BAB II

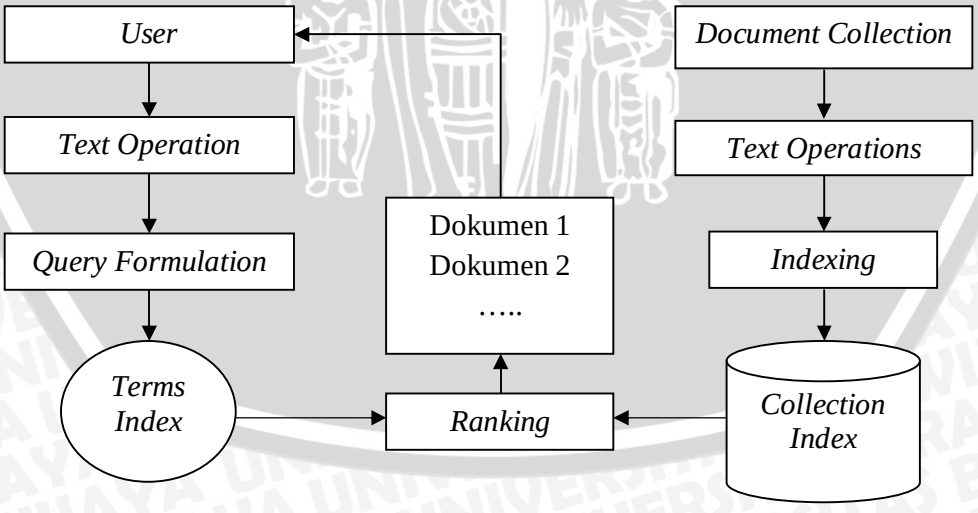
DASAR TEORI

Bab ini membahas dasar teori yang digunakan untuk menunjang penulisan skripsi mengenai Implementasi *Pattern Based Approach* pada *Question Answering System* beberapa dasar teori yang dimaksud adalah *Information Retrieval System, Natural Language Processing, Question Answering System, Pattern Learning Approach,* dan *Stemming* Arifin.

2.1 *Information Retrieval System*

Information Retrieval (IR) System adalah sistem yang digunakan untuk menemukan kembali (*retrieve*) informasi-informasi yang relevan terhadap kebutuhan pengguna dari suatu kumpulan informasi secara otomatis [BUN-08].

Sistem *IR* terutama berhubungan dengan pencarian informasi yang isinya tidak memiliki struktur. Demikian pula ekspresi kebutuhan pengguna yang disebut *query*, juga tidak memiliki struktur. Hal ini yang membedakan sistem *IR* dengan sistem *database*. Dokumen adalah contoh informasi yang tidak terstruktur. Isi dari suatu dokumen sangat tergantung pada pembuat dokumen tersebut.



Gambar 2.1 Bagian – bagian *Information Retrieval* (IR) *System*

Sumber: [BUN-08]

Dari gambar 2.1, terlihat bahwa terdapat dua proses operasi dalam sistem *IR*. Proses pertama dimulai dari koleksi dokumen dan proses kedua dimulai dari *query* pengguna. Proses pertama yaitu pemrosesan terhadap koleksi dokumen menjadi *database* indeks tidak ada ketergantungan dengan proses kedua. Sedangkan proses kedua tergantung dari keberadaan *database* indeks yang dihasilkan pada proses pertama.

Bagian-bagian dari sistem *IR* menurut gambar 1 meliputi :

1) ***Text Operations*** (operasi terhadap teks) yang meliputi pemilihan kata-kata dalam *query* maupun dokumen (*term selection*) dalam pentransformasian dokumen atau *query* menjadi *term index* (indeks dari kata-kata).
2) ***Query formulation*** (formulasi terhadap *query*) yaitu memberi bobot pada indeks kata-kata *query*.
3) ***Ranking*** (perangkingan), mencari dokumen-dokumen yang relevan terhadap *query* dan mengurutkan dokumen tersebut berdasarkan kesesuaiannya dengan *query*.
4) ***Indexing*** (pengindeksan), membangun *database* indeks dari koleksi dokumen. Dilakukan terlebih dahulu sebelum pencarian dokumen dilakukan.

Sistem *IR* menerima *query* dari pengguna, kemudian melakukan perangkingan terhadap dokumen pada koleksi berdasarkan kesesuaiannya dengan *query*. Hasil perangkingan yang diberikan kepada pengguna merupakan dokumen yang menurut sistem relevan dengan *query*. Namun relevansi dokumen terhadap suatu *query* merupakan penilaian pengguna yang subjektif dan dipengaruhi banyak factor seperti topik, pewaktuan, sumber informasi maupun tujuan pengguna.

