SISTEM TANYA JAWAB OTOMATIS DENGAN RELEVANCE VECTOR MACHINE STUDI KASUS: FRONT OFFICE

Gingga Ismu Muttaqin Hadiko[1], Ken Kinanti Purnamasari[2]

[1,2]Universitas Komputer Indonesia
Jl. Dipatiukur No. 112-116, Bandung 40132
E-mail : gingga_ismu_m@email.unikom.ac.id[1], ken.kinanti@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. Jawaban yang diperoleh dari sistem berisi informasi yang berasal dari sebuah sumber basis data. Sebuah sistem tanya jawab, diharapkan dapat membantu orang untuk mendapatkan informasi lebih mengenai hal tertentu tanpa harus bertanya kepada orang lain. Tujuan utama dari sistem jawab otomatis adalah untuk mencari jawaban yang tepat menggunakan bahasa alami yang ditentukan oleh manusia. Penelitian sebelumnya menggunakan metode SVM (Support Vector Machine) menghasilkan akurasi pengklasifikasian sebesar 46% dan akurasi jawaban sebesar 39%. Penelitian ini menggunakan metode RVM (Relevance Vector Machine) untuk mengklasifikasi pertanyaan ke dalam 7 kelas. Sebelum memasukin tahap utama, tahapan dalam *preprocessing* terdiri dari *case folding*, *cleansing* (penghilangan simbol), *tokenizing*, *stopword removal*, dan *stemming*. Hasil penelitian menunjukan bahwa akurasi pengklasifikasian sebesar 71,21% dan akurasi jawaban sebesar 66,67% .

Kata Kunci : Sistem Tanya Jawab, Relevance Vector Machine, One VS One, Cosine Similarity.

1. PENDAHULUAN

Suatu informasi dapat diperoleh dengan cara bertanya. Namun setiap orang memiliki keterbatasan dalam hal informasi dan waktu yang dimiliki untuk menjawab. Maka, sistem jawab otomatis dibangun dengan tujuan utamanya adalah untuk mencari jawaban yang tepat menggunakan bahasa alami yang ditentukan oleh manusia [5]. Jawaban yang diperoleh dari sistem berisi informasi yang berasal dari sebuah sumber basis data [4].

Penelitian sebelumnya, sistem tanya jawab pada kasus front office UNIKOM telah dilakukan dengan menggunakan metode SVM. Penelitian tersebut menghasilkan akurasi jawaban sebesar 39%. Salah satu faktor penyebab rendahnya akurasi jawaban terletak pada tahap klasifikasi pertanyaan. Pada tahap tersebut SVM mengklasifikasi pertanyaan dengan akurasi hanya sebesar 46% [2].

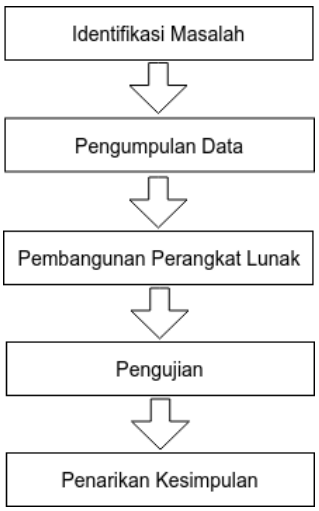
Oleh karena itu pada penelitian ini akan mencoba menggunakan metode klasifikasi yang berbeda. Berdasarkan penelitian yang dilakukan oleh Rafi dan Shaikh [3], metode RVM memiliki akurasi lebih baik dibandingkan SVM dengan hasil persentase f-measure untuk RVM adalah 92,33%, sedangkan untuk SVM adalah 90,29%. Berdasarkan keunggulan yang dimiliki RVM dibanding SVM, maka penelitian ini akan menganalisis akurasi pengklasifikasian pertanyaan dengan menggunakan RVM. Sehingga diharapkan dengan metode ini dapat menghasilkan akurasi pengklasifikasian dan jawaban yang lebih baik.

2. ISI PENELITIAN

Bagian ini menjelaskan penelitian yang dilakukan, mencakup metode penelitian, arsitektur sistem, *preprocessing*.

