

APLIKASI AUTOMATIC TEXT SUMMARIZER

Silvia, Pitri Rukmana, Vivi Regina Aprilia
Computer Science Department, Bina Nusantara
University
email: silvia.phang1@gmail.com,
pitrirukmana@gmail.com,
vieviemoochill@yahoo.com

ABSTRACT

The background of this research is information overload which is an effect of ease of information manipulation, storage and distribution. Massive amount of text documents available causes a decline in effectivity and efficiency of an individual when using information. Automatic Text Summarization can solve information overload by producing text document summaries. Purpose of this research is to create an Automatic Text Summarization algorithm and its implementation to create summaries of important information from text documents faster and can satisfy users' needs of relevant and consistent summaries. The algorithm is based on sentence features scoring and Genetic Algorithm for determining sentence feature weights. Implementation consists of training phase (read text input, pre-summarization, summarization, and Genetic Algorithm to produce learned sentence feature weights) and testing phase (read text input, pre-summarization, summarization, and saving summary). The algorithm is evaluated by calculating summarization speed, precision, recall, F-measure, and subjective evaluation. The results are Automatic Text Summarization algorithm which is able to text documents by extracting important sentences which represent contents of original text documents. Conclusions of this research are Automatic Text Summarization algorithm can create extractive summaries which represent important information from a single document in Indonesian with faster summarization speed compared to manual process.

Keywords: Automatic Text Summarization, Sentence Features, Genetic Algorithm, Latent Dirichlet Allocation

1. PENDAHULUAN

Information overload menjadi salah satu dampak kemudahan proses manipulasi, penyimpanan dan distribusi informasi. Bawden dan Robinson (2009:3-4) mendefinisikan information overload sebagai istilah untuk merepresentasikan keadaan efisiensi individu saat menggunakan informasi dalam aktivitasnya menjadi terhambat karena jumlah informasi relevan yang tersedia.

Jumlah dokumen teks elektronik yang disimpan di seluruh dunia tidak dapat dihitung banyaknya. Perkembangan internet berperan dalam penyebaran artikel dan dokumen teks. Dalam survei website Netcraft pada bulan Agustus 2013, mendapat respon dari 716.822.317 situs. Angka ini meningkat sebanyak 17.998.808 situs sejak Juli 2013. Berdasarkan kecenderungan 6 (enam) bulan terakhir, Netcraft memperkirakan akan ada 1 (satu) miliar situs dalam 18 bulan ke depan. Selain itu, jumlah tersebut masih harus ditambah dengan dokumen teks elektronik yang tidak terdapat di internet. Dengan banyaknya dokumen teks yang tersedia, mengakibatkan terlalu banyak informasi yang dituntut untuk dapat diakses dengan cepat dan tidak membutuhkan banyak waktu dalam pengambilan intisari yang dibutuhkan untuk mengambil keputusan berdasarkan informasi yang tersedia. Automatic Text Summarization adalah salah satu solusi untuk membantu pencarian inti dari dokumen atau artikel berupa deskripsi singkat (summary).

Menurut Jurafsky dan Martin (2006:24), Automatic Text Summarization adalah proses penyaringan (process of distilling) informasi yang paling penting dari suatu teks dokumen untuk menghasilkan sebuah versi singkat dari suatu tugas tertentu. Penelitian terhadap penerapan proses Automatic Text Summarization terus berkembang hingga saat ini terutama extractive summarization. Dalam extractive summarization tidak dilakukan perubahan struktur kalimat asli.

Awal penelitian ditandai dengan pembuatan metode term frequency oleh Luhn tahun 1958 dan Baxendale pada tahun yang sama, disusul oleh penelitian Edmunson di tahun 1969 (Jezek dan Steinberger, 2008:2). Setelah itu bermunculan berbagai metode Automatic Text Summarization, di antaranya yaitu metode: TF-IDF (Gupta dan Lehal 2010:261); Rhetorical Structure Theory

