DOCUMENT CLASSIFICATION USING EXTREME LEARNING MACHINES (ELM) METHODS IN INDONESIAN LANGUAGE

Rendi Nur Prasetyo Utomo[1], Ednawati Rainarli , S.Si., M.Si.[2]

[1,2]Teknik Informatika - Universitas Komputer Indonesia
Jl. Dipati Ukur No.112-116 Bandung
Email: rendinurprasetyo12@gmail.com[1], ednawati.rainarli@email.unikom.ac.id[2]

ABSTRACT

In this research, the document will be analyzed using the Extreme Learning Machines method (ELM). The data used in this research is an abstract of the thesis of Informatics Engineering of Unikom University, in .txt format. The data are used as input data for training and testing. This research aims to know the level of accuracy of document classification based on document content. Input data is processed through a preprocessing stage that continues with a feature selection using K-Means and synonyms dictionary. After obtaining a list of words that will be used as features, then document classification uses ELM. After that, the results will be used as input in ELM Training to create the ELM model. The ELM model produced from ELM Training is used to test abstract data on ELM Testing. Testing used 150 abstract of the thesis that consist of 100 abstract as training document and 50 abstract as test document. The first test was done to determine the classification model used. The second test is done to determine the number of clusters and parameters required in the classification. The conclusion that can be drawn is the classification of documents using the ELM method with a total of 6 clusters and 10 features resulting in a maximum accuracy of 42%. This research has not produced a high accuracy, but the K-Means process can reduce the number of inputs.

Keywords : Classification, Extreme Learning Machines (ELM), Preprocessing, K-Means, Feature selection.

1. INTRODUCTION

Today the availability of information that is growing rapidly indicates that the need to get information is getting bigger. Information needed to experience development ranging from general information to specific nature. The large amount of available document information encourages people to find ways to get the right information and documents in a short time.

Problems that arise when the amount of document data deposits becomes very large and unorganized results in certain document search processes are ineffective [1]. Therefore, a strategy for automatic grouping of documents is needed. Classification is one method that aims to define the class of an object class into one or more groups that have been previously known automatically based on the contents of the document.

Extreme Learning Machine (ELM) is one of the learning methods used for automatic classification. The main reason for the success of ELM is its ability to obtain model functions that provide better accuracy and speed and are capable of handling large amounts of data [2].

Huang [3] has proposed a new learning algorithm Single Hidden Layer Feedforward Neural Network (SLFN) called the Extreme Learning Machine (ELM). In this algorithm the input weight and hidden layer bias are randomly selected. The ELM formulation leads to a system solving of linear equations in terms of unknown weights connecting hidden layers to the output layer. The solution to this system of general linear equations is obtained using Moore-Penrose's inverse.

ELM is used to classify data in the form of text with a good level of accuracy and speed. But for the document classification process it is often found that results are not good because the number of words of each document is large and varied so it must be grouped first so that the accuracy of the proposed model is better [4]. In addition to the word grouping, character selection is also carried out in the form of a collection of representative words that will be used for input data. Therefore, this research will implement the Extreme Learning Machines (ELM) method to solve document classification problems.

2. THEORY

2.1 Preprocessing

Preprocessing is a stage that converts text into data that can be processed in the next stage. Preprocessing stages carried out in this research include filtering, folding cases, tokenizing, stopword removal, TF-IDF weight calculation and

normalization.

1. **Filtering**

Filtering is the process of removing symbols where text other than the characters "a" to "z" will be removed and otherwise it is considered a delimiter [5].

2. **Case Folding**

Case folding is the process of converting text into the same case. All letters in the text are changed to lowercase letters [5].

3. **Tokenizing**

Tokenizing is the process of cutting all words in a sentence and turning them into a collection of tokens [5].

4. **Stopword Removal**

Stopwords removal is the process of removing words that are general in nature and are considered as words that have no meaning.[5].

5. **Term Frequency – Inverse Document (TF-IDF)**

TF-IDF is the process of determining the weight of the term in a document based on the frequency of occurrence of the word. Inverse Document Frequency (IDF), is a reduction in the dominance of terms that often appear in various documents [6].

