Perancangan Sistem Pemeringkatan Jawaban Pada Forum Tanya Jawab Menggunakan *Textual Feature* dan *Semantic Similarity*

Answer Ranking System in Question-Answering Forum Using Textual Feature and Semantic Similarity

Lutfi Fitroh Hadi[1], Moch. Arif Bijaksana[2]

[12]Program Studi S1 Teknik Informatika, Fakultas Informatika, Universitas Telkom, Bandung

Jalan Telekomunikasi No. 1, Dayeuhkolot, Bandung 40257

[1]lutfi1304@gmail.com, [2]arifbijaksana@gmail.com

ABSTRAK

Kemajuan teknologi yang sangat pesat, tingginya jumlah pengguna internet di dunia, serta *basic* manusia sebagai makhluk sosial menjadi beberapa faktor munculnya situs-situs forum diskusi *online* atau forum tanya jawab di internet. Situs-situs tersebut memudahkan seseorang untuk mendapatkan jawaban atau solusi dari permasalahan yang dihadapi. Tetapi, seringkali ditemui jawaban yang tidak sesuai dengan pertanyaan. Penelitian ini akan mencoba membuat sebuah sistem pemeringkatan jawaban. *Dataset* yang digunakan berasal dari *SemEval2017 Task 3 Subtask A*. Pemberian peringkat jawaban dimulai dari *prepocessing data*, kemudian proses *feature extraction*. Lalu dilanjutkan dengan proses klasifikasi menggunakan *Support Vector Machine* (SVM). Nilai yang dihasilkan dari proses klasifikasi akan digunakan untuk memberikan peringkat jawaban. Pengukuran performansi pada aplikasi yang dibuat dalam tugas akhir ini menggunakan sistem penilaian yang sama seperti pada *SemEval2017* yaitu *Mean Average Precision* (MAP). Sistem yang dibangun pada penelitian ini memiliki performansi yang cukup baik, dengan nilai MAP terbesar yang didapat yaitu 72.3%. Jika dibandingkan dengan hasil dari SemEval 2016, sistem yang dibangun pada tugas akhir ini berada di peringkat ke-8 dari 13 peserta.

Kata kunci: *question answering system, semantic similarity, peringkat jawaban, support vector machine, mean average precision*

ABSTRACT

Technological advances, high number of internet users, and human nature as a social being become several factors in emersion of online forum discussion. Those kind of websites ease people in finding solution to their problem. But unfortunately, there are so many bad answers. Furthermore, the rating of an answers is given by human that makes it kind of subjective. This research will try to develop an answer ranking system. Dataset is obtained from SemEval 2017 Task 3 Subtask A. Ranking to an answers will be given through several processes, such as preprocessing data, and feature extraction. Then, preprocessed data will be classified using SVM. The score that is obtained from the classification process will be used as the ranking. Performance evaluation in this research will use the same as what is being used by SemEval 2017, that is Mean Average Precision (MAP). The system that is successfully developed here has a quite good performance, with an MAP score of 72.3%. Compared to the official SemEval 2016 result, this system is ranked 8th out of 13 contestants.

Keywords: *question answering system, semantic similarity, answer ranking, support vector machine, mean average precision*

1. Pendahuluan

Di era dengan kemajuan teknologi yang cepat seperti sekarang ini, seorang manusia dapat dengan mudah bersosialisasi atau berkomunikasi dengan orang lain. Di Indonesia sendiri, berdasarkan survey yang dilakukan oleh Asosiasi Penyelenggara Jasa Internet Indonesia (APJII) pada tahun 2016, pengguna internet di Indonesia mencapai angka 132.7 juta pengguna, atau setara dengan 51.7% dari total penduduk [1]. Ini menunjukkan ketergantungan manusia terhadap informasi yang tersedia di internet sangat besar.
Hal-hal di atas memicu munculnya beberapa forum diskusi *online* atau forum tanya-jawab di internet. Situs-situs seperti ini memberikan keuntungan dimana kita bisa mendapat banyak masukan terhadap suatu permasalahan sehingga tidak terpaku pada hanya satu solusi saja. Tetapi sayangnya situs-situs ini juga memiliki kelemahan dimana banyak dijumpai jawaban-jawaban yang buruk atau yang tidak relevan dengan pertanyaan yang diajukan. Selain itu, penilaian sebuah jawaban adalah baik atau buruk diberikan oleh pengguna lain sehingga jawaban yang tidak secara langsung berhubungan dengan pertanyaan bisa memiliki *rating* yang tinggi, atau jawaban bisa saja disusupi dengan tautan yang cenderung bersifat *spam*.

