Original Research

Automatic Text Summarization Berdasarkan Pendekatan Statistika pada Dokumen Berbahasa Indonesia

Christopher Setyawan [1]*, Njoto Benarkah [1], Vincentius Riandaru Prasetyo [1]

[1] jurusan Teknik Informatika Fakultas Teknik, Universitas Surabaya, Raya Kalirungkut Surabaya-Indonesia 60293

* corresponding author: christopher1setyawan@gmail.com

Abstract—*Propelled by the modern technological innovations data and text will be more abundant throughout the year. With this much text, automatic text summarization is needed now more than ever to help summarize a text. Automatic text summarization is defined as the creation of a shortened version of a text by a computer program, the product of this procedure still contains the most important points of the original text. Statistical approaches is one of automatic text summarization method. There is 5 statistical approaches that being used namely aggregation similarity method, frequency method, location method, title method (if text has a title), dan tf-based query method (if text doesn't have a title). Cosine similarity is used to calculate title method, aggregation similarity method, and tf- based query method. There is two type of validation, user validation and system validation. For system validation compare the similarity between human summary and summary generated by program, which result in accuracy of 76.7647% for summary with 30% length of the original journal. For user validation result in 82% accuracy. The conclusion based on user validation and system validation is statistical approaches is suitable for automatic text summarization.*

Keywords: *automatic text summarization, statistical approaches, Indonesian document, cosine similarity*

Abstrak— Dengan kemajuan teknologi jumlah data dan teks akan semakin melimpah sepanjang tahun. Dengan banyaknya teks ini dibutuhkan bantuan automatic text summarization untuk merangkum teks tersebut. Automatic text summarization didefinisikan sebagai versi singkat dari suatu teks menggunakan program komputer yang hasilnya masih memiliki informasi penting berupa gagasan dasar dan kata atau kalimat yang dapat merepresentasikan keseluruhan teks original. Salah satu metode dalam automatic text summarization adalah pendekatan statistika. Pendekatan statistika yang digunakan ada 5 yaitu aggregation similarity method, frequency method, location method, title method (bila teks memiliki judul), dan tf-based query method (bila teks tidak memiliki judul). Cosine similarity dipakai untuk perhitungan title method, tf-based query method, dan aggregation similarity method. Validasi dilakukan dengan dua macam validasi. Pertama adalah validasi sistem dengan membandingkan similaritas antara rangkuman program dan rangkuman manusia, yang menghasilkan akurasi 76.7647% untuk rangkuman dengan panjang 30% dari jurnal original. Kedua adalah validasi user yang menghasilkan akurasi 81%. Kesimpulannya berdasarkan validasi user dan validasi sistem yang cukup baik maka pendekatan statistika cocok dipakai dalam kasus automatic text summarization.

Kata kunci: *automatic text summarization, pendekatan statistika, cosine similarity, dokumen berbahasa Indonesia*

PENDAHULUAN

Manuel dan Moreno (2014) menuliskan *automatic text summarization* menurut definisi kamus *Oxford* adalah versi singkat dari suatu teks menggunakan program komputer yang hasilnya masih memiliki informasi penting berupa gagasan dasar dan kata atau kalimat yang dapat merepresentasikan keseluruhan teks original. Tujuan dari *automatic text summarization* adalah untuk membuat ringkasan yang singkat, mudah dipahami kalimatnya dan memiliki informasi penting yang ingin disampaikan dari teks original tersebut.

Dengan kemajuan teknologi yang semakin pesat, data dan teks akan semakin banyak dan melimpah seiring dengan bertambahnya tahun. (Garbade, 2018). Beberapa alasan mengapa diperlukannya *automatic text summarization* adalah mempersingkat waktu pembacaan, mempermudah proses seleksi, dan menghindari bias (Manuel dan Toreno, 2014). Alasan lainnya adalah karena *automatic text summarization* dapat merangkum teks-teks panjang dengan cepat dan akurat, suatu hal yang akan memakan waktu lama dan membutuhkan biaya besar bila dilakukan oleh manusia tanpa bantuan mesin. (Garbade, 2018). Berdasarkan hasil wawancara juga didapatkan hasil bahwa narasumber memiliki masalah saat ingin memahami dan mencari inti suatu teks dan masalah waktu baik dari waktu untuk memahami maupun waktu untuk merangkum.