Weighting is obtained from the frequency of the number of occurrences of a word contained in a document, term frequency (TF). A word or number of terms in the document collection, inverse document frequency (IDF).

$$IDFt = \log(N/df) \quad (1)$$

$$Wdt = TF * IDFt \quad (2)$$

where:

W = document weight d to word-t
d = document-d
t = word-t
N = number of documents
df = many documents containing each word

6. **Normalization**

Normalization is performed on document feature vectors to eliminate the influence of the notion that long documents are more relevant than short documents. With this normalization it can help normalize the value limit by standardizing values into intervals of 0 to 1. So that it can be written as follows:

$$\boldsymbol{w(word_i) = \frac{w(word_i)}{\sqrt{w^2(word_1) + w^2(word_2) + \cdots + w^2(word_n)}}} \quad (3)$$

2.2 K-Means

K-Means is a method of data clustering, which is classified as an unsupervised classification method. [7]. The algorithm for K-means is as follows:

1. Determine the number of clusters
2. Determine the value of the centroid

Determining the centroid value for the initial iteration is done randomly. While determining the value of the next iteration centroid, the following formula is used.

$$\overline{v}_{ij} = \frac{1}{N_i} \sum_{k=0}^{N_i} x_{kj} , \quad (4)$$

where:

$v\bar{ij}$ = centroid /cluste-i average for variable-j
Ni = amount of data that is a member of the cluster-i
i,k = index of the cluster
j = index dari variables
xkj = data-k value in the cluster for the variable-j

3. Calculate the distance between the data and the cluster center using Euclidean Distance.

$$De = \sqrt{(x1 - y1)^2 + (x2 - y2)^2 + \ \ldots + (xi - yi)^2} \quad (5)$$

De = euclidean distance.
i = amount of data
x = document weight
y = cluster center

4. Grouping Data

Determine cluster members by calculating the minimum distance of the object.

5. Repeat (stage 2) until the cluster members do not move to another cluster.

2.3 Extreme Learning Machines (ELM)

Extreme learning machine is an artificial neural network feedforward with one hidden layer or commonly referred to as a single hidden layer feedforward neural network (SLFNs) [8].

The ELM method was created to overcome the weaknesses of feedforward artificial neural networks, especially in terms of learning speed. The ELM algorithm does not exercise input or bias weight, ELM trains to obtain the output weight by using the least-squares solution and Moore-Penrose inverse in the linear system in general. By finding nodes that provide maximum output values, and parameters such as input weight and bias are randomly selected, so ELM has a learning speed that is fast and able to produce good generalization performance.

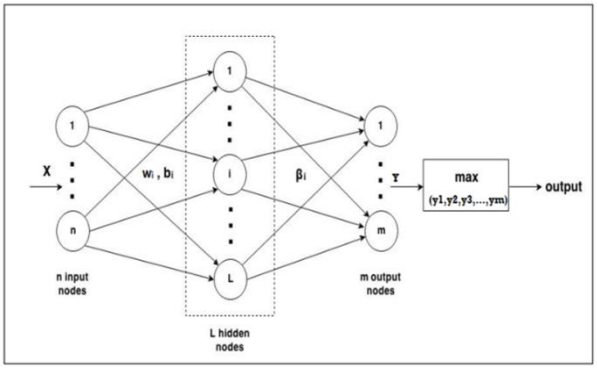

Figure 1 Architecture ELM

For samples N (x_i, y_i), where $x_i = [x_{i1}, x_{i2}, ..., x_{in}]^T \in R^n$ and $y_i = [y_{i1}, y_{i2}, ..., y_{im}]^T \in R^m$, such that $(x_i, y_i) \in R^n \times R^m$ where $(i = 1,2, ..., N)$, with L as the hidden nodes, and the activation function $g(x)$. The ELM output function for the given input x is [4]:

$$g_L(x_j) = \sum_{i=1}^{L} \beta_i g(w_i \cdot x_j + b_i) = y_j, j = 1, ..., N \quad (6)$$

Where, (w_i, b_i), $i = 1, ..., L$ with a random parameter where$w_i = [w_{i1}, w_{i2} \ ... \ w_{in}]$ is a vector weight that connects all input nodes to hidden nodes. b_i is the bias of a hidden node. $\beta = [\beta_1, ..., \beta_L]^T$ is the weight of the vector between the hidden node and the output node. $g(x)$ is the output vector that maps the n-dimension input space for the L dimensional space feature. The activation function $g(x)$ used is softsign, this function gives the output boundary between (0,1) the formula as follows:

$$g(w_i.x_j+b_i) = \frac{wi.xj+bi}{1+|wi.xj+bi|} \quad (7)$$

The format of equation (6) can be written as follows::

$$\boldsymbol{H\beta = Y} \quad (8)$$

where,

$$H = \begin{bmatrix} g(w_1.x_1 + b_1) & ... & g(w_L.x_1 + b_L) \\ g(w_1.x_2 + b_1) & ... & g(w_L.x_2 + b_L) \\ . & ... & . \\ . & ... & . \\ g(w_1.x_N + b_1) & ... & g(w_L.x_N + b_L) \end{bmatrix} \quad (9)$$

$$\beta = \begin{bmatrix} \beta_{11} & ... & \beta_{1m} \\ \beta_{21} & ... & \beta_{2m} \\ . & ... & . \\ . & ... & . \\ \beta_{L1} & ... & \beta_{Lm} \end{bmatrix} \quad Y = \begin{bmatrix} y_{11} & ... & y_{1m} \\ y_{N1} & ... & y_{2m} \\ . & ... & . \\ . & ... & . \\ y_{N1} & ... & y_{Nm} \end{bmatrix} \quad (10)$$

