KLASIFIKASI DOKUMEN MENGGUNAKAN METODE *EXTREME LEARNING MACHINES (ELM)* PADA DOKUMEN BERBAHASA INDONESIA

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]

ABSTRAK

Dalam penelitian ini akan dilakukan klasifikasi dokumen menggunakan metode Extreme Learning Machines (ELM). Data yang digunakan pada penelitian ini merupakan abstrak dari skripsi Prodi Teknik Informatika Unikom, dalam format .txt. Data tersebut dijadikan data masukan pelatihan dan pengujian. Penelitian ini bertujuan mengetahui tingkat akurasi dari klasifikasi dokumen berdasarkan isi dokumen. Data masukan diproses melalui tahap preprocessing yang dilanjutkan dengan seleksi fitur menggunakan K-Means dan kamus sinonim. List kata yang diperoleh digunakan sebagai fitur maka dilakukan klasifikasi dokumen menggunakan ELM. Hasil fitur akan digunakan sebagai inputan dalam ELM Training untuk membuat model ELM. Model ELM yang dihasilkan dari ELM Training selanjutnya digunakan untuk menguji data abstrak pada ELM Testing. Pengujian menggunakan 150 abstrak laporan skripsi prodi teknik informatika dimana 100 dokumen digunakan sebagai data latih dan 50 dokumen digunakan sebagai data uji. Pengujian pertama dilakukan untuk menentukan model klasifikasi yang dipakai. Pengujian kedua dilakukan untuk menentukan jumlah cluster dan nilai parameter yang diperlukan dalam klasifikasi. Kesimpulan yang dapat diambil dalah klasifikasi dokumen menggunakan metode ELM dengan jumlah 6 cluster dan 10 fitur menghasilkan akurasi maksimum sebesar 42%. Penelitian ini belum menghasilkan nilai akurasi yang tinggi tetapi proses K-Means dapat mengurangi jumlah inputan yang dipakai.

Kata kunci : Klasifikasi, *Extreme Learning Machines* (ELM), *Preprocessing*, *K-Means*, Seleksi fitur.

1. PENDAHULUAN

Dewasa ini ketersediaan informasi yang semakin berkembang pesat menandakan bahwa kebutuhan untuk mendapatkan informasi semakin besar. Baik informasi yang bersifat umum hingga yang bersifat khusus. Pencarian informasi yang dilakukan manusia diberbagai data dokumen bertujuan untuk mendapatkan informasi dan dokumen yang tepat dalam waktu yang singkat.

Masalah timbul apabila jumlah data dokumen sangat besar dan tidak teratur, maka akibatnya proses pencarian dokumen tidak efektif [1]. Oleh karena itu, diperlukan suatu cara untuk mengatasi masalah untuk mengelompokan dokumen-dokumen tersebut secara otomatis. Klasifikasi merupakan salah satu metode yang bertujuan untuk mengelompokkan kategori dari suatu dokumen ke dalam satu atau lebih kelompok yang telah dikenal sebelumnya secara otomatis berdasarkan isi dokumen.

Extreme Learning Machines (ELM) merupakan salah satu metode pembelajaran yang digunakan untuk klasifikasi otomatis. Alasan utama keberhasilan dari ELM adalah kemampuannya dalam memperoleh fungsi model yang memberikan tingkat akurasi dan kecepatan yang baik serta mampu menangani jumlah data yang besar [2].

Huang [3] telah mengusulkan algoritma pembelajaran baru *Single Hidden Layer Feedforward Neural Network* (SLFN) disebut *Extreme Learning Machine* (ELM). Di algoritma ini bobot input dan bias layer tersembunyi dipilih secara acak. Formulasi ELM mengarah ke pemecahan sistem persamaan linear dalam hal bobot yang tidak diketahui menghubungkan lapisan tersembunyi ke lapisan keluaran. Solusi dari sistem persamaan linear umum ini diperoleh menggunakan *Moore-Penrose pseudu inverse.*

ELM digunakan untuk klasifikasi data berupa text dengan tingkat akurasi dan kecepatan yang baik. Tetapi untuk proses klasifikasi dokumen seringkali ditemukan hasil yang kurang baik dikarenakan jumlah kata setiap dokumen yang besar dan bervariasi sehingga harus dikelompokkan terlebih dahulu agar tingkat akurasi model yang diusulkan menjadi lebih baik [4]. Selain dilakukan

pengelompokan kata juga dilakukan seleksi ciri berupa kumpulan perwakilan kata yang akan digunakan untuk data masukan.