2.1 Metode Penelitian

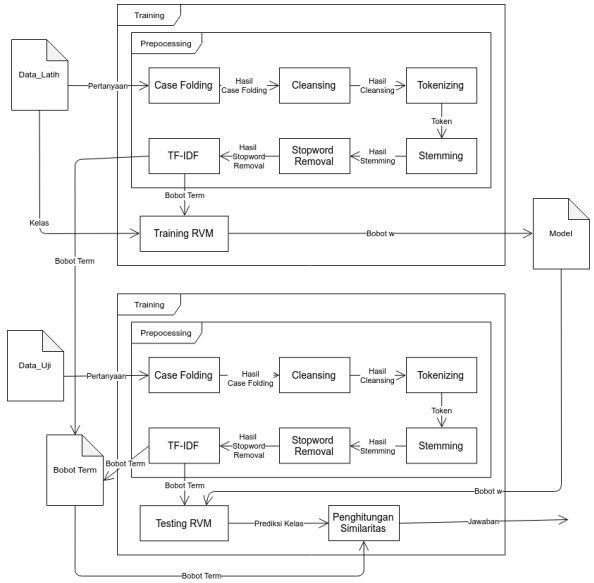
Metode yang akan digunakan pada penelitian ini adalah metode eksperimen. Metode ini dipilih karena pengujian hipotesis akan dilakukan menggunakan serangkaian percobaan. Adapun gambaran tahapan dalam penelitian ini seperti berikut.



Gambar 1. Tahapan Penelitian

2.2 Arsitektur Sistem

Sistem ini terdiri dari dua bagian utama, yaitu training dan testing. Data masukan berupa teks pertanyaan. Kemudian data tersebut diolah di tahap preprocessing untuk mendapatkan nilai bobot. Di tahap preprocessing terdiri dari case folding, cleansing, tokenizing, stopword removal, dan pembobotan. Pada bagian training, nilai bobot term akan dipetakan ke kelasnya masing-masing menggunakan metode RVM. Kelas-kelas tersebut dikelompokan berdasarkan jenis pertanyaan. Kemudian hasil dari training berupa nilai bobot w disimpan dalam sebuah file. Sedangkan pada bagian testing, nilai bobot term akan diprediksi kelasnya berdasarkan model yang telah dibuat. Prediksi kelas tersebut akan digunakan sebagai penentu data dari kelas mana yang akan digunakan pada tahap cosine similarity. Berikut ini merupakan gambaran secara umum sistem yang dibangun.



2.3 Preprocessing

Preprocessing adalah proses pengolahan data mentah sebelum diolah pada tahap utama. Tahapan dalam preprocessing pada penelitian ini terdiri dari *case folding*, *cleansing* (penghilangan simbol), *tokenizing*, *stopword removal*, dan *stemming*.

2.3.1 *Case Folding*

Case folding adalah proses menyeragamkan huruf dalam bentuk huruf kapital (uppercase) atau huruf kecil (lowecase) [6]. Berikut contoh case folding.

Tabel 1. Contoh Case Folding

Sebelum	ada berapakah fakultas di unikom?
Sesudah	ada berapakah fakultas di unikom

1.1.1 *Cleansing*

Cleansing adalah proses menghilangkan tanda baca [6]. Berikut contoh cleansing.

Tabel 2. Contoh Cleansing

Sebelum	ada berapakah fakultas di unikom?
Sesudah	ada berapakah fakultas di unikom

2.3.2 *Tokenizing*

Tokenizing adalah proses memecah kalimat menjadi beberapa kata [6]. Berikut contoh tokenizing.

Tabel 3. Contoh Tokenizing

Sebelum	ada berapakah fakultas di unikom
Sesudah	ada
	berapakah
	fakultas
	di
	unikom

2.3.3 *Stemming*

Stemming adalah proses mengubah suatu kata menjadi kata dasarnya [6]. Berikut contoh s*temming*.

Tabel 4. Contoh Stemming

Sebelum	**Sesudah**
ada	ada
berapakah	berapa
fakultas	fakultas
di	di
unikom	unikom

2.3.4 *Stopword Removal*

Stopword Removal adalah proses menghilangkan kata yang dianggap tidak penting. Berikut contoh s*topword* s*emoval* [6].

Tabel 5. Contoh Stopword Removal

Sebelum	Sesudah
ada	
berapa	berapa
fakultas	fakultas
di	
unikom	unikom

2.3.5 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 [7]. Tahap pertama dalam menghitung bobot dengan TF-IDF adalah menghitung banyaknya suatu kata dalam sebuah pertanyaan, *term frequency* (TF). Berikut contoh TF pada kata "kapan".

