007). Multi-label classification: An overview. *International Journal of Data Warehousing and Mining (IJDWM)*, *3*(3), 1-13.

Yang, K., Wang, H., Dai, G., Hu, S., Zhang, Y., & Xu, J. (2011, October). Determining the repeat number of cross-validation. In *Biomedical Engineering and Informatics (BMEI), 2011 4th International Conference on* (Vol. 3, pp. 1706-1710). IEEE.

Zhang, M. L., & Zhou, Z. H. (2006). Multilabel neural networks with applications to functional genomics and text categorization. *IEEE transactions on Knowledge and Data Engineering*, *18*(10), 1338-1351.

Zareapoor, M., & Seeja, K. R. (2015). Feature extraction or feature selection for text classification: A case study on phishing email detection. *International Journal of Information Engineering and Electronic Business*, *7*(2), 60.