Klasifikasi Multi Label pada Hadis Bukhari Terjemahan Bahasa Indonesia Menggunakan Mutual Information dan k-Nearest Neighbor

Afrian Hanafi[1], Adiwijaya[2], Widi Astuti[3]*
School of Computing[1], [2], [3]
Telkom University
Bandung, Indonesia
afrianhanafi@student.telkomuniversity.ac.id[1], adiwijaya@telkomuniversity.ac.id[2], widiwdu@telkomuniversity.ac.id[3]

***Abstract*— Hadith is the second source of law for Muslims after the Qur'an which comes from various forms of the words, actions and stipulations of the Prophet Muhammad or referred to as his sunnah. In order to make it easier for Muslims to apply the teachings of the hadiths, a classification system is needed that can categorize a hadith into a class or a combination of two of the three classes which called a multi-label classification. In building a text classification system, there are various classification techniques, one of which is k-Nearest Neighbor (KNN). KNN is a simple and effective classification method for text classification, but has a weakness in processing data with high vector dimensions so that the computation time is higher and the efficiency of text classification is very low. Mutual Information (MI) is used as a feature selection method to reduce vector dimensions because it has the ability to show how strong a feature is in making a correct prediction of a class. In this study Problem Transformation Method with the Binary Relevance (BR) approach is used so that the multi label classification process can be accomplished. The optimum results obtained in this study shows the value of hamming loss is 0.0886 or about 91.14% of data were correctly classified and computational time for 595 seconds by using MI as a feature selection, but without stemming.**

Keywords*— *multi-label classification, bukhari's hadith, k-nearest neighbor, mutual information, hamming loss

***Abstrak*— Hadis merupakan sumber hukum kedua bagi umat muslim yang setelah Al-Qur`an yang mana berasal dari berbagai bentuk ucapan, tindakan, dan ketetapan Nabi Muhammad SAW atau disebut sebagai sunnah-nya. Untuk mempermudah umat muslim dalam menerapkan ajaran yang ada pada hadis, diperlukan sebuah sistem klasifikasi yang dapat mengkategorikan suatu hadis kedalam suatu kelas ataupun gabungan dari dua diantara ketiga kelas atau disebut dengan klasifkasi *multi label*. Pada pembangunan sistem klasifikasi teks, terdapat berbagai teknik klasifikasi salah satunya yaitu *k-Nearest Neighbor* (KNN). KNN merupakan suatu metode klasifikasi yang sederhana dan efektif untuk klasifkasi teks, namun memiliki kelemahan dalam memproses data dengan dimensi vektor yang tinggi sehingga menyebabkan waktu komputasi menjadi lebih tinggi dan efisiensi dari klasifikasi teks sangat rendah. *Mutual Information* (MI) digunakan sebagai metode *feature selection* untuk mereduksi dimensi vektor karena memiliki kemampuan untuk menunjukan seberapa berpengaruh suatu fitur dalam melakukan prediksi yang tepat terhadap suatu kelas. Pada penelitian ini, *Problem Transformation Method* dengan pendekatan *Binary Relevance* (BR) digunakan agar proses klasifikasi multi label dapat dilakukan. Hasil optimum yang didapat pada penelitian ini menunjukan nilai *hamming loss* sebesar 0.0886 atau sekitar 91.14% data yang terklasifikasi dengan benar dan waktu komputasi selama 595 detik dengan menggunakan MI sebagai feature selection, namun tanpa stemming.**

Kata Kunci*— *klasifikasi multi-label, hadis bukhari, k-nearest neighbor, mutual information, hamming loss

I. PENDAHULUAN

Hadis merupakan sumber hukum kedua bagi umat muslim yang setelah Al-Qur`an yang mana berasal dari berbagai bentuk ucapan, tindakan, dan ketetapan Nabi Muhammad SAW atau disebut sebagai sunnah-nya [1][2]. Hadis juga berfungsi sebagai memperjelas dan menegaskan hukum-hukum yang ada di Al-Qur`an. Umumnya hadis tersebut diriwayatkan oleh para ahli hadis, salah satunya yaitu Imam Bukhari. Tujuannya supaya generasi yang akan datang dapat menerapkan nilai-nilai yang diterapkan oleh Nabi Muhammad SAW. Masing-masing hadis memiliki beberapa jenis ajaran yang dapat diterapkan dalam kehidupan umat muslim sehari-hari. Ajaran yang dimaksud seperti anjuran, informasi, hingga larangan-larangan [3]. Namun terdapat kendala dalam mempelajari hadis untuk diterapkan isi dan maknanya dikehidupan sehari-hari apabila umat Islam kesulitan dalam menggolongkan hadis ke dalam anjuran, larangan, dan informasi maupun gabungan dari kedua diantara ketiga jenis ajaran tersebut. Atas dasar itu, untuk mempermudah umat muslim dalam menerapkan ajaran yang ada pada hadis, diperlukan sebuah sistem klasifikasi yang dapat mengkategorikan dan mengindentifikasi suatu hadis kedalam suatu kelas ataupun gabungan dari dua diantara ketiga kelas atau disebut dengan klasifkasi *multi label*.

Pada pembangunan sistem klasifikasi teks, terdapat berbagai teknik klasifikasi salah satunya yaitu *k-Nearest Neighbor* (KNN). KNN merupakan suatu metode klasifikasi yang sederhana dan efektif untuk klasifkasi teks dengan pendekatan pembelajaran berbasis sampel atau disebut dengan *supervised learning*, yang mana pada tahapan klasifikasi

p-ISSN 2301-7988, e-ISSN 2581-0588
DOI : 10.32736/sisfokom.v9i3.980, Copyright ©2020
Submitted : 5 September 2020, Revised : 8 September 2020, Accepted : 9 September 2020, Published : 24 September 2020

menggunakan seluruh data latih untuk memprediksi label dari data uji. KNN memiliki kelemahan dalam memproses data dengan dimensi vektor yang tinggi sehingga menyebabkan waktu komputasi menjadi lebih tinggi dan efisiensi dari klasifikasi teks sangat rendah [4][5][6]. Salah satu solusi yang dapat dilakukan untuk meningkatkan kinerja dari KNN yaitu dengan mereduksi dimensi vektor [5]. Untuk mereduksi data dengan dimensi vektor yang tinggi perlu dilakukannya *feature selection* untuk menyeleksi kata atau fitur yang dianggap kurang relevan dalam pembentukan suatu model. *Mutual Information* (MI) digunakan sebagai metode *feature selection* karena memiliki kemampuan untuk menunjukan seberapa berpengaruh suatu fitur dalam melakukan prediksi yang tepat terhadap suatu kelas [7][8].