Term	tf																			
	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21
kapan	1	1	0	1	0	0	1	1	1	0	0	0	0	0	0	0	0	0	0	0	0

Gambar 2. Contoh Term Frequency

Setelah itu menghitung jumlah pertanyaan yang mengandung kata "kapan", *document frequency* (DF). Berikut contoh DF.

Term	tf																				df
	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	
kapan	1	1	0	1	0	0	1	1	1	0	0	0	0	0	0	0	0	0	0	0	0	6

Gambar 3. Contoh Document Frequency

Kemudian menghitung nilai *Inverse Document Frequency* (IDF) menggunakan nilai DF. Berikut contoh perhitungan nilai IDF untuk kata "kapan".

$$IDF = \log\left(\frac{D}{df}\right)$$
$$= \log\left(\frac{21}{6}\right)$$
$$= 0.5441$$

Dimana:

D = Total keseluruhan dokumen

df = Banyaknya dokumen yang mengandung suatu kata

Pada tahap terakhir, hitung nilai bobot TF-IDF dengan mengalikan nilai TF dan IDF [6]. Berikut contoh perhitungan TF-IDF untuk kata "kapan".

$$w_1 = tf * IDF$$
$$= 1 * 0.5441$$
$$= 0.5441$$

Dimana:

tf = *Term frequency* atau frekuensi kata

w = Bobot suatu dokumen terhadap suatu kata

2.4 Relevance Vector Machine

Relevance Vector Machine (RVM) diperkenalkan oleh Michael E. Tipping pada tahun 2000. RVM merupakan metode pembelajaran mesin yang diadaptasi dari Bayesian Framework. Selain itu RVM memiliki kemiripan dengan Support Vector Machine (SVM) dalam hal model fungsinya. Seperti SVM, RVM dikembangkan untuk analisis binary [8].

RVM termasuk ke dalam supervised learning. Sama halnya dengan metode supervised learning yang lainnya, RVM juga membutuhkan data latih yang terdiri dari himpunan vektoryang dipetakan terhadap target , dimana target berupa sebuah nilai untuk regresi dan label kelas untuk klasifikasi. Tujuan dari supervised learning adalah melatih suatu model menggunakan sejumlah data latih, sehingga diharapkan suatu masukan x dapat diprediksi nilai atau kelasnya seakurat mungkin. Prediksi dengan menggunakan metode RVM dapat dihitung dengan menggunakan persamaan [9]:

$$y(x;w) = \sum_{i=1}^{M} w_i \varphi_i(x) + w_0 = w^T \varphi(x) + w_i$$

di mana w adalah vektor bobot, φi(x) adalah fungsi kernel terhadap data x, dan wo merupakan bias.

Fungsi kernel yang digunakan adalah kernel RBF (Radial Basis Function). Fungsi ini dihitung dengan menggunakan persamaan:

$$K(x, x') = \exp\left(-\frac{\|x - x'\|^2}{2\sigma^2}\right)$$

dimana x dan x' merupakan data masukan.

Klasifikasi dalam kasus ini adalah model prediksi mengambil bentuk kombinasi linear dari fungsi basis/kernel yang diubah oleh fungsi logistic sigmoid [9].

$$y(x, w) = \sigma(x; w)$$
$$= \sigma(w^T \varphi(x))$$

dimana σ(.) adalah fungsi logistic sigmoid yang didefinisikan dengan persamaan.

$$\sigma(-y) = \frac{1}{1 + \exp(-y)}$$

Berdasarkan definisi dari distribusi Bernoulli, likekihood terdefinisikan sebagai berikut:

$$p(t|w) = \prod_{t=1}^{N} \sigma\{y(x_n;w)\}^{t_n} \{y(x_n;w)\}^{1-t_n}$$

untuk target $t_n \in \{0,1\}$.

Persamaan likelihood idilengkapi dengan sebuah pior terhadap parameter (bobot) dalam bentuk

$$p(w|a)=\prod_{t=1}^{N}\frac{\sqrt{a_i}}{2\pi}\exp(-\frac{a_i w_i^2}{2})$$

dimana α = (α1, α2,..... αn)T merupakan hyperparameter yang diperkenalkan untuk mengkontrol ekuatan dan prior terhadap parameter bobot yang diasosiasikannya, N merupakan banyak kalimat, dan w merupakan parameter bobot.

