

# SNHP3M

*by* Viny M

---

**Submission date:** 08-Nov-2020 03:29PM (UTC+0700)

**Submission ID:** 1439416860

**File name:** Paper\_SNHP3M\_Viny-Jeanny-Tony.pdf (283.27K)

**Word count:** 4216

**Character count:** 26779

# PERANGKAT AUTOMATIC TEXT SUMMARIZATION UNTUK GURU DAN SISWA SEKOLAH MENENGAH DI JAKARTA

Viny Christanti M., Jeanny Pragantha dan Tony<sup>1</sup>

## Abstrak:

Perancangan Perangkat Automatic Text Summarization ini dilakukan sebagai bagian dari rangkaian penelitian yang telah dilakukan selama 2 tahun oleh dosen dan mahasiswa. Penelitian dilakukan untuk mengetahui bagaimana membangun sebuah mesin peringkas otomatis untuk bahasa Indonesia. Setelah penelitian selesai dilakukan maka hasil dari penelitian tersebut diimplementasikan kepada masyarakat. Pada pengabdian ini dibuat aplikasi peringkas dokumen secara otomatis yang khusus diimplementasikan untuk guru, siswa SMU dan setingkatnya.

Aplikasi peringkas otomatis ini diberi nama SUM2, yang dapat digunakan untuk meringkas dokumen dalam bentuk teks secara otomatis. SUM2 ini dibuat untuk meringkas dokumen berbahasa Indonesia. SUM2 dibuat berdasarkan metode weight of feature dan menggunakan referensi sistem dari Stanford University. Dokumen yang dapat diringkas adalah dokumen umum yang berisi teks biasa.

Perancangan sistem peringkas dokumen Bahasa Indonesia menghasilkan hasil ringkasan dalam bentuk ekstraksi yaitu ringkasan yang terdiri dari unit-unit kalimat yang dianggap paling penting secara utuh tanpa mengubah susunan kata-kata yang ada di dalamnya. Proses peringkasan itu sendiri terdiri dari tiga tahap yaitu Preprocessing, Analyzing, dan Generating Summary. Aplikasi ini menghasilkan hasil ringkasan dalam bentuk ekstraksi dan telah diujicobakan terhadap dokumen pidato dan berita. Hasil penelitian sebelumnya telah menunjukkan bahwa aplikasi ini memiliki tingkat akurasi sebesar  $\pm 94\%$ .

## Kata Kunci:

Automatic Text Sumarization, Weight of Features, Named Entities Recognition

## Abstract:

This Automatic Text Summarization is the result of research that has been done in 2 years by lecturers and students. The study was conducted to determine how to build a machine to summarize Indonesian documents. Once the research completed, the results of these studies are implemented to the community. In this community service, Automatic Text Summarization implemented for teachers and high school students.

This automatic text summarization application is named SUM2, which can be used to summarize a document in text. SUM2 is made to summarize documents in Indonesian language. SUM2 made based method using the weight of features and reference systems from Stanford University. Documents that can be summarized are a general document that contains plain text.

The design of SUM2 produces summary results in extraction method, which is work by selecting a subset of existing words, phrases, or sentences in the original text to form the summary. These summarization processes consist of three stages: preprocessing, analyzing, and generating summary. We already tested this application on speeches and news documents. Results of this study have shown that this application have a level of accuracy of  $\pm 94\%$ .

## Keywords:

Automatic Text Sumarization, Weight of Features, Named Entities Recognition

<sup>1</sup> Fakultas Teknologi Informasi, Universitas Tarumanagara, viny@untar.ac.id.

## 1. Pendahuluan

Bahasa Indonesia merupakan bahasa pengantar sehari-hari di Indonesia. Pelajaran bahasa pun sudah dipelajari sejak usia dini. Bahasa dapat digunakan secara lisan maupun tertulis. Secara lisan pelajaran bahasa Indonesia dapat diperoleh mulai dari rumah hingga sekolah. Tentunya bahasa lisan dan tertulis dapat memiliki perbedaan dalam penggunaannya. Secara informal umumnya bahasa Indonesia dapat diperoleh dalam kehidupan sehari-hari. Secara lisan bahasa digunakan manusia untuk bercakap-cakap dan berkomunikasi. Secara formal bahasa dapat diperoleh pada pelajaran bahasa Indonesia di sekolah.

Penulisan bahasa pun bergantung dari penggunaannya. Tentunya bahasa formal memiliki aturan yang baku sehingga dapat digunakan untuk penulisan berita, artikel atau lainnya. Pada tingkat SD, pelajaran bahasa Indonesia yang baku sudah diajarkan. Pelajaran bahasa Indonesia terus dipelajari sampai tingkat sekolah menengah bahkan sampai di perguruan tinggi. Setiap siswa akan diajarkan bagaimana menyusun kalimat yang baik dan benar, bagaimana membuat karangan, bagaimana memahami sebuah artikel atau bagaimana membuat ringkasan.

Walaupun pelajaran mengenai tata bahasa Indonesia yang baik dan benar telah dipelajari sejak kecil, namun terkadang menangkap makna tulisan dapat berbeda. Makna kalimat yang sama dapat disusun dengan kalimat yang berbeda antara satu orang dengan yang lain. Hal ini tentunya dapat mengakibatkan subjektivitas dalam penilaian. Guru bahasa yang satu dengan yang lain dapat pula memberikan jawaban atau penilaian yang berbeda dalam membaca hasil pekerjaan siswanya. Masalah ini terkadang dapat menimbulkan kekeliruan dalam memberikan nilai.

