

Penerapan String Matching Pada Pemilihan Jawaban untuk Autoreply Frequently Asked Question pada SMS

Riandy Rahman Nugraha (13511014)¹

Program Studi Teknik Informatika

Sekolah Teknik Elektro dan Informatika

Institut Teknologi Bandung, Jl. Ganessa 10 Bandung 40132, Indonesia

¹riandyrn@students.itb.ac.id

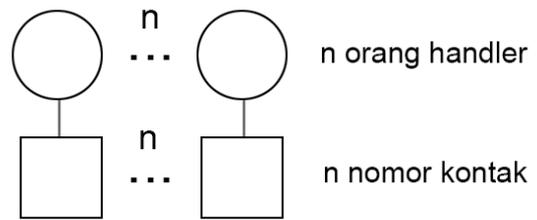
Abstrak—Sebagai contact person dari sebuah event yang baik, setiap pertanyaan yang diajukan oleh partisipan/non-partisipan haruslah dijawab dengan baik. Namun, hal ini akan menjadi masalah besar jika banyak sekali orang yang bertanya (implikasinya contact person tersebut harus menjawab pertanyaan-pertanyaan yang diajukan satu persatu sebanyak jumlah dari orang yang bertanya tersebut). Namun, biasanya pada pertanyaan-pertanyaan yang diajukan banyak pertanyaan-pertanyaan serupa yang muncul, sehingga sebenarnya jawaban dari pertanyaan-pertanyaan tersebut juga sama. Pada makalah ini akan dibahas mengenai penerapan string matching dalam menjawab pertanyaan-pertanyaan yang sering diajukan tersebut (sering disebut juga sebagai F.A.Q - Frequently Asked Question) melalui media SMS.

Index Terms—Contact Person, String Matching, F.A.Q, SMS.

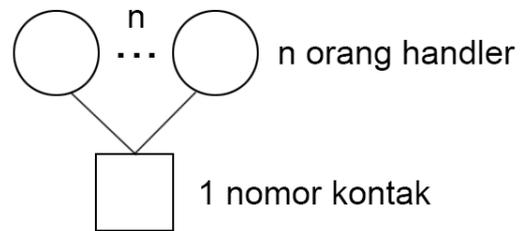
I. PENDAHULUAN

Sebagai contact person dari suatu event, terutama suatu event yang besar, sudah menjadi hal yang lumrah jika banyak orang yang bertanya mengenai informasi terkait dengan event kepada contact person tersebut. Tentu sudah menjadi sebuah keharusan untuk melayani setiap pertanyaan-pertanyaan tersebut dengan baik. Namun, hal ini akan menjadi persoalan yang cukup besar jika terdapat puluhan, ratusan, atau bahkan ribuan pertanyaan-pertanyaan yang harus dijawab dalam satu waktu. Tentu membutuhkan cost yang tidak sedikit (terutama cost tenaga dan waktu) untuk melayani setiap pertanyaan-pertanyaan tersebut.

Salah satu solusi untuk menangani persoalan menjawab pertanyaan dalam jumlah yang besar adalah dengan cara menambah personil contact person. Bisa dengan secara eksplisit mencantumkan lebih dari satu contact person atau juga bisa hanya dengan menggunakan satu contact saja namun sebenarnya yang menangani pertanyaan-pertanyaan tersebut ada beberapa orang. Berikut ini adalah ilustrasi mengenai kedua solusi tersebut (persegi merepresentasikan nomor kontak dan lingkaran merepresentasikan orang yang akan menangani pertanyaan-pertanyaan yang akan diterima oleh nomor kontak tersebut).



Gambar 1.1 Solusi n nomor kontak, n orang handler



Gambar 1.2 Solusi 1 nomor kontak, n orang handler

Meskipun solusi-solusi diatas sudah cukup baik untuk menyelesaikan persoalan mengenai menjawab pertanyaan dalam jumlah besar ini, namun menurut hemat penulis masih ada solusi yang lebih efektif lagi, yakni dengan cara memanfaatkan mesin untuk menjawab pertanyaan-pertanyaan tersebut.

Tentu tidak semua pertanyaan dapat dijawab oleh mesin, pada makalah ini pertanyaan yang akan dijawab oleh mesin dibatasi hanya pada pertanyaan-pertanyaan yang sering muncul saja atau sering juga disebut sebagai Frequently Asked Question (F.A.Q). Mesin dapat menjawab Frequently Asked Question ini dengan memanfaatkan strategi algoritma string matching untuk mencari kata kunci dari setiap pertanyaan kemudian dilakukan pencarian jawaban yang relevan berdasarkan kata kunci yang ditemukan di pertanyaan tersebut. Secara prinsip mekanisme ini hampir mirip dengan mekanisme chatbot sederhana, namun lebih praktis karena digunakan untuk menyelesaikan persoalan di domain yang lebih nyata. Berikut ini adalah ilustrasi solusi menjawab pertanyaan dengan menggunakan mesin penjawab.

string matching.