Berdasarkan penjabaran diatas, maka pada penelitian ini dilakukan klasifikasi *multi label* pada Hadis Bukhari Terjemahan Bahasa Indonesia kedalam suatu kelas ataupun gabungan dari dua diantara ketiga kelas yang ada yaitu anjuran, larangan dan informasi dengan menggunakan KNN sebagai metode klasifikasi dan MI sebagai metode *feature selection.* Namun umumnya klasifikasi teks dengan KNN hanya dapat dilakukan untuk data dengan *single label*. Sehingga *Problem Transformation Method* juga perlu digunakan agar proses klasifikasi *multi label* dapat dilakukan. *Binary Relevance* (BR) merupakan salah satu pendekatan yang ada pada *Problem Transformation Method*. Dasar algoritma dari BR yaitu untuk mendekomposi permasalahan *multi label* menjadi permasalahan *single label* dan mengintegrasikan hasil klasifikasinya kembali menjadi bentuk *multi label* [9]. Metode evaluasi diperlukan untuk mengetahui hasil performansi dari sistem yang dibangun. Pada penelitian ini, *hamming loss* digunakan untuk mengukur performansi dari hasil klasifikasi *multi label* yang didapatkan.

II. TINJAUAN PUSTAKA

A. Penelitian Terkait

Penelitian mengenai klasifikasi Hadis Bukhari terjemahan Bahasa Indonesia dengan kelas anjuran, larangan dan informasi sudah pernah dilakukan beberapa kali menggunakan pendekatan dan metode yang beragam salah satunya penelitian oleh S. Al Faraby [10] yang berfokus pada permasalahan klasifikasi *single label* dengan menggunakan *Support Vector Machine* dan *Artificial Neural Network*. Pada penelitian ini juga dibangun *classfier* dasar menggunakan metode *rule-based*. Hal tersebut disebabkan karena kelas atau kategori yang dimaksud cukup baru pada saat penelitian dilakukan. Hasil yang didapat pada metode *rule-based classifier* yaitu dengan nilai *f1-score* sebesar 0.69, sedangkan nilai *f1-score* yang optimum dengan pendekatan pembelajaran mesin dihasilkan oleh model *Support Vector Machine* sebesar 0.88 dengan *kernel* linear.

Untuk kasus klasifikasi *multi label* telah dilakukan oleh H. Prasetyo [11] yang mana pada penelitiannya menggunakan *Backpropagation Neural Network* dan *Mutual Information.* Penelitian ini mendapatkan nilai *hamming loss* sebesar 0.1330 dengan data yang telah melalui proses *stemming*. Sedangkan nilai *hamming loss* yang didapat tanpa melalui proses *stemming* sebesar 0.1317, sehingga penggunaan proses *stemming* pada data *multi label* memiliki performansi lebih rendah dibandingkan proses tanpa *stemming* dengan selisih nilai *hamming loss* sebesar 0.0013. Nilai *hamming loss* optimum yang didapat pada penelitian ini yaitu sebesar 0.0954 yang mana tanpa menggunakan *Mutual Information*. Namun disisi lain, *Mutual Information* memiliki pengaruh baik terhadap proses klasifikasi yaitu pada waktu komputasi. Penggunaan *Mutual Information* sebagai metode *feature selection* membutuhkan waktu komputasi selama 2175 detik atau 1603 detik lebih cepat dibandingkan tanpa menggunakan *Mutual Information* yang membutuhkan waktu komputasi selama 3778 detik.

Penelitian klasifikasi teks lainnya pada ayat-ayat Al-Qur`an telah dilakukan oleh Abdullahi O. Adeleke [12], yang mana penelitannya menerapkan tiga metode klasifikasi yaitu, *K-Nearest Neighbor, Support Vector Machine* dan *Naïve Bayes*. Pada eksperimen yang dilakukan, *K-Nearest Neighbor* memperoleh nilai akurasi yang tinggi secara umum dibandingkan metode klasifikasi yang lain yaitu sebesar 77.5%.

Adapun penelitian klasifikasi teks dengan menggunakan *K-Nearest Neighbor* lainnya juga dilakukan pada penelitian [13]. Penelitian ini berhasil menggunakan *K-Nearest Neighbor* untuk klasifikasi multi label pada topik berita Indonesia. Nilai parameter *k* yang diuji antara 5-23, sedangkan parameter *k* = 11 menghasilkan nilai *hamming loss* paling optimum dibandingkan nilai *k* lainnya yaitu sebesar 0.1116. Ia juga menyebutkan bahwa menentukan nilai *k* merupakan hal yang sangat penting untuk mendapatkan hasil klasifikasi yang optimum. Jika nilai *k* terlalu kecil, maka hasil klasifikasi akan lebih dipengaruhi oleh *noise*. Sedangkan apabila nilai *k* terlalu tinggi, maka akan mengurangi efek *noise* pada klasifikasi.

B. Klasifikasi Teks

Klasifikasi teks atau kategorisasi teks merupakan suatu pendekatan teknologi yang ditujukan untuk menentukan apakah suatu teks termasuk kedalam satu atau lebih kategori berdasarkan isi dari teks tersebut yang mengacu terhadap pemodelan kategori yang diberikan [14][15][16].

Proses dari klasifikasi teks itu sendiri bervariasi, hal tersebut dikarenakan beberapa penelitian memiliki kepentingan dan kebutuhan yang berbeda-beda. Namun secara umum proses klasifikasi teks terdiri dari enam tahapan, diantaranya yaitu: pengumpulan data, *pre-processing*, *feature selection*, *feature extraction*, pembangunan *classifier* dan evaluasi performansi [17].