Meringkas merupakan kegiatan merangkum atau menyarikan dokumen. Ringkasan dari suatu dokumen tetap mengandung intisari penting dari dokumen asli. Dengan adanya ringkasan, diharapkan pembaca dapat dengan cepat dan mudah memahami makna sebuah teks tanpa harus membaca keseluruhan teks. Pekerjaan meringkas biasanya dapat dilakukan dengan membaca secara keseluruhan, kemudian menyusun kerangka berpikir, mencari kalimat utama dan menyusun kembali kalimatnya.

Pelajaran meringkas sudah diajarkan ditingkat SD, sedangkan pada tingkat sekolah menengah pelajaran meringkas dilakukan kepada dokumen yang lebih formal dan jumlah yang banyak. Hasil ringkasan antar satu siswa dengan siswa lain tentunya tidak sama, bergantung dari pandangan setiap siswa. Walaupun demikian tetap ada panduan yang harus diikuti. Pada tingkat SMU pelajaran bahasa Indonesia diarahkan kepada bahasa yang lebih formal sesuai tata bahasa. SMU merupakan tingkatan sekolah yang sudah mengenal pelajaran bahasa Indonesia lebih matang. Sehingga implementasi aplikasi ringkasan ini dapat lebih mudah dan cepat untuk digunakan. Mereka sudah lebih banyak memiliki pengetahuan mengenai meringkas.

Saat ini perolehan informasi yang cepat dan tepat sangat dibutuhkan. Dokumen elektronik dalam jumlah besar dapat membuat manusia sulit memperoleh informasi dengan cepat. Cara memperoleh informasi dengan cepat adalah dengan mengetahui terlebih dahulu intisari dari sebuah artikel atau dokumen. Teknologi informasi dapat digunakan untuk memperoleh intisari dari sebuah dokumen. Namun pemanfaatan teknologi akan bergantung pada bahasa yang digunakan. Salah satu implementasi dalam masalah tersebut adalah dengan adanya mesin peringkas secara otomatis. Peringkas otomatis dapat membantu manusia dalam memperoleh intisari sebuah dokumen. Namun peringkas otomatis akan berbeda sesuai bahasa yang ada dalam dokumen tersebut.

Permasalahan tersebut dapat diatasi dengan membuat *Automatic Text Summarization (ATS)*. Secara sederhana, peringkas dokumen otomatis adalah sebuah proses untuk menghasilkan ringkasan dari suatu dokumen yang tetap mengandung intisari penting dari dokumen asli secara otomatis, yang berarti user hanya cukup memasukkan dokumen yang ingin diringkas, lalu sistem yang akan memproses dokumen tersebut dan menghasilkan ringkasan dokumen.

Terdapat dua pendekatan pada peringkas teks, yaitu ekstraksi dan abstraksi. Pada teknik ekstraksi, sistem menyalin informasi yang dianggap paling penting dari teks asli menjadi ringkasan.

1 Sedangkan teknik abstraksi melibatkan parafrase dari teks asli. Teknik abstraksi membuat ringkasan dengan menciptakan kalimat-kalimat baru yang merepresentasikan intisari teks sumber dalam bentuk berbeda dengan kalimat-kalimat pada teks sumber. Pada umumnya, abstraksi dapat meringkas teks lebih kuat daripada ekstraksi, tetapi dalam pembuatan sistem, akan lebih sulit dikembangkan karena mengaplikasikan teknologi *natural language generation* yang merupakan bahasan yang dikembangkan tersendiri.

Suatu ringkasan dapat bersifat *general*, yaitu ringkasan yang berupaya mengambil sebanyak mungkin informasi penting yang mampu menggambarkan keseluruhan isi teks. Selain itu dapat juga berupa informasi yang diambil untuk ringkasan berdasar pada query masukan yang didefinisikan pengguna sistem. *Query-oriented* atau *user-oriented summarization* mencoba mengambil informasi yang relevan dengan *query* pengguna dan menampilkannya dalam bentuk ringkasan.

Tujuan dari pembuatan aplikasi peringkas otomatis ini adalah mempercepat proses pembelajaran bagaimana cara meringkas dan membantu siswa dalam belajar meringkas. Aplikasi ini juga dapat bermanfaat bagi para guru untuk dijadikan salah satu alat rekomendasi untuk menilai sebuah ringkasan. Apabila sebuah ringkasan sudah terdiri dari kalimat utama tersebut maka ringkasan tersebut dapat diartikan sudah memiliki inti dari informasi sebuah dokumen tersebut. Tidak menutup kemungkinan ATS ini juga dapat digunakan oleh siswa untuk memudahkan dalam memperoleh informasi. Sehingga siswa dapat menggunakan secara pribadi untuk membaca buku atau referensi elektronik lainnya.

