

APLIKASI BOTQA UNTUK MENINGKATKAN CARA INTERAKSI MANUSIA DAN MESIN

Emma Utami¹, Sri Hartati²

¹STMIK AMIKOM Yogyakarta

²Program Studi Magister Ilmu Komputer, Sekolah Pascasarjana, Universitas Gadjah Mada

e-mail: ¹emma@nrar.net, ²shartati@ugm.ac.id

ABSTRAKSI

Artificial Intelligence atau kecerdasan buatan merupakan bagian dari ilmu pengetahuan komputer yang ditujukan untuk perancangan otomatisasi tingkah laku cerdas dalam sistem kecerdasan komputer. Dengan demikian diharapkan komputer bisa menirukan beberapa fungsi otak manusia, seperti berpikir dan belajar. Kecerdasan merupakan kemampuan untuk memperoleh, mengerti dan menggunakan kemampuan yang dimilikinya untuk menyelesaikan persoalannya yang dihadapinya.

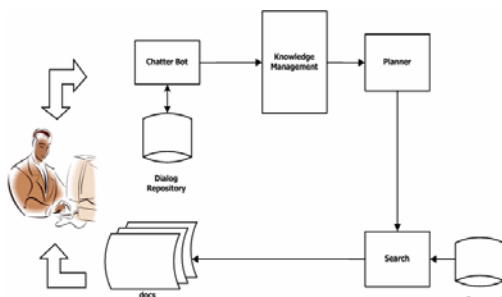
Salah satu alternatif penggunaan jasa komunikasi melalui internet yang sangat diminati saat ini adalah fasilitas Internet Relay Chat (IRC) atau chatting atau question answer (QA). QA yang umumnya dilakukan oleh dua orang atau lebih yang keduanya harus terhubung dengan jaringan internet. Namun saat ini dengan bantuan kecerdasan buatan memungkinkan conversation dapat dilakukan oleh satu orang saja (www.zabaware.com). Dengan menerapkan kecerdasan buatan maka seseorang dapat melakukan percakapan dengan sebuah "robot" atau program yang dibuat sedemikian rupa sehingga dapat saling berinteraksi dengannya seperti halnya percakapan dengan orang lain. Aplikasi percakapan yang dibuat pada penelitian ini dinamakan BotQA yang merupakan singkatan dari roBot Question Answer.

Hasil penelitian menunjukkan bahwa aplikasi yang dibuat dapat melakukan percakapan menggunakan kalimat bahasa Indonesia sederhana dan respon yang diberikan disertai dengan animasi ekspresi wajah.

Kata kunci: artificial intelligence, question answer, robot, bahasa Indonesia

1. PENGANTAR

Secara harfiah, chat dapat diartikan sebagai obrolan, namun dalam dunia komputer istilah ini merujuk pada kegiatan komunikasi melalui sarana baris-baris tulisan yang diketikkan melalui keyboard. Salah satu arti penting dari Chatterbot adalah pengertian bot. Bot merupakan singkatan dari kata robot yang berarti bekerja. Bot merupakan sebuah program yang mengandung sejumlah data dimana jika diberikan masukan maka akan memberikan jawaban. Perkataan robot saat ini tergantikan oleh kata agent, karena suatu robot biasanya dirancang untuk misi-misi tertentu. Chatterbot merupakan salah satu bentuk robot atau bot (www.simonlaven.com). Chatterbot dirancang untuk dapat berkomunikasi dengan manusia sebagai penggunaanya (www.whatis.techtarget.com). Chatterbot menggunakan kecerdasan buatan untuk mensimulasikan percakapan dengan penggunanya (www.chatterbot.net). Chatterbot dirancang untuk dapat mendekati sifat manusia. Konsep dasar Chatterbot adalah sebagai berikut:



Gambar 1. Konsep dasar aplikasi Chatterbot

Seiring dengan perkembangannya, Chatterbot tidak hanya digunakan sebagai lawan chatting yang harus dilakukan melalui jaringan internet, namun dapat juga digunakan secara *offline* (www.zabaware.com) dan difungsikan sebagai asisten yang dapat mengingat beberapa hal, misalnya alamat dan nomor telepon serta dapat juga dijadikan teman yang dapat diajak ngobrol beberapa hal (Campbell, 2005).

Supaya Chatterbot terlihat memiliki emosi maka diperlukan suatu animasi ekspresi wajah yang dipadukan dengan teknik pemrograman (Tatai, et al, 2002). Dengan adanya animasi ekspresi wajah disini diharapkan dapat membuat aplikasi ini menjadi lebih hidup karena seolah-olah aplikasi ini mirip dengan manusia yang memiliki ekspresi dan emosi (Abbattista, et al, 2004).

2. BotQA

Aplikasi BotQA ini merupakan aplikasi *Question Answer* yang sifatnya *offline*, jadi tidak seperti IRC yang harus dijalankan melalui jaringan internet. Input dari pengguna untuk aplikasi BotQA ini hanya berupa teks, jadi tidak menggunakan input suara seperti halnya orang yang sedang bicara. Begitu juga dengan outputnya, namun disini ditambahkan ekspresi wajah berupa gambar animasi karakter diam, senang, sedih, marah dan umum yang dibuat menggunakan Poser 5.