Oleh karena itu, pada penelitian ini akan mengimplementasikan metode *Extreme Learning Machines* (ELM) untuk menyelesaikan permasalahan klasifikasi dokumen.

2. LANDASAN TEORI

2.1 *Preprocessing*

Preprocessing adalah sebuah tahapan yang mengubah teks menjadi data yang dapat diolah pada tahap berikutnya. Tahapan *preprocessing* yang dilakukan pada penelitian ini meliputi *filtering, case folding, tokenizing, stopword removal,* perhitungan bobot TF-IDF dan normalisasi.

1) Filtering

Filtering adalah proses menghilangkan symbol dimana teks selain karakter "a" sampai "z" akan dihilangkan dan selain itu dianggap delimiter [5].

2) Case Folding

Case folding adalah proses mengkonversi teks menjadi case yang sama. Semua huruf dalam teks diubah menjadi huruf kecil [5].

3) Tokenizing

Tokenizing adalah proses memotong semua kata dalam kalimat dan mengubahnya menjadi kumpulan *token* [5].

4) Stopword Removal

Stopwords removal adalah proses menghilangkan kata yang bersifat umum dan dianggap sebagai kata yang tidak memiliki makna.[5].

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

TF-IDF adalah proses menentukan bobot *term* pada suatu dokumen berdasarkan frekuensi kemunculan kata. *Inverse Document Frequency* (IDF), adalah pengurangan dominasi *term* yang sering muncul diberbagai dokumen [6].

Pembobotan diperoleh dari frekuensi jumlah kemunculan sebuah kata yang terdapat di dalam sebuah dokumen, *term frequency* (TF). Sebuah kata atau jumlah kemunculan *term* di dalam koleksi dokumen, *inverse document frequency* (IDF).

$$I\ D\ F\ t = \log(N\ /d\ f\ ) \quad (1)$$

$$Wdt \ = \ TF * I\ D\ F\ t \quad (2)$$

Dimana:

W = bobot dokumen ke-d terhadap kata ke-t
d = dokumen ke-d
t = kata ke-t
N = jumlah dokumen
df = banyak dokumen yang mengandung tiap kata

6) Normalisasi

Normalisasi dilakukan terhadap vektor fitur dokumen untuk menghilangkan pengaruh anggapan bahwa dokumen panjang lebih relevan dibandingkan dokumen pendek. Dengan normalisasi ini dapat membantu menormalkan batas nilai dengan melakukan standarisasi nilai ke dalam interval 0 sampai dengan 1. Sehingga dapat dituliskan sebagai berikut:

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

2.2 *K-Means*

K-Means adalah metode pengelompokkan data dengan cara menghitung jarak terdekat data dengan *centroid* [7]. Adapun algoritma dari *K-means* sebagai berikut:

1. Tentukan jumlah *cluster*
2. Pilih nilai *centroid*

Pemilihan *centroid* awal iterasi dilakukan secara *random*. Sedangkan menentukan *centroid* iterasi selanjutnya menggunakan rumus sebagai berikut.

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

dimana:

$v\bar{i}j$ = *centroid* cluster ke-i untuk variabel ke-j
Ni = banyak data anggota *cluster* ke-i
i,k = indeks *cluster*
j = indeks variabel
xkj = data ke-k dalam cluster variabel ke-j

3. Hitung jarak data dengan pusat *cluster* menggunakan *Euclidean Distance*.

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

De = *euclidean distance*.
i = jumlah data
x = bobot dokumen.
y = pusat *cluster*.

4. Pengelompokkan Data

Menentukan anggota cluster dengan memperhitungkan jarak minimum objek.