Aplikasi Peringkas Otomatis untuk dokumen bahasa Indonesia ini diberi nama Sum2. Aplikasi ini merupakan hasil penelitian yang dilakukan oleh dosen dan mahasiswa FTI Untar. Aplikasi ini dibuat dan dibagikan kepada para guru dan siswa sebagai implementasi hasil penelitian. Khalayak mitra yang menjadi sasaran adalah sekolah-sekolah di Jakarta Barat. Sekolah-sekolah di Jakarta Barat merupakan sekolah yang paling dekat dengan Universitas Tarumanagara. Berdasarkan permintaan mitra maka, pengabdian ini dilakukan dengan memberikan aplikasi dan mengadakan pelatihan di FTI Untar.

## 4 Materi dan Metode

### 2.1. Ringkasan

Ringkasan adalah penyajian bacaan dalam bentuk singkat dengan mempertahankan urutan isi dan sudut pandang atau memendekkan bacaan dengan mengamati inti sari bacaan itu tanpa mengubah struktur wacana (Irman, M, 2008). Ringkasan merupakan suatu cara yang efektif untuk menyajikan suatu karangan yang panjang dalam bentuk yang singkat. Manfaat dari membuat ringkasan adalah memudahkan pembaca memperoleh informasi yang penting. Selain itu ringkasan juga bermanfaat bagi pembaca agar lebih mudah mengingat atau tidak mudah lupa.

Dalam membuat ringkasan terdapat beberapa tahapan yang perlu diperhatikan. Langkah-langkah dalam membuat ringkasan adalah sebagai berikut:

1. Membaca naskah asli
2. Membaca ulang naskah jika perlu
3. Mencatat gagasan utama
4. Mengadakan reproduksi
5. Kalimat disusun menjadi suatu wacana yang jelas dan dapat diterima akal sehat, dan sekaligus menggambarkan kembali isi dari karangan aslinya

Lima tahapan tersebut dapat saja menghasilkan kalimat yang berbeda namun memiliki inti yang sama.

Terdapat dua jenis ringkasan yaitu abstraksi dan ekstraksi. Dalam membuat abstraksi, kelima langkah diatas perlu dilakukan untuk mendapatkan hasil ringkasan yang tersusun dengan kalimat yang baik dan benar. Sedangkan ekstraksi merupakan ringkasan yang hanya digunakan untuk mencari intisari, sehingga tahap kelima terkadang tidak dilakukan. Kelemahan dari ringkasan bentuk

ekstraksi adalah susunan kalimat tidak enak dibaca dan susunan kalimat yang dihasilkan tidak memiliki alur yang baik (Gupta, V., & Lehal, G. S., 2010).

## 2.2. Automated Text Summarization (ATS)

Secara sederhana, peringkasan dokumen otomatis adalah sebuah proses untuk menghasilkan ringkasan dari suatu dokumen yang tetap mengandung intisari penting dari dokumen asli secara otomatis (Fattah, M. A., & Ren, F., 2008). Manusia hanya cukup memasukkan dokumen yang ingin diringkaskan, lalu sistem yang akan memproses dokumen tersebut dan menghasilkan ringkasan dokumen.

Dengan adanya ringkasan, diharapkan pembaca dapat dengan cepat dan mudah memahami makna sebuah teks tanpa harus membaca keseluruhan teks. Hal ini dapat menghemat waktu pembaca karena dapat menghindari pembacaan teks yang tidak relevan dengan informasi yang diharapkan oleh pembaca, terutama ketika sangat banyak informasi yang tersedia. Tetapi, bila ringkasan dibuat secara manual tentunya akan membuang lebih banyak waktu lagi dibandingkan dengan ringkasan otomatis. Oleh karena itu, para peneliti information retrieval melakukan penelitian mengenai sistem peringkasan dokumen otomatis yang disebut juga sebagai *Automated Text Summarization* (ATS). Para peneliti berharap hasil penelitiannya dapat menyamai atau bahkan mengungguli ringkasan dokumen manual yang dibuat oleh manusia.

Terdapat dua pendekatan pada peringkasan teks, yaitu ekstraksi dan abstraksi (Basagic, R., Krupic, D., & Suzic, B., 2009). Pada teknik ekstraksi, sistem menyalin informasi yang dianggap paling penting dari teks asli menjadi ringkasan. Sedangkan teknik abstraksi melibatkan parafrase dari teks asli. Teknik abstraksi membuat ringkasan dengan menciptakan kalimat-kalimat baru yang merepresentasikan intisari teks sumber dalam bentuk berbeda dengan kalimat-kalimat pada teks sumber. Pada umumnya, abstraksi dapat meringkas teks lebih kuat daripada ekstraksi, tetapi dalam pembuatan sistem, akan lebih sulit dikembangkan karena mengaplikasikan teknologi *natural language generation*.

Pada umumnya proses peringkasan dokumen terdiri dari 3 tahap yaitu *pre-processing*, *analysis* dan *generating summary* (Bawakid, A., & Oussalah, M., 2008). Tahap *pre-processing* atau pemrosesan awal adalah tahap untuk mempersiapkan dokumen sehingga dapat diproses lebih lanjut. Tahap analisis kalimat adalah tahap untuk menghitung bobot dari masing-masing kalimat. Metode yang digunakan pada tahapan ini adalah metode *Weight of Features*. Sedangkan tahap terakhir adalah *Generating Summary*, yaitu mengurutkan kalimat berdasarkan bobot dari setiap kalimat yang telah diperoleh (*Ordering Sentences*). Tahapan meringkas dokumen secara otomatis ini, disusun berdasarkan tata cara meringkas secara manual oleh manusia.