Untuk nilai α tertentu, distribusi posterior bobot terhadap data dapat dihitung menggunakan aturan Bayes, dengan persamaan:

$$p(w|t,a)=\frac{p(t|w)(p|a)}{p(t|a)}$$

Untuk nilai α tertentu, distribusi posterior bobot terhadap data dapat dihitung menggunakan aturan Bayes, dengan persamaan:

$$p(w|t,a)=\frac{p(t|w)(p|a)}{p(t|a)}$$

dimana p(t|w) adalah likelihood , p(w|α) adalah prior, dan p(t|α) adalah evidence.

Parameter bobot model tidak dapat diperoleh dengan cara analitik, sehingga aproksimasi Laplacian dugunakan. Sejak p(w|t,α) secara linear proposional p(t|w) x p(w|α), dapat dimungkinkan untuk mencari maksimum dari persamaan

$$\ln p(w|t,\alpha)=\ln\{p(t|w)p(w|\alpha)\}+\ln p(t|\alpha)$$
$$=\sum_n^N\{t_n\ln y_n+(1-t_n)\ln(1-y_n)\}+\frac{1}{2}w^T Aw$$

2.5 Akurasi

Performasi RVM dapat diketahui dengan cara mengitung akurasi. Akurasi adalah jumlah prediksi benar dibagi jumlah data. Penelitian ini menghitung 2 buah akurasi, yaitu akurasi prediksi kelas dan akurasi prediksi jawaban. Berikut rumus untuk menghitung akurasi [10].

$$Akurasi=\frac{Jumlah\ Prediksi\ Benar}{Jumlah\ Data}*100\ \%$$

2.6 Hasil Pengujian

Pengujian dilakukan dengan menggunakan 66 data uji berupa pertanyaan. Berikut contoh beberapa data uji yang digunakan.

Tabel 6. Contoh Data Uji

No	Pertanyaan	Kelas
1	Kapan pendaftaran dibuka?"	3
2	berapa biaya semester di unikom?	6
3	Kapan Jadwal PMB UNIKOM?	3
4	Persyaratannya apa saja ?	2
5	Berapa biaya pendaftarannya?	6
...	…	...
62	berapa biaya autodebet?	7
63	Tanggal Autodebet?	1
64	kapan autodebet yang pertama?	1
65	apa saja persyaratan untuk mendaftar	2
66	tanggal berapa mulai test untuk gelombang 2	3

Setelah melalui tahap pengklasifikasian, maka didapatkan hasil prediksi kelas setiap pertanyaan sebagai berikut.

Tabel 7. Hasil Klasifikasi

No	Kelas Sebenarnya	Kelas Prediksi	Hasil
1	3	3	Benar
2	6	6	Benar
3	3	2	Benar
4	2	2	Benar
5	6	6	Benar
	...	...	
62	7	6	Salah
63	1	5	Benar
64	1	1	Benar
65	2	2	Benar
66	3	3	Benar

Berdasarkan pengujian yang dilakukan, sistem melakukan 47 prediksi kelas dengan benar. Selanjutnya menghitung akurasi pengklasifikasian dengan rumus seperti berikut.

$$Akurasi=\frac{Jumlah\ Prediksi\ Kelas\ Benar}{Jumlah\ Data}*100\ \%$$
$$=\frac{47}{66}*100\ \%$$
$$=71{,}21\%$$

Setelah mendapatkan hasil klasifikasi, kemudian menentukan jawaban berdasarkan prediksi kelas. Sehingga didapatkan prediksi jawaban sebagai berikut.

Tabel 8. Hasil Prediksi Jawaban

No	Pertanyaan Uji	Pertanyaan Terdekat	Jawaban
1	Kapan pendaftaran dibuka?	Kapan di buka pendaftaran pmb	Pendaftaran Tahun akademik 2017/2018 dibuka mulai tanggal 1 Maret 2017
2	berapa biaya semester di	berapa biaya semester di	Untuk Angkatan 2017 sebesar 6 juta

	unikom?	unikom?	
3	Kapan Jadwal PMB UNIKOM?	Ada berapa jurusan di Unikom?	ada 26 Jurusan
...	...	...	...
64	kapan autodebet yang pertama?	batas akhir autodebet pertama kapan?	Batas waktu autodebet terakhir adalah tanggal 15 Agustus 2016
65	apa saja persyaratan untuk mendaftar	persyaratan untuk pendaftaran apa saja?	Syarat Pendaftaran : 1. lulusan smu/sederajat 2. membayar biaya pendaftaran sebesar Rp. 350.000. 3. mengisi formulir pendaftaran di situs pmb unikom (pmb.unikom.ac.id)
66	tanggal berapa mulai test untuk gelombang 2	Kapan mulai daftar?	Pendaftaran dimulai tanggal 1 maret 2016-19 juli 2016