5. Lakukan perulangan (tahap 2) hingga anggota cluster tidak berpindah ke cluster lain.

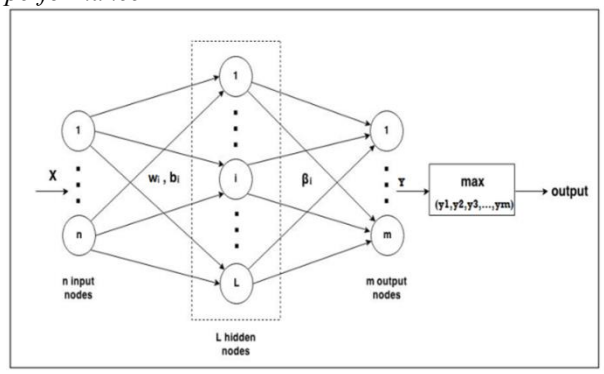
2.3 *Extreme Learning Machines* (ELM)

Extreme learning machine merupakan bagian dari jaringan saraf tiruan dengan satu *hidden layer* maupun *multi layer* [8].

Metode ELM dibuat untuk mengatasi kelemahan-kelemahan dari jaringan saraf tiruan *feedforward* terutama dalam hal *learning speed*. Algoritma ELM tidak melatih bobot input ataupun bias, ELM melatih

untuk memperoleh bobot keluarannya dengan menggunakan *norm-least-squares solution* dan *moore-penrose* inverse pada sistem linier secara umum.

Dengan menemukan node yang memberikan nilai output maksimal, dan parameter-parameter seperti *input weight* dan bias dipilh secara *random*, sehingga ELM memiliki *learning speed* yang cepat dan mampu menghasilkan *good generalization performance*.



Gambar 1 Arsitektur ELM

Untuk sampel N (x_i, y_i), di mana $x_i = [x_{i1}, x_{i2}, ..., x_{in}]^T \in R^n$ dan $y_i = [y_{i1}, y_{i2}, ..., y_{im}]^T \in R^m$, sedemikian rupa sehingga $(x_i, y_i) \in R^n \times R^m$ dimana $(i = 1,2, ..., N)$, dengan L sebagai hidden nodes, dan fungsi aktivasi $g(x)$. Fungsi keluaran ELM untuk input x yang diberikan adalah [4]:

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

Dimana, (w_i, b_i), $i = 1, ..., L$ dengan parameter secara acak dimana $w_i = [w_{i1}, w_{i2} ... w_{in}]$ adalah bobot vektor yang menghubungkan semua node input ke simpul tersembunyi. b_i adalah bias dari simpul tersembunyi. $\beta = [\beta_1, ..., \beta_L]^T$ adalah beratnya vektor antara *node* tersembunyi dan *node output*. $g(x)$ adalah vektor output yang memetakan ruang *input n-dimension* untuk fitur ruang dimensi L. Fungsi aktivasi $g(x)$ yang digunakan adalah softsign, fungsi ini memberikan batasan keluaran antara (0,1) rumusnya sebagai berikut:

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

Format dari persamaan (6) dapat dituliskan sebagai berikut:

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

Dimana,

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

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

Pelatihan kesalahan terkecil (konsep serupa di SVM) dan norma bobot keluaran terkecil dapat dicapai dengan ELM dan dapat direpresentasikan sebagai berikut:

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

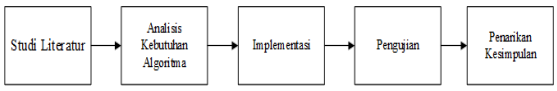
Solusi kuadrat terkecil minimal dari sistem linier di atas diberikan oleh:

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

dimana H^+ adalah *Moore-Penrose invers* matriks H. H (disebut ruang fitur ELM, juga dikenal sebagai *hidden layer output matrix*) merupakan matriks bobot penghubung input *layer* dan *hidden layer* maka matriks H mempunyai ukuran N x L. Penentuan nilai elemen-elemen matriks tersebut dilakukan secara *random*. Kemudian setiap nilai input tersebut diproses pada *hidden layer* menggunakan fungsi aktivasi dan nilai tersebut dihimpun dalam sebuah matriks H dengan ordo N x L.

