PENERAPAN METODE SVM UNTUK SISTEM TANYA JAWAB PADA KASUS *FRONT-OFFICE*

Tommy Edityantama Hutapea[1], Kania Evita Dewi[2]
[1,2] Universitas Komputer Indonesia
Jl.Dipatiukur No. 112-116, Coblong, Bandung, Jawa Barat, 40132
E-mail : t.edityantama@email.unikom.ac.id[1], kania.evita.dewi@email.unikom.ac.id[2]

ABSTRAK

Sistem Tanya Jawab *(Question Answering System)* merupakan sebuah sistem pada komputer yang dapat menjawab pertanyaan menggunakan bahasa alami yang biasa dipakai manusia secara otomatis. Penelitian sebelumnya tentang sistem tanya jawab sudah dilakukan sebelumnya salah satunya menggunakan rule based menggunakan bahasa indonesia, namun pada penelitian tersebut belum menghasilkan akurasi yang baik yaitu sebesar 60%. Adapun penelitian lain tentang sistem tanya jawab yang menggunakan metode pendekatan *machine learning* yaitu metode SVM, namun pada penelitian tersebut menghasilkan akurasi sebesar 80% menggunakan bahasa Inggris, bahasa Arab, dan bahasa Inggris-Arab. Sehingga pada penelitian ini, akan menerapkan metode pendekatan *machine learning* SVM untuk sistem tanya jawab di *front office* dengan Bahasa Indonesia. Tahapan yang dilakukan sistem untuk mendapatkan hasil jawaban dibagi menjadi tiga tahap, yaitu *Preprocessing*, Klasifikasi dan Prediksi jawaban. Klasifikasi adalah tahapan awal dimana data latih akan diklasifikasi sesuai dengan jenis pertanyaan. kemudian mencari jawaban yang ada di dalam database berdasarkan kedekatan kalimat pertanyaan dengan yang ada pada database. Hasil dari main process ini adalah jawaban yang telah diisikan sebelumnya pada setiap data latih. Hasil menunjukkan akurasi yang didapat pada pendekatan *machine learning* svm untuk sistem tanya jawab pada kasus front office mendapat nilai akurasi sebesar 39%. Nilai akurasi tersebut disebabkan karena kurangnya data latih.

Kata kunci : Sistem Tanya Jawab, *Support Vector Machine*, Bahasa Indonesia, SVM

1. PENDAHULUAN

Sistem Tanya Jawab (Question Answering System) merupakan sebuah sistem pada komputer yang dapat menjawab pertanyaan menggunakan bahasa alami yang biasa dipakai manusia secara otomatis. Jawaban yang diperoleh dari sistem berisi informasi yang berasal dari sebuah sumber basis data [1]. Sebuah sistem tanya jawab, diharapkan dapat membantu orang untuk mendapatkan informasi lebih mengenai hal tertentu tanpa harus bertanya kepada orang lain [2].

Pada penelitian sebelumnya tentang sistem tanya jawab dalam bahasa Indonesia sudah dilakukan, salah satunya dengan menggunakan rule base [2] [3]. Pada penelitian tersebut belum menghasilkan akurasi yang cukup baik yaitu 60% dan memiliki kekurangan yaitu hanya mendeteksi kata kunci yang terdapat pada rule, sehingga jika menemukan kata kunci selain pada rule maka menghasilkan jawaban tidak relevan.

Mengatasi kata kunci yang tidak terdapat pada rule dapat diterapkan metode pendekatan Machine Learning [4]. Pada penelitian tersebut menjelaskan dengan menggunakan pendekatan Machine Learning dapat menghasilkan sistem yang mencakup ribuan fitur pertanyaan dan mengklasifikasikan pertanyaan tersebut secara otomatis dan meningkatkan produktivitas sistem tanya jawab.

Adapun penelitian lain tentang sistem tanya jawab yang menggunakan 5 metode pendekatan Machine Learning yaitu K-Nearest Neighbor, Naive Baiyes, Support Vector Machine (SVM), Sparse Network of Winnows (SNoW) dan Decision Tree (DT) [5]. Pada penelitian tersebut metode SVM menghasilkan akurasi sebesar 87,4%. Penelitian tersebut menggunakan bahasa Inggris.

Adapun penelitian lainnya mengenai sistem tanya jawab dengan menggunakan metode SVM. Hasil yang didapat yakni nilai akurasi diatas 80-90% [6] [7] [8] [9].Pada penelitian tersebut menggunakan bahasa Inggris-Arab.