Gambar 2. Contoh gambar ekspresi wajah

Pada aplikasi BotQA ini terdapat data-data jawaban dari bermacam-macam pertanyaan atau balasan dari bermacam-macam obrolan. Data-data tersebut tersimpan dalam file-file berekstensi .txt yang dibuat menggunakan Notepad, jadi bukan berbentuk tabel-tabel seperti pada aplikasi-aplikasi lain yang membutuhkan basis data.

Aplikasi BotQA ini dapat berkomunikasi dengan pengguna menggunakan bahasa Indonesia. Tata bahasa yang digunakan tidak sepenuhnya menggunakan bahasa formal namun bahasa Indonesia yang sering digunakan sehari-hari. Jawaban atau respon yang diberikan oleh aplikasi BotQA ini tidaklah selalu eksak atau optimal, tetapi lebih bersifat cukup.

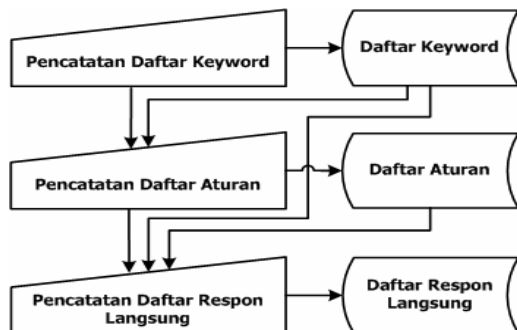
Aplikasi BotQA ini hanya dapat digunakan oleh satu user, bukan merupakan aplikasi yang bersifat multi user. Jadi user hanya dapat berkomunikasi dengan aplikasi ini saja, tidak dapat berkomunikasi dengan user lain. Aplikasi BotQA ini dibuat menggunakan Visual Basic 6.0.

Penelusuran jawaban dari obrolan pengguna dilakukan secara Breadth-First Search. Pada metode *Breadth-First Search* pencarian dilakukan pada semua node dalam setiap level secara berurutan dari kiri ke kanan. Jika pada satu level belum ditemukan solusi maka pencarian dilanjutkan pada level berikutnya (Firebaugh, 1988).

3. ALGORITMA BotQA

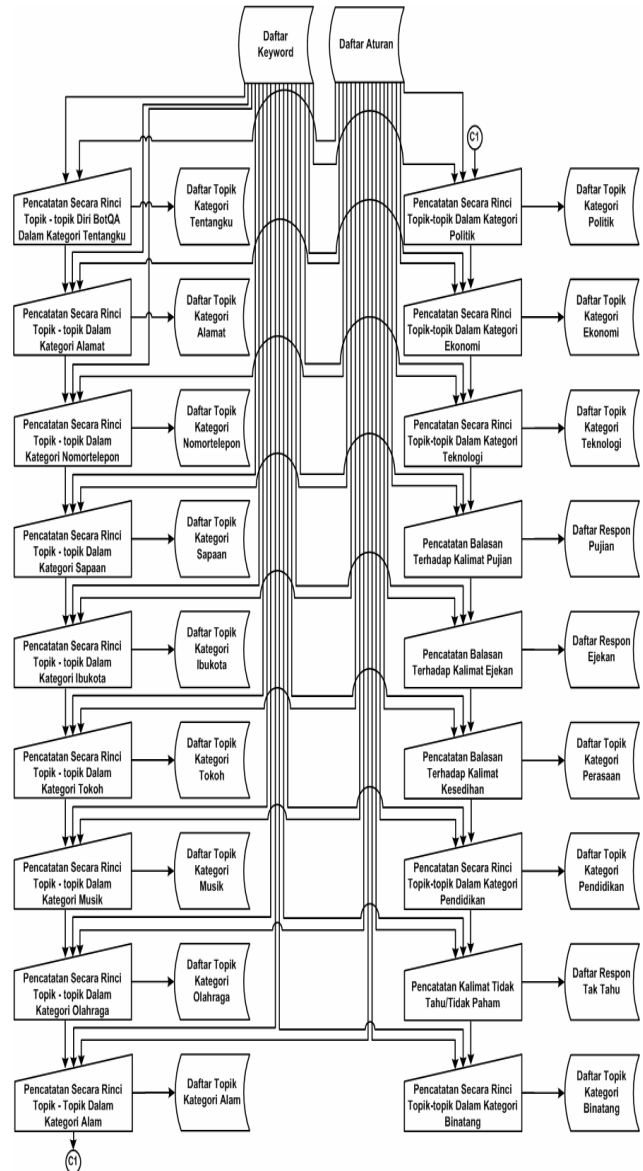
Penelitian ini mempunyai tujuan membuat sistem yang dapat membangun jawaban secara otomatis dari suatu dialog non verbal dilengkapi dengan animasi ekspresi wajah. Untuk mencapai tujuan tersebut maka disusun algoritma BotQA sebagai berikut:

- Mencatat kata-kata yang dijadikan daftar keyword.
- Membuat daftar aturan produksi.
- Membuat daftar keyword yang akan direspon langsung.