Pada *automatic text summarization* terdapat istilah kompresi τ yang didapatkan dari rasio perbandingan panjang rangkuman dan panjang teks original, panjang teks ini bisa berarti karakter, kata, ataupun kalimat. Hasil *automatic text summarization* terbaik ditemukan saat kompresi τ = 15 sampai dengan 30 % dari panjang teks original atau kurang dari sepertiga panjang teks original (Manuel dan Moreno, 2014).

Ada dua macam automatic text summarization yaitu metode *extractive* dan metode *abstractive*. Metode *extractive* adalah metode yang mengambil kata atau kalimat pada teks original kemudian memberikan ringkasan, pada teknik ini tidak ada perubahan kata atau kalimat pada teks. Sedangkan metode *abstractive* adalah metode yang memberikan kata baru yang tidak ada pada teks original lalu menggabungkan dan menyusun kata baru tersebut dengan kata-kata original untuk membuat kalimat baru dan memberikan ringkasan dari kalimat-kalimat baru tersebut. (Garbade, 2018). Pendekatan statistika adalah metode *extractive*.

Sebelum dilakukan pendekatan statistika langkah pertama yang perlu dilakukan adalah *preprocessing*. *Prepoccessing* yang digunakan adalah segmentasi, *stemming*, *stopword*, dan *tokenizing*. Segmentasi adalah memecah teks menjadi paragraf, lalu dari paragraf dipecah menjadi kalimat nantinya hasil rangkuman akan dikembalikan menjadi kalimat-kalimat asli dari segmentasi ini. *Stemming* digunakan untuk merubah kata dalam tiap kalimat hasil segmentasi menjadi kata dasarnya contoh kata perubahan menjadi kata ubah. *Stopword* digunakan untuk menghapus kata yang kurang penting atau *irrelevant* contoh kata ini, dari, ke, pada. *Tokenizing* merupakan proses untuk memisah kalimat menjadi kata.

Langkah selanjutnya setelah *preprocessing* adalah ekstrasi fitur. Ekstrasi fitur yang digunakan adalah *term frequency raw (tf raw)* dan *term frequency – inverse document frequency (tf idf). Term frequency – inverse document frequency* adalah perkalian dari *term frequency row (tf raw)* yang dipakai untuk menghitung jumlah kemunculan kata / *term* untuk tiap kalimat pada teks. Sedangkan *inverse document frequency* adalah perhitungan untuk menentukan bobot pada suatu kata / term dalam suatu teks. Nantinya hasil dari nilai ekstrasi fitur tf-idf ini digunakan untuk perhitungan similaritas, dan untuk beberapa metode dalam pendekatan statistika.

Similaritas dalam pendekatan statistika digunakan untuk menghitung kemiripan antara kalimat dengan kalimat lainnya, atau kalimat dengan judul. Similaritas yang digunakan adalah *cosine similarity*. Untuk proses perhitungannya didapatkan dari nilai-nilai dari ekstrasi tf-idf sebelumnya.