Oleh karena itu, pada penelitian ini bermaksud untuk mengembangkan sistem tanya jawab pada Front Office UNIKOM dengan menerapkan metode pendekatan machine learning Support Vector Machine (SVM) dalam bahasa indonesia. Penelitian ini diharapkan dapat menghasilkan dan akurasi yang cukup baik.

1.1. Sistem Tanya Jawab

Sistem Tanya Jawab (*Question Answering System*) merupakan sebuah sistem pada komputer yang dapat menjawab pertanyaan menggunakan bahasa alami yang biasa dipakai manusia secara otomatis. Jawaban yang diperoleh dari sistem berisi informasi yang berasal dari sebuah sumber basis data [1]. Sebuah sistem tanya jawab, diharapkan

dapat membantu orang untuk mendapatkan informasi lebih mengenai hal tertentu tanpa harus bertanya kepada orang lain [2].

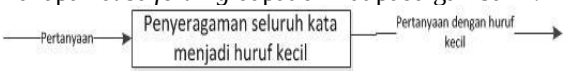
Tujuan utama dari sistem jawab otomatis adalah untuk mencari jawaban yang tepat menggunakan bahasa alami yang ditentukan oleh manusia. Saat ini sistem tanya jawab telah dikembangkan dalam berbagai bidang seperti pada asisten pribadi, pelayanan konsumen yang diterapkan pada pelayanan jasa guna untuk mendapatkan informasi dan memenuhi kebutuhan konsumen.

1.2. *Preprocessing*

Preprocessing adalah proses pengolahan data raw sebelum diolah pada tahap selanjutnya. Pada penelitian ini data yang digunakan adalah data pertanyaan berupa text dari hasil wawancara dari kuesioner yang tidak terstruktur. Oleh karena itu, proses ini sangat dibutuhkan karena akan digunakan untuk mengubah data yang tidak terstruktur menjadi terstruktur (normalisasi) agar dapat diproses pada tahap selanjutnya. Pada penelitian ini tahapan dalam *preprocessing* terdiri dari *casefolding*, *cleansing* (penghilangan simbol), *tokenizing* dan *remove stopword*, dan pembobotan TF-IDF.

1.2.1. *Case Folding*

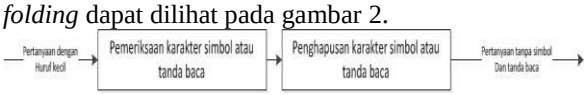
Case Folding adalah proses menyeragamkan semua teks ke dalam case sama menjadi huruf kecil (lowercase) atau huruf besar (uppercase) [10]. Tahapan *case folding* dapat dilihat pada gambar 1.



Gambar 1 Proses Tahapan *Case Folding*

1.2.2. *Cleansing*

Cleansing merupakan proses membersihkan kata-kata yang tidak diperlukan untuk mengurangi noise. Kata yang dihilangkan adalah url, hashtag (#), username (@username), dan email. Selain itu juga tanda baca seperti titik(.), koma(,), dan tanda baca lainnya akan dihilangkan. [10]. Tahapan *case folding* dapat dilihat pada gambar 2.



Gambar 2 Proses Tahapan *Cleansing*

1.2.3. *Tokenizing*

Tokenizing adalah proses untuk memisahkan kata dalam teks menjadi beberapa urutan kata yang dipisahkan melalui spasi atau karakter lainnya [10]. Tahapan *case folding* dapat dilihat pada gambar 3.



Gambar 3 Proses Tahapan *Tokenizing*

1.2.4. *Remove Stopwords*

Remove Stopword merupakan proses penghilangan kata tidak penting yang terdapat dalam *stoplist*. Kata-kata yang dihilangkan tersebut menunjukan kurang relevansinya dengan teks dan didefinisikan dalam *stopword list* [10]. Tahapan *Remove Stopwords* dapat dilihat pada gambar 4.