14

## 2.3. Named Entity Recognition (NER)

*Named Entity Recognition* (NER) atau pengenalan nama entitas merupakan sebuah proses dalam *information extraction* yang berfungsi untuk mengklasifikasi entitas ke dalam kategori yang telah ditentukan dalam teks. NER dapat dianggap sebagai proses klasifikasi kata ke dalam kategori yang sesuai. Pada umumnya NER fokus dalam mengklasifikasi kategori seperti nama orang, lokasi serta organisasi.

NER dapat dianggap sebagai proses klasifikasi sehingga metode yang digunakan untuk klasifikasi juga berlaku untuk pengenalan nama entitas. Contoh metode yang dapat digunakan untuk melakukan pengenalan adalah *Naive Bayes*, *Hidden Markov Model* dan *Conditional Random Field*. Beberapa program NER terkemuka yang telah dibuat adalah *NER Stanford*, *Lingpipe*, *GATE* dan lain-lain. Pada aplikasi ini digunakan NER buatan *Stanford* untuk melakukan proses pengenalan.

*Stanford NER* merupakan sebuah sistem dengan metode *Conditional Random Field* (CRF) yang digunakan untuk memberikan entitas pada token seperti nama orang, nama organisasi, atau nama tempat. *Stanford NER* dikembangkan oleh *The Stanfords Natural Language Processing Group*

dari Universitas Stanford. Stanford NER mulai dikembangkan oleh NLP group pada tahun 2003 (The Stanford NLP Group, 2010).

#### 2.4. Weight of Features

Proses mencari kalimat inti dari sebuah dokumen dilakukan dengan menghitung bobot dari masing-masing kalimat dalam dokumen. Proses perhitungan bobot kalimat dilakukan dengan menggunakan kombinasi linear dari setiap bobot fitur (*Weight of Features*) yang telah dihitung (Bawakid, A., & Oussalah, M., 2008). Perhitungan bobot dari setiap kalimat dilakukan untuk mengetahui seberapa pentingnya suatu kalimat. Semakin besar nilai bobot dari sebuah kalimat, maka semakin banyak informasi penting yang terkandung dalam kalimat tersebut. Sebaliknya semakin kecil nilai bobot dari suatu kalimat, semakin sedikit informasi penting yang terkandung dalam kalimat tersebut.

Perhitungan yang digunakan untuk memperoleh hasil ringkasan dengan metode Weight of Features (Basagic, R., Krupic, D., & Suzic, B., 2009):

a. Banyaknya Kalimat

Banyaknya kalimat yang terdapat dalam sebuah dokumen menjadi salah satu komponen yang diperlukan dalam melakukan penghitungan bobot untuk setiap kalimat dengan metode *Weight of Features*.

$$N = \text{banyaknya kalimat} \dots (1)$$

b. Bobot Posisi Kalimat

Pada proses ini setiap kalimat akan diberikan bobot posisi kalimat berdasarkan posisi sebuah kalimat dalam dokumen. Bobot posisi kalimat dihitung sesuai rumus berikut:

$$P(S_i) = \text{bobot posisi kalimat} \dots (2)$$

Misalkan dalam sebuah dokumen terdapat  $i$  kalimat. Maka bobot posisi kalimat adalah sebagai berikut :

Bobot posisi kalimat pertama :  $P(s_1) = \frac{1}{i}$   
Bobot posisi kalimat kedua :  $P(s_2) = \frac{(i-1)}{i}$   
Bobot posisi kalimat ketiga :  $P(s_3) = \frac{(i-2)}{i}$   
Bobot posisi kalimat keempat :  $P(s_4) = \frac{(i-3)}{i}$   
Bobot posisi kalimat kelima :  $P(s_5) = \frac{(i-4)}{i}$   
dan seterusnya.

c. Banyaknya entitas dalam dokumen

Banyaknya entitas dalam sebuah dokumen diperoleh dari proses *Named Entities Recognition*. Dalam melakukan pemberian entitas sistem akan melihat nama orang, nama organisasi, nama tempat dan waktu (hari, tanggal, bulan dan tahun). Proses ini dilakukan dengan menghitung ada berapa banyak entitas dalam satu buah dokumen.

$$NE = \text{banyaknya entitas dalam sebuah dokumen} \dots (3)$$

d. Banyaknya entitas dalam sebuah kalimat.

Banyaknya entitas dalam sebuah dokumen diperoleh dari proses *Named Entities Recognition*. Dalam melakukan pemberian entitas sistem akan melihat nama orang, nama organisasi, nama tempat dan waktu (hari, tanggal, bulan dan tahun). Proses ini dilakukan dengan menghitung ada berapa banyak entitas dalam satu buah kalimat.

$$F_{NE}(S_i) = \text{banyaknya entitas dalam sebuah dokumen} \dots (4)$$

Sebuah kalimat yang mengandung lebih banyak entitas akan memiliki bobot kalimat yang lebih besar jika dibandingkan dengan kalimat yang tidak mengandung entitas.

