JURUSAN TEKNIK INFORMATIKA

FAKULTAS TEKNOLOGI INFORMASI

INSTITUT TEKNOLOGI SEPULUH NOPEMBER

USULAN TUGAS AKHIR

1. IDENTITAS PENGUSUL

NAMA : Sayval Balad

NRP : 5108 100 148

DOSEN WALI : Yudhi Purwananto, S.Kom.,M.Sc.

1. JUDUL TUGAS AKHIR

**“Pengembangan Sistem Temu Kembali Informasi Berbasis Agent untuk Pencarian Berita Online”**

1. LATAR BELAKANG

Berita itu sangat penting, dalam bentuk dan ruang lingkup apapun. Berita akan menambah wawasan, tanpa berita kita tidak akan tahu apa yang terjadi di dunia luar dan kejadian-kejadian penting yang terjadi saat itu, dan dalam berita juga terkadang terdapat beberapa informasi penting yang bisa membantu kehidupan sehari-hari, bahkan bisa mencegah jatuhnya korban jiwa pada suatu kejadian bencana, misalnya ada berita mengenai perkiraan terjadi gempa pada dini hari dari sumber yang terpercaya, jika kita mengetahui berita tersebut, kita bisa segera mempersiapkan diri sebelum bencana gempa bumi tersebut terjadi dan memberitahu masyarakat sekitar. Dengan demikian mereka yang mengetahui berita bisa mengungsi dan mengurangi kerugian yang diterima. Itulah pentingnya berita.

Berita saat ini sangat mudah didapatkan, tidak hanya terbatas pada surat kabar atau melalui internet, siaran televisi ataupun radio, dan dengan topik yang bermacam-macam. Salah satu sumber berita adalah melalui internet. Internet menyediakan banyak sekali situs yang digunakan khusus untuk berita, seperti detik.com, biasanya bentuk berita dalam situs sejenis berbentuk artikel, sehingga masyarakat mampu mengakses berita dengan mudah.

Namun ada fitur penting yang tidak disediakan oleh situs penyedia berita saat ini yaitu pencarian topik berita yang spesifik sedang dicari oleh pencari berita. Banyaknya berita yang ditampilkan dalam situs tersebut bisa membuat bingung pencari berita yang menginginkan berita mengenai suatu permasalahan tertentu. Mereka harus mencari berita yang diinginkan secara manual, dengan menelusuri berita yang jumlahnya tidak sedikit itu satu per satu.

Untuk mengatasi hal ini diperlukan aplikasi yang mampu memfilter berita berdasarkan topik yang diinginkan menggunakan kata kunci. Misalnya, kita ingin mencari berita mengenai informatika its, dengan demikian kata kuncinya adalah “informatika its”, dan output berita yang akan muncul berupa berita mengenai informatika its, bisa berupa berita mengenai dosen di informatika its, mahasiswa, kondisi bangunan, jumlah mahasiswa, semua yang berhubungan dengan kata kunci tersebut.

Pada tugas akhir ini akan menerapkan konsep *agent* pada aplikasi pencarian berita. Dengan menggunakan agent, pencarian berita pada situs berita di internet yang membutuhkan waktu yang lama dan melelahkan karena banyaknya berita bisa dihilangkan. Agent yang akan mencarikan berita dari situs berita dari web penyedia berita internet kemudian diproses sesuai dengan kata kunci menggunakan metode *Information Retrieval* (Sistem Temu Kembali Informasi) berbasis TF-IDF. Hasil keluaran dari proses ini berupa list berita yang diinginkan sesuai dengan kata kunci yang di masukkan.

1. TUJUAN

Tujuan pembuatan tugas akhir ini adalah merancang dan membuat aplikasi perangkat lunak berbasis agentmenggunakan framework JADE untuk mempermudah *user* dalam mencari informasi berita berdasarkan topik diinginkan menggunakan kata kunci dengan tujuan untuk mempermudah pencarian berita.