3. METODE PENELITIAN

Metodologi penelitian yang akan digunakan dalam penelitian ini dari beberapa tahap seperti pada Gambar 2 berikut:



Gambar 2 Metode Penelitian

1) Studi Literatur

Studi literatur dalam penelitian ini mempelajari teori - teori melalui buku, artikel, jurnal dan bahan lain yang berkaitan dengan metode untuk mengklasifikasi dokumen dan algoritma yang digunakan.

2) Analisis Kebutuhan Algoritma

Menganalisis kebutuhan algoritma mulai dari

inputan (dokumen yang mengandung kata-kata yang dibutuhkan untuk penelitian), proses (tahapan praproses inputan yang diubah menjadi hasil sesuai format yang dipakai dalam penelitian), sampai output yang dihasilkan untuk dapat diimplementasikan pada program.

3) Implementasi

Pada tahap ini dilakukan pembangunan dari klasifikasi dokumen untuk dokumen berbahasa Indonesia. Dimulai dari tahapan preprocessing, seleksi menggunakan algoritma K-Means dan implementasi ELM untuk menghasilkan klasifikasi, analisis kebutuhan non fingsional, analisis kebutuhan fungsional dan perancangan sistem (struktur tabel, perancangan antarmuka dan jaringan semantik).

4) Pengujian

Pengujian dilakukan dengan membandingkan hasil klasifikasi oleh sistem dengan klasifikasi awal (manual) sehingga mendapatkan hasil perbandingan. Hasil perbandingan tersebut akan menjadi evaluasi apakah sistem berjalan cukup baik atau tidak.

5) Penarikan Kesimpulan

Pada tahap ini dilakukan penarikan kesimpulan yang didapat dari hasil klasifikasi yang dihasilkan oleh sistem untuk dokumen berbahasa Indonesia.

4. HASIL DAN PEMBAHASAN

4.1 HASIL

Data masukan untuk *training* adalah data abstrak (judul dan isi) skripsi dengan ekstensi .txt. Jumlah data untuk *training* sebanyak 100 dokumen dibagi menjadi 5 kategori yaitu A, B, C, D, dan E yang jumlahnya sama banyak. Data abstrak *training* dilakukan *preprocessing* untuk setiap dokumen yang ada di data *training* sehingga mendapat hasil pembobotan TF-IDF. *Preprocessing* yang dilakukan adalah *filtering, case folding, tokenizing, stopword removal,* TF-IDF, dan normalisasi.

Seleksi Fitur menggunakan algoritma *K-Means*. Dari hasil pembobotan kata dengan TF-IDF pada tiap dokumen sehingga kata pada semua dokumen digunakan sebagai data inputan *K-Means clustering*.

Setelah mendapatkan hasil cluster data. Selanjutnya adalah menentukan fitur kata yang akan digunakan untuk proses klasifikasi. Menggunakan cara sebagai berikut [4]:

1) Pilih kata secara acak dari daftar kata dari cluster. Siapkan daftar sinonim awal kata menggunakan kamus sinonim. Misalkan kata “metode”.
2) Periksa kata “metode” baik dalam daftar kata cluster dan dalam daftar sinonim. Jika kata “metode” ditemukan maka buat *new synonym-list* (NSL) dan tambahkan semua kata bersinonim ke NSL dan hapus dari daftar kata dari cluster. Dengan cara ini NSL dapat dibuat.
3) Ulangi langkah 1 dan 2 sampai daftar kata dari cluster terperiksa semua. Sehingga menghasilkan daftar NSL dari daftar kata cluster tersebut.
4) Ulangi langkah 1 sampai langkah 4 untuk setiap cluster. Pilih kata dikenal sebagai perwakilan kata dari masing-masing cluster dari daftar NSL yang memiliki nilai TF-IDF terbesar dan pada akhirnya, kombinasikan semua fitur dari setiap cluster sebagai seleksi fitur untuk data inputan.