Metode pendekatan statistika yang digunakan ada 5 yaitu *title method, location method, aggregation similarity method, freguency method, tf-based query method* (Ko dan Seo, 2008). Title method dilakukan apabila teks memiliki judul, skor tiap kalimat dilakukan dengan perhitungan similaritas antara judul dengan masing -masing kalimatnya menggunakan *cosine similarity*. *Tf-based query method* akan dilakukan apabila teks tidak memiliki judul. Kata-kata yang memiliki jumlah frekuensi kemunculan tertinggi dalam teks akan dipilih untuk menggantikan fungsi judul, kemudian skor kalimat akan dihitung dari similaritas antara kalimat dengan kata-kata yang menggantikan judul ini. Bila metode ini dipakai *title method* tidak akan digunakan dan juga sebaliknya. Pada *aggregation similarity method* skor kalimat akan dihitung dari total similaritas satu kalimat dengan seluruh kalimat lainnya yang ada pada teks tersebut. Pada *frequency method* dihitung skor bobot suatu kalimat dari total bobot tiap kata yang dimiliki oleh suatu kalimat, bobot ini didapatkan dari tf-idf. Pada *location method* akan memberikan skor tiap kalimat terhadap lokasi kalimat dalam sebuah teks, biasanya kalimat pada awal teks lebih penting dan dapat menjadi ringkasan yang baik sehingga memiliki skor yang lebih tinggi. Terakhir dari ke 4 metode yang akan dipakai untuk merangkum teks tersebut akan dihitung skor akhirnya untuk masing-masing kalimat. Kalimat yang memiliki skor akhir tinggi inilah yang akan dijadikan hasil rangkuman.

https://doi.org/10.24123/saintek.v2i1.4045

METODE

Metode penelitian dilakukan dengan menganalisa keadaan saat ini, analisa sistem sejenis dan kebutuhan sistem. Keadaan saat ini dianalisa dengan melakukan wawancara terhadap 3 mahasiswa dan 1 guru SD. Dari narasumber mengatakan pernah menemui teks yang cukup panjang dalam perkuliahan atau saat mengajar dan pernah mengalami masalah dalam memahami teks tersebut. Keempat dari narasumber mengatakan biasanya membuat rangkuman terlebih dahulu untuk memahami suatu teks, setelah selesai merangkum kemudian narasumber juga akan menghafal dan mempelajari materi dari hasil rangkumanya. Ada 2 narasumber yang mengatakan cukup sulit untuk memahami inti teks dan malas saat ingin merangkum. Keempat dari narasumber mengeluhkan tentang waktu yang cukup lama untuk memahami teks lalu merangkum kemudian mempelajarinya lagi. Kesimpulan inti dari permasalah-permasalahan narasumber adalah tentang waktu dan memahami serta mencari inti materi.

Analisis sistem sejenis yang dibahas menggunakan penelitian yang berkaitan dengan *automatic text summarization* dengan menggunakan beberapa macam metode. Penelitian pertama oleh Tardan, Erwin, Eng dan Muliady (2013), dengan membandingkan hasil metode *semantic* dengan *euclidean*, *semantic* dengan *cosine* dan pendekatan statistika dengan *word frequency* dan pendekatan statistika dengan *jaccard*. Pendekatan statistika yang digunakan adalah *title method* dan *location method*. Hasilnya kompresi dengan pendekatan statistika memiliki nilai tertinggi, tetapi *semantic* memiliki tingkat hasil subjektivitas lebih tinggi.

Penelitian kedua dilakukan oleh Kyoomarsi, Khosravi, Eslami, & Davoudi (2010). Metode yang digunakan adalah *fuzzy logic* dengan menggunakan *Mamdani*. Kesimpulan yang didapatkan peneliti adalah kelebihan dari metode yang dipakai adalah hasil rangkuman lebih mirip dengan rangkuman manusia, dan kelemahannya adalah proses untuk mendesain variabel dan rules sangat memakan waktu.

Penelitian ketiga dilakukan oleh Darmawan & Wahono (2015). Metode yang digunakan adalah ekstraksi kata kunci dengan nilai tf-idf dan menggabungkanya dengan *cosine similarity*. Kesimpulan yang didapatkan oleh peneliti adalah penggunaan *cosine similarity* dapat meningkatkan kepaduan antar kalimat pada rangkuman, meningkatkan hasil rangkuman dan kompresi terbaik yang didapatkan dari hasil percobaan adalah 50%.