C. Klasifikasi Multi Label

Klasifikasi multi label pada dasarnya merupakan suatu bagian permasalahan dari klasifikasi teks, yang mana masing-masing dokumen dapat tergolong kedalam beberapa kelas. Klasifikasi *multi label* ini berbeda dengan klasifikasi *single label*. Klasifikasi *single label* bertujuan untuk mengklasifikasikan suatu dokumen hanya kedalam satu kelas saja [13][18][19]. Dalam kasus *multi label* ini, tiap dokumen yang ada pada data latih memiliki satu set label, dengan tujuan untuk memprediksi suatu set label pada tiap dokumen yang

p-ISSN 2301-7988, e-ISSN 2581-0588
DOI : 10.32736/sisfokom.v9i3.980, Copyright ©2020
Submitted : 5 September 2020, Revised : 8 September 2020, Accepted : 9 September 2020, Published : 24 September 2020

belum diketahui kelasnya [20].

Oleh karena itu, untuk mengatasi permasalahan klasifikasi multi label ini dapat dilakukan dengan algoritma Binary Relevance (BR). BR merupakan salah satu pendekatan yang ada pada Problem Transformation Method. Dasar dari algoritma ini yaitu untuk mendekomposisi permasalahan multi label menjadi permasalahan single label agar proses klasifikasi dapat dilakukan [9].

D. Pre-processing

Pre-processing merupakan tahapan awal yang perlu dilakukan sebelum dilakukan suatu klasifikasi teks. Hal tersebut bertujuan untuk mengolah struktur teks agar lebih optimal sehingga informasi yang didapat dari teks memiliki kualitas yang baik. Teks *pre-processing* umumnya terdiri dari *case folding*, *tokenizing*, *stopword removal*, dan *stemming* [21].

E. Mutual Information (MI)

MI merupakan salah satu metode yang sudah digunakan secara luas sebagai untuk melakukan *feature selection*. MI mengukur berapa banyak informasi yang ada pada fitur, sehingga dapat diketuahui pengaruh fitur tersebut untuk membuat keputusan klasifikasi yang tepat [22][7]. Rumus perhitungan nilai MI secara formal dapat dilihat pada persamaan (1).

$$I(U,C) = \sum_{et\in\{1,0\}} \sum_{ec\in\{1,0\}} P(U = et, C = ec) \log^2 \frac{P(U = et, C = ec)}{P(U = et)P(C = ec)} \quad (1)$$

Dimana variabel *U* merupakan variabel acak dengan nilai e_t = 1 (dokumen mengandung *term* t) dan e_t = 0 (dokumen tidak mengandung *term* t), sedangkan *C* untuk merupakan variabel acak dengan nilai e_c = 1 (dokumen berada dikelas *c*) dan e_c = 0 (dokumen tidak berada dikelas *c*). Persamaan diatas (1) dapat dijabarkan menjadi seperti persamaan (2).

$$I(U,C) = \frac{N11}{N} log_2 \frac{N.N11}{N1.N1} + \frac{N01}{N} log_2 \frac{N.N01}{N0.N1} + \frac{N10}{N} log_2 \frac{N.N10}{N1.N0} + \frac{N00}{N} log_2 \frac{N.N00}{N0.N0} \quad (2)$$

Sebagai contoh, N10 merupakan jumlah dokumen yang mengandung *t* (e_t = 1) dan tidak dalam kelas *c* (e_c = 0).

Keterangan:
N = Jumlah dokumen yang memiliki et dan ec atau (N = N00 + N01 + N10 + N11).
N1. = Jumlah dokumen yang memiliki et atau (N1. = N10 + N11).
N1 = Jumlah dokumen yang memiliki ec atau (N1 = N01 + N11).
N0. = Jumlah dokumen yang tidak memiliki et atau (N0. = N01 + N00).
N0= Jumlah dokumen yang tidak memiliki ec atau (N0 = N10 + N00).

F. Term Frequency and Invers Document Frequency (TF-IDF)

TF-IDF merupakan statistik numerik yang dimaksudkan untuk menggambarkan seberapa penting suatu kata atau *term* dalam suatu dokumen pada suatu *collection* atau *corpus*. TF-IDF digunakan sebagai metode pembobotan *term* dengan menggunakan *term-frequency* (jumlah *term* yang terdapat pada tiap dokumen) serta *inverse document frequency* (invers jumlah dokumen yang memuat suatu *term*) [23]. Rumus perhitungan untuk melakukan pembobotan dengan TF-IDF dapat dilihat pada persamaan (3).

$$W_{ij} = tf_{ij} \times \log\left(\frac{D}{df_i}\right) \quad (3)$$

Keterangan:
w_{ij} = Bobot kata ke-*i* pada dokumen ke-*j*
tf_{ij} = merupakan jumlah term ke-*i* pada dokumen ke-*j*
D = Jumlah dokumen keseluruhan
df_i = merupakan jumlah dokumen yang mengandung term ke-*i*

G. Classifier

Pada penelitian ini, *classifier* yang digunakan adalah *K-Nearest Neighbor* (KNN). Metode KNN bekerja dengan cara mencari sejumlah *k* objek data yang memiliki jarak paling dekat dengan data yang sedang diklasifikasi, kemudian data tersebut akan digolongkan kedalam suatu kategori berdasarkan voting dengan probabilitas kategori tertinggi [24][25].

H. k-Fold Cross Validation

Merupakan sebuah proses yang cukup popular untuk mengestimasi performansi dari suatu algoritma klasifikasi maupun membandingkannya berdasarkan sebuah data set. Proses ini dilakukan dengan cara membagi data menjadi k bagian yang mana tiap bagian tersebut akan dijadikan data tes secara bergantian. Ketika suatu data dijadikan data tes, maka bagian yang lain akan dijadikan data latih. Performansi suatu algoritma klasifikasi akan dievaluasi berdasarkan hasil dari rata-rata nilai performansi k yang didapat [26]. Pada penelitian ini, digunakan nilai $k = 4$ seperti pada penelitian [27].