1.2.5. Pembobotan TF-IDF

Metode TF-IDF merupakan metode untuk menghitung bobot setiap kata yang paling umum digunakan pada *information retrieval*. Metode ini juga terkenal efisien, mudah dan memiliki hasil yang akurat [11]. Metode ini akan menghitung bobot setiap *term* dalam sebuah dokumen dengan rumus.

$$w_i = tf_i * IDF_{t_i} \quad \textbf{(1)}$$

Dimana :

d : dokumen ke-d

t : *term* ke-t dari kata kunci

tf_i : *term* frekuensi/ frekuensi kata

W_i : bobot dokumen ke-d terhadap term ke-t

Nilai IDF_{t_i} didapatkan dari:

$$IDF_{t_i} = log(\frac{D}{df_i}) \quad \textbf{(2)}$$

Dimana:

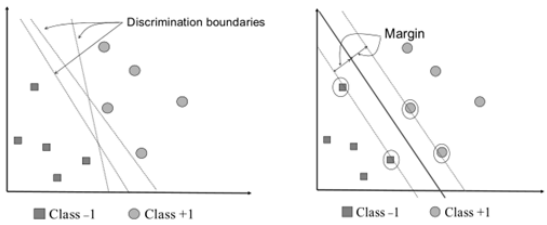
D : total keseluruhan dokumen,

df_i : banyaknya dokumen yang mengandung kata yang dicari.

1.3. *Support Vector Machine*

Support Vector Machine (SVM) adalah sistem pembelajaran yang menggunakan ruang hipotesis berupa fungsi-fungsi *linier* dalam sebuah ruang fitur (*feature space*) berdimensi tinggi, dilatih dengan algoritma pembelajaran yang didasarkan pada teori optimasi dengan mengimplementasikan *learning bias* yang berasal dari teori pembelajaran statistik. Teori yang mendasari SVM sendiri sudah berkembang sejak 1960-an, tetapi baru diperkenalkan oleh Vapnik, Boser dan Guyon pada tahun 1992 dan sejak itu SVM berkembang dengan pesat.

Konsep klasifikasi dengan SVM dapat dijelaskan secara sederhana sebagai usaha mencari hyperplane terbaik yang berfungsi sebagai pemisah dua buah kelas data pada ruang input. Gambar 1 memperlihatkan beberapa pola yang merupakan anggota dari dua buah class : +1 dan –1. data yang tergabung pada kelas –1 disimbolkan dengan warna merah (kotak), sedangkan pattern pada class +1, disimbolkan dengan warna kuning(lingkaran) [12].



Gambar 4 Batas Keputusan SVM

Adapun data yang berada pada bidang pembatas ini disebut *support vector*. Dalam contoh di atas, dua kelas dapat dipisahkan oleh sepasang bidang pembatas yang sejajar. Bidang pembatas pertama membatasi kelas pertama sedangkan bidang batas kedua membatasi kelas kedua [13].

Setiap data latih dinyatakan oleh (x_i, y_i), dimana *i*=1,2,...,N, dan $x_i = \{x_{i1}, x_{i2}, .., x_{iq}\}^T$ merupakan atribut (fitur) set untuk data latih ke-*i*, $y_i \in \{-1,+1\}$ menyatakan label kelas. *Hyperplane* klasifikasi linear SVM, dinotasikan dengan

$$\boldsymbol{w}.\boldsymbol{x}_i + b = 0 \quad \textbf{(3)}$$

w dan b adalah parameter model. $\boldsymbol{w}.\boldsymbol{x}_i$ merupakan inner-product dalam antara w dan xi. Data xi yang masuk ke dalam kelas ke -1 adalah data yang memenuhi pertidaksamaan berikut:

$$\boldsymbol{w}.\boldsymbol{x}_i + b \leq -1 \quad \textbf{(4)}$$

Sementara data xi yang masuk ke dalam kelas +1 adalah data yang memenuhi pertidaksamaan berikut:

$$\boldsymbol{w}.\boldsymbol{x}_i + b \geq +1$$

Margin optimal dihitung dengan memaksimalkan jarak antara hyperplane dan data terdekat. Masalah ini diformulasikan ke dalam *Quadratic Programming* (QP) problem, dengan meminimalkan invers persamaan. Optimalisasi ini dapat diselesaikan dengan Lagrange Multiplier dengan persamaan 5.

$$\max_{\alpha} L_D = \sum_{i=1}^{n} \alpha_i - \frac{1}{2} \sum_{i=1,j=1}^{n} \alpha_i \alpha_j y_i y_j x_i x_j \quad \textbf{(5)}$$