1. PERMASALAHAN

Adapun yang menjadi permasalahan utama dalam tugas akhir ini :

1. Bagaimana membuat suatu agent untuk mengumpulkan berita dari situs penyedia berita yang telah ditentukan.
2. Bagaimana mengolah data yang data telah dikumpulkan untuk dipilih berita yang terkait dengan kata kunci yang diinputkan oleh user.
3. Mengimplementasikan cara kerja dari metode untuk filter berita online, yaitu metode Information Retrieval berbasis TF-IDF.
4. BATASAN MASALAH

Dari permasalahan yang telah diuraikan di atas, terdapat beberapa batasan masalah terhadap tugas akhir ini, yaitu :

1. Data berupa berita didapatkan melalui web HTML buatan sendiri.
2. Penelitian ini tidak membahas keamanan dari sistem berbasis agent.
3. Aplikasi ini dibangun dengan menggunakan bahasa pemrograman Java dan framework JADE.
4. RINGKASAN ISI TUGAS AKHIR

Tugas akhir bertujuan membangun suatu sistem untuk memberikan list berita yang diinginkan dari beberapa berita yang ada pada *desktop application*. Fungsi dari sistem ini adalah untuk memudahkan pencari berita dalam menemukan berita yang diinginkannya dalam situs penyedia berita.

Terdapat 4 komponen dalam sistem, yaitu :

1. *Sender* : Mengirim kata kunci dari berita yang diinginkan ke agent A.

Jumlah sender bisa lebih dari 1

1. Agent A : Untuk menerima kata kunci dari sender, dan mengirim kata

kunci tersebut ke agent B

1. Agent B : menerima kata kunci dari agent A, mengumpulkan berita

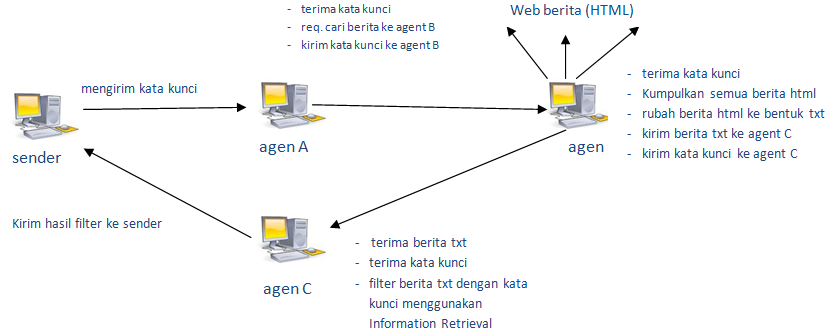
dari situs penyedia berita, merubah data berita dari bentuk HTML ke teks, mengirim data berita dalam bentuk teks ke

agent C dan kata kunci (crawled)

1. Agent C : menerima data berita dari agent B, melakukan filter kepada data berita menggunakan kata kunci dengan metode

Information Retrieval berbasis TF-IDF, mengirim data berita yang sudah difilter ke sender (filtering).

Sender mengirimkan data mengenai informasi berita yang diinginkan / kata kunci melalui agent A. Kemudian agent A akan memberikan request kepada agent B untuk mengumpulkan semua berita dari situs penyedia berita yang sudah ditentukan dari internet dan mengirimkan kata kunci dari sender. Berita yang telah dikumpulkan oleh agent B yang sudah dalam bentuk teks beserta kata kunci kemudian dikirimkan ke agent C untuk dilakukan proses pemilihan berita yang berkaitan dengan kata kunci yang telah diberikan sebelumnya menggunakan metode Information Retrieval berbasis TF-IDF. Hasil dari pemrosesan tersebut berupa list berita yang berkaitan dengan kata kunci yang diinginkan oleh user. Kemudian list berita tersebut dikirimkan kepada sender oleh agent sehingga useryang menggunakan aplikasi ini mendapatkan berita yang diiinginkan.



Gambar Struktur Kerja Agent

1. **Agent**

Agent yang dimaksud disini adalah program yang akan melakukan suatu proses yang telah ditentukan oleh pembuat agent ketika menerima suatu perintah yang akan mengaktifkan proses tersebut. Agent bekerja secara *autonomous* / independen dalam suatu *environment*[4].