I. Hamming Loss

Salah satu metode yang dapat dilakukan untuk melakukan evaluasi performansi adalah metode Hamming Loss. Metode ini ditujukan untuk menghitung banyaknya kesalahan klasifikasi terhadap data yang diuji. Semakin kecil nilai yang dihasilkan, maka semakin baik performa dari klasifikasi tersebut dan begitu juga sebaliknya. Persamaan untuk melakukan perhitungan Haming Loss dapat dilihat pada persamaan (4).

p-ISSN 2301-7988, e-ISSN 2581-0588
DOI : 10.32736/sisfokom.v9i3.980, Copyright ©2020
Submitted : 5 September 2020, Revised : 8 September 2020, Accepted : 9 September 2020, Published : 24 September 2020

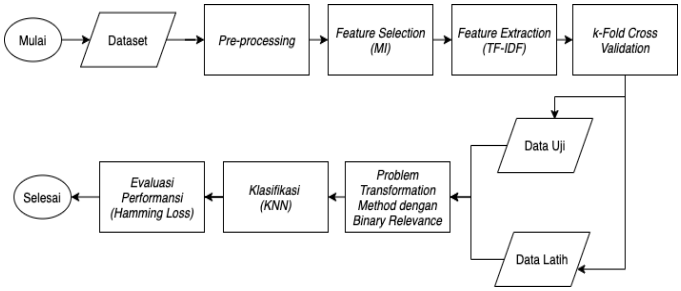
$$HammingLoss(h) = \frac{1}{p}\sum_{i=1}^{p}\frac{1}{Q}|h(x_i) \triangle Y_i| \qquad (4)$$

Dimana p merupakan jumlah banyak data yang akan digunakan pada klasifikasi, Q merupakan jumlah label kelas yang ada pada tersebut, dan $|h(x)\triangle Y|$ adalah jumlah banyaknya kesalahan yang terjadi pada klasifikasi.

III. PERANCANGAN SISTEM

A. System Overview

Pada penelitian ini, dibangun sebuah sistem yang dapat mengklasifikasikan topik Hadis Bukhari Terjemahan Bahasa Indonesia secara *multi label* menggunakan k-Nearest Neighbor dan Mutual Information. Gambar 1 menunjukan gambaran sistem yang dirancang secara umum.



Gambar 1. Gambaran Umum Sistem

B. Dataset

Dataset yang digunakan pada penelitian ini merupakan data Hadis Bukhari terjemabahan Bahasa Indonesia sebanyak 1064 hadis yang mana tiap datanya telah dilabeli kelas secara manual oleh Muhammad Yuslan Abu Bakar [28]. Data hadis ini bersifat *multi label* yang terdiri dari tiga kelas diantaranya ada tiga, yaitu anjuran, larangan dan informasi. Label dari masing-masing data dapat terdiri dari satu atau dua dari ketiga kelas yang ada. Representasi dataset *multi label* yang digunakan dalam penelitian ini dapat dilihat pada Tabel I.

TABEL I. REPRESENTASI DATASET MULTI LABEL

Hadis	Anjuran	Larangan	Informasi
Kenapa orang-orang mengarahkan pandangan mereka ke langit ketika mereka sedang shalat? Suara beliau semakin tinggi hingga beliau bersabda: Hendaklah mereka menghentikannya atau Allah benar-benar akan menyambar penglihatan mereka.	1	0	1

C. Pre-processing

Sebelum dilakukannya proses klasifikasi, perlu dilakukan *pre-processing* terhadap dataset. *Pre-processing* dilakukan untuk membuat data yang diproses menjadi lebih optimal dan meningkatkan performansi sistem yang akan dibangun. Pada penelitian ini, tahapan yang akan dilakukan pada pre-processing dengan tahapan sebagai berikut*:*

1) *Noise Removal* merupakan tahap yang dilakukan untuk menghapus komponen pada data yang dianggap tidak dibutuhkan, seperti spasi, angka, dan tanda baca.
2) *Case Folding* dilakukan yaitu untuk merubah semua huruf yang ada pada data menjadi huruf kecil.
3) *Tokenizing* dilakukan pemenggalan kata dari tiap data yang akan diolah.
4) *Stopword Removal* merupakan tahap yang dilakukan untuk menghapus kata-kata yang dianggap tidak memiliki pengaruh besar. Ciri dari kata tersebut biasanya memiliki frekuensi kemunculan yang jauh lebih tinggi dibandingkan dengan kata lainnya.
5) *Stemming* merupakan proses untuk mengembalikan kata-kata yang didapat dari hasil tokenizing menjadi kata dasar. Tujuan dari proses ini untuk menghilangkan awalan, akhiran, imbuhan dan lain sebagainya dari suatu kata.

D. Feature Selection

Setelah dilakukannya pembersihan data melalui tahap *pre-processing*, fitur yang ada pada data selanjutnya dilakukan *feature selection. Feature selection* merupakan suatu tahap yang dilakukan untuk memilih *feature* dengan pengaruh yang tinggi dan menyingkirkan feature yang kurang berpengaruh atau dianggap kurang relevan dalam pembangunan model klasifkasi. *Feautre selection* yang akan digunakan pada penelitian ini adalah *Mutual Information* (MI), dengan mengukur jumlah informasi suatu variabel terhadap suatu kelas. Persamaan untuk melakukan perhitungan MI dapat dilihat pada persamaan (2).

Setelah didapatkan nilai MI untuk masing-masing kata, maka selanjutnya akan diurutkan bersarkan nilai yang paling tinggi. Tabel II memperlihatkan contoh hasil perhitungan MI.

TABEL III. HASIL PERHITUNGAN MI

No. Fitur	Fitur	Nilai MI
1	jangan	0.0817
2	kalian	0.0482
3	hendak	0.0278

E. Feature Extraction

Tahap selanjutnya setelah didapatkan nilai fitur yang diseleksi yaitu pembobotan fitur atau *feature extraction.* Metode *feature extraction* yang digunakan adalah *Term Frequency and Invers Document Frequency* (TF-IDF), yang mana rumus perhitungannya dapat dilihat pada persamaan (3). Tabel III memperlihatkan contoh hasil perhitungan TF-IDF.

Setelah bobot dari masing-masing fitur yang diseleksi didapatkan, maka selanjutnya akan dibentuk matriks baru yang merepresentasikan tiap fitur beserta bobot akhirnya. Ukuran

p-ISSN 2301-7988, e-ISSN 2581-0588
DOI : 10.32736/sisfokom.v9i3.980, Copyright ©2020
Submitted : 5 September 2020, Revised : 8 September 2020, Accepted : 9 September 2020, Published : 24 September 2020

matriks sebesar $d \times t$, dimana d merupakan jumlah banyaknya dokumen dan t merupakan jumlah banyaknya *term*/fitur. Representasi matriks TF-IDF dapat dilihat pada Tabel IV.