1.4. *Cosine Simlarity*

Cosine similarity digunakan untuk menghitung pendekatan relevansi *query* terhadap dokumen. Semakin besar nilai kesamaan vektor *query* dengan vektor dokumen maka *query* tersebut dipandang semakin relevan dengan dokumen. Cosinus dari 0^0 adalah 1, dan kurang dari 1 untuk setiap sudut lainnya. Dengan demikian dua vektor dengan orientasi yang sama memiliki kesamaan cosinus dari 1, dua vektor pada 90^0 memiliki kesamaan 0. Cosine similarity terutama digunakan dalam ruang positif, dimana hasilnya dibatasi (0,1). Cosine similarity kemudian memberi tolak ukur seberapa mirip dua dokumen [14].

$$CS(d1,d2) = \frac{\sum_{t=1}^{n} W_{t.d_1} W_{t.d_2}}{\sqrt{\sum_{t=1}^{n} W_{t.d_1}^2} \sqrt{\sum_{t=1}^{n} W_{t.d_2}^2}} \quad \textbf{(6)}$$

Dimana :

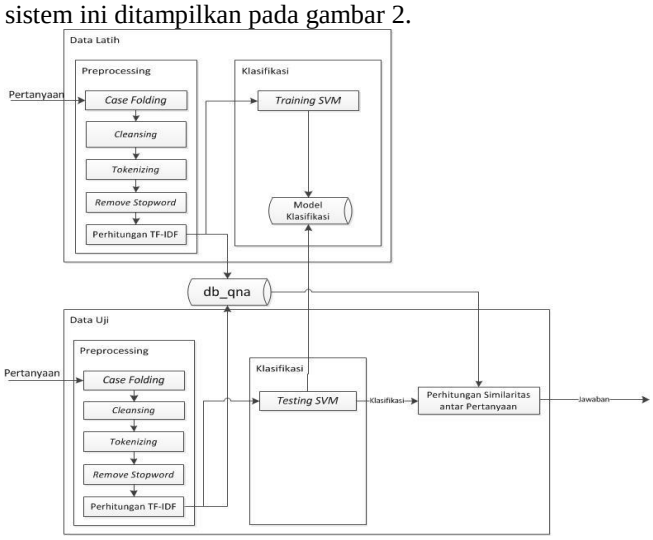
$W_{t.d_1}$ = bobot term t dalam dokumen d_1.
$W_{t.d_2}$ = bobot term t dalam dokumen d_2.

2. ISI PENELITIAN

Pada penelitian ini, akan mengembangkan sistem tanya jawab pada kasus *front-office* dengan menerapkan metode SVM. Data masukan berupa pertanyaan akan melalui tahap preprocessing, kemudian klasifikasi dengan metode SVM, dan pencarian jawaban dengan metode *cosine similarity*. Adapun jumlah data yang digunakan dalam penelitian ini, yaitu 146 data yang terdiri dari 82 data latih dan 64 data uji.

2.1. Analisis Sistem

Adapun beberapa proses yang akan berjalan pada sistem ini, dimana dalam sistem ini bagi menjadi beberapa bagian yaitu pelatihan dan pengujian. Berikut Proses yang dilakukan dalam sistem ini ditampilkan pada gambar 2.



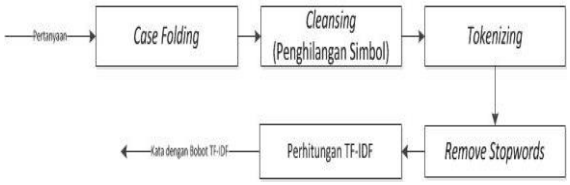
Gambar 5 Arsitektur Sistem

2.2. Analisis Data Masukan

Data Masukan yang digunakan adalah data berupa pertanyaan. Data masukan tersebut akan melalui proses tahapan *preprocessing*.

2.2. Analisis Tahap *Preprocessing*

Ada beberapa tahapan yang dilakukan dalam tahap *preprocessing*. Tahap *preprocessing* yang dilakukan adalah *Case Folding*, *Cleansing*, *Tokenizing*, *Remove Stopword*, dan pembobotan TF-IDF. Berikut gambaran dari tahapan *Preprocessing* yang ditampilkan pada gambar 3.



Gambar 6 Tahapan *Preprocessing*

2.2.1. Case Folding

Case Folding adalah proses menyeragamkan semua teks ke dalam case sama menjadi huruf kecil (lowercase) atau huruf besar (uppercase) [10].