V. PEMILIHAN JAWABAN F.A.Q SEBAGAI PERSOALAN STRING MATCHING

Pada dasarnya kita dapat memandang persoalan pemilihan jawaban F.A.Q ini sebagai persoalan string matching. Yakni mencari string yang paling cocok dengan F.A.Q yang terdaftar kemudian mengirimkan jawabannya berdasarkan jawaban dari F.A.Q yang dianggap paling cocok. Namun demikian setiap pertanyaan yang diajukan dapat berbeda-beda bentuknya, misalnya saja untuk menanyakan tentang perlengkapan apa saja yang harus dibawa saat acara si penanya dapat bertanya dengan berbagai macam bentuk kalimat misalkan saja sebagai berikut: "Apa saja spek acaranya?", "Apa yang harus dibawa saat acara?", dan lain sebagainya, namun pada intinya berbagai macam bentuk kalimat tersebut menanyakan satu hal yang sama: barang-barang apa saja yang harus dibawa oleh peserta pada saat acara.

Hampir tidak mungkin untuk melakukan enumerasi seluruh bentuk kalimat untuk suatu pertanyaan tertentu, selain itu jika dilakukan pun tentu akan memerlukan banyak memori untuk penyimpanannya. Namun demikian, di setiap pertanyaan terdapat kata kunci yang dapat mengindikasikan apa sebenarnya maksud dari pertanyaan tersebut. Dengan memanfaatkan sifat ini sebenarnya kita tidak perlu menyimpan seluruh bentuk kalimat pertanyaan untuk satu maksud pertanyaan tertentu, cukup yang disimpan adalah kata kuncinya saja. Jadi hanya dengan melihat kata kuncinya saja kita sudah tahu apa maksud dari pertanyaan tersebut.

Jadi gagasan dasarnya adalah yang akan dicocokkan oleh program adalah kata kunci dari pertanyaan yang diajukan kemudian program akan mencocokkannya dengan kata kunci yang ada di basis data lalu menggenerate jawaban dari pertanyaan tersebut berdasarkan atas data jawaban yang terdapat di dalam database. Namun demikian tentu masih terdapat berbagai kendala dalam pencocokan string ini, misalnya saja bagaimana agar jawaban yang dihasilkan oleh program memang sesuai dengan konteks yang ditanyakan oleh penanya. Pada bagian selanjutnya akan dijelaskan gagasan mengenai mekanisme program secara lebih mendetail.

VI. PENERAPAN STRING MATCHING DALAM MENENTUKAN JAWABAN DARI F.A.Q

Berikut ini adalah algoritma yang penulis usulkan sebagai solusi penentuan jawaban dari F.A.Q (penulisan notasi algoritmik secara lengkap dapat dilihat pada lampiran A.2).

Misalkan untuk merepresentasikan baris tabel pertanyaan dan jawaban pada lampiran A.1 kita menggunakan struct sebagai berikut.

```
PertanyaanStruct {  
    Keywords: array of string;
```

```
Jawaban: string;  
}
```

Dengan memanfaatkan `PertanyaanStruct` ini kita dapat membuat algoritma mesin penjawab F.A.Q sebagai berikut (notasi algoritmik selengkapnya dapat dilihat pada lampiran A.2).

```
/** Notasi Algoritmik Mesin Penjawab */  
loadAllPertanyaanInDB (AllPertanyaanDBStruct)  
input (PertanyaanPartisipan)  
foreach (PertanyaanStruct Row:  
AllPertanyaanDBStruct)  
    foreach (string Keyword: Row.Keywords)  
        if (KMP (Keyword, PertanyaanPartisipan))  
            output (Row.Jawaban);
```

Notasi Algoritmik Mesin Penjawab

Pada awalnya isi dari database pertanyaan disalin terlebih dahulu ke array `AllPertanyaanDBStruct`. Dengan cara ini tidak perlu dilakukan pembacaan berulang kali ke file karena seluruh data pertanyaan sudah terdapat di memori. Setelah berhasil melakukan penyalinan isi database, program akan menunggu masukan dari user yang dalam hal ini adalah SMS dari penanya. Masukan dari user akan dimasukkan ke variable `PertanyaanPartisipan`.