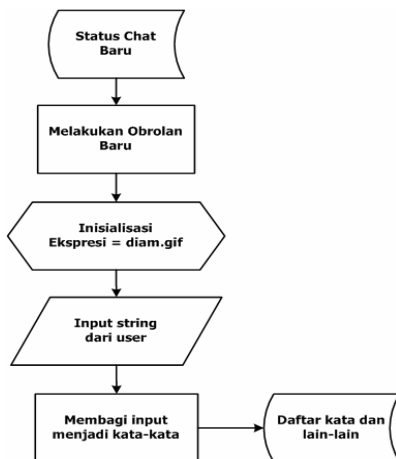
Gambar 3. Flowchart mencatat daftar keyword, aturan produksi dan daftar keyword yang bisa direspon langsung

- Mencatat secara rinci daftar jawaban dalam setiap kategori.



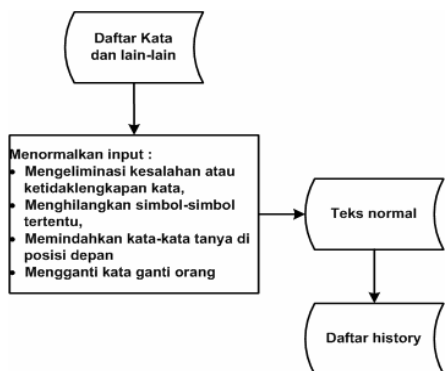
Gambar 4. Flowchart mencatat daftar jawaban dalam setiap kategori

- Pencatatan status suatu percakapan baru dan disimpan dalam *chat* baru.
- Menginisialisasi ekspresi wajah dengan *diam.gif* sebelum menerima masukan dari pengguna.
- Pengguna memberikan masukan teks untuk aplikasi.
- Membagi input teks menjadi kata-kata dan disimpan dalam *daftar kata*.



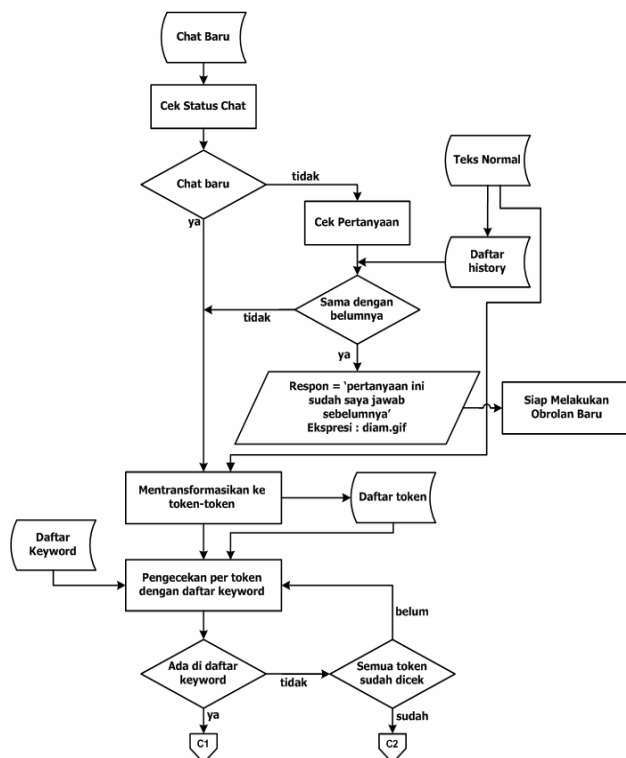
Gambar 5. Flowchart pembagian obrolan menjadi daftar kata

- i. Menormalisasikan input dengan mengeliminasi kesalahan atau ketidaklengkapan kata, menghilangkan simbol-simbol tertentu, memindahkan kata-kata tanya ke posisi depan dan mengganti kata ganti orang.



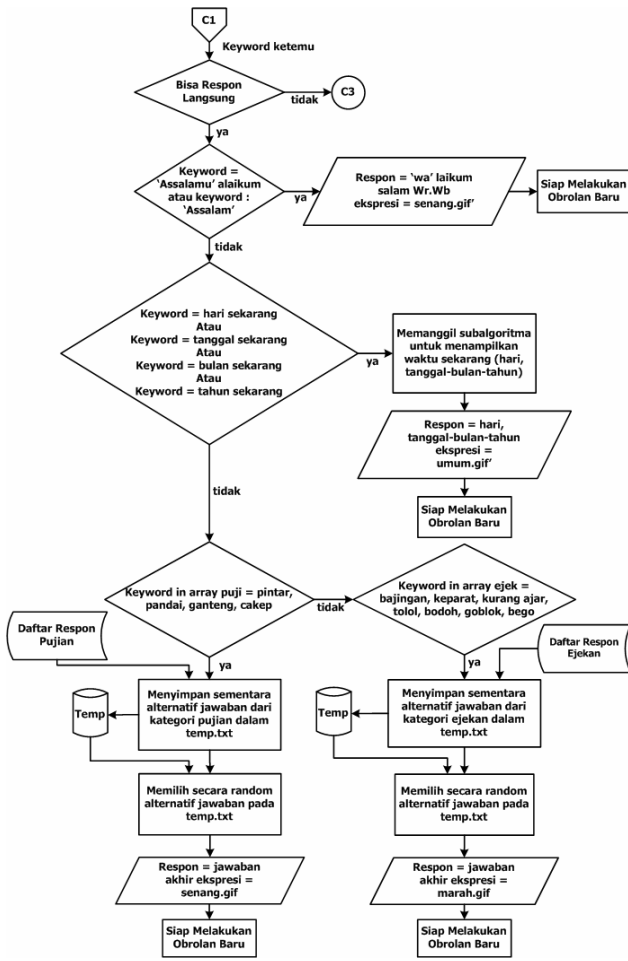
Gambar 6. Flowchart menormalisasikan input