2.2 *Natural Language Processing*

Natual Language Processing (NLP) atau pengolahan bahasa alami merupakan salah satu bidang ilmu *Artificial Intelligence* (Kecerdasan Buatan) yang mempelajari komunikasi antara manusia dengan komputer melalui bahasa alami. *NLP* mencoba membuat komputer mampu memahami suatu perintah yang dituliskan dalam bentuk bahasa sehari-hari dan diharapkan komputer juga

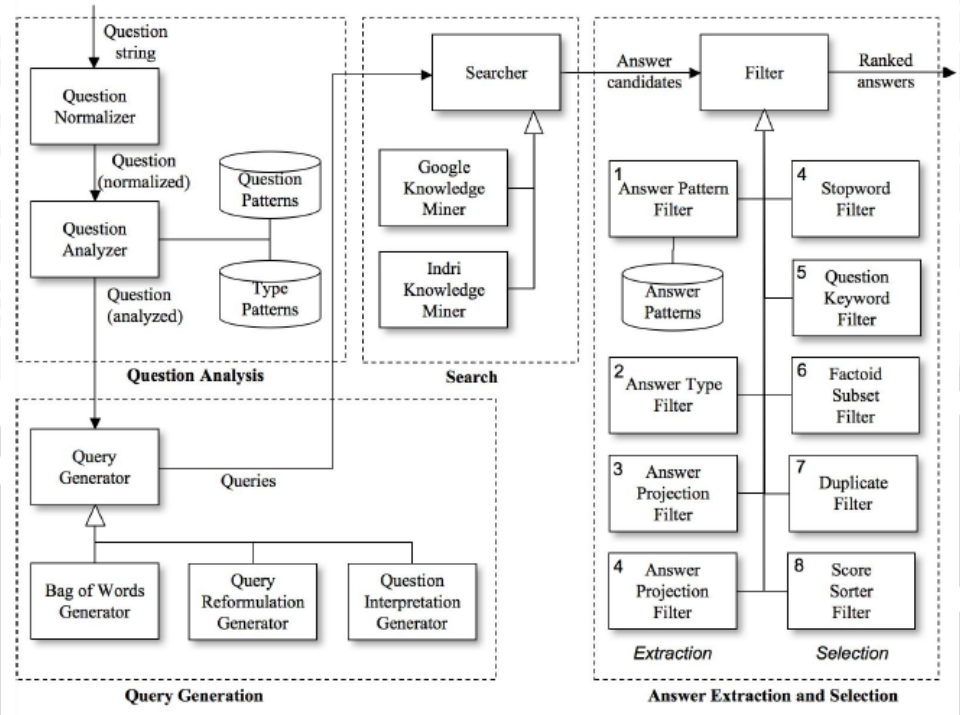
merespon dalam bahasa yang mirip dengan bahasa *natural*. Setelah komputer bisa memahami perintah dalam bahasa natural, maka diharapakan sistem komputer juga dapat memberikan respon dalam bahasa natural pula. [FAT-09]

Natural Language processor tidak memperdulikan bagaimana suatu kalimat diinputkan ke komputer. Tugasnya adalah mengekstrak informasi dari kalimat. *NLP* tidak bisa digunakan sendirian, kecuali dalam riset. Tetapi *NLP* dapat menyediakan *front end* untuk program komputer yang lain-terutama *database manager* dan *generelized problem solver*. Juga untuk menjalankan system operasi, banyak riset yang mengarah ke *NLP driven Operating System*. Dan yang paling utama adalah pada pengembangan robotika, dimana dituntut interaksi yang efektif antara mesin dan manusia. [FAT-09]

2.3 *Question Answering System*

Question Answering System merupakan sebuah sistem yang secara otomatis menjawab suatu pertanyaan yang diajukan dalam bahasa alami (*natural language*). Untuk memperoleh jawaban dari pertanyaan, *Question Answering System* dapat menggunakan basisdata ataupun koleksi dokumen dalam bahasa alami. Hasil kembalian dari *Question Answering System* adalah berupa kutipan teks singkat atau bahkan frase sebagai jawabannya.

Question Answering System merupakan kombinasi antara *Information Retrieval (IR)* dengan *Natural Language Processing (NLP)*. *Question Answering System* memiliki tujuan menampilkan jawaban berdasarkan *query* dalam bentuk pertanyaan yang diajukan oleh pengguna [ARM-11]. Perbedaan yang mendasar antara *Question Answering System* dengan *IR* terletak pada masukan (*query*) dan keluaran yang dihasilkan. Pada *IR query* yang dimasukkan berupa kata atau kalimat pertanyaan dan keluaran yang dihasilkan adalah dokumen yang dianggap relevan oleh sistem. Sedangkan pada *Question Answering System*, *query* berupa kalimat tanya dan keluarannya berupa jawaban (entitas) yang dianggap sesuai oleh sistem sehingga memungkinkan sistem tidak mengembalikan jawaban apapun.