Setelah semua parameter diperoleh maka bobot kalimat dapat dihitung dengan rumus *Weight of Features* sebagai berikut (Basagic, R., Krupic, D., & Suzic, B., 2009):

$$Score(i) = \frac{(F_{NE}(s_i) + 1) P(s_i)}{N (NE + 1)} \quad (5)$$

Keterangan :

Score(i) = bobot untuk kalimat ke i  
 $F_{NE}(s_i)$  = banyaknya entitas dalam kalimat ke i  
 $P(s_i)$  = bobot posisi kalimat untuk kalimat ke i  
 $N$  = banyaknya kalimat dalam sebuah dokumen  
 $NE$  = banyaknya entitas dalam sebuah dokumen

## 2.5. Final Filtering

*Final Filtering* merupakan sebuah proses pengembalian hasil ringkasan dari sebuah dokumen awal yang *user* masukkan ke dalam sistem (Basu, R., Krupic, D., & Suzic, B., 2009). Semua kalimat yang sudah dihitung bobot kalimatnya akan diurutkan dari yang terbesar hingga yang terkecil. Sistem akan mengembalikan hasil ringkasan sesuai dengan *compression rate* yang dipilih oleh *user*.

*Compression rate* (nilai kompresi) pada proses peringkasan akan menentukan panjang ringkasan yang dihasilkan. Biasanya diukur berdasarkan persentase dari teks sumber, misalnya ringkasan sepanjang 25% atau 50% dari teks sumber. Selain itu dapat pula diukur berdasarkan jumlah kata, misalnya ditentukan ringkasan sepanjang 10 kata. Biasanya, panjang ringkasan tidak lebih dari setengah teks sumber.

Pada aplikasi ini digunakan nilai kompresi sebesar 75%, 50% dan 25% dari teks sumber. Artinya jika *user* memilih nilai kompresi sebesar 25% pada dokumen yang terdiri dari 12 kalimat, maka sistem akan memilih 9 kalimat yang mempunyai bobot tertinggi yang akan dikembalikan kepada *user* sebagai hasil ringkasan.

Dalam menghasilkan ringkasan yang memenuhi alur pembacaan dokumen, maka sistem harus mengembalikan ringkasan sesuai dengan urutan kalimat pada dokumen aslinya. Dengan demikian setelah didapat kalimat-kalimat dengan bobot tertinggi maka sistem harus mengurutkan kalimat-kalimat tersebut berdasarkan nomor kalimat pada dokumen asli.

23

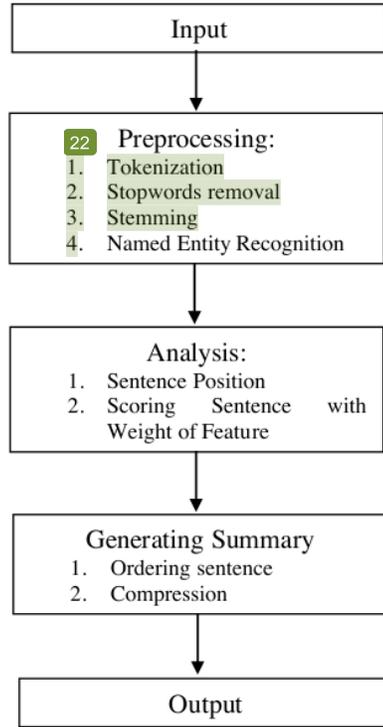
## 3. Hasil dan Pembahasan

### 3.1. Alur sistem peringkasan

Program aplikasi yang dirancang adalah program peringkasan dokumen untuk Bahasa Indonesia dengan menggunakan metode *Weight of Features*. Sistem diawali dengan memberikan input berupa file teks. Setelah itu sistem melakukan tahap *preprocessing* yang berfungsi untuk memberikan entitas pada kata dan menghitung jumlah entitas yang terdapat dalam setiap kalimat. Setelah itu dilakukan analisis data yang mencakup penghitungan bobot posisi kalimat (*sentences position weight*) dan penghitungan bobot dari setiap kalimat. Setelah diperoleh bobot dari setiap kalimat, dilakukan proses *generating summary* dengan mengurutkan kalimat berdasarkan bobot dari setiap kalimat dan dilakukan kompresi.

Gambar 1 adalah alur proses dari perancangan ini. Alur proses pembuatan peringkasan dengan metode *Weight of Features* adalah sebagai berikut:

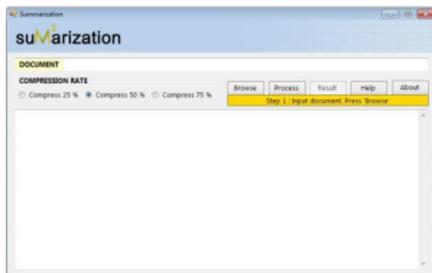
- Input dokumen.
- Proses *Named Entities Recognition* pada dokumen yang telah di input dengan bantuan program Stanford NER.
- Menghitung jumlah entitas yang terdapat pada setiap kalimat.
- Menghitung jumlah entitas yang terdapat pada sebuah dokumen.
- Menghitung jumlah kalimat yang terdapat pada dokumen.
- Memberikan bobot posisi kalimat untuk setiap kalimat.
- Menghitung bobot setiap kalimat dengan metode *Weight of Features*.
- Mengurutkan kalimat mulai dari bobot kalimat terbesar hingga bobot terkecil
- Melakukan kompresi dokumen sesuai dengan nilai kompresi yang dipilih oleh *user*.