(Suneetha, 2011:291-292); cluster-based (Gholamrezazadeh, Salehi dan Gholamzadeh, 2009:4); machine learning (Gupta dan Lehal, 2010:262); graph (Kumar dan Salim, 2012:136-137); Latent Semantic Analysis (Gong dan Liu, 2001:23); sentence features dan pembobotan Genetic Algorithm (Suanmali, Salim, dan Binwahlan, 2011:1-8); dan lexical chains dan Genetic Algorithm (Berker dan Gungor, 2012:595-599). Sedangkan untuk peringkasan teks berbahasa Indonesia, graph dan algoritma exhaustive (Budhi, Intan, Silvia, dan Stevanus, 2007:1-6); cluster-based (SIDoBI buatan Prasetyo, Uliniansyah, dan Riandi pada tahun 2008); dan Latent Semantic Analysis (Aristoteles, Herdiyeni, Ridha dan Adisantoso, 2012:1-6). Pemakaian Latent Semantic Analysis dalam bidang NLP saat ini mulai digantikan oleh Latent Dirichlet Allocation.

Permasalahan yang ditemukan yaitu bagaimana algoritma dapat digunakan oleh banyak pihak untuk membuat ringkasan informasi penting dengan lebih cepat, relevan dan konsisten dengan dokumen teks aslinya.

Pembuatan algoritma ini diharapkan dapat membantu pembuatan ringkasan informasi penting dari suatu dokumen teks dengan lebih cepat dan dapat memenuhi kebutuhan pengguna akan ringkasan yang relevan dan konsisten dengan dokumen teks aslinya melalui ekstraksi kalimat-kalimat penting yang merepresentasikan isi dokumen teks. Implementasi algoritma berperan sebagai sarana pembuatan ringkasan informasi penting dari dokumen teks tunggal (single document) berbahasa Indonesia yang dapat diakses oleh banyak pihak dan juga sebagai bahan untuk pengembangan penelitian Automatic Text Summarization berbahasa Indonesia di masa yang akan datang.

Berdasarkan analisis penelitian terdahulu, maka akan dibuat algoritma extractive Automatic Text Summarization untuk single document berbahasa Indonesia menggunakan sentence features dengan Latent Dirichlet Allocation dan Genetic Algorithm. Algoritma ini berdasarkan pada sentence features algoritma Suanmali, Salim, dan Binwahlan yang dipublikasikan pada tahun 2011 dan Genetic Algorithm pada penelitian Berker dan Gungor pada tahun 2012 yang ditambahkan dengan Latent Dirichlet Allocation dan modifikasi-modifikasi lainnya.

2. METODE

Related Works

Jezek dan Steinberger (2008:2) menyatakan bahwa awal mula Automatic Text Summarization ditandai dengan publikasi ekstraksi kalimat dari teks dengan menggunakan metode term frequency oleh Luhn pada tahun 1958. Metode Luhn berdasarkan pada asumsi bahwa frekuensi kata di dalam teks merupakan indikasi terhadap tingkat kepentingannya. Beberapa poin penting yang hingga sekarang masih digunakan yaitu langkah stemming kata menjadi bentuk dasarnya sebagai langkah pertama, dan kemudian penghapusan stop words. Pada tahun yang sama, Baxendale menambahkan ide untuk menggunakan posisi kalimat sebagai salah satu faktor penentu. Ia meneliti 200 paragraf dan menemukan bahwa 85% kalimat inti paragraf terdapat pada kalimat pertama, dan 7% terdapat pada kalimat terakhir. Disimpulkan bahwa topik inti paragraf akan terdapat pada salah satu dari kalimat pertama atau kalimat terakhir paragraf.

Perkembangan penting berikutnya ditunjukkan melalui metode Edmundson pada 1969 yang menjumlahkan bobot term frequency, posisi kalimat, frasa judul dan key phrases. Contoh key phrases tersebut yaitu "important", "results are", "paper introduces", dan lain-lain.

Penelitian algoritma Automatic Text Summarization terus dilakukan dan dapat dibagi menjadi beberapa metode. Gupta dan Lehal (2010:261) menjelaskan metode Term Frequency-Inverse Document Frequency (TF-IDF) menggunakan teori bahwa term di dalam dokumen proporsional secara berbanding terbalik dengan jumlah dokumen dalam korpus yang mengandung term tersebut. Salah satu sistem ekstraksi yang menggunakan metode ini adalah ANES yang dibuat pada tahun 1995.