TABEL IIIII. HASIL PERHITUNGAN TF-IDF

Fitur	tf			df	$\frac{D}{df}$	idf	$tf \times idf$		
	D1	D2	D3				D1	D2	D3
jangan	1	2	0	2	1.5	0.176	0.176	0.352	0
kalian	2	0	1	2	1.5	0.176	0.352	0	0.176
hendak	0	0	1	1	3	0.477	0	0	0.477

TABEL IVV. REPRESENTASI MATRIKS TF-IDF

	jangan	kalian	hendak
D1	0.176	0.352	0
D2	0.352	0	0
D3	0	0.176	0.477

Untuk melakukan proses klasifikasi, perlu dilakukannya *problem transformation*. Pendekatan *problem transformation* yang digunakan pada penelitian ini adalah *Binary Relevance* (BR). BR dilakukan dengan tujuan untuk mendekomposisi data yang masih dalam bentuk *multi label* menjadi *single label* agar proses klasifikasi dapat dilakukan. Contoh representasi dataset *multi label* dapat dilihat pada Tabel I, sedangkan hasil dekomposisi data menjadi *single label* dapat dilihat pada Tabel V, Tabel VI dan Tabel VII.

TABEL V. HASIL DEKOMPOSISI DATA MENJADI SINGLE LABEL PADA KELAS ANJURAN

Hadis	Anjuran
Kenapa orang-orang mengarahkan pandangan mereka ke langit ketika mereka sedang shalat? Suara beliau semakin tinggi hingga beliau bersabda: Hendaklah mereka menghentikannya atau Allah benar-benar akan menyambar penglihatan mereka.	1

TABEL VI. HASIL DEKOMPOSISI DATA MENJADI SINGLE LABEL PADA KELAS LARANGAN

Hadis	Larangan
Kenapa orang-orang mengarahkan pandangan mereka ke langit ketika mereka sedang shalat? Suara beliau semakin tinggi hingga beliau bersabda: Hendaklah mereka menghentikannya atau Allah benar-benar akan menyambar penglihatan mereka.	0

TABEL VII. HASIL DEKOMPOSISI DATA MENJADI SINGLE LABEL PADA KELAS INFORMASI

Hadis	Informasi
Kenapa orang-orang mengarahkan pandangan mereka ke langit ketika mereka sedang shalat? Suara beliau semakin tinggi hingga beliau bersabda: Hendaklah mereka menghentikannya atau Allah benar-benar akan menyambar penglihatan mereka.	1

F. Klasifikasi dengan k-Nearest Neighbor (KNN)

Setelah dillakukannya *Problem Transformation Methods* menggunakan pendekatan *Binary Relevance*, maka selanjutnya dibangun model klasifikasi menggunakan metode KNN. Berikut merupakan langkah-langkah algoritma KNN yang dilakukan secara garis besar:

1) Menentukan jumlah *k.*
2) Menghitung jarak suatu objek data terhadap seluruh data latih.
3) Mengurutkan hasil nilai jarak secara *ascending.*
4) Pilih beberapa objek tetangga terdekat sebanyak *k.*
5) Melakukan voting berdasarkan kelas dengan frekuensi tertinggi pada *k* tetangga terdekat.

Untuk menghitung jarak antar data digunakan metode *euclidean distance* dengan persamaannya dapat dilihat pada persamaan (5).

$$D(x,y) = \sqrt{\sum_{i=1}^{n}(x_i - y_i)^2} \quad (5)$$

Dimana D merupakan nilai jarak antar data, x dan y merupakan nilai dari masing-masing titik.

G. Evaluasi Performansi

Setelah dilakukannya proses klasifikasi, selanjutnya dilakukan tahap evaluasi performansi untuk mencari tahu seberapa baik hasil dari klasifikasi. Metode yang digunakan pada penelitian ini menggunakan *Hamming Loss* (HL). HL bekerja dengan cara menghitung nilai kesalahan dari hasil klasifikasi. Persamaan untuk melakukan perhitungan HL dapat dilihat pada persamaan (4).

TABEL VIII. CONTOH HASIL KLASIFIKASI

Dokumen	Hasil Klasifkasi			Nilai Seharusnya		
	Anjuran	Larangan	Informasi	Anjuran	Larangan	Informasi
1	1	0	0	1	0	1
2	0	1	1	1	0	0
3	1	0	1	1	0	1

Jika dilihat pada Tabel VIII, terjadi 1 kesalahan pada dokumen 1 dan 3 kesalahan pada dokumen 2. Total kesalahan dari 3 dokumen yaitu sebanyak 4 kesalahan. Berikut merupakan nilai HL yang didapat dari contoh hasil klasifikasi.

$$\frac{1}{p}\sum_{i=1}^{p}\frac{1}{Q}|h(x_i) \triangle Y_i| = \frac{1}{3}.\frac{1}{3}.4 = 0.444$$

Semakin kecil atau mendekati 0 hasil yang didapat dari nilai HL, maka semakin baik.

IV. HASIL DAN PEMBAHASAN

Tahap pengujian dilakukan untuk dapat melakukan analisis dan evaluasi terhadap performansi yang dihasilkan dari sistem yang dibangun. Fokus pengujian ini terdiri dari tiga tahap yaitu

p-ISSN 2301-7988, e-ISSN 2581-0588
DOI : 10.32736/sisfokom.v9i3.980, Copyright ©2020
Submitted : 5 September 2020, Revised : 8 September 2020, Accepted : 9 September 2020, Published : 24 September 2020

tahap *pre-processing*, tahap *feature selection*, dan tahap klasifikasi.

A. Hasil Pengujian

Pada penelitian ini, dilakukan pembagian data uji dan data latih menggunakan *4-Fold Cross Validation* dengan data sejumlah 1064 hadis. Data uji terdiri dari satu bagian *fold* atau 266 data hadis, sedangkan data latih terdiri dari tiga bagian *fold* atau 798 data hadis yang mana masing-masing *fold*-nya telah tersebar data dengan berbagai kelas yang ada.

Skenario pengujian pertama adalah menguji penggunaan proses *stemming* pada tahap *pre-processing.* Pada skenario ini menggunakan *7-nearest neighbor* dan 472 fitur atau sebesar 10% dari seluruh fitur yang telah diseleksi menggunakan *mutual information* sebagai parameternya. Hasil pengujian pada skenario pertama dapat dilihat pada Tabel IX.