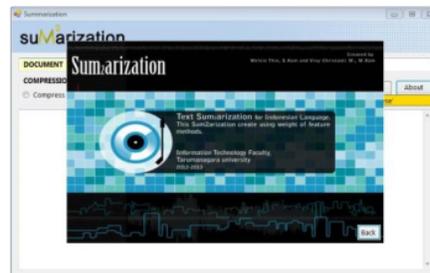
Gambar 1. Alur sistem yang dirancang

### 3.2. Hasil Program Aplikasi

Program aplikasi ini dibuat dengan menggunakan bahasa program Visual basic .net dan Java. Hasil aplikasi dapat dilihat pada Gambar 2 dan 3. Pada aplikasi disediakan manual program untuk memudahkan guru dan siswa dalam menggunakan aplikasi tersebut.



Gambar 2. Tampilan awal aplikasi SUM2



Gambar 3. Tampilan pembuat SUM2

Pada tabel 1, dapat dilihat bentuk dari dokumen asli yang digunakan dan dokumen hasil ringkasan dengan nilai kompresi sebesar 25%. Hasil ringkasan ini sudah diuji dengan metode Question & Answer (Q&A) Evaluation untuk mengetahui banyaknya informasi yang dikandung oleh dokumen hasil ringkasan. Caranya adalah dengan membuat beberapa pertanyaan mengenai informasi yang dikandung dalam dokumen asli. Kemudian responden diberikan hasil ringkasan dengan nilai kompresi 75%. Berdasarkan dokumen dengan nilai kompresi 75%, responden menjawab pertanyaan

lalu dihitung banyaknya jawaban yang benar. Hal yang sama dilakukan untuk dokumen hasil ringkasan dengan nilai kompresi 50% dan 25%.

Dokumen asli	Hasil ringkasan dengan kompresi 25%
<p>Salah satu makhluk hidup yang dianggap sebagai hewan paling cerdas, ternyata dapat menderita masalah psikologi terkait dengan ditangkapnya dan dikurungnya mereka di dalam kolam. Penemuan ini ditemukan Lori Marino, seorang ahli syaraf di Universitas Emory, Atlanta, Amerika Serikat.</p> <p>Menurut Lori Marino, lumba-lumba bisa trauma jika mereka terus disimpan atau hanya berenang dengan wisatawan.</p> <p>"Kompleksitas dan kecerdasan otak yang dimiliki lumba-lumba menunjukkan bahwa praktek berpotensi merusak psikologis lumba-lumba dan menyajikan sebuah gambaran salah tentang informasi kapasitas intelektual alami mereka," kata Marino.</p> <p>Lumba-lumba yang canggih, sadar diri, makhluk sangat cerdas dengan kepribadian individu, otonomi dan kehidupan batin. Mereka sangat rentan terhadap penderitaan dan trauma psikologis. Selama ini dikatakan bahwa interaksi lumba-lumba dengan manusia bisa memiliki kualitas penyembuhan, seperti orang cacat memiliki manfaat kesehatan hanya dengan berenang bersama lumba-lumba, seperti dilansir Telegraph, Kamis (18/2).</p> <p>Mark Simmons, direktur The Whale and Dolphin Conservation Society berkata, "Saya pikir kami harus sangat berhati-hati, bukan hanya berenang dengan lumba-lumba, tetapi juga memperhatikan mereka dari perahu-perahu juga." Menurut Simmons, manusia perlu berpikir ulang tentang bagaimana interaksi dengan lumba-lumba di alam liar.</p> <p>"Sayangnya salah satu indikator kecerdasan adalah bahwa mereka dapat menderita dengan cara yang sama seperti yang kita derita. Aku cukup yakin mereka menderita dalam cara-cara yang sangat mirip dengan kita," kata Simmons. Dr Marino menambahkan, "Banyak otak lumba-lumba modern secara signifikan lebih besar daripada kita sendiri." Ciri-ciri anatomi otak mereka menunjukkan sebuah "kompleks" kecerdasan</p>	<p>Salah satu makhluk hidup yang dianggap sebagai hewan paling cerdas, ternyata dapat menderita masalah psikologi terkait dengan ditangkapnya dan dikurungnya mereka di dalam kolam. Penemuan ini ditemukan Lori Marino, seorang ahli syaraf di Universitas Emory, Atlanta, Amerika Serikat.</p> <p>Menurut Lori Marino, lumba-lumba bisa trauma jika mereka terus disimpan atau hanya berenang dengan wisatawan.</p> <p>"Kompleksitas dan kecerdasan otak yang dimiliki lumba-lumba menunjukkan bahwa praktek berpotensi merusak psikologis lumba-lumba dan menyajikan sebuah gambaran salah tentang informasi kapasitas intelektual alami mereka," kata Marino.</p> <p>Lumba-lumba yang canggih, sadar diri, makhluk sangat cerdas dengan kepribadian individu, otonomi dan kehidupan batin. Mereka sangat rentan terhadap penderitaan dan trauma psikologis. Selama ini dikatakan bahwa interaksi lumba-lumba dengan manusia bisa memiliki kualitas penyembuhan, seperti orang cacat memiliki manfaat kesehatan hanya dengan berenang bersama lumba-lumba, seperti dilansir Telegraph, Kamis (18/2).</p> <p>Mark Simmons, direktur The Whale and Dolphin Conservation Society berkata, "Saya pikir kami harus sangat berhati-hati, bukan hanya berenang dengan lumba-lumba, tetapi juga memperhatikan mereka dari perahu-perahu juga." Menurut Simmons, manusia perlu berpikir ulang tentang bagaimana interaksi dengan lumba-lumba di alam liar.</p> <p>"Sayangnya salah satu indikator kecerdasan adalah bahwa mereka dapat menderita dengan cara yang sama seperti yang kita derita.</p>