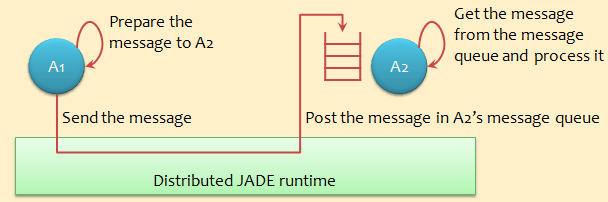
Agent bisa diatur untuk terus melakukan proses tanpa henti, berhenti pada interval waktu tertentu, atau berhenti ketika sudah menyelesaikan tugasnya.

Agent memiliki *set of behavior* / action yang berisi sub proses, dan behavior ini bisa ditambahkan berapapun pada agent sesuai dengan kebutuhan. Behavior juga bisa dijalankan bersamaan / *multithreaded*, dan behavior mampu mengaktifkan behavior lainnya ketika selesai melakukan proses[4].

Agent juga mampu berkomunikasi dengan agent lainnya. JADE adalah salah satu framework yang memfasilitasi fungsi agent[4].

Contoh potongan *class code* sederhana JADE.

|  |
| --- |
| **import** jade.core.Agent;  **public** **class** HelloWorld **extends** Agent {  **protected** **void** setup() {  System.out.println(“Hello World!”);  System.out.println(“My name is: “ + getLocalName());  }  } |

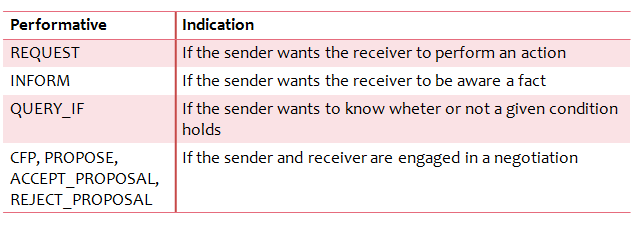


Gambar Pengiriman Pesan menggunakan JADE [4]

Komunikasi antar agent menggunakan bantuan dari framework JADE, jenisnya *Asynchronous message passing*, jadi message bisa dikirimkan dari sender lebih dari satu, sender menggunakan *message queue* dan memprosesnya secara bergantian[4].

Format message dalam sistem yang dikirimkan menggunakan aturan *ACL Language*, formatnya antara lain :

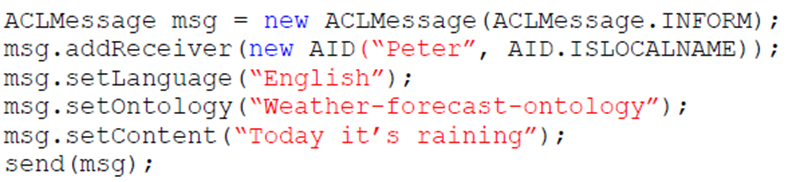
* 1. Sender dari message
  2. List *receiver*
  3. Maksud dari pengiriman (*performative*), mengapa sender mengirim message.



Gambar List dari Performative [4]

Sedangkan format yang akan diketikkan user sebagai berikut :

1. *Content* (Informasi yang akan dikirimkan)
2. *Content Language* (*syntax* yang digunakan untuk mengekspresikan content)
3. *Ontology* (perbendaharaan kata yang digunakan di content beserta maknanya)
4. Beberapa field untuk mengontrol beberapa pembicaraan yang terjadi bersamaan dan menentukan timeout untuk menerima pesan seperti *conversation-id, reply-with, reply-to, reply-by*[4]*.*



Gambar Contoh *Sending Message Code[4]*

1. **Sistem temu kembali informasi**

* **Information Retrieval**

Information retrieval merupakan teknik pencarian berupa material (dokumen) yang berada dalam sekumpulan dokumen yang sangat banyak, dapat berupa text, gambar. Fungsinya adalah untuk menemukan kembali informasi-informasi yang relevan terhadap kebutuhan pengguna dari suatu kumpulan informasi secara otomatis. Salah satu contoh aplikasi yang menggunakan Information Retrieval ini adalah search engine. Ciri khas information retrieval ini adalah hasil pencariannya ditampilkan secara ranking, beda dengan string matching biasa yang digunakan untuk mencari sebuah dokumen melalui proses pencocokan string[6].