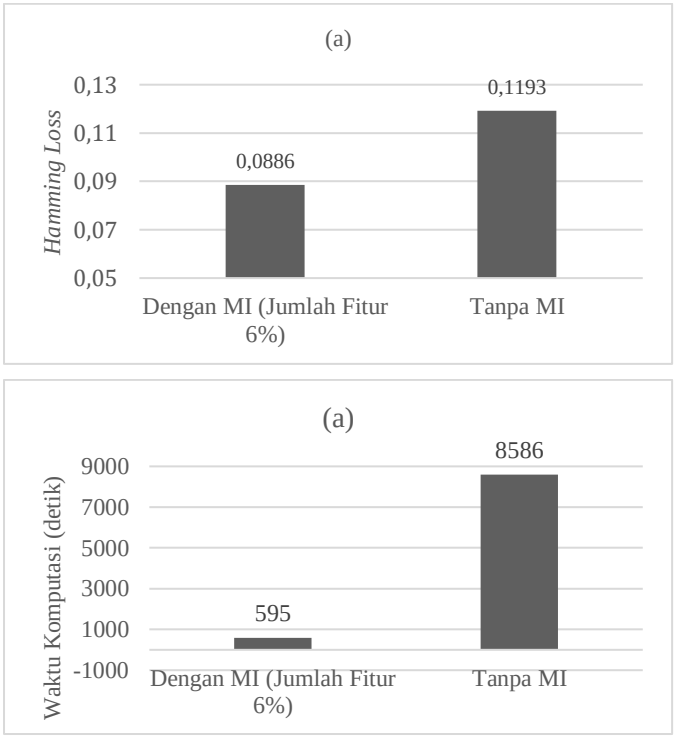
TABEL IX. HASIL PENGUJIAN SKENARIO 1

Metode	*Hamming Loss*					Waktu Komputasi (detik)
	Fold 1	*Fold* 2	*Fold* 3	*Fold* 4	Rata-rata	
Stemming	0.1015	0.1203	0.0964	0.0864	0.1011	957
Tanpa *Stemming*	0.0864	0.1052	0.0764	0.0927	**0.0902**	**899**

Setelah ditentukannya metode terbaik pada tahap *pre-processing*, pada skenario berikutnya dilakukan pengujian pertama yaitu menguji kinerja *mutual information* sebagai metode *feature selection* dengan tujuan untuk mengoptimasi hasil klasifikasi terhadap nilai *hamming loss* dan waktu komputasi dengan cara mencari tahu berapa banyak fitur yang telah diseleksi yang perlu digunakan agar meraih hasil optimum. Pada pengujian pertama untuk skenario ini, dilakukan pengujian dengan penggunaan fitur dimulai sebesar 2%, 6%, lalu 10% sampai 100% dengan kelipatan sebesar 10% pada setiap proses klasifikasi. Data yang digunakan merupakan data tanpa *stemming* yang menghasilkan 4716 fitur. Hasil pengujian pertama pada skenario kedua dapat dilihat pada Gambar 2.

Gambar 2. (a) Hasil pengujian penggunaan fitur pada mutual information terhadap hamming loss, (b) Hasil pengujian penggunaan fitur pada mutual information terhadap waktu komputasi

Pengujian kedua yang terdapat pada skenario ini yaitu membandingkan performa klasifkasi menggunakan persentase jumlah fitur optimum pada *mutual information* yang didapat dari pengujian sebelumnya dengan performa klasifkasi tanpa *mutual information* guna mengetahui dampak yang dihasilkan dari metode *feature selection* tersebut. Hasil pengujian ini pada skenario kedua dapat dilihat pada Gambar 3.



Gambar 3. (a) Hasil pengujian dengan atau tanpa mutual information terhadap hamming loss, (b) Hasil pengujian dengan atau tanpa mutual information terhadap waktu komputasi

Setelah ditentukannya persentase atau jumlah fitur optimum yang digunakan pada *mutual information*, selanjutnya perlu dilakukan skenario pengujian terakhir yang bertujuan untuk mencari tahu nilai parameter k optimum dengan kemungkinan *noise* yang rendah dalam pembangunan model klasifikasi *k-nearest neighbor* terhadap nilai *hamming loss.* Pengujian dilakukan sebanyak 7 kali, yang dimulai dari $k = 3$ sampai $k = 15$. Hasil pengujian pada skenario ketiga dapat dilihat pada Tabel X.

TABEL X. HASIL PENGUJIAN SKENARIO 3

K-Fold	*Hamming Loss*						
	$k = 3$	$k = 5$	$k = 7$	$k = 9$	$k = 11$	$k = 13$	$k = 15$
Fold 1	0.0952	0.0839	0.0814	0.0914	0.0927	0.0964	0.0977
Fold 2	0.1077	0.1077	0.1090	0.1115	0.1190	0.1265	0.1240
Fold 3	0.0852	0.0814	0.0802	0.0852	0.0839	0.0877	0.0914
Fold 4	0.0852	0.0852	0.0839	0.0877	0.0939	0.0939	0.0952
Rata-rata	0.0933	0.0895	**0.0886**	0.0939	0.0973	0.1011	0.1020

p-ISSN 2301-7988, e-ISSN 2581-0588
DOI : 10.32736/sisfokom.v9i3.980, Copyright ©2020
Submitted : 5 September 2020, Revised : 8 September 2020, Accepted : 9 September 2020, Published : 24 September 2020

B. Analisis Hasil Pengujian

Skenario pengujian pertama dilakukan untuk melihat pengaruh proses *stemming* pada tahap *pre-processing* terhadap nilai *hamming loss* dan waktu komputasi. Hasil pengujian pada skenario ini dapat dilihat pada Tabel IX yang mana menunjukan bahwa pengujian tanpa proses *stemming* menghasilkan nilai *hamming loss* lebih baik yaitu sebesar 0.0902 atau sekitar 90.98% data yang terklasifikasikan dengan benar dan waktu komputasi lebih rendah yaitu selama 899 detik. Hal ini disebabkan karena dengan digunakannya proses *stemming*, maka setiap kata yang ada pada hadis diubah menjadi kata dasarnya dan menghilangkan ciri khusus yang ada pada data atau hadis. Sebagai contoh pada kata yang memiliki akhiran "-lah" seperti "hendak-lah" menunjukan suatu anjuran, namun apabila dilakukan proses *stemming* maka kata tersebut berubah menjadi "hendak" yang cenderung menunjukan suatu informasi, sehingga menyebabkan terjadinya pergeseran makna pada kata tersebut apabila dilakukan generalisasi dengan proses *stemming*. Analisis contoh dampak terhadap penggunaan *stemming* dapat dilihat pada Lampiran 1.