### 3.3. Pembahasan hasil program aplikasi

Aplikasi SUM2 ini sudah dapat menghasilkan ringkasan sesuai dengan nilai kompresi yang dipilih oleh *user*. Apabila dokumen asli terdiri dari 13 kalimat maka dokumen output dengan nilai kompresi 75% akan terdiri dari 4 kalimat, dokumen output dengan nilai kompresi 50% terdiri dari 7 kalimat dan dokumen output dengan nilai kompresi 25% terdiri dari 10 kalimat.

Penyebab umum terjadinya kurang tepatnya hasil ringkasan yang diperoleh adalah tidak ditemukannya kata penting dalam kalimat sehingga kalimat tidak dikenali sebagai kalimat penting. Penyebab lainnya adalah entitas yang diberikan oleh program kurang lengkap. Sebagai contoh, kata "lumba-lumba" seharusnya dapat dikenali sebagai kata yang berisi informasi nama binatang. Namun

dikarenakan pada tahap pelatihan, aplikasi belum mendapatkan informasi entitas nama binatang tersebut, maka program tidak dapat mengenalinya.

### 3.4. Pembahasan Pelaksanaan pelatihan

20

Setelah hasil program aplikasi selesai dibuat dan telah diuji coba maka tahap selanjutnya adalah memberikan program tersebut kepada para guru dan siswa. Selain memberikan program secara langsung kepada guru dan siswa, tim juga memberikan pelatihan secara gratis bagaimana program aplikasi ini dapat digunakan. Pelatihan diadakan pada tanggal 17-18 Desember 2013 di FTI Untar. Pelatihan dilaksanakan secara paralel yang terdiri dari 3 ruangan. 2 ruangan untuk siswa dan 1 ruangan untuk guru. Guru dan siswa diundang untuk mengikuti acara tersebut. Pelatihan ini diadakan bersamaan dengan acara seminar Open Source Software yang diadakan pada tanggal 17-18 Desember 2013. Acara dilangsungkan bukan disekolah mitra namun dilaksanakan di FTI Untar sehingga memudahkan para peserta dalam pengadaan fasilitas komputer.

Pelatihan diberikan oleh 3 orang dosen dan 3 orang asisten mahasiswa. Jeanny P., Tony, Viny Christanti M. Robertus, Fx. Adrian dan Alfin. Pelatihan aplikasi ini berhasil diikuti oleh 18 sekolah yang terdiri dari 31 siswa dan 20 guru. Nama sekolah yang mengikuti acara ini dapat dilihat pada tabel 2.

Tabel 2. Sekolah yang diberikan dan mengikuti acara pelatihan aplikasi peringkask

No.	Nama Sekolah	No.	Nama Sekolah
1	SMAN 56	10	SMA Sang Timur
2	SMAN 61	11	SMA Santo Andreas
3	SMAN 42	12	SMA AA
4	SMAN 4	13	SMAN 31
5	SMAN 25	14	SMA Pelita Kudus
6	SMAN 77	15	SMA Pelita II
7	SMAN 5	16	SMAN 13
8	SMAN 56	17	SMAN 68
9	SMAN Tri Ratna	18	SMAN 110

Gambar 4 adalah foto hasil pelaksanaan pelatihan Aplikasi Peringkask ini. Pada setiap ruang, para peserta memiliki antusiasme yang berbeda-beda. Peserta guru lebih memberikan antusias terhadap pelatihan ini. Mereka merasakan sekali bahwa kebutuhan akan aplikasi ini sangat berguna baik bagi guru. Para guru banyak menanyakan bagaimana proses peringkask dari aplikasi ini. Selain menanyakan prosesnya, para guru pun memberikan masukan yang sangat banyak atas keterbatasan aplikasi ini. Sedangkan para siswa hanya mengikuti pelatihan tersebut dengan tenang. Para siswa lebih melihat kepada kemajuan teknologi bahwa meringkask dapat dilakukan secara otomatis.



Gambar 4. Suasana acara pelatihan

#### 4. Simpulan atau Implikasi

Sistem yang dirancang dapat menghasilkan dokumen ringkasan dengan tingkat kompresi 25%, 50% dan 75%. Pengujian yang dilakukan terhadap hasil ringkasan dengan metode *Question & Answer Evaluation* menunjukkan bahwa banyaknya informasi yang dikandung dalam dokumen hasil ringkasan dipengaruhi oleh nilai kompresi. Secara rata-rata ringkasan dengan nilai kompresi 25% mengandung 94% informasi, ringkasan dengan nilai kompresi 50% mengandung 77.95% informasi dan ringkasan dengan nilai kompresi 75% mengandung 56.04% informasi dari dokumen asli.

Pelatihan mendapat sambutan yang cukup baik. Guru-guru berpendapat bahwa aplikasi ini sangat membantu mereka dalam mendapatkan referensi mengenai hasil ringkasan yang baik. Namun beberapa kelemahan masih dirasakan dari aplikasi ini. Seperti ketidakmampuan program untuk meringkas banyak dokumen, aplikasi belum dapat meringkas dokumen dalam bentuk word dan aplikasi belum dapat merangkum dokumen yang mengandung rumus.