Pada metode Rhetorical Structure Theory, Suneetha (2011:291-292) menjelaskan bahwa dalam koneksi logika antara bagian-bagian berbeda dalam teks dan menginterpretasikan koneksi tersebut. Informasi ini mengacu pada struktur discourse dan karakter dokumen utama.

Untuk metode selanjutnya yaitu cluster-based, Kumar dan Salim (2012:135-136) mengungkapkan bahwa clustering adalah mengelompokkan objek yang serupa ke dalam kelas-kelas tertentu. Metode ini banyak

digunakan dalam multiple document summarization.

Metode berikutnya adalah machine learning. Gupta dan Lehal (2010:262) memberi pernyataan bahwa kalimat-kalimat diklasifikasikan sebagai kalimat summary dan kalimat non-summary berdasarkan pada keberadaan kriteria-kriteria tertentu. Probabilitas klasifikasi "dipelajari" secara statistik dari training data dengan menggunakan aturan Bayes, SVM, dan lain-lain.

Sedangkan untuk metode graph, Kumar dan Salim (2012:136-137) menyatakan bahwa graph digunakan untuk merepresentasikan koneksi antara objek-objek yang ada. Kalimat merupakan objek dalam graph dan koneksi merupakan kesamaan antar kalimat. Contoh implementasinya adalah TextRank.

Metode lain yang digunakan yaitu Latent Semantic Analysis. Manning, Raghavan, dan Schutze (2009:407) mendefinisikan Singular Value Decomposition (SVD) sebagai teknik yang dapat menemukan dimensi orthogonal dari data multidimensional. SVD banyak digunakan di berbagai bidang termasuk pada pemrosesan gambar dan Latent Semantic Analysis (LSA). Salah satu contoh implementasinya dibuat oleh Gong dan Liu (2001:22-23).

Dalam metode Genetic Algorithm Based Sentence Extraction for Text Summarization, Suanmali, Salim, dan Binwahlan (2011:1-8) melakukan ekstraksi ringkasan dengan memberikan nilai untuk setiap sentence features yang dimiliki kalimat dan menggunakan Genetic Algorithm (GA) dalam proses training dokumen guna mendapatkan bobot atau proporsi masing-masing feature. Metode tersebut dibagi menjadi dua tahap yaitu preprocessing (pemotongan kalimat, tokenization, eliminasi stop words, dan stemming) serta summarization. Pada summarization, setiap kalimat akan diberi nilai antara 0 sampai 1 untuk setiap sentence features. Maksud dari sentence features adalah kategori kriteria penilaian kalimat berdasarkan karakteristik yang dimilikinya. Terdapat 8 features yaitu title feature, sentence length, term weight, sentence position, sentence to sentence similarity, proper noun, thematic word, dan numerical data. Genetic Algorithm (GA) untuk

training dokumen demi menentukan bobot sentence features.

Metode lain milik Berker dan Gungor (2012:595-599) yaitu Using Genetic Algorithms with Lexical Chain for Automatic Text Summarization menggunakan lexical chains dan pembobotan features dengan Genetic Algorithm. Sentence features yang digunakan adalah sentence location (F1), sentence relative length (F2), average TF (F3), average TF-IDF (F4), similarity to title (F5), cue words (F6), named entities (F7), numerical data (F8), sentence centrality (F9), synonym links (F10), dan co-occurrence links (F11). Untuk penentuan bobot features digunakan Genetic Algorithm. Bagi setiap dokumen, nilai features dihitung per kalimat. Pada iterasi Genetic Algorithm, bobot feature awal dibuat secara acak. Nilai kalimat dihitung dan ringkasan diekstraksi serta dievaluasi untuk setiap dokumen. Proses ini diulangi dan rata-rata presisi menunjukkan performa dari iterasi tersebut. Hasil iterasi terbaik akan dipilih oleh GA. Setiap kromosom dalam populasi merupakan vektor dari bobot features dengan representasi binary. Panjang kromosom 48 bits yang mewakili 12 features, setiap feature bernilai antara 0 sampai 15 dan direpresentasikan dalam 4 bits. Total kromosom dalam populasi adalah 1000. Untuk setiap generasi operator mating/crossover memilih 50 kromosom dengan fitness tertinggi dan memasukkannya ke populasi baru untuk generasi berikutnya. Sisa 950 kromosom lagi akan diproduksi oleh pasangan parents yang terpilih melalui roulette wheel weighting. Algoritma ini dijalankan sebanyak 100 generasi dan kromosom terbaik akan dipilih menjadi bobot features.