Pada tugas akhir ini, penulis mencoba membuat sebuah aplikasi yang dapat memberikan peringkat terhadap jawaban-jawaban dari sebuah pertanyaan. Proses yang dilakukan untuk mendapatkan ranking jawaban berupa *prepocessing data*, *feature extraction*, klasifikasi, dan pemberian peringkat.
Tahapan *preprocessing data* yang dilakukan adalah *Tag Removal, Case Folding, Tokenizing, Stopword Removal, Part of Speech (POS) Tagging, dan Lemmatization*. *Feature extraction* yang digunakan adalah *textual feature* dan *semantic similarity feature*. Sedangkan pada proses klasifikasi akan menggunakan algoritma *Support Vector Machine* (SVM) sebagai *classifier*. Untuk evaluasi performansi sistem, parameter yang digunakan sama seperti pada *SemEval2017* yaitu *Mean Average Precision* (MAP).

Penelitian ini akan menggunakan *Semantic Similarity* untuk mempertahankan kesamaan makna kalimat, dan juga *Textual Feature* untuk mendapatkan ciri khas dari sebuah jawaban yang baik atau buruk. Sementara pada proses klasifikasi, akan menggunakan SVM.

2. Dasar Teori dan Perancangan Sistem

2.1 Dataset

Dataset yang digunakan didapat dari *SemEval2017 Task 3*. Contoh data dapat dilihat pada Gambar 1 dan statistik persebaran kelas pada dataset dapat dilihat pada Tabel 1. *Dataset* ini terbagi menjadi 2 bagian yaitu data latih untuk proses *learning* pada *classifier*, dan data uji untuk menguji hasil klasifikasi. Data uji inilah yang kemudian akan dilakukan perangkingan.

```
<RelQuestion RELQ_ID="Q388_R14" RELQ_CATEGORY="Advice and Help" RELQ_DATE="2009-05-17
22:23:52" RELQ_USERID="U2935" RELQ_USERNAME="Yui">
    <RelQSubject>Schengen Visa @ Greece Embassy</RelQSubject>
    <RelQBody>Do you know how long it will take to get Schengen Visa from Embassy of
    Greece? 3 days? 1 week? any idea? Thanks.</RelQBody>
</RelQuestion>

<RelComment RELC_ID="Q388_R14_C1" RELC_DATE="2009-05-17 22:33:01" RELC_USERID="U188"
RELC_USERNAME="novita77" RELC_RELEVANCE2RELQ="Good">
    <RelCText>standard is 2 weeks maximum, but when I apply my schengen visa from Hungarian
    Embassy took 5 weeks.</RelCText>
</RelComment>

<RelComment RELC_ID="Q388_R14_C2" RELC_DATE="2009-05-17 22:36:20" RELC_USERID="U2935"
RELC_USERNAME="Yui" RELC_RELEVANCE2RELQ="Bad">
    <RelCText>Thanks Novita for your quick reply.</RelCText>
</RelComment>
```

Gambar 1 Contoh Data

Pada *dataset*, terdapat beberapa atribut penting yaitu:

- *RelQSubject* : Subjek dari pertanyaan
- *RelQBody* : isi dari pertanyaan, biasanya lebih spesifik daripada atribut *RelQSubject*
- *RELC_RELEVANCE2RELQ* : Kelas dari jawaban
- *RelCText* : isi dari jawaban

Tabel 1 Statistik Sebaran Kelas Data Latih dan Data Uji

Item	Data Latih	Data Uji
Pertanyaan	**1000**	**293**
Good	3804	1523
PotentiallyUseful	1642	0
Bad	4554	1407
Total Jawaban	**10000**	**2930**

Berdasarkan analisis pada persebaran data, maka proses klasifikasi dilakukan terhadap 2 kelas, dimana data dengan kelas *Potentially Useful* dianggap sebagai data *Bad* [2]. Hal ini dilakukan karena pada sistem yang akan dibangun, hanya akan mencari mana jawaban yang baik (*good*) dan mana jawaban yang buruk (*bad*).

2.2 Textual Feature

1. *Cosine Similarity*

Tingkat kemiripan teks didapatkan dengan cara menghitung nilai kosinus antara 2 vektor menggunakan Persamaan 1 berikut:

$$CosineSimilarity = \frac{\sum_i^n u_i \cdot v_i}{\sqrt{\sum_i^n (u_i)^2} \cdot \sqrt{\sum_i^n (v_i)^2}} \quad (1)$$

Dengan:
n = jumlah kata pada kalimat
u = kalimat pertama
v = kalimat kedua

Jika kedua teks yang dibandingkan tidak memiliki kata yang sama, maka nilai *Cosine Similarity* akan menjadi 0.

2. *Question Mark*

Fitur ini akan memeriksa apakah jawaban yang diberikan mengandung tanda tanya (?) atau tidak.

3. *Booster Words*

Booster words biasanya digunakan untuk menekankan suatu bagian dari teks. Jawaban dengan kategori baik cenderung memiliki kata-kata yang bersifat ajakan atau anjuran, seperti "*try*", "*recommend*", "*advise*" [3].