Saran selanjutnya adalah program ini terus dikembangkan agar dapat mendatangkan manfaat bagi seluruh masyarakat terutama didunia pendidikan. Adanya aplikasi ringkasan ini diharapkan dapat membantu siswa untuk dapat memperoleh informasi dari banyak sumber dengan cepat. Para siswa dan guru juga berharap agar program aplikasi ini dapat dibuat dengan bentuk yang lebih memudahkan *user* dalam menggunakannya. Misalkan dapat langsung membaca dari buku dan hasilnya dapat disimpan.

#### 6 Daftar Pustaka

- Basagic, R., Krupic, D., & Suzic, B. (2009). Automatic text summarization. *Graz University of Technology*.
- Bawakid, A., & Oussalah, M. (2008). A semantic summarization system: University of Birmingham at TAC 2008. In *Proceedings of the First Text Analysis Conference*.
- Budhi, G. S., Intan, R., Silvia, R., & Stevanus, R. R. (2007). Indonesian Automated Text Summarization. *Proceeding ICSIT*, 26-27.
- Fattah, M. A., & Ren, F. (2008). Automatic text summarization. *World Academy of Science, Engineering and Technology*, 37, 2008.
- Gupta, V., & Lehal, G. S. (2010). A survey of text summarization extractive techniques. *Journal of Emerging Technologies in Web Intelligence*, 2(3), 258-268.
- Irman, M., Prastowo, Tri Wahyu dan Nurdin. (2008). Bahasa Indonesia 2: untuk SMA/MAK Semua Program Keahlian Kelas XI. Jakarta: Pusat Perbukuan, Departemen Pendidikan dan Kebudayaan.
- The Stanford Natural Language Processing Group (2010), *Stanford Named Entity Recognizer*, diakses dari <http://nlp.stanford.edu/software/CRF-NER.shtml>.

# SNHP3M

---

## ORIGINALITY REPORT

---

**18%**

SIMILARITY INDEX

**17%**

INTERNET SOURCES

**8%**

PUBLICATIONS

**5%**

STUDENT PAPERS

---

## PRIMARY SOURCES

---

<b>1</b>	<b>sukmanapradibtha.wordpress.com</b> Internet Source	<b>5%</b>
<b>2</b>	<b>dmcittelkom.files.wordpress.com</b> Internet Source	<b>2%</b>
<b>3</b>	<b>docplayer.info</b> Internet Source	<b>2%</b>
<b>4</b>	<b>journal.ugm.ac.id</b> Internet Source	<b>1%</b>
<b>5</b>	<b>www.scribd.com</b> Internet Source	<b>1%</b>
<b>6</b>	<b>www.inderscienceonline.com</b> Internet Source	<b>1%</b>
<b>7</b>	<b>ethesis.nitrkl.ac.in</b> Internet Source	<b>&lt;1%</b>
<b>8</b>	<b>manualzz.com</b> Internet Source	<b>&lt;1%</b>
<b>9</b>	<b>thesis.binus.ac.id</b> Internet Source	<b>&lt;1%</b>

---

10	<a href="http://vdocuments.site">vdocuments.site</a> Internet Source	<1%
11	Submitted to Universitas Muria Kudus Student Paper	<1%
12	<a href="http://www.neliti.com">www.neliti.com</a> Internet Source	<1%
13	<a href="http://www.fundepe.com">www.fundepe.com</a> Internet Source	<1%
14	<a href="http://mafiadoc.com">mafiadoc.com</a> Internet Source	<1%
15	<a href="http://senopatihansyah.blogspot.com">senopatihansyah.blogspot.com</a> Internet Source	<1%
16	Ilmi Usrotin Choiriyah. "GAYA KEPEMIMPINAN DAN KINERJA PEGAWAI (Studi Pada Dinas Sosial Kabupaten Malang)", JKMP (Jurnal Kebijakan dan Manajemen Publik), 2015 Publication	<1%
17	<a href="http://text-id.123dok.com">text-id.123dok.com</a> Internet Source	<1%
18	<a href="http://mjcoolz21.blogspot.com">mjcoolz21.blogspot.com</a> Internet Source	<1%
19	<a href="http://fe.ugm.ac.id">fe.ugm.ac.id</a> Internet Source	<1%
20	<a href="http://suryapambudi04.blogspot.com">suryapambudi04.blogspot.com</a>	

Internet Source

<1%

21

[jurnal.batan.go.id](http://jurnal.batan.go.id)

Internet Source

<1%

22

Francesco Camastra, Angelo Ciaramella, Antonio Maratea, Le Hoang Son, Antonino Staiano. "Chapter 3 Semantic Maps for Knowledge Management of Web and Social Information", Springer Science and Business Media LLC, 2020

Publication

<1%

23

[pt.scribd.com](http://pt.scribd.com)

Internet Source

<1%

24

[tintafajar.wordpress.com](http://tintafajar.wordpress.com)

Internet Source

<1%

25

[elib.tiho-hannover.de](http://elib.tiho-hannover.de)

Internet Source

<1%

26

[urbanoustics.blogspot.com](http://urbanoustics.blogspot.com)

Internet Source

<1%

Exclude quotes On

Exclude matches Off

Exclude bibliography On