Metode-metode yang disebutkan di atas merupakan metode untuk bahasa Inggris. Sedangkan untuk penelitian Automatic Text Summarization bahasa Indonesia belum terlalu banyak dilakukan. Beberapa di antaranya yaitu metode graph dan algoritma exhaustive milik Budhi, Intan, Silvia, dan Stevanus (2007:1-6) yang menggunakan konsep virtual graph. Proses yang dilakukan di antaranya adalah menggunakan TF-IDF (Term Frequency-Inverse Document Frequency) dan algoritma exhaustive untuk membuat graph.

Sedangkan pada tahun 2008, Prasetyo, Uliniansyah, dan Riandi dari Badan Pengkajian dan Penerapan Teknologi (BPPT) membuat aplikasi bernama SIDoBI (Sistem

Ikhtisar Dokumen untuk Bahasa Indonesia), yang mampu meringkas dokumen menjadi suatu ikhtisar. Aplikasi ini memakai MEAD yang menggunakan metode cluster centroids.

Aristoteles, Herdiyeni, Ridha dan Adisantoso (2012:1-6) membuat Automatic Text Summarizer berbahasa Indonesia dengan menggunakan algoritma genetic. Terdapat sebelas komponen yang diperhatikan yaitu posisi kalimat (f1), kata kunci positif dalam kalimat (f2), kata kunci negatif dalam kalimat (f3), kemiripan dengan kalimat lain (f4), kemiripan dengan judul (f5), keberadaan entitas nama (f6), keberadaan data numerik (f7), panjang relatif kalimat (f8), path dari node (f9), penjumlahan kemiripan untuk setiap node (f10), dan komponen latent semantic (f11). Analisis bobot features menunjukkan bahwa hanya melalui penggunaan fitur kata kunci positif (f2), kemiripan dengan kalimat lain (f4), kemiripan dengan judul (f5) dan komponen latent semantic (f11) dapat memberi hasil yang mirip dengan penggunaan kesebelas features. Semua komponen digunakan untuk melakukan training model Genetic Algorithm untuk mendapat kombinasi bobot yang tepat bagi setiap komponen.

Proposed Method

Metode yang digunakan dalam penulisan ini terdiri atas 4 (empat) bagian pokok yaitu analisis, perancangan, implementasi dan evaluasi.

Dalam metode analisis, dilakukan studi pustaka, analisis state of the art, dan analisis aplikasi sejenis (Open Text Summarizer, Text Compactor, dan Essential Summarizer). Komponen utama aplikasi yaitu berbasis web, memiliki input text area, pilihan rasio ringkasan, dan penyimpanan summary.

Algoritma dirancang berdasarkan pada pemberian nilai sentence features dalam Genetic Algorithm Based Sentence Extraction for Text Summarization oleh Suanmali, Salim dan Binwahlan pada tahun 2011 dan penerapan Genetic Algorithm untuk pembobotan features kalimat pada Using Genetic Algorithms with Lexical Chains for Automatic Text Summarization milik Berker dan Gungor pada tahun 2012. Terdapat beberapa modifikasi seperti penggunaan LDA topic modeling, metode lemmatization milik Stephen, Christiandy, dan Rolando (2013) yang menggantikan stemming, dan

penyesuaian-penyesuaian lain untuk menangani teks bahasa Indonesia.

Implementasi pada penelitian ini terdiri atas dua tahap, yaitu training yaitu tahap untuk menentukan bobot sentence features (melibatkan proses read text input, pre-summarization, summarization, dan Genetic Algorithm untuk menghasilkan learned sentence feature weights) dan testing yaitu tahap pembuatan ringkasan teks pengguna (melibatkan read text input, pre-summarization, summarization, dan saving summary).