Dari permasalahan diatas akan dibuat suatu program *automatic text summarization* untuk mempersingkat waktu dan memudahkan *user* dalam memahami teks. *User* hanya perlu meng-*upload file* yang ingin dirangkum atau *copy paste* pada bagian yang disediakan atau mengetik manual teks yang ingin dirangkum. Program akan menggunakan pendekatan statistika dengan *cosine similarity*.

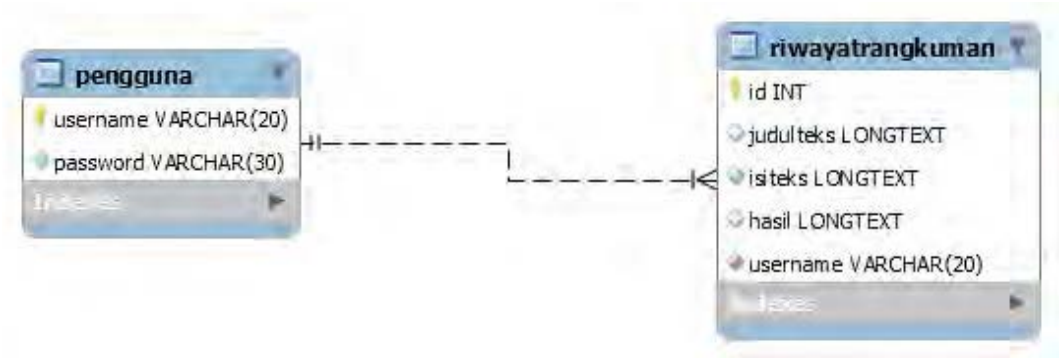
HASIL DAN PEMBAHASAN

Pada tahap ini dilakukan desain dan implementasi program, kemudian dilakukan ujicoba yaitu verifikasi dan validasi untuk mengetahui tingkat akurasi program dalam pembuatan rangkuman.

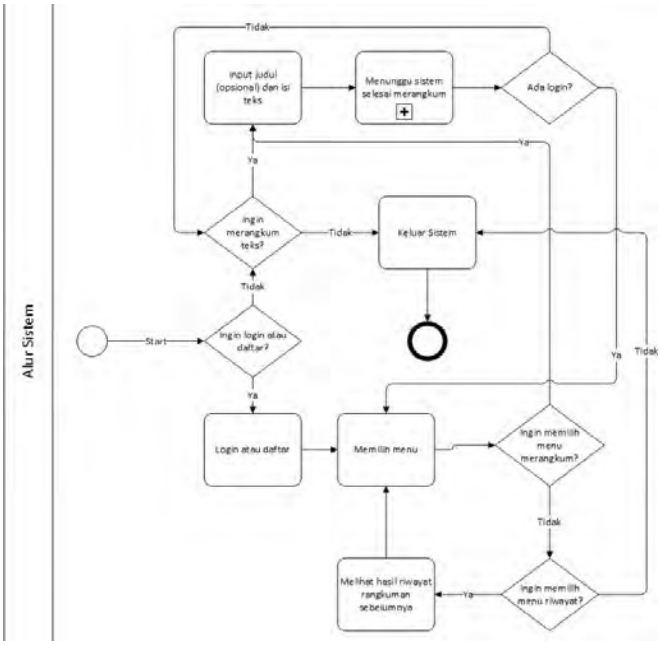
Desain pembuatan *automatic text summarization* meliputi desain *database*, desain proses, dan desain antarmuka. Untuk ERD karena *automatic text summarization* sebenarnya tidak perlu disimpan dalam *database*, maka *database* hanya akan berjalan apabila *user login* ke sistem, dan hanya digunakan untuk menyimpan data *user* dan *history* rangkuman yang dilakukan oleh *user*, proses inti *automatic text summarization* tidak membutuhkan *database* dapat dilihat pada Gambar 1.