Seluruh data hasil *cluster* dan sinonim diproses sistem, kemudian bobot w dan bias b ditentukan dengan bilangan acak yang terletak pada rentang (0,1) sehingga bisa didapatkan matriks H dan Y (8) kemudian bobot β dihitung dengan menggunakan rumus (12). Setelah dilakukan *training*, maka model klasifikasi yang diperoleh adalah sinonim *list*, nilai w, b serta β untuk digunakan dalam proses ELM *testing*. Lakukan *testing* dengan menggunakan data *testing* kemudian hitung *output* (6).

Pengujian dilakukan terhadap dokumen abstrak skripsi berjumlah 50 buah berkategorikan A, B, C, D dan E. Setiap kelas memiliki data sejumlah 10 buah. Pengujian akurasi dilakukan dua skenario yaitu pengujian pertama dilakukan untuk menentukan model klasifikasi yang dipakai. Pengujian kedua dilakukan untuk menentukan jumlah cluster dan nilai parameter yang diperlukan dalam klasifikasi. Sedangkan untuk melihat performansi maka hasil klasifikasi dari metode ELM akan dibandingkan dengan beberapa jumlah *cluster* dan parameter yang dipakai dalam pengujian.

Table 1 Perbandingan Performansi

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

Tabel 1 adalah perbandingan hasil pengujian dari metode klasifikasi ELM. Jumlah *cluster* dan fitur yang digunakan memberikan hasil yang berbeda. Hal ini terlihat pada akurasi yang didapatkan. Akurasi maksimum akurasi terbesar 42% dengan proses *testing* menggunakan 6 *cluster* dan arsitektur ELM yang terdiri dari 10 node pada *input layer*, 5 node pada *hidden layer,* dan 5 node pada *output layer* . Maksimum akurasi klasifikasi ELM dapat dilihat pada tabel 2.

Table 2 Hasil Pengujian Maksimum

No	Kelas	Benar	Salah	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
Jumlah		21	29	50
Akurasi		42%	58%	100%

$$\text{Akurasi} = \frac{Data\ Benar}{Banyak\ Data} \text{ x } 100\%$$
$$= \frac{21}{50} \text{ x } 100\%$$
$$= 42\%$$

4.2 PEMBAHASAN

Berdasarkan hasil yang diperoleh pada Tabel 1 maka dapat dilihat penggunaan jumlah *cluster* dan seleksi fitur pada ELM akan membuat nilai akurasi yang didapat berbeda. ELM yang mempunyai kelebihan memiliki kinerja baik untuk data yang berdimensi tinggi, pada kasus ini ternyata tidak lebih baik dari pada SVM [9]. Nilai akurasi ELM berbeda pada setiap model klasifikasi memperoleh akurasi maksimum sebesar 42%.

Model klasifikasi yang dipakai menentukan akurasi data yang diuji. Penggunaan proses *clustering* tidak signifikan meningkatkan performansi Penelitian ini belum menghasilkan nilai akurasi yang tinggi tetapi proses K-Means dapat mengurangi jumlah inputan yang dipakai. Hasil pengujian ini berbeda dengan yang diperoleh dari [10] K-Means berpengaruh terhadap optimalisasi tingkat akurasi model SVM dalam mengklasifikasikan kategori tema tugas akhir.

5. KESIMPULAN

Berdasarkan hasil implementasi dan pengujian maka dapat ditarik kesimpulan klasifikasi laporan skripsi dengan metode *Extreme Learning Machines* dengan seleksi fitur menghasilkan akurasi maksimum sebesar 42%. Dari hasil tersebut dapat dilihat bahwa akurasi ELM masih rendah dibanding dengan SVM. Penentuan model dari ELM juga mempengaruhi keakuratan sistem. Pemilihan parameter yang kurang tepat pada hidden layer (w dan b) akan memberikan hasil pengenalan berbeda dikarenakan nilai parameter tersebut ditentukan secara random.

Untuk penelitian selanjutnya, perlu dilakukan pengujian *Extreme Learning Machine* dengan metode-metode machine learning lainnya sebagai perbandingan. Memperluas ruang lingkup penelitian dengan menambahkan jenis objek yang akan dikenali. Menambahkan jumlah dataset untuk data *training* dan *testing*.

DAFTAR PUSTAKA

[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.