Jika terdapat masukan, maka dimulailah proses pencarian keyword yang ada dalam database di string `PertanyaanPartisipan`, jika ditemukan kata kunci yang cocok maka jawaban akan ditampilkan (dalam hal ini dikirim kembali ke penanya).

Namun ada hal yang perlu diperhatikan, yakni proses pencocokan keyword itu sendiri. Proses pencocokan kata kunci yang dimaksud oleh penulis tidak berarti pada satu pertanyaan yang diperiksa hanya satu kata kunci saja yang dicocokkan, namun bisa saja lebih dari satu kata kunci. Misalkan begini, mengacu pada pertanyaan no 1 pada Lampiran A.1, yakni: "Bagaimana cara mengikuti/mendaftar kegiatan ini?" di kolom Kata/Pattern Kunci penulis menuliskan "cara & daftar" artinya adalah untuk mencapai pertanyaan "Bagaimana cara mengikuti/mendaftar kegiatan ini?" maka perlu dilakukan pengecekan apakah di string pertanyaan ditemukan kata kunci "cara" dan kata kunci "daftar" tidak? Jika iya, maka berdasarkan data pada table dapat disimpulkan bahwa yang ditanyakan oleh penanya adalah pertanyaan "Bagaimana cara mengikuti/mendaftar kegiatan ini?" dan program dapat mengeluarkan jawaban yang tepat, yakni "Silakan kirim pesan ke 085793174788 dengan format Nama_Fakultas_Angkatan".

Meskipun algoritma ini sudah cukup baik dalam memberikan solusi, namun masih memiliki kelemahan, yakni algoritma ini hanya dapat memproses satu pertanyaan saja. Padahal seringkali orang-orang bertanya lebih dari satu pertanyaan dalam satu kali SMS. Untuk memenuhi kebutuhan ini maka algoritma mesin penjawab dapat dimodifikasi sebagai berikut (notasi algoritmik secara lengkap dapat dilihat pada lampiran A.3).

```

/** Notasi Algoritmik Mesin Penjawab */
loadAllPertanyaanInDB (AllPertanyaanDBStruct)
input (PertanyaanPartisipan)
pecahPertanyaanJadiArrayPertanyaan (PertanyaanPartisipan, Questions)
foreach (string Question: Questions) {
  foreach (PertanyaanStruct Row:
    AllPertanyaanDBStruct)
    foreach (string Keyword: Row.Keywords)
      if (KMP (Keyword, PertanyaanPartisipan))
        insertToArray (Answers, Row.Jawaban);
}
output (Answers)

```

Notasi Algoritmik Mesin Penjawab Versi 2

Algoritma hasil modifikasi ini sebenarnya tidak jauh berbeda dengan algoritma sebelumnya. Hanya saja pada algoritma ini ditambahkan tokenizer untuk memecah string pertanyaan menjadi beberapa pertanyaan. Selain itu ditambahkan pula iterasi untuk setiap pertanyaan agar dapat diproses. Setiap kali akhir pemrosesan pertanyaan, jika jawaban ditemukan maka jawaban tersebut akan dimasukkan ke dalam sebuah array jawaban. Pada saat akhir dari algoritma, semua jawaban akan diberikan kepada pengguna (dalam hal ini adalah penanya).

Pada bagian selanjutnya akan dijelaskan mengenai proses perhitungan/penentuan jawaban dari pertanyaan dari penanya dengan menggunakan contoh kasus.

VII. PERHITUNGAN STRING MATCHING UNTUK MENENTUKAN JAWABAN F.A.Q

Misalkan kita ingin mencari jawaban dari string pertanyaan berikut menggunakan algoritma mesin penjawab F.A.Q yang penulis usulkan (versi 2): "Assalamu'alaykum, kak, mau nanya kalau acara ini wajib? Soalnya kemungkinan sy g bisa dateng acara ini".

Pertama string ini akan dipecah dulu menjadi 2 kalimat, yakni "Assalamu'alaykum, kak, mau nanya kalau acara ini wajib?" dan "Soalnya kemungkinan sy g bisa dateng acara ini". Setelah itu akan dilakukan pencocokan kata kunci yang ada di database apakah kata kunci tersebut ditemukan pada kalimat atau tidak.

Pada kalimat "Assalamu'alaykum, kak, mau nanya kalau acara ini wajib?" ditemukan kata kunci "acara" dan "wajib" yang sesuai dengan baris ke 3 pada database, maka jawaban untuk baris ke 3 ini dimasukkan ke dalam array Answers.