Pada skenario kedua terdiri dari dua pengujian yang dilakukan untuk melihat pengaruh *mutual information* pada tahap *feature selection* terhadap nilai *hamming loss* dan waktu komputasi. Hasil pengujian pertama pada skenario ini dapat dilihat pada Gambar 4, dimana semakin banyak jumlah fitur yang digunakan maka kian besar nilai *hamming loss* yang didapat, serta waktu komputasi yang dibutuhkan juga kian meninggi. Persentase penggunaan fitur optimum yang didapat dari hasil pengujian ini yaitu sebesar 6% atau sebanyak 283 fitur digunakan, yang berarti memiliki hasil performansi yang sama dengan hasil pengujian pada skenario pertama yaitu dengan nilai *hamming loss* sebesar 0.0886 atau sekitar 91.14% data yang terklasifikasi dengan benar dan waktu komputasi selama 595 detik. Hal ini disebabkan karena pada 6% jumlah fitur tersebut merupakan fitur yang berpengaruh dan menggambarkan suatu kelas pada hadis, sehingga semakin besar jumlah fitur yang digunakan maka kian banyak fitur yang kurang berpengaruh dapat menyebabkan penurunan performansi dari proses klasifikasi. Namun apabila jumlah fitur yang digunakan terlalu kecil, maka terdapat kemungkinan bahwa fitur yang berpengaruh lainnya tidak terpilih atau tidak digunakan, sehingga juga dapat menyebabkan penurunan performansi dari proses klasifikasi. Pengujian kedua pada skenario ini yaitu membandingkan hasil optimum dari penggunaan *mutual information* sebagai metode *feature selection* dengan proses klasifikasi tanpa *mutual information*. Hasil pengujian kedua pada skenario ini dapat dilihat pada Gambar 5, dimana selisih dari perbandingan tersebut terhadap nilai *hamming loss* yaitu sebesar 0.0307, namun selisih terhadap waktu komputasi sangat signifikan yaitu selama 7.891 detik. Hal ini disebabkan karena tanpa adanya tahap *feature selection*, maka jumlah fitur yang digunakan sangat banyak dan fitur tersebut tidak semuanya menggambarkan atau berpengaruh terhadap suatu kelas, sehingga membuat performansi menurun dan waktu komputasi menjadi lebih tinggi.

Skenario ketiga melakukan pengujian parameter k dalam pembangunan model klasifikasi *k-nearest neighbor* terhadap nilai *hamming loss.* Hasil pengujian pada skenario ini dapat dilihat pada Tabel X, dimana nilai $k = 7$ menghasilkan performansi optimum dengan nilai hamming loss sebesar 0.0886 atau sekitar 91.14% data yang terklasifikasi dengan benar. Berdasarkan hasil tersebut, maka semakin tinggi nilai k maka performansi yang dihasilkan akan meningkat, namun apabila nilai k sudah mencapai performansi yang optimum maka performansi setelahnya cenderung menurun karena semakin besar nilai k maka semakin banyak tetangga yang digunakan untuk melakukan proses klasifikasi, sehingga kemungkinan noise semakin tinggi.

V. KESIMPULAN

Berdasarkan pengujian dan analisis yang telah dilakukan pada beberapa skenario yang dibuat, maka dapat diambil kesimpulan bahwa klasifikasi *multi label* pada hadis Bukhari terjemahan Bahasa Indonesia menggunakan *mutual information* dan *k-nearest neighbor* berhasil dibangun dengan nilai *hamming loss* optimum sebesar 0.0886 atau sekitar 91.14% data yang terklasifikasi dengan benar dengan waktu komputasi selama 595 detik. Rangkaian metode serta parameter yang digunakan pada penelitian ini yaitu menggunakan jumlah fitur sebesar 6% atau sekitar 283 fitur, *pre-processing* tanpa *stemming* dan nilai $k = 7$ sebagai parameter yang digunakan pada proses klasifikasi.

Adapun penggunaan *stemming* pada tahap *pre-processing*, tidak memberikan hasil performansi yang lebih baik dibandingkan dengan proses tanpa *stemming.* Hal ini disebabkan karena dengan digunakannya proses *stemming*, maka setiap kata yang ada pada hadis diubah menjadi kata dasarnya dan menghilangkan ciri khusus yang ada pada data atau hadis. Sedangkan untuk penggunaan *mutual information* sebagai metode *feature selection* terbukti memiliki pengaruh yang baik dibandingkan dengan proses tanpa *mutual information* atau *feature selection*. Hal ini disebabkan karena tanpa adanya tahap *feature selection*, maka jumlah fitur yang digunakan sangat banyak dan fitur tersebut tidak semuanya menggambarkan atau berpengaruh terhadap suatu kelas, sehingga membuat performansi menurun dan waktu komputasi menjadi lebih tinggi.

Beberapa saran yang dapat diterapkan untuk penelitian selanjutnya yaitu melakukan penambahan data yang dilabeli oleh ahli hadis dan pesebaran data, agar selain bertambahnya variasi data, jumlah data dengan tiap label yang ada itu seimbang, sehingga tidak terjadi ketidakseimbangan data. Jika tidak memungkinkan untuk mengatasi ketidakseimbangan data, maka perlu diterapkan suatu metode untuk menangani permasalahan tersebut.

DAFTAR PUSTAKA

[1] K. A. Aldhlan, A. M. Zeki, A. M. Zeki, and H. A. Alreshidi, "Novel mechanism to improve hadith classifier performance," in *Proceedings - 2012 International Conference on Advanced Computer Science Applications and Technologies, ACSAT 2012*, 2013, pp. 512–517.

p-ISSN 2301-7988, e-ISSN 2581-0588
DOI : 10.32736/sisfokom.v9i3.980, Copyright ©2020
Submitted : 5 September 2020, Revised : 8 September 2020, Accepted : 9 September 2020, Published : 24 September 2020

[2] M. D. Purbolaksono, F. D. Reskyadita, Adiwijaya, A. A. Suryani, and A. F. Huda, "Indonesian text classification using back propagation and sastrawi stemming analysis with information gain for selection feature," *Int. J. Adv. Sci. Eng. Inf. Technol.*, no. 1, pp. 234–238, 2020.