Secara garis besar, proses yang terjadi dalam Information Retrieval :

1. Memberi indeks pada dokumen
2. *Parsing*

Mengambil *term* dari dokumen dengan cara memotong string input berdasarkan tiap kata yang menyusunnya.

1. *Stopword removing*

Menghilangkan stopword pada string input yang menyusun dokumen.

Kata tugas, kata hubung, kata bantu termasuk dalam stopword

1. *Stemming*

Proses untuk menggabung atau memecahkan suatu kata menjadi kata dasar.

1. Pembobotan (*Term Weighting*)

Term dapat berupa kata, frase atau unit hasil indexing dokumen. Karena setiap kata memiliki tingkat kepentingan yang berbeda dalam dokumen, maka untuk setiap kata tersebut diberikan sebuah indikator, yaitu *term weight*. Salah satu contoh metode term weighting adalah TF-IDF.

1. Memberikan peringkat dokumen berdasarkan ukuran kesamaan

Setelah bobot masing-masing dokumen diketahui, maka dilakukan proses pemeringkatan atau perankingan dokumen berdasarkan besarnya tingkat kerelevanan (kesesuaian) dokumen terhadap kata kunci, dimana semakin besar nilai bobot dokumen terhadap kata kunci maka semakin besar tingkat similaritas dokumen tersebut terhadap kata kunci yang dicari[6].

* **TF-IDF**

TF-IDF merupakan metode term weighting, yang melakukan perhitungan bobot term *t* dalam dokumen dengan mengalikan nilai *Term Frequency* dengan *Inverse Document Frequency[6]*.

1. Term Frequency (TF) adalah faktor yang menentukan bobot term pada suatu dokumen berdasarkan jumlah kemunculannya dalam dokumen tersebut. Nilai jumlah kemunculan suatu kata (term frequency) diperhitungkan dalam pemberian bobot terhadap suatu kata. Semakin besar jumlah kemunculan suatu term (TF tinggi) dalam dokumen, semakin besar pula bobotnya dalam dokumen atau akan memberikan nilai kesesuaian yang semakin besar.

TF dihitung berdasarkan jumlah kemunculan kata di dokumen. Contohnya jika muncul lima kali maka kata tersebut bernilai lima (*raw TF)*.

1. Inverse Document Frequency (IDF) adalah pengurangan dominansi term yang sering muncul di berbagai dokumen. Hal ini diperlukan karena term yang banyak muncul di berbagai dokumen, dapat dianggap sebagai term umum (*common term*) sehingga tidak penting nilainya. Sebaliknya faktor kejarangmunculan kata (*term scarcity*) dalam koleksi dokumen harus diperhatikan dalam pemberian bobot.

IDF dihitung dengan menggunakan formula

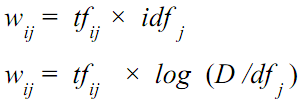


Keterangan :

*D* Jumlah semua dokumen dalam koleksi

*Df* j Jumlah dokumen yang mengandung term *t*j

Rumus umum dari TF-IDF adalah penggabungan formula dari raw TF dan IDF dengan cara mengalikan :



Keterangan :

*W* ij Bobot term tj terhadap dokumen di

*W* ij Jumlah kemunculan term *t*j dalam dokumen di

*D* Jumlah semua dokumen

*df* j Jumlah dokumen yang mengandung term *t*j (minimal ada satu kata yaitu term *t*j)

Berapapun besarnya nilai t*f*ij , apabila D = d*f*j maka akan didapatkan hasil nol untuk perhitungan idf. Untuk itu dapat ditambahkan nilai 1 pada sisi idf, sehingga perhitungan bobotnya menjadi sebagai berikut :



Contoh Sederhana Information Retrieval yang menggunakan metode TF-IDF :

Pengguna memberi kata kunci / *query* : gold silver truck

Sehingga didapat *query term* (*Q*) :

* + - gold
    - silver
    - truck

Koleksi dokumen :

Dokumen 1 (*d1*) = Shipment of gold damaged in a fire.