The smallest training error (a similar concept in SVM) and the smallest output weight norm can be achieved with ELM and can be represented as follows:

$$minimize : \ \| \mathbf{H}\boldsymbol{\beta} - \mathbf{Y} \|^2 \ and \ \| \boldsymbol{\beta} \| \quad (11)$$

The minimum minimum squares solution of the linear system above is given by:

$$\boldsymbol{\beta = H^{+}Y} \quad (12)$$

where H^+ Moore-Penrose inverse matrix H.

H (called ELM feature space, also known as hidden layer output matrix) is a weighting matrix connecting input layers and hidden layers so that the H matrix has a size of N x L. Determination of the value of the matrix elements is done randomly. Then each input value is processed in the hidden layer using the activation function and the value is collected in an H matrix with order N x L.

3. Research Methods

The research methodology will be used in this research from several stages as shown in Figure 2 below:

Figure 2 Research Methods

1) Literature Study

Literature studies in this research study theories through books, articles, journals and other materials related to methods for classifying documents and algorithms used.

2) Analysis of Algorithm Requirements

Analyzing algorithm needs starting from input (documents that contain words needed for research), processes (stages of input preprocesses that are converted into results according to the format used in the research), until the outputs produced can be implemented in the program.

3) Implementation

At this stage development is carried out from the classification of documents to Indonesian-language documents. Starting from the preprocessing stage, the selection uses the K-Means algorithm and the implementation of ELM to generate classification, non-functional requirements analysis, functional requirements analysis and system design (table structure, interface design and semantic networks).

4) Testing

Testing is done by comparing the results of the classification of the system with the initial classification (manual) so that the results of the comparison. The results of the comparison will be an evaluation of whether the system is running well or

not.

5) Conclusions

At this stage conclusions are obtained from the results of the classification produced by the system for Indonesian-language documents.

4. Results and Discussion

1. Results

Data input for training is abstract data (title and content) thesis with extension .txt. The amount of data for training as many as 100 documents is divided into 5 categories, which are A, B, C, D, and E which are equal in number. Training abstract data is made preprocessing for each document in the training data so that the TF-IDF is weighted. Preprocessing is done by filtering, folding case, tokenizing, stopword removal, TF-IDF, and normalization.

Feature Selection uses the K-Means algorithm. From the results of word weighting with TF-IDF on each document so that the words in all documents are used as K-Means clustering input data.

After getting the data cluster results. Next is to determine the word features that will be used for the classification process. Use the following method [4]:

1) Select words randomly from the list of words from the cluster. Prepare a list of initial synonyms for words using dictionary synonyms. Example the word "metode".
2) Check the word "method" both in the cluster word list and in the list of synonyms. If the word "method" is found then create a new synonym list (NSL) and add all the words synonymous to NSL and delete from the list of words from the cluster. In this way NSL can be created.
3) Repeat steps 1 and 2 to the list of words from the cluster examines all. So that it generates an NSL list from the cluster word list.
4) Repeat steps 1 through step 4 for each cluster. Select words are known as representative words from each cluster from the NSL list which has the largest TF-IDF value and in the end, combine all the features of each cluster as feature selection for input data.

All data from the cluster and synonym are processed by the system, then the weights w and bias b are determined by random numbers located in the range (0.1) so that we can obtain H and Y (8) then the weight of β is calculated using formula (12). After training, the classification model obtained is a synonym list, w, b and β values for use in the ELM testing process. Perform testing using data testing, then calculate output (6).

Tests are carried out on thesis abstract documents totaling 50 categorized A, B, C, D and E. Each class has 10 pieces of data. Accuracy testing is done in two scenarios, the first test is done to determine the classification model used. The second test is done to determine the number of clusters and the value of the parameters needed in the classification. Whereas to see performance, the classification results from the ELM method will be compared with several numbers of clusters and the parameters used in testing.