- j. Mengecek status percakapan, apakah percakapan pertama atau tidak.
k. Apabila bukan percakapan pertama maka perlu dicek apakah pertanyaan atau pernyataannya sama dengan sebelumnya dengan memperhatikan *daftar history* jika ya maka sistem akan merespon dengan menjawab bahwa obrolan tersebut sudah ditanggapi sebelumnya kemudian kembali ke langkah *f* (siap melakukan obrolan baru). Jika tidak maka lanjutkan ke langkah *l*.
l. Apabila percakapan pertama maka langkah selanjutnya adalah mentransformasikan ke dalam token-token dan disimpan dalam *daftar token*. Pengecekan per token dengan daftar keyword. Jika ada dalam daftar keyword maka lanjutkan ke langkah *m* s/d *o*. Jika tidak ada dalam daftar keyword maka lanjutkan ke langkah *s*.
m. Pengecekan apakah keyword yang ditemukan dapat direspon langsung atau diolah terlebih dahulu untuk mendapatkan jawaban. Jika ya maka lanjutkan ke langkah *n*. Jika tidak ada maka lanjutkan ke langkah *o*.



Gambar 7. Flowchart pengecekan status obrolan dan keyword

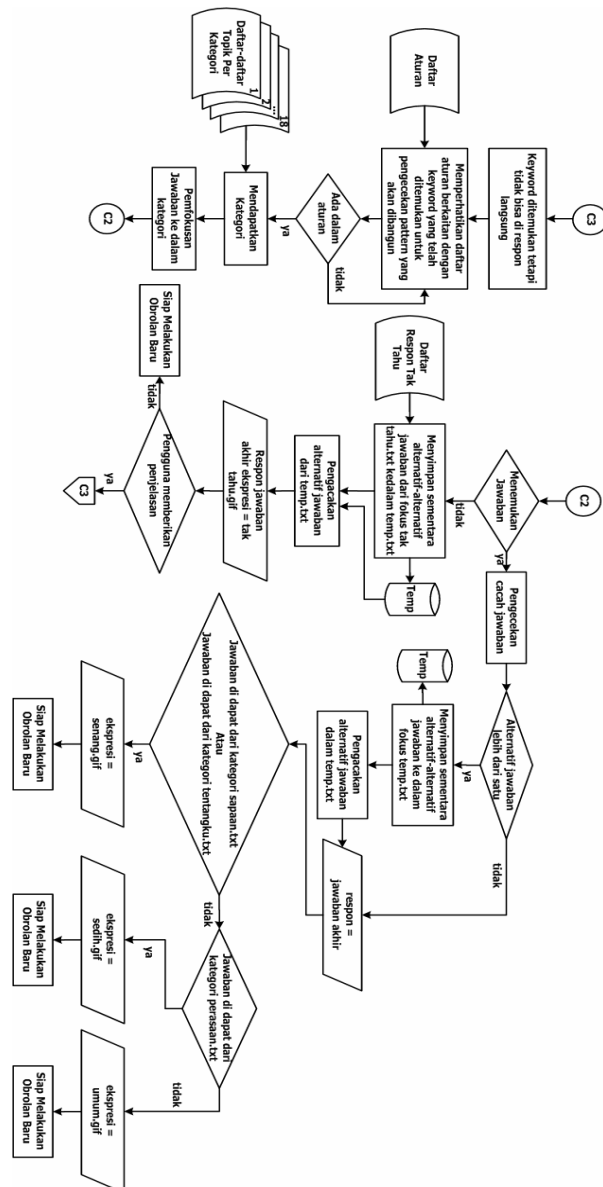
- n. Jika keywordnya adalah ucapan 'Assalammu'alaikum Wr. Wb' maka responnya adalah 'Wa'alaikum Wr. Wb' dengan animasi ekspresi wajah *senang.gif*. Jika keywordnya adalah hari sekarang, tanggal sekarang, bulan sekarang atau tahun sekarang maka sistem akan menjalankan subalgoritma untuk menentukan waktu aktual. Setelah waktu sekarang diketahui maka sistem akan menyajikannya sebagai jawaban dan menampilkan animasi ekspresi *umum.gif* dan kembali ke langkah *f* (siap melakukan obrolan baru).
Jika keywordnya adalah kata-kata pujian seperti *pintar*, *pandai*, *ganteng* atau *cakap* maka sistem akan menyimpan sementara alternatif-alternatif jawaban yang ada dalam *pujian.txt* ke dalam *temp.txt*, kemudian memilih secara random alternatif jawaban dalam *temp.txt*. Jawaban akhir akan dilengkapi dengan animasi ekspresi *senang.gif* dan kembali ke langkah *f* (siap melakukan obrolan baru).
Jika keywordnya adalah kata-kata ejekan seperti *bajingan*, *keparat*, *kurang ajar*, *tolol*, *bodoh*, *goblok* atau *bego* maka sistem akan menyimpan sementara alternatif-alternatif jawaban yang ada dalam *ejekan.txt* ke dalam *temp.txt*, kemudian memilih secara random alternatif jawaban dalam *temp.txt*. Jawaban akhir akan dilengkapi dengan animasi ekspresi *marah.gif* dan kembali ke langkah *f* (siap melakukan obrolan baru).