Dokumen 2 (*d2*) = Delivery of silver arrived in a silver truck.

Dokumen 3 (*d3*) = Shipment of gold arrived in a truck.

Total dokumen (*D*) : 3

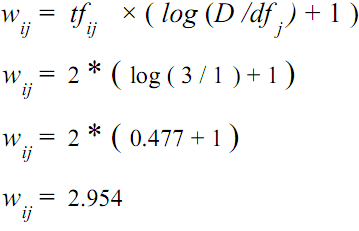
Dilakukan proses Information Retrieval :

1. Memberi indeks dokumen
2. Parsing untuk tiap query dan dokumen
3. Menghilangkan stopword, maka pada ketiga dokumen kata of, in, dan a dihapus
4. Stemming, sehingga didapat term-term sebagai berikut :

|  |  |
| --- | --- |
| * Ship * Damage * Deliver * Arrive | * Gold * Fire * Silver * Truck |

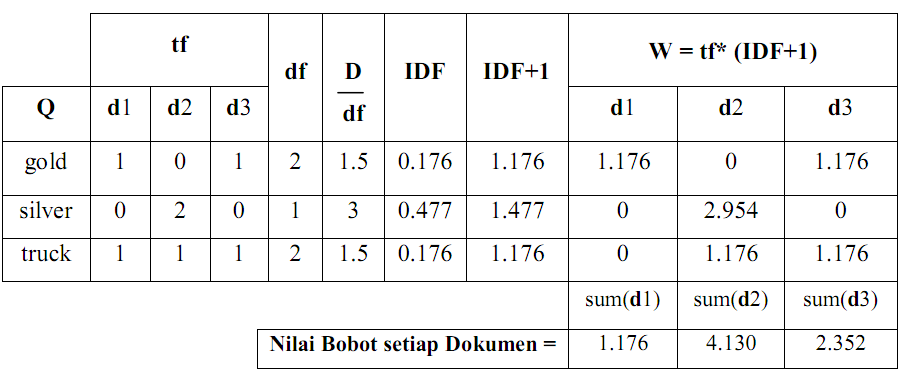
1. Pembobotan

Semakin besar nilai perhitungan bobot yang diperoleh maka semakin tinggi nilai smilaritas dokumen terhadap query. Contoh untuk perhitungan bobot (*w*) term query silver, dalam dokumen 2 (*d2*) = Delivery of silver arrived in a silver truck, yaitu : jumlah kemunculan term silver dalam dokumen 2 (*d2*) adalah sebanyak dua kali (*tf* = 2), total dokumen yang ada sebanyak tiga dokumen (*D*=3), dari ketiga dokumen dalam koleksi, term silver muncul pada dokumen 2 (*d2*), sehingga total dokumen yang mengandung term silver adalah satu dokumen (*df* = 1), sehingga dapat diperoleh nilai bobot term silver pada dokumen 2 (*d2*).



Dengan menggunakan cara yang sama seperti contoh query silver diatas, nilai bobot (*w*) untuk query lainnya bisa dicari dalam masing-masing dokumen :

1. Pemeringkatan (Perangkingan) Dokumen



Gambar 5. Perhitungan Pembobotan TF-IDF Term Query dalam Setiap Dokumen[6]

Dengan demikian dapat dihasilkan daftar dokumen teranking berdasarkan nilai kesesuaian (*similarity*) antara dokumen dan query masukan yang kemudian akan diberikan kepada pengguna. Dari hasil pembobotan dan perankingan dapat diketahui bahwa dokumen 2 (*d2*) memiliki tingkat relevansi tertinggi kemudian disusul dengan dokumen 3 (*d3*) lalu dokumen 1 (*d1*)[6].