Table 1 Comparison of Performance

Cluster	Jumlah Fitur per *Cluster*	Akurasi (%)
C5	5	26
	10	30
C6	5	38
	10	42
C7	5	30
	10	36

Table 1 is a comparison of the test results from the ELM classification method. The number of clusters and features used gives different results. This can be seen in the accuracy obtained. Maximum accuracy with the greatest accuracy 42% of the testing process use 6 clusters and ELM architecture consisting of 10 nodes in the input layer, 5 nodes in the hidden layer, and 5 nodes in the output layer. The maximum accuracy of ELM classification can be seen in table 2.

Table 2 Maximum Test Results

No	Class	True	False	Total
1	A	7	3	10
2	B	4	6	10
3	C	1	9	10
4	D	1	9	10
5	E	8	2	10
Total		21	29	50
Accuracy		42%	58%	100%

$$\text{Accuracy} = \frac{True\ Data}{Total\ Data} \text{ x } 100\%$$
$$= \frac{21}{50} \text{ x } 100\%$$
$$= 42\%$$

2. Discussion

Based on the results obtained in Table 1, it can be seen that the use of cluster numbers and feature selection in ELM will make the accuracy values obtained differ. ELM, which has the advantage of having good performance for high-dimensional data, in this case it was not better than SVM [9]. The

accuracy of different ELM values in each classification model has a maximum accuracy of 42%. The classification model used determines the accuracy of the data tested. The use of clustering processes does not significantly improve performance. This research has not produced a high accuracy value, but the K-Means process can reduce the number of inputs used. The results of this test are different from those obtained from [10] K-Means affect the optimization of the accuracy level of the SVM model in classifying.

5. Conclusion

Based on the results of the implementation and testing, it can be concluded the thesis report classification with the Extreme Learning Machines method with feature selection resulting in a maximum accuracy of 42%. From these results it can be seen that the accuracy of ELM is still low compared to SVM. The determination of the model from ELM also affects the accuracy of the system. The selection of parameters that are less precise in the hidden layer (w and b) will give different recognition results because the parameter values are randomly determined.

For further research, another preprocessing process such as stemming is needed with the hope of increasing the accuracy of the data used.. Extending the scope of research by adding the type of object to be recognized. Add the number of datasets for training and testing data.

REFERENCES

[1] Herny Februariyanti and Eri Zuliarso, "Jurnal Teknologi Informasi DINAMIK," *Klasifikasi Dokumen Berita Teks Bahasa Indonesia menggunakan Ontologi*, vol. 17, no. 1, pp. 14-23, Januari 2012.

[2] R Singh and S Balasundaram, "Application of Extreme Learning Machine Method for Time Series Analysis," *International Journal of Electrical and Computer Engineering*, 2007.

[3] G. B. Huang, Q. Y. Zhu, and C. K. Siew, "Extreme Learning Machine:Theory and Applications," no. 70, pp. 449-465, 2006.

[4] Rajendra Kumar Roul, Shashank Gugnani, and Shan Mit Kalpeshbhai, "Clustering Based Feature Selection using Extreme Learning Machines for Text Classification," *International Symposium on Parallel and Distributed Computing (ISPDC)* , 2016.

[5] Budiawan Wijakso, Lailil Muflikhah, and Achmad Ridok, "Klasifikasi Jurnal Ilmiah Berbahasa Inggris Berdasarkan Abstrak Menggunakan Algoritma ID3," 2013.

[6] Eko Budi Setiawan and Aji Teja Hartanto, "Implementasi Metode Maximum Marginal Relevance (MMR) dan Algoritma Steiner Tree untuk Menentukan Storyline Dokumen Berita," *ULTIMATICS*, vol. 8, no. 1, 2016.

[7] Nur Wakhidah, "Clustering Menggunakan K-Means Algorithm," *Jurnal Transformatika*, vol. 8 (1), pp. 33-39, 2010.

[8] Z. L Sun, T. M Choi, K. F Au, and Y Yu, "Sales Forecasting using Extreme Learning Machines with Application in Fashion Retailling," *Elsevier Decision Support Systems*, no. 46, pp. 411-419, 2008.

[9] Nelly Indriani, Ednawati Rainarli, and Kania Evita Dewi, "Peringkasan dan Support Vector Machine pada Klasifikasi Dokumen," *JURNAL INFOTEL Informatika - Telekomunikasi - Elektronika*, vol. 9, no. 4, November 2017.

[10] Oman Somantri, Slamet Wijono, and Dairoh , "Metode K-Means untuk Optimasi Klasifikasi Tema Tugas Akhir Mahasiswa Menggunakan Support Vector Machine (SVM)," *Scientific Journal of Informatics*, vol. 3, no. 1, Mei 2016.