4. *URL*

Fitur ini akan memeriksa apakah jawaban yang diberikan memiliki tautan menuju situs lain.

2.3 Semantic Similarity

Algoritma yang digunakan untuk menghitung Semantic Similarity adalah Resnik Algorithm. Algoritma ini menghitung kesamaan antara 2 string berdasarkan *information content* yang terdapat di dalamnya. Perhitungan *similarity* dilakukan terhadap 2 buah kata yang memiliki POS yang sama. Formula yang digunakan pada algoritma Resnik dapat dilihat pada Persamaan 2.

$$Resnik\ (c_1, c_2) = \mathrm{IC}\big(LCS(c_1, c_2)\big) \quad (2)$$

Dengan:
c_1,c_2 = kata yang dibandingkan
LCS = *Lowest Common Subsumer*
IC = *Information Content*.

2.4 WordNet

WordNet merupakan kamus yang berisi setiap kata di dalam Bahasa Inggris. *WordNet* mengelompokkan kata ke dalam empat kategori yaitu *noun*, *verb*, *adjective*, dan *adverb*. Kata-kata yang terdapat di dalam database *WordNet* memiliki keterhubungan satu sama lain, dengan unit terkecil di dalam *WordNet* adalah *synsets*. *Synsets* merupakan struktur kata yang memiliki makna sama (*synonymous meaning*) [4].

2.5 Evaluasi Performansi Sistem

Evaluasi performansi sistem akan menggunakan *Mean Average Precision* (MAP). MAP memberikan sebuah nilai tunggal terhadap titik *Recall*. MAP hanya akan menghitung dokumen yang relevan, dan level *recall* tidak ditetapkan secara baku sehingga tidak ada interpolasi [5]. Formula yang digunakan pada MAP dapat dilihat pada Persamaan 3.

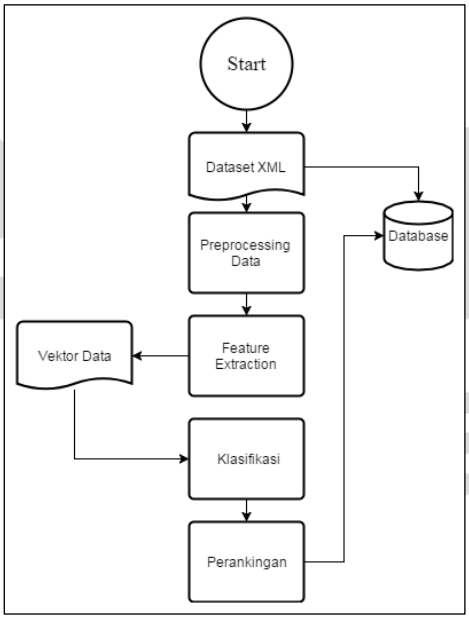
$$MAP = \frac{\sum_1^Q AvgPrecision(Q)}{Q} \quad (3)$$

Dimana Q adalah jumlah query.

2.6 Perancangan Sistem

Flowchart sistem dapat dilihat pada Gambar 2. Secara umum, gambaran kerja sistem adalah sebagai berikut:

1. Sistem menerima *input* (*dataset*) berupa teks dengan format XML.
2. *Preprocessing data* (*tag removal*, *case folding*, *tokenizing*, *stopword removal*, *part of speech tagging*, *lemmatization*)
3. Data hasil *preprocessing* akan dilakukan ekstraksi fitur
4. Hasil *preprocessing* dan ekstraksi fitur akan disimpan ke dalam file untuk mempersingkat waktu pengujian.
5. Data pada proses no. 4 akan digunakan untuk proses klasifikasi dan pemberian peringkat jawaban
6. Setelah peringkat didapatkan, sistem akan menampilkan kembali pertanyaan beserta jawabannya berurut mulai dari yang memiliki ranking paling besar ke paling kecil



Gambar 2 Alur Kerja Sistem

3. Pengujian dan Hasil

Pengujian sistem dilakukan beberapa kali dengan tujuan untuk:

1. Mengetahui kombinasi parameter dan kernel SVM yang dapat memberikan nilai MAP terbaik, skenario dapat dilihat pada Tabel 2
2. Mengetahui fitur mana yang memiliki pengaruh paling besar terhadap sistem, skenario dapat dilihat pada Tabel 3

3.1 Skenario Pengujian

Pada skenario pertama, dilakukan pengujian terhadap SVM dengan mengubah tipe kernel dan beberapa parameter seperti parameter *C*, *gamma*, dan *degree*.