Gambar 8. Flowchart keyword ditemukan dan dapat direspon langsung

- o. Memperhatikan daftar aturan berkaitan dengan keyword yang telah ditemukan untuk pengecekan pattern yang akan dibangun. Setelah mendapatkan kategori maka dilakukan pemfokusan ke dalam file yang menyimpan daftar jawaban yang berkaitan dengan kategori tersebut. Jika jawaban ditemukan maka lanjutkan ke langkah *p*. Jika jawaban tidak ditemukan maka lanjutkan ke langkah *q*.
- p. Pengecekan cacah jawaban. Jika cacah jawaban yang ditemukan lebih dari satu maka alternatif-alternatif jawaban yang ada disimpan sementara dalam *temp.txt* dan dilakukan pemilihan salah satu jawaban secara random. Ketika sistem menampilkan jawaban akhir maka animasi ekspresi wajah yang ditampilkan bergantung dari kategori percakapan. Jika kategori percakapan adalah sapaan maka animasi ekspresi wajah yang ditampilkan adalah *senang.gif* dan kembali ke langkah *f* (siap melakukan obrolan baru). Jika kategori percakapan adalah selain sapaan dan berita kesedihan maka animasi ekspresi wajah yang ditampilkan adalah *sedih.gif*. Jika kategori percakapan adalah selain sapaan dan berita kesedihan maka animasi ekspresi wajah yang ditampilkan adalah *umum.gif* dan kembali ke langkah *f* (siap melakukan obrolan baru).

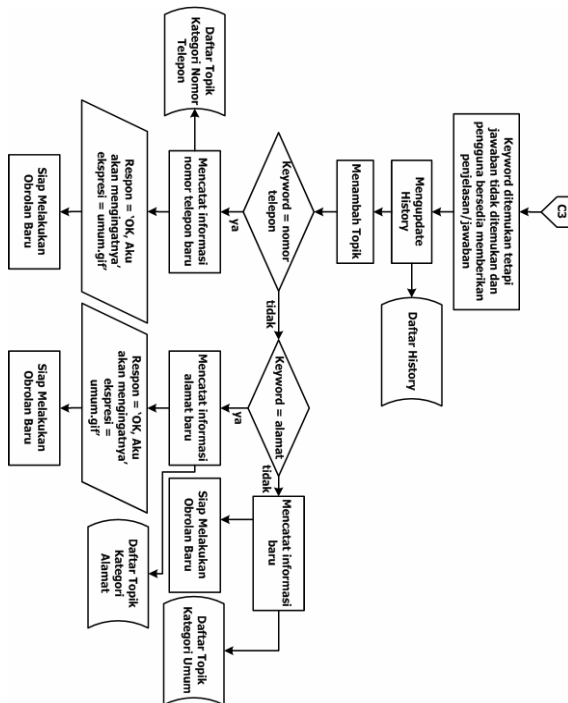
q. Jika jawaban tidak ditemukan maka alternatif-alternatif jawaban yang menyatakan ketidaktahuan dari file *taktahu.txt* disimpan sementara dalam *temp.txt* dan selanjutnya akan dipilih secara random untuk menentukan jawaban akhir. Jawaban akhir tersebut ditampilkan bersama animasi ekspresi wajah *taktahu.gif*. Jika pengguna memberikan penjelasan mengenai pertanyaan yang tidak ditemukan jawabannya tersebut maka lanjutkan ke langkah *r*. Jika pengguna tidak memberikan penjelasan mengenai pertanyaan yang tidak ditemukan jawabannya tersebut maka kembali ke langkah *f* (siap melakukan obrolan baru).



- 1. Daftar topik kategori Tentangku
- 2. Daftar topik kategori Alamat
- 3. Daftar topik kategori Nomor Telepon
- 4. Daftar topik kategori Sapaan
- 5. Daftar topik kategori Ibukota
- 6. Daftar topik kategori Tokoh
- 7. Daftar topik kategori Musik
- 8. Daftar topik kategori Olahraga
- 9. Daftar topik kategori Alam
- 10. Daftar topik kategori Politik
- 11. Daftar topik kategori Ekonomi
- 12. Daftar topik kategori Teknologi
- 13. Daftar topik kategori Pujian
- 14. Daftar topik kategori Ejekan
- 15. Daftar topik kategori Kesedihan
- 16. Daftar topik kategori Pendidikan
- 17. Daftar topik kategori Binatang
- 18. Daftar topik kategori Umum