Pada tahap training, proses pertama yaitu input judul, isi dokumen, dan rasio akan divalidasi terlebih dahulu. Lalu proses kedua yaitu pre-summarization akan dijalankan. Langkah-langkah pre-summarization adalah: pemisahan isi dokumen teks ke dalam paragraf; NLTK tokenizer digunakan untuk sentence and word tokenization; konversi ke huruf kecil; eliminasi stop words; dan lemmatization dengan dictionary lookup ke database MySQL kamus bahasa Indonesia.

Proses ketiga adalah summarization yang terdiri dari kalkulasi nilai features TF-ISF, lokasi kalimat, panjang relatif kalimat, LDA topic modeling, title similarities, keyword similarities, sentence cohesion, dan numerical data. Basis penghitungan untuk title similarities, keyword similarities, dan sentence cohesion adalah LDA Topic Modeling dan Jensen-Shannon Divergence.

Kemudian nilai total features setiap kalimat akan dihitung dan sejumlah kalimat dengan nilai tertinggi akan diekstraksi sesuai dengan input rasio ringkasan. Nilai features setiap kalimat akan digunakan dalam training Genetic Algorithm untuk menemukan bobot setiap feature.

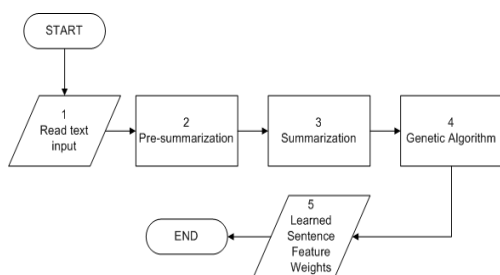
Pada proses summarization ini tidak ada perbedaan antara tahap training maupun testing kecuali pada proses sentence extraction. Dalam tahap training, masing-masing nilai feature setiap kalimat akan ditampung untuk menjadi input dalam penentuan bobot feature oleh Genetic Algorithm. Dalam Genetic Algorithm ditetapkan bahwa jumlah generasi yaitu 100, populasi 1000 kromosom, representasi kromosom binary dengan panjang 28 bits (setiap 4 bits mewakili nilai bobot sentence feature dengan rentang 0-15), fitness function adalah presisi rata-rata dari 100 dokumen, elitist selection akan meloloskan 50

kromosom, pemilihan parents untuk proses crossover melalui roulette wheel weighting dengan crossover rate sebesar 0,8 dan mutation rate sebesar 0,2.

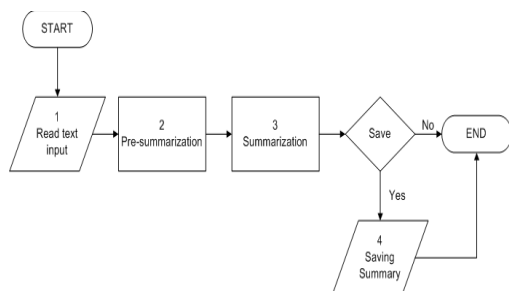
Sedangkan dalam tahap testing, nilai kalimat merupakan penjumlahan dari perkalian bobot feature (yang didapat dari training Genetic Algorithm) dengan masing-masing nilai sentence feature. Bobot sentence features hasil training akan digunakan dalam proses testing.

Pada tahap testing disediakan fasilitas penyimpanan ringkasan dalam bentuk plain text (.txt) atau PDF (.pdf).

Alur kerangka berpikir algoritma digambarkan dalam skema flowchart training pada Gambar 1 dan flowchart testing pada Gambar 2.



Gambar 1 Flowchart Training Algoritma Automatic Text Summarization



Gambar 2 Flowchart Testing Algoritma Automatic Text Summarization

Pada tahap evaluasi dilakukan uji coba kecepatan pembuatan ringkasan, presisi, recall, F-measure, dan evaluasi subjektif.

3. HASIL DAN PEMBAHASAN

Untuk pengujian, algoritma Automatic Text Summarization berbahasa Indonesia diimplementasikan dengan menggunakan Python, web framework Django, package

NLTK, dan library Gensim pada localhost environment.