Tabel 2 Skenario Pengujian Parameter SVM

No	Kernel	C	Gamma	Degree
1	Linear	1.0	0.2	3
2	Linear	0.5	0.2	3
3	Linear	0.1	0.2	3
4	RBF	1.0	0.2	3
5	RBF	0.5	0.2	3
6	RBF	0.1	0.2	3
7	RBF	1.0	0.3	3
8	Polynomial	1.0	0.2	3
9	Polynomial	0.5	0.2	4
10	Polynomial	0.1	0.2	2
11	Sigmoid	1	0.3	3
12	Sigmoid	0.5	0.5	4
13	Sigmoid	0.1	1.0	5
14	Sigmoid	0.01	0.3	2

Setelah mendapatkan kombinasi parameter yang tepat untuk SVM, selanjutnya parameter tersebut digunakan untuk melakukan klasifikasi terhadap kombinasi fitur.

Tabel 3 Skenario Pengujian Penggunaan Fitur

No	Skenario
1	Tanpa Cosine Similarity
2	Tanpa Ekstraksi Tanda Tanya
3	Tanpa Ekstraksi Booster Words
4	Tanpa Ekstraksi URL
5	Tanpa Semantic Similarity

3.2 Hasil Klasifikasi

Hasil pengujian skenario untuk mencari parameter yang tepat dapat dilihat pada Tabel 4

Tabel 4 Hasil Pengujian Parameter SVM

Skenario	MAP(%)	Precision(%)	Recall(%)	F-Measure(%)	Akurasi (%)
1	72.1	64	59	55	58.6
2	72.3	65	59	56	59.1

3	**72.3**	**65**	**59**	**56**	**59.2**
4	71.7	66	62	60	61.6
5	71.5	66	62	60	61.5
6	71.3	66	61	59	61.2
7	70.8	66	62	60	61.9
8	71.8	65	58	54	58.3
9	71.8	65	57	52	56.8
10	72.2	65	60	57	59.5
11	52.2	41	41	40	40.6
12	52.2	41	41	40	41.1
13	50.8	41	42	40	41.6
14	52.2	41	42	40	41.9

Berdasarkan 14 kali pengujian, nilai MAP terbesar diperoleh dengan *kernel Linear* dengan parameter *C*=0.1, *Gamma*=0.2, dan *Degree*=3.

3.3 Pengaruh Fitur

Selanjutnya, dilakukan pengujian terhadap skenario kedua menggunakan pengaturan SVM yang telah didapatkan sebelumnya. Hasil dari pengujian skenario kedua dapat dilihat pada Tabel 5

Tabel 5 Hasil Pengujian Penggunaan Fitur

Skenario	MAP(%)	Precision(%)	Recall(%)	F-Measure(%)	Akurasi(%)
1	72.2	65	59	55	58.6
2	**66.3**	**58**	**50**	**39**	**50.1**
3	72.2	66	61	59	61.3
4	72.3	65	59	56	58.9
5	**69.8**	**58**	**50**	**39**	**50.1**

Dari hasil pengujian terhadap *textual feature* dan *semantic similarity* terjadi perubahan nilai MAP yang signifikan pada skenario ke-dua yaitu tanpa ekstraksi tanda tanya dan pada skenario ke-lima yaitu tanpa *semantic similarity*. Sementara itu pada skenario ke-empat yaitu tanpa ekstraksi URL, tidak terjadi perubahan nilai MAP.

4. Kesimpulan

Berdasarkan pengujian yang telah dilakukan, dapat diambil kesimpulan sebagai berikut:

1. Sistem yang dapat membantu mencari jawaban yang baik terhadap pertanyaan dan melakukan perangkingan terhadap jawaban tersebut sudah berhasil dibangun dengan nilai MAP tertinggi dicapai pada angka 72.3%
2. Fitur yang memiliki pengaruh paling besar adalah ekstraksi tanda tanya dan *semantic similarity*. Adanya tanda tanya pada jawaban dapat membedakan sebuah jawaban adalah baik atau buruk. Sementara pada semantic *similarity*, semakin besar nilai *similarity* maka jawaban tersebut cenderung termasuk ke dalam kelas "*Good*"

Daftar Pustaka

[1] Asosiasi Penyelenggara Jasa Internet Indonesia, "Infografis Penetrasi dan Perilaku Pengguna Internet Indonesia," 2006.

[2] Preslav Nakov et al., "SemEval-2016 Task 3: Community Question Answering," in *SemEval 2016*, 2016.

[3] Luh Putri Ayu Ningsih, Ade Romadhony, and Mochammad Arif Bijaksana, "Pemeringkatan Jawaban pada Community Question Answering dengan Tekstual Fitur dan PemodelanTopik," in *Telkom University*, Bandung, 2016.

[4] George Miller, "WordNet: A Lexical Database for English," *Communications of the ACM*, vol. 38, no. 11, pp. 39-41, 1995.

[5] Keneilwe Zuva, "Evaluation of information retrieval systems.," 2012.