Assalamu'alaykum, kak, mau nanya kalau **acara** ini **wajib**?

kata kunci acara dan wajib ditemukan pada string

Gambar 7.1 Pattern Ditemukan

Pada kalimat "Soalnya kemungkinan sy g bisa dateng acara ini" tidak ditemukan kata kunci yang sesuai (perhatikan bahwa meskipun ada kata kunci "acara" dalam kalimat ini, namun di database tidak pertanyaan yang dimaksudkan hanya dengan kata kunci "acara" saja). Berarti tidak ada solusi yang dimasukkan ke dalam array

Answers.

Karena seluruh kalimat sudah diperiksa, maka seluruh jawaban yang ada di dalam array Answers dikirimkan kepada penanya. Jawaban yang dikirimkan kepada penanya adalah: "Ya, acara ini wajib bagi anggota unit/aktivis salman". Jawaban yang diberikan cukup relevan dengan pertanyaan yang diberikan, bukan?

Mari kita tinjau persoalan lain, misalkan kita ingin mencari jawaban dari string berikut: "Kak, apa saja peralatan yang harus dibawa saat acara? Apa saya boleh menyusul datangnya? Soalnya saya ada ujian". String ini akan dipecah menjadi 3 kalimat: "Kak, apa saja peralatan yang harus dibawa saat acara?", "Apa saya boleh menyusul datangnya?", dan "Soalnya saya ada ujian".

Pada kalimat pertama ditemukan kata kunci "yang" dan "dibawa" maka jawabannya adalah "Alat tulis, pakaian yang menutupi aurat, air minum, sepatu". Pada kalimat kedua ditemukan kata kunci "boleh" dan "menyusul" maka jawabannya adalah "Ya, silakan hadir di tengah-tengah acara". Pada kalimat ketiga tidak ditemukan kata kunci yang sesuai. Jawaban akhir dari pertanyaan ini adalah "Alat tulis, pakaian yang menutupi aurat, air minum, sepatu. Ya, silakan hadir di tengah-tengah acara".

VIII. KESIMPULAN

Algoritma yang penulis usulkan adalah algoritma yang sangat bergantung pada kata kunci yang tersedia di database. Jika kata kuncinya tidak tepat maka sudah tentu jawaban dari algoritma ini juga kurang tepat, sehingga memang diperlukan analisis yang cukup mendalam saat melakukan pengisian tabel kata kunci beserta jawabannya.

Selain itu untuk menyelesaikan persoalan menjawab F.A.Q ini sebaiknya tidak sepenuhnya *reply* jawaban diserahkan kepada mesin, sebaiknya dibarengi juga dengan operator manusia. Apabila mesin tidak dapat menjawab pertanyaan atau terdapat kesalahan pada jawaban yang dihasilkan oleh mesin, operator manusia dapat menanggulangnya.

Namun bagaimanapun ini adalah gagasan solusi untuk membantu para contact person untuk melaksanakan tugasnya melayani pertanyaan demi pertanyaan dengan baik. Selain itu memang gagasan ini dapat diaplikasikan. Hanya saja memang diperlukan analisis yang mendalam pada saat pembuatan database kata kunci dan jawabannya.

IX. UCAPAN TERIMA KASIH

Penulis mengucapkan terima kasih kepada Bapak Rinaldi Munir dan Bu Masayu Leilia Khodra sebagai dosen pengajar mata kuliah Strategi Algoritma.

Penulis juga mengucapkan terima kasih kepada kawan-kawan di Masjid Salman ITB yang memberikan kesempatan kepada penulis untuk berada di divisi MSDM yang banyak sekali menangani partisipan, terutama dalam hal distribusi informasi kepada partisipan.

Tanpa adanya kesempatan penulis menjadi MSDM di kegiatan Menara Salman ITB 2013, inspirasi untuk

menulis makalah ini mungkin tidak akan pernah ada.

Selain itu penulis juga mengucapkan terimakasih sebanyak-banyaknya kepada teman-teman di Asrama Salman ITB yang terus menyemangati penulis untuk bersabar menyelesaikan makalah ini ditengah-tengah persiapan kegiatan kaderisasi Salman yang selanjutnya: LMD (Latihan Mujtahid Dakwah) 171 yang dimana penulis juga memegang peranan yang cukup penting disana.