2.2.2. Cleansing

Cleansing merupakan proses membersihkan kata-kata yang tidak diperlukan untuk mengurangi noise. Kata yang dihilangkan adalah url, hashtag (#), username (@username), dan email. Selain itu juga tanda baca seperti titik(.), koma(,), dan tanda baca lainnya akan dihilangkan. Pada penelitian ini, proses *cleansing* untuk menghilangkan simbol seperti tanda baca dan *emoticon* dalam teks. Berikut daftar cleansing yang ditampilkan pada tabel 1.

Tabel 1 Daftar *Cleansing*

Karakter	Keterangan
?	Tanda tanya
!	Tanda Seru
“	Tanda Kutip
☺	Ekspresi Senyum
☹	Ekspresi sedih
😐	Ekspresi Datar
.	titik
,	Koma
;	Titik koma

2.2.3. Tokenizing

Tokenizing adalah proses untuk memisahkan kata dalam teks menjadi beberapa urutan kata yang dipisahkan melalui spasi atau karakter lainnya. Pada penelitian ini, karakter pemisah menggunakan (‘ ‘) spasi.

2.2.4. Remove Stopwords

Remove Stopword merupakan proses penghilangan kata tidak penting yang terdapat dalam *stoplist*. Kata-kata yang dihilangkan tersebut menunjukan kurang relevansinya dengan teks dan didefinisikan dalam *stopword list*. Berikut daftar Stopword/Stoplist yang ditampilkan pada tabel 2.

Tabel 2 Daftar *Stopword/Stoplist*

No	*Stopwords*	No	*Stopwords*
1	yang	18	dari
2	untuk	19	menjadi
3	saya	20	masih
4	harus	21	aja
5	saja	22	itu
6	terdapat	23	dalam
7	setiap	24	pada
8	di	25	ini
9	tentang	26	yg
10	ada	27	ke
11	jika	28	mau
12	cara	29	kok
13	tetap	30	soalnya
14	dengan	31	apakah
15	hanya	32	katanya
16	seperti	33	kata
17	atau		

2.2.5. Pembobotan dengan TF-IDF

Pada tahap ini, setelah proses remove stopwords akan dilakukan proses pembobotan dengan menggunakan TF-IDF. Proses pembobotan yang dilakukan terhadap term pada kalimat pertanyaan dengan rumus (1), sehingga didapatkan bobot masing-masing term pada kalimat yang nanti akan digunakan proses pelatihan dengan SVM.

2.3. *Support Vector Machine*

Pada penelitian ini, proses klasifikasi menggunakan metode SVM. Klasifikasi dilakukan setelah proses *preprocessing.* Kelas Klasifikasi yang digunakan adalah 7 kelas dan metode SVM yang digunakan adalah One Vs One dengan kernel RBF. Adapun 2 tahapan proses klasifikasi yaitu proses pelatihan dan pengujian. Berikut adalah penjelasan dari proses pelatihan.

2.3.1. Analisis Pelatihan SVM

Pada proses tahapan pelatihan SVM, data masukan yaitu data kelas dan data latih. Dimana pada data latih berjumlah 82 data latih. Kemudian, dilanjutkan pada proses kernelisasi dan perhitungan untuk mendapatkan nilai support vector dan mendapatkan bidang pemisah untuk proses pengujian data. Adapun proses tahapan pelatihan SVM yang digambarkan pada gambar 4.



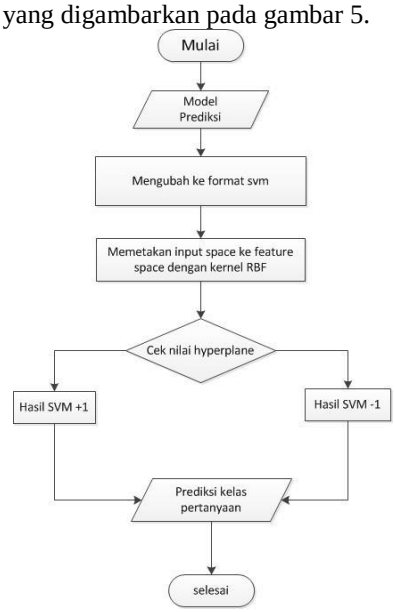
Gambar 7 Proses Tahapan Pelatihan SVM

2.3.2. Analisis Pengujian SVM

Pada Analisis pengujian data, data uji di proses dengan tahapan *preprocessing* hingga proses pembobotan TF-IDF seperti pada proses pelatihan. Kemudian akan dilakukan klasifikasi dengan SVM.