Gambar 9. Flowchart keyword ditemukan tetapi tidak dapat langsung direspon

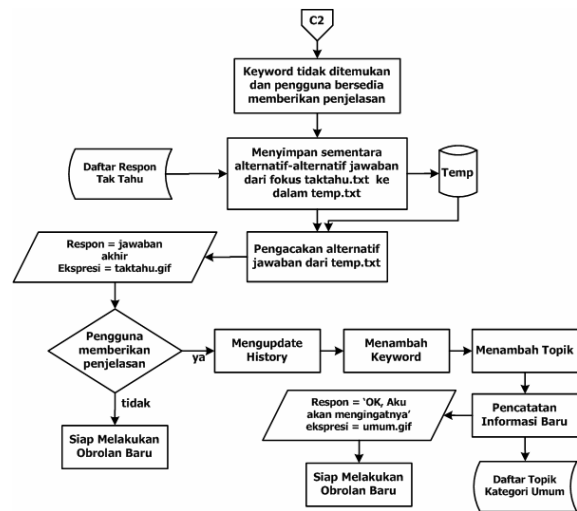
r. Sistem akan mengupdate *daftar history* bahwa pertanyaan yang sudah masuk tersebut belum ditemukan jawabannya. Kemudian *daftar jawaban* akan ditambah dengan ketentuan jika keywordnya adalah *nomor telepon* maka informasi nomor telepon baru tersebut akan dicatat ke dalam *nmrtelp.txt* dan sistem akan merespon dengan kalimat '*OK, Aku akan mengingatnya*' dengan tampilan animasi ekspresi wajah *umum.gif* dan kembali ke langkah f (siap melakukan obrolan baru). Jika keywordnya adalah *alamat* maka informasi alamat baru tersebut akan dicatat ke dalam *alamat.txt* dan sistem akan merespon dengan kalimat '*OK, Aku akan mengingatnya*' dengan tampilan animasi ekspresi wajah *umum.gif* dan kembali ke langkah f (siap melakukan obrolan baru).



Gambar 10. Flowchart keyword ditemukan tetapi jawaban tidak ditemukan dan pengguna bersedia memberikan penjelasan

s. Jika semua token sudah dicek tetapi keywordnya tidak ditemukan maka alternatif-alternatif jawaban yang menyatakan ketidaktahuan dari file *taktahu.txt* disimpan sementara dalam *temp.txt* dan selanjutnya akan dipilih secara random untuk menentukan jawaban akhir. Jawaban akhir tersebut ditampilkan bersama animasi ekspresi wajah *taktahu.gif*. Jika pengguna memberikan penjelasan mengenai pertanyaan yang tidak ditemukan jawabannya tersebut maka lanjutkan ke langkah t. Jika pengguna tidak memberikan penjelasan mengenai pertanyaan yang tidak ditemukan jawabannya tersebut maka kembali ke langkah f (siap melakukan obrolan baru).

t. Sistem akan mengupdate *daftar history* bahwa pertanyaan yang sudah masuk tersebut belum ditemukan jawabannya. Kemudian *daftar jawaban* akan ditambah dengan mencatat informasi baru tersebut ke dalam *umum.txt* dan sistem akan merespon dengan kalimat '*OK, Aku akan mengingatnya*' dengan tampilan animasi ekspresi wajah *umum.gif* dan kembali ke langkah f (siap melakukan obrolan baru).



Gambar 11. Flowchart keyword tidak ditemukan dan pengguna bersedia memberikan penjelasan

4. PROSES PENELUSURAN JAWABAN

Berikut proses dialog antara pengguna dan aplikasi BotQA:

Pengguna: *yg membuat alam siapa ?*

a. Membagi input menjadi kata-kata:

*yg
membuat
alam
siapa
?*

b. Menormalkan input dengan mengeliminasi kesalahan atau ketidaklengkapan kata, menghilangkan simbol-simbol tertentu, memindahkan kata-kata tanya ke posisi depan dan mengganti kata ganti orang.

*siapa
yang
membuat
alam*

c. Mentransformasikan ke dalam token-token.

d. Menentukan keyword:

- *siapa* tidak bisa dijadikan keyword
- *membuat* bisa dijadikan keyword

e. Memperhatikan rule untuk keyword "*membuat*" untuk pengecekan pattern yang dibangun.

```
<pattern> membuat <pattern>
bagaimana membuat *
kapan membuat *
dimana membuat *
→ siapa membuat *
.....
.....
```

-
siapa membuat kamu aku
*siapa membuat lagu **
*siapa membuat lukisan **
*siapa membuat teori **
*siapa membuat goal **
 → *siapa membuat alam*

- f. Pattern yang dibangun “*siapa membuat alam*” termasuk dalam kategori *alam*
 g. Setelah mendapatkan kategori *alam* kemudian pencarian jawaban difokuskan ke dalam file *alam.txt*

Luas hutan di Indonesia semakin berkurang
Indonesia termasuk daerah beriklim tropis

- *Tuhan membuat alam dan isinya*
 h. Mendapat jawaban akhir: “*Tuhan membuat alam dan isinya*”
 i. Untuk obrolan kategori *alam* ekspresi wajah yang ditampilkan adalah *umum.gif*.