Untuk desain proses pertama saat *user* masuk kedalam *website*, *user* bisa *login* atau tidak. Apabila *user* ingin merangkum maka setelah *user* memasukkan *input* teks atau *file* dan judul (opsional) prosesnya adalah menunggu sistem selesai merangkum. Pada proses ini dicek terlebih dahulu apakah format *file* benar atau adakah teks yang dikirim (txt, docx) apabila format tidak benar akan ditampilkan pesan *error*. Apabila format benar dilakukan proses *automatic text summarization* mulai dari *preprocessing*, ekstraksi fitur, perhitungan *cosine*

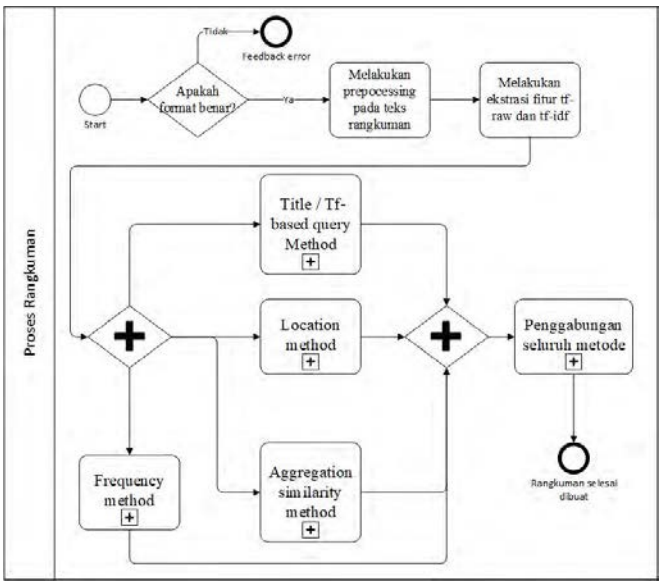
similarity, dan metode yang dipakai dalam pendekatan statistika. Setelah itu sistem akan menampilkan hasil rangkuman yang telah dibuat. Apabila pada awal *user login* hasil rangkuman akan disimpan dalam *database*, dan *user* dapat melihat riwayat rangkuman yang sudah dirangkum oleh *user* tersebut. Dapat dilihat pada Gambar 2 dan Gambar 3. Untuk *user interface* halaman awal tanpa *login* dapat dilihat pada Gambar 4.



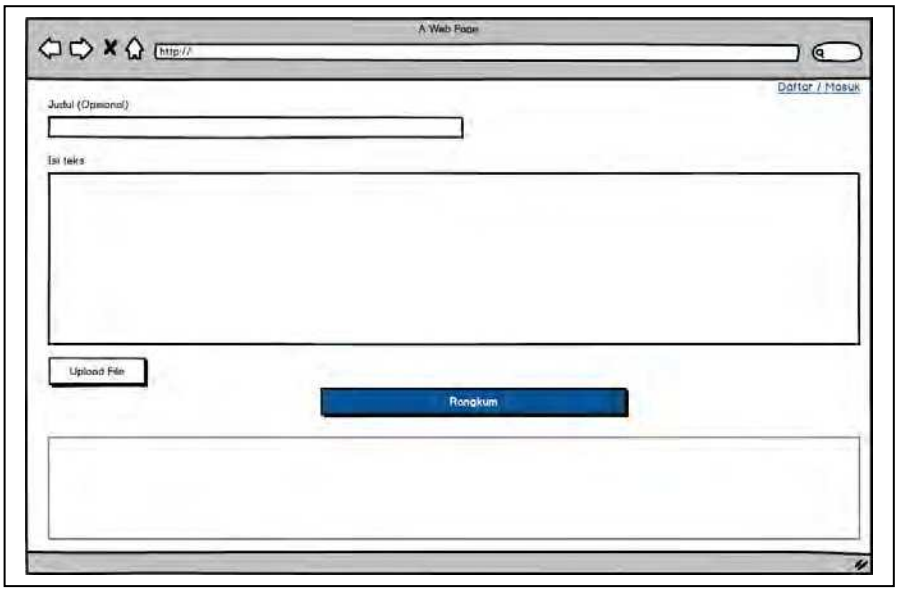
Gambar 1. ERD program automatic text summarization.