1. METODOLOGI

Beberapa tahapan metodologi yang akan dilakukan dalam Tugas Akhir ini, di antaranya sebagai berikut:

1. Studi Literatur

Tahap ini merupakan tahap pengumpulan informasi yang diperlukan untuk pengerjaan tugas akhir, konsep-konsep yang berkaitan dengan rumusan masalah, termasuk di dalamnya mempelajari konsep agent beserta *framework* yang digunakan.

1. Perancangan Sistem

Tahap ini merupakan tahapan analisis dan desain perangkat lunak yang akan dikembangkan dengan mengacu pada proses bisnis dan data yang telah diperloleh. Tahap ini merupakan tahap yang paling penting pada bentuk awal atau prototipe akan diimplementasikan. Pada tahap ini dilakukain desain sistem dan desain proses-proses yang ada.

Pada tahapan ini dilakukan analisa awal dan pendefinisian kebutuhan sistem untuk mengetahui masalah yang sedang dihadapi. Dari proses tersebut selanjutnya dirumuskan rancangan sistem yang dapa memberi pemecahan masalah tersebut.

1. Implementasi

Pada tahap ini dilakukan implementasi rancangan sistem yang telah dibuat. Tahapan ini merealisasikan apa yang terdapat pada tahapan sebelumnya sehingga menjadi sebuah aplikasi yang sesuai dengan yang telah direncanakan.

1. Pengujian dan Evaluasi

Melakukan ujicoba dan evaluasi prototipe perangkat lunak untuk mencari masalah yang mungkin timbul, mengevaluasi jalannya program, dan mengadakan perbaikan jika terdapat kekurangan.

1. Penyusunan Buku Tugas Akhir

Tahap terakhir merupakan penyusunan laporan yang memuat dokumentasi mengenai pembuatan serta hasil dari ujicoba dari perangkat lunak yang telah dibuat. Secara garis besar, Buku Tugas Akhir yang nantinya akan dibuat terdiri dari beberapa bagian antara lain

1. Pendahuluan
   1. Latar Belakang
   2. Permasalahan
   3. Batasan Tugas Akhir
   4. Tujuan
   5. Metodologi
2. Tinjauan Pustaka
3. Desain dan Implementasi
4. Uji Coba dan Evaluasi
5. Kesimpulan dan Saran
6. Daftar Pustaka
7. JADWAL KEGIATAN

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| No. | Kegiatan | Bulan | | | | | | | | | | | | | | | | | |
| 1 | | | | 2 | | | | 3 | | | | | 4 | | | | |
| 1. | Studi Literatur |  |  |  |  |  |  |  |  |  |  |  |  |  | |  |  |  |
| 2. | Perancangan Sistem |  |  |  |  |  |  |  |  |  |  |  |  |  | |  |  |  |
| 3. | Implementasi & Pembuatan Sistem |  |  |  |  |  |  |  |  |  |  |  |  |  | |  |  |  |
| 4. | Uji Coba dan Evaluasi |  |  |  |  |  |  |  |  |  |  |  |  |  | |  |  |  |
| 5. | Penyusunan Buku Tugas Akhir |  |  |  |  |  |  |  |  |  |  |  |  |  | |  |  |  |

Keterangan : Bulan 1 dimulai pada Maret 2012

1. DAFTAR PUSTAKA
2. Bellifemine, Fabio., Caire, Giovanni., Trucco, Tiziana., Rimassa, Giovanni., JADE PROGRAMMER’S GUIDE, TILab S.p.A. 2005.
3. Jennings, N. R., 2000, On agent-base software engineering. Artificial Intelligence.
4. Wooldridge, M., Jennings, N. R., Intelligent agents: theory and practice. The Knowledge Engineering Review. 1995.
5. Caire, Giovanni., Jade Tutorial Jade Programming For Beginners, Telecom Italia S.p.A. 2009.
6. Tf-Idf::Information Retrieval and Text Mining. <http://www.tfidf.com/> (diakses tanggal 16 maret 2012)
7. Zafikri, A., Implementasi Metode Term Frequency Inverse Document Frequency (TF-IDF) Pada Sistem Temu Kembali Informasi. Skripsi Terpublikasi. Medan: Universitas Sumatera Utara. 2008.