[3] G. Mediamer, adiwijaya@telkomuniversity ac id Adiwijaya, and S. Al Faraby, "Development of rule-based feature extraction in multi-label text classification," *Int. J. Adv. Sci. Eng. Inf. Technol.*, vol. 9, no. 4, pp. 1460–1465, 2019.

[4] S. Jiang, G. Pang, M. Wu, and L. Kuang, "An improved K-nearest-neighbor algorithm for text categorization," *Expert Syst. Appl.*, vol. 39, no. 1, pp. 1503–1509, Jan. 2012.

[5] Z. Yong, L. Youwen, and X. Shixiong, "An Improved KNN Text Classification Algorithm Based on Clustering," 2009.

[6] Adiwijaya, M. N. Aulia, M. S. Mubarok, W. Untari Novia, and F. Nhita, "A comparative study of MFCC-KNN and LPC-KNN for hijaiyyah letters pronounciation classification system," in *2017 5th International Conference on Information and Communication Technology, ICoIC7 2017*, 2017.

[7] L. G. Irham, A. Adiwijaya, and U. N. Wisesty, "Klasifikasi Berita Bahasa Indonesia Menggunakan Mutual Information dan Support Vector Machine," *J. Media Inform. Budidarma*, vol. 3, no. 4, p. 284, 2019.

[8] M. A. Ulfa, B. Irmawati, and A. Y. Husodo, "Twitter Sentiment Analysis using Na¨ive Bayes Classifier with Mutual Information Feature Selection," *J. Comput. Sci. Informatics Eng.*, vol. 2, no. 2, pp. 106–111, Dec. 2018.

[9] M. L. Zhang and Z. H. Zhou, "A review on multi-label learning algorithms," *IEEE Transactions on Knowledge and Data Engineering*, vol. 26, no. 8. IEEE Computer Society, pp. 1819–1837, 2014.

[10] S. Al Faraby, E. R. R. Jasin, A. Kusumaningrum, and Adiwijaya, "Classification of hadith into positive suggestion, negative suggestion, and information," in *Journal of Physics: Conference Series*, 2018, vol. 971, no. 1.

[11] H. Prasetyo, Adiwijaya, and W. Astuti, "Klasifikasi Multi -Label pada Hadis Bukhari dalam Terjemahan Bahasa Indonesia Menggunakan Mutual Information dan Backpropagation Neural Network," vol. 6, no. 2, pp. 9086–9098, 2019.

[12] A. O. Adeleke, N. A. Samsudin, A. Mustapha, and N. M. Nawi, "Comparative Analysis of Text Classification Algorithms for Automated Labelling of Quranic Verses," vol. 7, no. 4, 2017.

[13] N. Isnaini, Adiwijaya, M. S. Mubarok, and M. Y. A. Bakar, "A multi-label classification on topics of Indonesian news using K-Nearest Neighbor," in *Journal of Physics: Conference Series*, 2019, vol. 1192, no. 1.

[14] X. F. Zhang, H. Y. Huang, and K. L. Zhang, "KNN text categorization algorithm based on semantic centre," in *Proceedings - 2009 International Conference on Information Technology and Computer Science, ITCS 2009*, 2009, vol. 1, pp. 249–252.

[15] 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.

[16] R. Bintang Purnomoputra and U. Novia Wisesty, "Sentiment Analysis of Movie Reviews using Naïve Bayes Method with Gini Index Feature Selection," *OPEN ACCESS J DATA SCI APPL*, vol. 2, no. 2, pp. 85–094, Nov. 2019.

[17] J. Kaur and J. Saini, "A Study of Text Classification Natural Language Processing Algorithms for Indian Languages," *VNSGU J. Sci. Technol.*, vol. 4, no. 1, pp. 162–167, 2015.

[18] R. A. Pane, M. S. Mubarok, N. S. Huda, and Adiwijaya, "A multi-lable classification on topics of Quranic verses in English translation using multinomial naive bayes," in *2018 6th International Conference on Information and Communication Technology, ICoICT 2018*, 2018, pp. 481–484.

[19] A. M. K. Izzaty, M. S. Mubarok, N. S. Huda, and Adiwijaya, "A multi-label classification on topics of quranic verses in English translation using Tree Augmented Naïve Bayes," in *2018 6th International Conference on Information and Communication Technology, ICoICT 2018*, 2018, pp. 103–106.

[20] M. L. Zhang, J. M. Peña, and V. Robles, "Feature selection for multi-label naive Bayes classification," *Inf. Sci. (Ny).*, vol. 179, no. 19, pp. 3218–3229, Sep. 2009.

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

[22] G. Doquire and M. Verleysen, "Mutual information-based feature selection for multilabel classification," *Neurocomputing*, vol. 122, pp. 148–155, Dec. 2013.

[23] A. Yusuf and T. Priambadha, "Support Vector Machines yang Didukung K-Means Clustering dalam Klasifikasi Dokumen," *JUTI J. Ilm. Teknol. Inf.*, vol. 11, no. 1, p. 15, Jan. 2013.

[24] Suyanto, *Data Mining Untuk Klasifikasi dan Klasterisasi Data*, 1st ed. Penerbit Informatika, 2017.

[25] N. Octaviani Faomasi Daeli, "Sentiment Analysis on Movie Reviews Using Information Gain and K-Nearest Neighbor," *OPEN ACCESS J DATA SCI APPL*, vol. 3, no. 1, pp. 1–007, May 2020.

[26] T. T. Wong, "Performance evaluation of classification algorithms by k-fold and leave-one-out cross validation," *Pattern Recognit.*, vol. 48, no. 9, pp. 2839–2846, Sep. 2015.

[27] I. K. Syuriadi, W. Astuti, F. Informatika, and U. Telkom, "Klasifikasi Teks Multi Label pada Hadis dalam Terjemahan Bahasa Indonesia Berdasarkan Anjuran , Larangan dan Informasi menggunakan TF-IDF dan KNN."

[28] 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," in *Proceedings of the 2018 International Conference on Asian Language Processing, IALP 2018*, 2019, pp. 344–350.

p-ISSN 2301-7988, e-ISSN 2581-0588
DOI : 10.32736/sisfokom.v9i3.980, Copyright ©2020
Submitted : 5 September 2020, Revised : 8 September 2020, Accepted : 9 September 2020, Published : 24 September 2020