Pada metode *Multiclass Support Vector Machine One Vs One* dibangun sejumlah $k = \frac{k(k-1)}{2}$

buah SVM biner, dengan k adalah jumlah kelas. Untuk setiap model klasifikasi ke-i dilatih dengan menggunakan keseluruhan data, untuk mencari solusi permasalahan.Dimana pada data uji berjumlah 64 data. Kemudian, dilanjutkan pada proses kernelisasi dan perhitungan dengan bidang pemisah yang didapatkan dari pelatihan data sebelumya. Hasil Akhir dari pengujian SVM adalah keputusan kelas pertanyaan. Adapun proses tahapan pengujian SVM yang digambarkan pada gambar 5.



Gambar 8 Proses Tahapan Pengujian SVM

2.4. *Cosine Similarity*

Pada penelitian ini, metode cosine similarity digunakan sebagai pencarian jawaban data pengujian. Metode ini akan mencari kedekatan antara data uji terhadap data latih. Hasil akhir dari metode ini adalah prediksi jawaban terhadap data uji.

2.6. Pengujian Akurasi

Pada pengujian akurasi pada penelitian sistem tanya jawab dengan metode SVM dihitung dari ketepatan dari pertanyaan dan jawaban yang diberikan. Pada pengujian ini diberikan data latih sebanyak 82 pertanyaan dan data uji sejumlah 64. Adapun Hasil Akurasi yang didapatkan yang ditampilkan pada tabel 3.

Tabel 3 Hasil Pengujian

Proses	Jumlah Data pertanyaan yang digunakan
Pelatihan	82
Pengujian	64

Jumlah jawaban yang benar = 25
Jumlah klasifikasi pertanyaan yang benar = 40
Jumlah pertanyaan = 64

$$akurasi = \frac{jumlah\ jawaban\ yang\ benar}{jumlah\ pertanyaan} x100\%$$

$$akurasi = \frac{25}{64} x100\%$$
$$akurasi = 39\%$$

3. PENUTUP

3.1. KESIMPULAN

Hasil kesimpulan dari pengujian menunjukkan penerapan metode svm untuk sistem tanya jawab pada kasus front-office menghasilkan akurasi 39% dan data pertanyaan yang diklasifikasi yang menghasilkan akurasi 40%. Hal ini dipengaruhi jumlah pada data latih beserta pola bahasa yang berbeda dari data uji terhadap data latih walaupun memiliki maksud yang sama terhadap data latih.

3.2. SARAN

Dalam penelitian tugas akhir ini, masih banyak kekurangan yang kedepannya dapat diperbaiki untuk penelitian yang lebih lanjut. Adapun saran penulis yang ingin disampaikan adalah :

1. Untuk penelitian selanjutnya disarankan menambah data untuk pelatihan.
2. Untuk penelitian selanjutnya disarakan untuk tidak mencoba menggunakan *stopwords* apakah mempengaruhi akurasi atau tidak.
3. Disarankan untuk menggunakan metode klasifikasi yang berbeda seperti Fuzzy KNN-C.

DAFTAR PUSTAKA

[1] R. A. Ewerton J. Wantroba, "An Interactive Question Answer System with dialogue for a Receptionist avatar," *2th Latin American Robotics Symposium, 2015 Third Brazilian Symposium on Robotic*, pp. 360-365, 2015.

[2] Y. K. Maharlika, "Rule Based Method Implementation For Question Answerin System in A Front Office Study Case," *Perpustakaan UNIKOM*, 2016.

[3] R. H. Gusmita, Y. Durachman, S. Harun, A. F. Firmansyah, H. T. Sukmana dan A. Suhaemi, "A Rule-Based Question Answering System on Relevant Documents of Indonesian Quran Translation," *International Conference on Cyber and IT Service Management (CITSM)*, pp. 104-107, 2014.

[4] F. S. Gharehchopogh dan Y. Lotfi, "Machine Learning based Question Classification Methods in the Question Answering Systems," *International Journal of Innovation and Applied Studies*, vol. 4, pp. 264-273, 2013.

[5] A. Mathur dan M. Haider, "Question Answering System : A Survey," *International Conference on Smart Technologies and Management*, pp. 44-57, 2015.