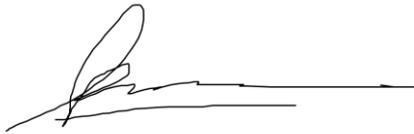
REFERENCES

- [1] Rinaldi Munir, "Diktat Kuliah Strategi Algoritma", 2011
- [2] <http://www-igm.univ-mlv.fr/~lecroq/string/index.html>, diakses pada 18 Desember 2013

PERNYATAAN

Dengan ini saya menyatakan bahwa makalah yang saya tulis ini adalah tulisan saya sendiri, bukan saduran, atau terjemahan dari makalah orang lain, dan bukan plagiasi.

Bandung, 16 Desember 2013



Riandy Rahman Nugraha (13511014)

Lembar Lampiran

Lampiran A.1 – Tabel F.A.Q Menara Salman ITB 2013

Tabel F.A.Q Menara Salman 2013			
No	Pertanyaan	Kata/Pattern Kunci	Jawaban
1	Bagaimana cara mengikuti/mendaftar kegiatan ini?	cara & daftar; bagaimana & daftar;	Silakan kirim pesan ke 085793174788 dengan format Nama_Fakultas_Angkatan
2	Apa saja perlengkapan yang harus dibawa?	perlengkapan; spek; yang & dibawa;	Alat tulis, pakaian yang menutupi aurat, air minum, sepatu
3	Apakah acara ini wajib?	acara & wajib	Ya, acara ini wajib bagi anggota unit/aktivis salman
4	Apakah acara ini terbuka untuk umum?	terbuka untuk umum	Ya, acara ini terbuka juga untuk umum
5	Apakah acara ini gratis?	acara & gratis	Ya, acara ini gratis
6	Apakah boleh menyusul?	boleh & menyusul	Ya, silakan hadir di tengah-tengah acara
7	Apa saja kegiatan acara ini?	kegiatan & acara	Pengenalan sejarah masjid salman ITB, materi, dan outbound
8	Apakah boleh izin di tengah acara?	izin & tengah & acara;	Silakan konsultasikan dengan panitia saat acara berlangsung

Lampiran A.2 – Notasi Algoritmik Mesin Penjawab versi 1

```
procedure loadAllPertanyaanInDB (AllPertanyaanStruct: array of PertanyaanStruct)  
//load semua pertanyaan dari database dan mengisikannya ke AllPertanyaanStruct
```

```
function KMP(Keyword: string, Pertanyaan: string): boolean  
//akan mengembalikan nilai true jika ditemukan Keyword pada Pertanyaan
```

```
PertanyaanStruct {  
    Keywords: array of string;  
    Jawaban: string;  
}
```

Algoritma Mesin Penjawab:

Deklarasi:

```
AllPertanyaanDBStruct: array(1..n) of PertanyaanStruct  
PertanyaanPartisipan: string
```

Program:

```
loadAllPertanyaanInDB (AllPertanyaanDBStruct)  
input (PertanyaanPartisipan)  
foreach (PertanyaanStruct Row: AllPertanyaanDBStruct)  
    foreach (string Keyword: Row.Keywords)  
        if (KMP (Keyword, PertanyaanPartisipan))  
            output (Row.Jawaban);
```

Lampiran A.3 – Notasi Algoritmik Mesin Penjawab versi 2

```
procedure loadAllPertanyaanInDB (AllPertanyaanStruct: array of PertanyaanStruct)  
//load semua pertanyaan dari database dan mengisikannya ke AllPertanyaanStruct
```

```

function KMP(Keyword: string, Pertanyaan: string): boolean
//akan mengembalikan nilai true jika ditemukan Keyword pada Pertanyaan

procedure pecahPertanyaanJadiArrayPertanyaan(PertanyaanPartisipan: string,
Questions: array of string)
//memecah PertanyaanPartisipan menjadi beberapa pertanyaan, pertanyaan diisikan
ke Questions

procedure insertToArray(Answers: array of string, Jawaban: string)
//memasukkan Jawaban ke array Answers

Algoritma Mesin Penjawab (Modified):
Deklarasi:
AllPertanyaanDBStruct: array(1..n) of PertanyaanStruct
PertanyaanPartisipan: string
Questions: array(1..n) of string
Answers: array(1..n) of string

Program:
loadAllPertanyaanInDB(AllPertanyaanDBStruct)
input(PertanyaanPartisipan)
pecahPertanyaanJadiArrayPertanyaan(PertanyaanPartisipan, Questions)
foreach(string Question: Questions) {
  foreach(PertanyaanStruct Row: AllPertanyaanDBStruct)
    foreach(string Keyword: Row.Keywords)
      if(KMP(Keyword, PertanyaanPartisipan))
        insertToArray(Answers, Row.Jawaban);
}
output(Answers)

```