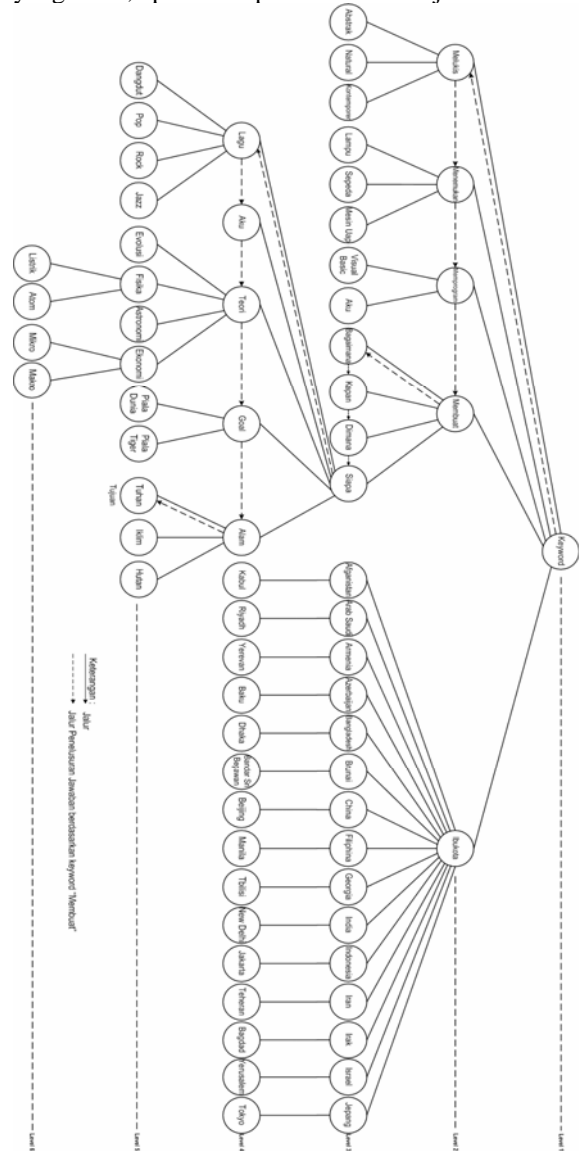
Ilustrasi proses penelusuran jawaban dari obrolan di atas ditunjukkan oleh gambar 12. Tree ditelusuri secara Breadth-First Search. Tree ditelusuri dari node paling kiri yaitu node “*melukis*”. Karena keyword yang dimaksud bukan “*melukis*” maka penelusuran dilanjutkan pada node-node berikutnya dalam satu level yaitu level 2 hingga ditemukan keyword “*membuat*”. Setelah memilih keyword “*membuat*” maka dengan menggunakan aturan produksi dari node “*membuat*” untuk pengecekan pattern yang akan dibangun. Dari node “*membuat*” penelusuran diteruskan pada node di bawahnya dimulai dari node paling kiri yaitu node “*bagaimana*”, kemudian dilanjutkan ke node-node disebelah kanannya (masih dalam satu level) sampai dipilih node “*siapa*”. Dari node “*siapa*” pada level 3 penelusuran dilanjutkan pada level di bawahnya yaitu level 4, dimulai dari node “*lagu*” dan node-node dalam satu level disebelah kanannya sampai ke node “*alam*”. Dengan demikian pattern yang dihasilkan adalah “*siapa membuat alam*”. Pattern “*siapa membuat alam*” termasuk dalam kategori “*alam*”. Setelah mendapatkan kategori yang dicari, dari node “*alam*” pencarian jawaban dilanjutkan pada node level 5. salah satu node pada level 5 yaitu “*Tuhan*” merupakan sebuah solusi sehingga penelusuran berhenti dan jawaban untuk obrolan dari pengguna “*yg membuat alam siapa?*” diperoleh, yaitu “*Tuhan membuat alam dan isinya*”.

5. Hasil Uji Coba

Dalam Gambar 13 adalah contoh percakapan yang dilakukan oleh pengguna dengan aplikasi BotQA.

Pada kasus ini tidak ditemukan nomor telepon dari orang yang dimaksud tetapi apabila pengguna bersedia memberikan informasi nomor telepon baru maka informasi tersebut akan disimpan dalam *NmrTelp.txt*.

Aplikasi tidak mengetahui jawaban dari pertanyaan yang dilontarkan oleh pengguna maka ketika pengguna menyampaikan penjelasan yang berhubungan dengan pertanyaan tersebut selanjutnya akan diingat oleh aplikasi. Oleh aplikasi jawaban yang bersifat penjelasan di atas disimpan ke dalam file *Umum.txt*, sehingga pada saat ditanyakan hal yang sama, aplikasi dapat memberikan jawaban.



Gambar 12. Penelusuran jawaban untuk obrolan “yg membuat alam siapa”



Gambar 13. Percakapan pertama



Gambar 14. Percakapan kedua



Gambar 15. Percakapan ketiga



Gambar 16. Percakapan keempat



Gambar 17. Percakapan kelima

Pertanyaan di atas mempunyai alternatif jawaban lebih dari satu, sehingga alternatif-alternatif jawaban tersebut disimpan sementara dalam file *Temp.txt* kemudian jawaban yang akan dipilih secara random.