[6] H. Abdelmasser, M. Ragah, B. Farouk, R. Mohamed, A. Mohammed and N. El-Makky,

"Al-Bayan : An Arabic Question Answering for the Holy Quran," *Workshop on Arabic Natural Language Processing (ANLP)*, pp. 57-64, 2014.

[7] W. Ahmed, A. PV dan B. Anto, "Web-Based Arabic Question Answering System using Machine Learning Approach," *International Journal of Advanced Computer Science*, vol. 8, 2017.

[8] S.-J. Yen, Y.-C. Wu, J.-C. Yang, Y.-S. Lee, C.-J. Lee dan J.-J. -Liu, "A Support Vector Machine-Based Context Ranking Model For Question Answering," *Information Sciences 224*, pp. 77-87, 2013.

[9] N. Othman dan R. Faiz, "Question Answering Passage Retrieval and Re-rangking Using N-grams and SVM," *Computacion y Sistemas*, vol. 20, pp. 483-494, 2016.

[10] A. M. Bachtiar dan A. R. Sentiaji, "Analisis Sentimen Terhadap Acara Televisi Berdasarkan Opini Publik," *Jurnal Ilmiah Komputer dan Informatika*, 2014.

[11] A. A. Maarif, "Penerapan Algoritma TF-IDF Untuk Pencarian Karya Ilmiah," 2015.

[12] E. Prasetyo, Data Mining, Mengolah Data Menjadi Informasi Menggunakan Matlab, Yogyakarta: ANDI, 2014.

[13] K. Sembiring, "Penerapan Teknik Support Vector Machine untuk Pendeteksian Intrusi pada jaringan," pp. 2-26, 2007.

[14] J. Pardede dan J. Sinataria, "Implementasi Maximum Marginal Relevance Dan Matriks Cosine Similarity Pada Aplikasi Peringkasan Dokumen," *Itenas Library*, pp. 1349-1352.

[15] Suprapto, "Bahasa Pemrograman untuk Sekolah Menengah Kejuruan," Direktorat Pembinaan Sekolah Menengah Kejuruan Direktorat Jenderal Manajemen Pendidikan Dasar dan Menengah Departemen Pendidikan Nasiona, 2008.

[16] R. A. Yuana, "Dasar-dasar Pemrograman PHP".

[17] M. Fowler, UML Distilled Edisi 3, Panduan Singkat Bahasa Pemodelan Objek Standar, vol. 53, Yogyakarta: ANDI Jogyakarta, 2013, pp. 1689-1699.

[18] I. Fathansyah, Basis Data, Bandung: Informatika Bandung, 2016.

[19] T. Karlita, S.Kom.,M.Kom, A. S. Ahsan, S.Kom dan A. R. Barakbah, S.Kom.,Ph.D, "Logika dan Algoritma," 2013.

[20] C.-W. Hsu dan C.-J. Lin, "A Comparison of Methods for Multi-Class Support Vector Machines," *Department of Computer Science and Informatics Engineering Nationa Taiwan University*, 2002.

[21] O. Nurdiana, J. dan D. Nursantika, "Perbandingan Metode Cosine Similarity dengan Metode Jaccard Similarity pada Aplikasi Pencarian Terjemah Al-Qur'an dalam Bahasa Indonesia," vol. 1, pp. 59-63, 2016.

[22] E. Prasetyo, Data Mining Konsep dan Aplikasi menggunakan Matlab, Yogyakarta: ANDI Yogyakarta, 2012.

[23] H. Tohari, Astah : Analisis serta perancangan sistem informasi melalui pendekatan UML, Yogyakarta: Andi Yogyakarta, 2014.

[24] "UML : ATM Case Study".

[25] A. Sunyoto, Membangun Web dengan Teknologi Asyncronous, Yogyakarta: ANDI, 2007.

[26] I. H. Witten, Text Mining, New Zealand, 2002.

[27] J. M. Pricila, "Comparison of Approaches multiclass SVM classification Articles Speak Indonesia," *Perpustakaan UNIKOM*, 2016.

[28] R. S. Pressman,Ph.D dan B. R. Maxim,Ph.D, Software Engineering, A Practicioner's Approach, Eight Edition, Mc Graw Hill Education, 2015.

[29] I. Sommerville, Software Engineering Edisi 8 (Rekayasa Perangkat Lunak), New York: Mc Graw Hill, 2015.

[30] Sugiyono, Metode Penelitian Kuantitatif, Kualitatif Dan R&D, Alfabeta, 2012.