Dilakukan uji kecepatan pembuatan ringkasan untuk 50 dokumen. Kumpulan dokumen tersebut merupakan artikel dari situs kompas.com, detik.com, tempo.co, gatra.com, chip.co.id, dan femina.co.id. Rincian uji kecepatan pembuatan ringkasan dapat terangkum pada tabel berikut ini:

Tabel 1. Gambaran Umum Uji Kecepatan Pembuatan Ringkasan

Artikel	Waktu Tercepat	Waktu Terlama	Rata-rata Waktu
50 artikel	2,395 detik (5 kalimat, 143 kata)	3,642 detik (33 kalimat, 571 kata)	2,85062 detik (rata-rata 14,94 kalimat, 280,12 kata)

Dapat disimpulkan bahwa jumlah kalimat dan kata dalam dokumen memiliki pengaruh terhadap lama waktu pembuatan ringkasan, tetapi tidak secara mutlak menentukan durasi tersebut.

Kemudian uji algoritma dilakukan dalam kalkulasi presisi, recall, dan F-measure untuk 50 artikel, ringkasan referensi yang digunakan adalah ringkasan manual yang dibuat oleh 29 orang dengan rasio berkisar antara 15%-30%. Sedangkan ringkasan sistem dibuat dengan tiga rasio yaitu 10%, 20%, dan 30%. Berikut ini adalah rangkuman dari uji presisi, recall, dan F-measure terhadap 50 dokumen dengan rasio ringkasan sistem 10%, 20%, dan 30%.

Tabel 2. Presisi, Recall, dan F-measure 50 Artikel

Rasio	Rata-rata Presisi	Rata-rata Recall	Rata-rata F-measure
10%	0,66666	0,258	0,372025
20%	0,60274	0,4213	0,495946
30%	0,53448	0,58134	0,556926

Dapat disimpulkan bahwa nilai rata-rata presisi menurun dan recall membaik seiring dengan bertambahnya rasio ringkasan.

F-measure digunakan untuk menyatakan kualitas ringkasan sistem dengan memadukan presisi dan recall. Terlihat nilai F-measure tertinggi didapatkan saat rasio ringkasan 30%.

Untuk evaluasi subjektif, sejumlah 55 artikel asli dan ringkasan rasio 30% dari tahap testing disajikan kepada khalayak umum melalui kuesioner. Kuesioner disebarluaskan melalui spreadsheet Google Docs dari tanggal 28 Desember 2013 hingga 4 Januari 2014. Pengguna diminta untuk menilai apakah ringkasan tersebut sudah mewakili informasi yang terpenting dari artikel asli. Pilihan jawaban yaitu: sangat tidak mewakili; tidak mewakili; cukup; mewakili; sangat tidak mewakili.

Total respon yang diterima untuk 55 artikel tersebut adalah sebanyak 645 jawaban dengan rincian sebagai berikut:

Tabel 3. Hasil Kuesioner Evaluasi Subjektif

Pilihan Jawaban	Jumlah Respon	Persentase (%)
Sangat tidak mewakili	5	0,775
Tidak mewakili	35	5,426
Cukup	154	23,876
Mewakili	330	51,163
Sangat mewakili	121	18,760

Dari hasil tersebut dapat disimpulkan bahwa sebanyak 69,923% merupakan respon positif terhadap keterwakilan artikel asli oleh ringkasan sistem rasio 30%. Respon negatif sejumlah 6,201% dan netral sejumlah 23,876%. Perbedaan signifikan antara respon positif dengan respon negatif dan netral menunjukkan bahwa pengguna merasa bahwa kualitas mayoritas ringkasan sudah memuaskan karena sudah mewakili informasi penting yang dibutuhkan dari artikel asli.

4. KESIMPULAN

Berdasarkan hasil dan pembahasan uji algoritma Automatic Text Summarization untuk bahasa Indonesia ini dapat disimpulkan bahwa algoritma dapat menghasilkan ringkasan extractive yang mewakili informasi penting dari suatu dokumen teks tunggal (single document) berbahasa Indonesia dengan lebih cepat (berkisar antara 2,395-3,642 detik

untuk dokumen teks yang terdiri dari 5-33 kalimat). Dari tiga rasio ringkasan yang diuji, nilai F-measure tertinggi dicapai oleh ringkasan dengan rasio 30% yaitu sebesar 0,556926 dengan presisi 0,53448 dan recall 0,58134.