6. KESIMPULAN

Dari hasil uji coba aplikasi BotQA dapat disimpulkan bahwa:

- Aplikasi yang dibuat mempunyai kemampuan seperti yang ditunjukkan pada tabel 1.
- Tata bahasa dari kalimat masukan pengguna harus ditentukan dalam bentuk aturan produksi

sehingga tidak sembarang kalimat dapat diproses.

- Adanya pengorganisasian informasi baru berupa jawaban-jawaban dari pengguna, sistem dapat menjawab pertanyaan-pertanyaan yang berhubungan dengan informasi tersebut.
- Aplikasi ini dalam menjawab pertanyaan atau merespon pernyataan pengguna disertai dengan animasi ekspresi wajah yang ditampilkan bergantung dari isi masukan pengguna dan kalimat jawaban yang ditemukan.

Tabel 1. Daftar kemampuan BotQA

No	Daftar Kemampuan
1	Mengingat tanggal aktual
2	Mengingat nama hari aktual dalam seminggu
3	Mengingat nama bulan aktual dalam setahun
4	Mengingat tahun aktual
5	Mengingat identitas diri BotQA.
6	Menjawab kalimat salam "Assalamu'alaikum"
7	Menjawab kalimat sapaan-sapaan.
8	Mengingat kata-kata ejekan atau cemoohan dan memberikan tanggapan
9	Mengingat kata-kata pujian dan memberikan tanggapan
10	Mengingat hal-hal yang berhubungan dengan perasaan dan memberikan tanggapan
11	Mengingat daftar alamat yang telah dimasukkan oleh programmer atau daftar alamat yang diperoleh dari pembicaraan dengan user.
12	Mengingat daftar nomor telepon yang telah dimasukkan oleh programmer atau daftar nomor telepon yang diperoleh dari pembicaraan dengan user.
13	Mengingat data-data ibukota dari negara-negara yang ada di seluruh penjuru dunia.
14	Mengingat beberapa nama tokoh, artis, penemu dan lain-lain dari seluruh penjuru dunia.
15	Mengingat hal-hal yang berhubungan dengan musik.
16	Mengingat hal-hal yang berhubungan dengan olahraga.
17	Mengingat hal-hal yang berhubungan dengan alam.
18	Mengingat hal-hal yang berhubungan dengan politik.
19	Mengingat hal-hal yang berhubungan dengan ekonomi.
20	Mengingat hal-hal yang berhubungan dengan teknologi.
21	Mengingat hal-hal yang berkaitan dengan pendidikan.
22	Mengingat kata-kata tidak tahu atau tidak paham
23	Mengingat hal-hal yang berhubungan dengan binatang peliharaan.
24	Mengingat informasi baru yang didapatkan dari pengguna.
25	Memilih secara random alternatif-alternatif jawaban.
26	Apabila aplikasi tidak mengetahui jawaban dari pertanyaan yang dilontarkan oleh pengguna maka ketika pengguna menyampaikan penjelasan yang berhubungan dengan pertanyaan tersebut selanjutnya akan diingat oleh BotQA.
27	Menjawab atau merespon pertanyaan atau pernyataan pengguna disertai dengan animasi ekspresi wajah yang akan ditampilkan bergantung dari masukan pengguna dan data jawaban yang ditemukan.
28	Mengingat pertanyaan atau pernyataan seorang pengguna yang telah diberikan kepadanya dalam satu sesi percakapan dan mengingat pertanyaan atau pernyataan yang mana saja yang sudah berhasil ditemukan jawabannya

7. SARAN

Supaya aplikasi BotQA ini dapat berkembang menjadi lebih baik, berikut ini diberikan saran dan harapan yaitu:

- a. Dalam memberikan pertanyaan-pertanyaan atau dalam melakukan pembicaraan dengan aplikasi BotQA ini hendaknya menggunakan kata-kata yang lengkap dan jelas. Hal tersebut dilakukan agar aplikasi ini memberikan jawaban yang bagus atau mendekati tepat.
- b. Aplikasi BotQA dapat dikembangkan atau diintegrasikan dengan *speech recognition* sehingga input dan output dari aplikasi ini tidak hanya berbentuk teks namun juga ada suaranya.

PUSTAKA

- Abbattista, F., Catucci, G., Semeraro, G., & Zambetta, F., 2004, *SAMIR: A Smart 3D Assistant on the Web*, www.psychology.org/File/PSYCHOLOGY_JOURNAL_2_1_ABBATTISTA.pdf/downloads (2006, Oktober, 10)
- Campbell, June, 2005, *Billy and Daisy My New Friends* (pdf), www.computoredge.com/downloads (2006, Oktober, 10)
- Firebaugh, Morris W. (1998). *Artificial Intelligent A Knowledge-Based Approach*, PWS-KENT Publishing Company
- Tatai, G., Laufer, L., Szalo, A., & Csordas, A., 2002, *On Transmitting Emotions – Experiments on Multimodal Presence over the Internet* (pdf), www.presence-research.org/papers/Tatai.pdf/downloads (2006, Oktober, 10)
- www.chatterbot.net (2006, Oktober, 10)
- www.simonlaven.com (2006, Oktober, 10)
- www.whatis.techtarget.com (2006, Oktober, 10)
- www.zabaware.com (2006, Oktober, 10)