Gambar 2. Desain proses alur pengguna.



Gambar 3. Desain proses rangkuman sistem.



Gambar 4. Desain user interface halaman awal.

Implementasi sistem menggunakan bahasa pemrograman *PHP*. *Database* dibuat dengan *MySql*. Verifikasi pengecekan metode perhitungan manual dengan hasil rangkuman yang dibuat oleh program. Untuk validasi ada 2 yaitu validasi sistem dan validasi *user*. Pada validasi sistem dilakukan dengan memilih 10 teks yang akan dihitung kemiripannya antara rangkuman manusia dengan hasil rangkuman yang dibuat oleh program. Pada validasi *user* dilakukan dengan meminta 10 *user* untuk memilih teks/ dokumen yang ingin dirangkum, kemudian menilai dari rentang 0-10 hasil rangkuman yang dibuat oleh program, apakah menurut *user* hasil rangkuman memuaskan, memiliki informasi penting dan inti teks dan apakah menurut *user* cukup singkat. Rata-rata dari validasi *user* adalah 81% dan rata-rata cukup puas dengan hasil rangkuman. Selanjutnya juga dibandingkan validasi *user* jika memakai *title method* dan *location method* saja menggunakan data teks dan *user* yang sama seperti saat dilakukan validasi user asal. Terakhir perbandingan dengan *semantic* untuk 1 data teks percobaan dengan meminta 10 *user* menilai hasil dari 2 rangkuman tersebut.

Pada validasi sistem dibandingkan antara hasil rangkuman 15%, 20%, 25%, dan 30% dari teks original untuk menentukan tingkat kompresi yang memiliki hasil paling baik. Hasil rangkuman ini akan dihitung similaritasnya dengan rangkuman yang dibuat oleh manusia. Hasilnya dapat dilihat pada tabel 1.

Tabel 1
Hasil Validasi Sistem

Teks Ke-	Similaritas rangkuman 15% dan rangkuman manusia	Similaritas rangkuman 20% dan rangkuman manusia	Similaritas rangkuman 25%dan rangkuman manusia	Similaritas rangkuman 30% dan rangkuman manusia
1	0.641963	0.641963	0.6959227	0.7135533
2	0.7381301	0.8000847	0.8153694	0.814961
3	0.5282002	0.5282002	0.6542861	0.6925869
4	0.8315932	0.8669614	0.8773525	0.8909504
5	0.6342576	0.685882	0.7609974	0.7741126
6	0.7335139	0.7264132	0.7877683	0.8080817
7	0.5278007	0.5732882	0.6087139	0.6376366
8	0.5422262	0.5713861	0.6461512	0.6850196
9	0.7055405	0.7193056	0.8040864	0.7998335
10	0.8276997	0.8426774	0.8469788	0.8597309
Rata-rata	0.67109251	0.69561618	0.74976267	0.76764665

Dari hasil tabel 6.7 maka untuk hasil rangkuman dengan panjang 15% memiliki rata-rata nilai similaritas 0.67109251 yang berarti memiliki tingkat akurasi 67.1093%. Untuk hasil rangkuman dengan panjang 20% memiliki rata-rata nilai similaritas 0.69561618 yang berarti memiliki tingkat akurasi 69.5616%. Untuk hasil rangkuman dengan panjang 25% memiliki rata-rata similaritas 0.74976267 yang berarti memiliki tingkat akurasi 74.9763%. Terakhir untuk hasil rangkuman dengan panjang 30% memiliki rata-rata similaritas 0.76764665 yang berarti memiliki akurasi sebesar 76.7647%. Berdasarkan hasil percobaan tersebut menandakan rangkuman dengan panjang 30% yang paling bagus untuk digunakan dalam rangkuman. Hasil percobaan juga menunjukkan bahwa untuk validasi sistem didapatkan hasil yang cukup memuaskan.