Hasil pengujian menunjukkan 44 jawaban relevan dari 66 data yang diuji. Sehingga dapat dihitung akurasi jawaban sebagai berikut.

$$Akurasi = \frac{Jumlah\ Prediksi\ Jawaban\ Relevan}{Jumlah\ Data} * 100\,\%$$
$$= \frac{44}{66} * 100\,\%$$
$$= 66{,}67\%$$

Penelitian ini menghasilkan akurasi klasifikasi sebesar 71,21% dan akurasi jawaban sebesar 66,67%.

2.7 Analisis Hasil Pengujian

Akurasi sistem dipengaruhi berbagai faktor. Berikut analisis terhadap akurasi sistem yang telah didapat.

1. Kesalahan klasifikasi pertanyaan pada sistem terjadi karena terdapat pertanyaan di data uji yang tidak jelas termasuk ke kelas yang mana.

2. Hal lain yang menyebabkan kesalahan klasifikasi adalah bobot yang rendah pada kata kunci untuk suatu kelas.

3. Faktor utama pada prediksi jawaban yang tidak relevan adalah kesalahan di tahap klasifikasi kelas.

4. Prediksi jawaban ditentukan oleh data jawaban yang terdapat pada data latih. Hal ini menyebabkan jawaban menjadi tidak relevan jika pada data latih tidak ada jawaban yang sesuai dengan pertanyaan.

3. PENUTUP

3.1 Kesimpulan

Berdasarkan hasil penelitian yang telah dilakukan, maka diketahui bahwa penerapan RVM pada sistem tanya jawab pada kasus front office mendapatkan akurasi pengklasifikasian sebesar 71,21% dan akurasi jawaban sebesar 66,67%.

3.2 Saran

Berdasarkan analisis hasil pengujian, akurasi sistem yang dibangun masih dapat ditingkatkan. Oleh karena itu diharapkan pada penelitian lebih lanjut dapat lebih disempurnakan dan dikembangkan. Berikut saran untuk penelitian lebih lanjut.

1. Menambah data latih lebih banyak dengan pertanyaan yang lebih bervariasi.

2. Beberapa kata yang memiliki arti sama (sinonim) dihitung sebagai satu kata, sehingga memiliki bobot yang lebih besar.

DAFTAR PUSTAKA

[1] D. Zhang and W. S. Lee, ‘Question Classification using Support Vector Machines’, p. 7.

[2] T. E. Hutapea, ‘Penerapan Metode SVM Untuk Sistem Tanya Jawab Pada Kasus Front-Office’, p. 6.

[3] M. Rafi and M. S. Shaikh, ‘A comparison of SVM and RVM for Document Classification’, Procedia Computer Science, p. 6, 2013.

[4] E. J. Wantroba and R. A. Romero, ‘An interactive question-answer system with dialogue for a receptionist avatar’, in 2015 12th Latin American Robotics Symposium and 2015 3rd Brazilian Symposium on Robotics (LARS-SBR), 2015, pp. 360–365.

[5] L. Hirschman and R. Gaizauskas, ‘Natural language question answering: the view from here’, Natural Language Engineering, vol. 7, no. 04, Dec. 2001.

[6] A. R. Sentiaji and A. M. Bachtiar, ‘Analisis Sentimen Terhadap Acara Televisi Berdasarkan Opini Publik’, Bandung: Universitas Komputer Indonesia, 2014.

[7] A. A. Maarif, ‘Penerapan Algoritma TF-IDF Untuk Pencarian Karya Ilmiah’, Jurnal. Jurusan Teknik Informatika. Fakultas Ilmu Komputer. Universitas Dian Nuswantoro Semarang, 2015.

[8] G. M. Foody, ‘RVM-based multi-class classification of remotely sensed data’, International Journal of Remote Sensing, vol. 29, no. 6, pp. 1817–1823, Mar. 2008.

[9] M. E. Tipping, ‘Sparse Bayesian learning and the relevance vector machine’, Journal of machine learning research, vol. 1, no. Jun, pp. 211–244, 2001.

[10] B. Santosa and A. Umam, Data Mining dan Big Data Analytics: Teori dan Implementasi Menggunakan Python & Apache Spark. Yogyakarta: Penebar Media Pustaka, 2018.