Saran-saran yang dapat dipertimbangkan untuk perbaikan atau pengembangan penelitian selanjutnya berkaitan dengan algoritma Automatic Text Summarization berbahasa Indonesia yaitu menambahkan feature proper noun untuk bahasa Indonesia. Selain itu juga algoritma lemmatization yang digunakan dalam tahap pre-summarization dapat ditingkatkan akurasi sehingga lebih dari 98%. Algoritma Automatic Text Summarization ini juga dapat dijadikan sebagai dasar dalam pengembangan algoritma untuk multiple document summarization berbahasa Indonesia. Analisis redundansi berperan penting dalam jenis summarization tersebut. Selain itu perlu juga dibuat corpus dokumen dan ringkasan ideal bahasa Indonesia yang dibuat oleh ahli agar standarisasi uji algoritma dan evaluasi algoritma Automatic Text Summarization dapat diwujudkan.

UCAPAN TERIMA KASIH

Bapak Drs. Marcus Bambang Walgito SF.Th. selaku dosen bahasa Indonesia di Universitas Bina Nusantara, dan seluruh responden yang telah berpartisipasi dalam evaluasi implementasi algoritma.

DAFTAR PUSTAKA

- [1] Aristoteles, Hardiyeni, Ridha, and Adisantoso. 2012. Text Feature Weighting for Summarization of Documents in Bahasa Indonesia Using Genetic Algorithm. *International Journal of Computer Science Issues* 9(1):1-6.
- [2] Bawden and Robinson. 2009. The Dark Side of Information: Overload, Anxiety and Other Pathologies. *Journal of Information Science* 35(2):180-191.
- [3] Berker and Gungor. 2012. Using Genetic Algorithms with Lexical Chains for Automatic Text Summarization. *ICAART* 1:595-600.
- [4] Budhi, Intan, Silvia, and Stevanus. 2007. Indonesian Automated Text Summarization. *Proceeding ICSIT* 2007.

- [5] Gholamrezazadeh, Salehi, and Gholamzadeh. 2009. A Comprehensive Survey on Text Summarization System. Proceedings of CSA 9:1-6.
- [6] Gong and Liu. 2001. Generic Text Summarization Using Relevance Measure and Latent Semantic Analysis. Proceedings of The 24th International ACM SIGIR Conference on Research and Development in Information Retrieval hlm. 19-25.
- [7] Gupta and Lehal. 2010. A Survey of Text Summarization Extractive Techniques. Journal of Emerging Technologies in Web Intelligence 2(3):258-268.
- [8] Jezek and Steinberger. 2008. Automatic Text Summarization (The State of The Art 2007 and New Challenges). Znalosti 2008 hlm. 1-12.
- [9] Jurafsky and Martin. 2006. Speech and Language Processing: An Introduction To Natural Language Processing, Computational Linguistics, And Speech Recognition 2nd Edition. New Jersey: Pearson Prentice Hall.
- [10] Kumar and Salim. 2012. Automatic Multi Document Summarization Approaches. Journal of Computer Science 8(1):133-140.
- [11] Manning, Raghavan, and Schutze. 2009. Introduction to Information Retrieval. Cambridge: Cambridge University Press.
- [12] Netcraft. 2013. Web Server Survey. Retrieved August 18, 2013, from <http://news.netcraft.com/archives/2013/08/09/august-2013-web-server-survey.html>.
- [13] Suhartono, Christiandy, and Rolando. 2014. "Lemmatization Technique in Bahasa: Indonesian Language". Journal of Software 9 (5), 1202-1209, 2014.
- [14] Suanmali, Salim, and Binwahlan. 2011. Genetic Algorithm Based Sentence Extraction for Text Summarization. International Journal of Innovative Computing 1(1):1-22.
- [15] Suneetha. 2011. Automatic Text Summarization: The Current State of The Art. International Journal of Science and Advanced Technology 1(9):283-293.