Untuk validasi *user* jika menggunakan *title method* dan *location method* didapatkan akurasi 65% lebih rendah dari validasi *user* yang memiliki akurasi 81%. Maka penggunaan penggabungan *title method/ tf-based query method*, *location method*, *frequency method*, *aggregation similarity method* terbukti lebih meningkatkan akurasi hasil rangkuman dibandingan dengan penggunaan *title method* dan *location method*. Untuk validasi perbandingan dengan *semantic* didapatkan hasil akurasi 76% untuk *semantic* dan 78% untuk program. Dari pendapat *user* didapatkan hasil bahwa *semantic* lebih unggul di segi kesinambungan kalimat dan lebih berurutan, sedangkan pendekatan statistika yang digunakan program lebih unggul di segi kalimat yang lebih dianggap penting dan berbobot, serta informasi yang lebih jelas.

SIMPULAN

Dari hasil uji coba dan evaluasi didapatkan kesimpulan dan saran. Berdasarkan hasil validasi user memiliki nilai rata-rata 81%. Berdasarkan hasil validasi sistem hasil terbaik ada pada rangkuman dengan panjang 30% dari jurnal asli dengan nilai rata-rata 76.7647%. Penggunaan penggabungan *title method/ tf- based query method*, *location method*,

aggregation similarity method, dan *frequency method* dapat meningkatan akurasi rangkuman. Berdasarkan pendapat *user* hasil rangkuman *semantic* lebih unggul di segi kesinambungan kalimat dan kalimat lebih beruntun, sedangkan hasil rangkuman pendekatan statistika lebih unggul di segi informasi yang lebih jelas dan kalimat yang lebih penting.

Saran dapat ditambah atau digabungkan dengan metode *semantic* untuk meningkatkan kesinambungan antar kalimatnya. Bisa juga digabungkan dengan metode lainnya yang bisa merangkai kalimat sendiri dengan mengambil hanya beberapa kata penting dalam kalimat asal sehingga rangkuman lebih lagi memiliki kesinambungan dan lebih seperti hasil rangkuman manusia. Lalu menggunakan suatu metode untuk mengenali rujukan kata, agar kata rujukan yang seharusnya mengacu pada kata di kalimat sebelumnya menjadi jelas dan tidak rancu.

PUSTAKA ACUAN

Darmawan, R., & Wahono, R. S. (2015). Hybrid Keyword Extraction Algorithm and Cosine Similarity for Improving Sentences Cohesion in Text Summarization. Journal of Intelligent Systems, 1(2), 109 – 114. Retrieved from http://journal.ilmukomputer.org/index.php/jis/article/view/44

Garbade, M. J. (2018, September 19). A Quick Introduction to Text Summarization in Machine Learning. Retrieved from https://towardsdatascience.com/a- quick-introduction-to-text-summarization-in-machine-learning- 3d27ccf18a9f

Ko, Y., & Seo, J. (2008). An effective sentence-extraction technique using contextual information and statistical approaches for text summarization. Pattern Recognition Letters, 29(9), 1366 – 1371. DOI: 10.1016/j.patrec.2008.02.008

Kyoomarsi, F., Khosravi, H., Eslami, E., & Davoudi, M. (2010). Extraction-based text summarization using fuzzy analysis. Iranian Journal of Fuzzy System, 7(3), 15 – 32. DOI: 10.1007/978-3-540-79187-4_11

Manuel,J., & Moreno, T. (2014). Automatic Text Summarization. DOI:10.1002/9781119004752

Tardan, P. P., Erwin, E., Eng, K. I., & Muliady, W. (2013). Automatic Text Summarization Based on Semantic Analysis Approach for Documents in Indonesia Languange. 2013 International Conference on Information Technology and Electrical Engineering (ICITEE), 47 – 52. DOI: 10.1109/ICITEED.2013.6676209