Gambar 2.2 Arsitektur *Question Answering System* pada *OpenEphyra*

Sumber: [SCH-07]

2.3.1 Arsitektur *Question Answering System*

Question Answering System yang dikembangkan dengan tujuan, sumber informasi, dan teknik yang berbeda dapat memiliki arsitektur yang berbeda pula. Gambar 2.2 menunjukkan arsitektur dari *Question Answering System* pada *OpenEphyra*. *OpenEphyra* adalah *framework open source* yang digunakan untuk membangun *Question Answering System*. *Question Answering System* pada *OpenEphyra* memiliki empat bagian utama, yaitu *Question Analysis, Query Generation, Search,* dan *Answer Extraction and Selection.* [SCH-07]

2.3.2 *Question Analysis*

Pertanyaan yang diinputkan oleh pengguna seringkali tidak sesuai dengan aturan dan persyaratan yang ditentukan. Hal ini dapat mengakibatkan sistem sulit mengenali maksud dari pertanyaan yang diajukan. Oleh karena itu diperlukan

proses yang mengubah pertanyaan menjadi format yang dapat dikenali oleh *Question Answering System*.

Question Analysis memproses input pertanyaan dari pengguna menjadi pertanyaan yang memenuhi syarat-syarat yang telah ditentukan. Bagian ini mengandung dua buah proses, yaitu *Question Normalizer* dan *Question Analyzer*.

2.3.3 *Query Generation and Search*

Representasi pertanyaan perlu diubah menjadi *string query* yang untuk selanjutnya *query* tersebut digunakan untuk mencari dokumen yang relevan. Pada *OpenEphyra* digunakan tiga buah metode *query* generator, yaitu *Bag of Words*, *Query Reformulation*, dan *Question Interpretation Generator*. Sedangkan pada *search engine* digunakan *Google Knowledge Miner* dan *Indri Knowledge Miner*.

2.3.4 *Answer Extraction and Selection*

Dokumen-dokumen yang dikembalikan oleh *search engine* dianalisis lebih lanjut, yaitu dengan membuang dokumen yang tidak memenuhi kriteria yang ditentukan. Hal ini dilakukan dengan memproses dokumen-dokumen ke dalam *filter* yang telah ditentukan. *Filter* tersebut juga memberikan skor atau nilai untuk dapat melakukan *ranking* pada dokumen sesuai dengan tingkat relevansinya. Hasil yang paling tinggi skornya adalah kandidat yang paling mungkin untuk menjadi jawaban dari pertanyaan yang diajukan.

2.4 *Pattern Based Approach*

OpenEphyra adalah salah satu contoh *Question Answering System* yang menggunakan *Pattern Based Approach* untuk penggolongan pertanyaan. *OpenEphyra* dapat mempelajari pasangan pertanyaan-jawaban dan menggunakan *IR System* standar untuk mengambil potongan teks yang sesuai untuk ekstraksi pola. Dari hasil pengujian metode *Pattern Based Approach* pada *Question Answering* berbahasa Indonesia didapatkan hasil akurasi interpretasi sebesar 90% dan akurasi jawaban sebesar 35 % [TOB-10].

Dua tahap utama dalam *Pattern Based Approach* yaitu

1. Pertama adalah dengan mempelajari pola pertanyaan dari *template* pertanyaan sesuai dengan tipe pertanyaannya. Tujuannya adalah untuk mengintepretasikan pertanyaan dan mengubahnya ke dalam bentuk *query*. Template pertanyaan ini dikembangkan secara *manual* yang independen untuk setiap *natural language*.
2. Langkah kedua adalah dengan mempelajari pola jawaban dari pasangan pertanyaan-jawaban. Tujuannya adalah untuk mengekstrak kandidat jawaban dari dokumen yang relevan dan mengurutkannya.

2.5 *Stemming* Arifin

Stemmin Arifin adalah metode *stemming* yang dikembangkan oleh Arifin dan Setiono. Metode ini lebih sederhana jia dibandingkan denga metode Nazief dan Adriani, tetapi tetap menggunakan pendekatan yang sama dengan menggunakan kamus, dan secara progresif menghilangkan imbuhan. Pendekatan ini berusaha menghilangkan awalan dan dilanjutkan dengan akhiran, dan berhenti bila kata hasil *stemming* ditemukan pada kamus atau jumlah imbuhan yang dihilangkan mencapai maksimum 2 awalan dan 3 akhiran. Jika kata tidak dapat ditemukan setelah semua awalan dan akhiran dihilangkan, imbuhan akan dikembalikan pada kata dalam semua kombinasi yang memungkinkan sehingga kemungkinan kesalahan pada *stemming* dapat diminimalisir. Pada pengujian *stemming* menggunakan metode Arifin pada kata-kata dalam bahasa Indonesia didapatkan tingkat kebenaran sebesar 87,7% [ASI-05].