

Interview Result Extraction For Functional Requirements Identification Using TextRank Algorithm

Dika Rizky Yunianto¹, Annisa Rahmania Putri², Vivi Nur Wijyaningrum³

^{1,2,3} Jurusan Teknologi Informasi, Politeknik Negeri Malang, Malang, 65151, Indonesia

Informasi Artikel

Diterima : 25 Februari 2026
Revisi : 15 Maret 2026
Publikasi : 31 Maret 2026

Kata Kunci:

TextRank,
Text Summarization
Requirements Engineering
Software Development

ABSTRAK

Identifikasi kebutuhan fungsional merupakan tahap kritis dalam pengembangan sistem informasi yang menentukan kesesuaian sistem dengan kebutuhan pengguna. Wawancara merupakan teknik dominan untuk menggali kebutuhan, namun hasil wawancara berupa teks panjang dan tidak terstruktur memerlukan waktu analisis yang lama dan rentan subjektivitas jika dilakukan manual. Penelitian ini bertujuan menerapkan algoritma TextRank untuk merangkum hasil wawancara secara otomatis agar informasi penting diperoleh lebih ringkas. Penelitian dilakukan melalui tahapan preprocessing teks, perhitungan kemiripan antar kalimat (word overlap), pembentukan graf, serta pemeringkatan kalimat menggunakan algoritma PageRank. Kalimat penting dipilih berdasarkan variasi threshold ($k = 3, 5, 7, 9$). Tiga metode dibandingkan: TextRank standar, TextRank dengan pembobotan TF-IDF, dan TextRank dengan preprocessing stemming. Evaluasi menggunakan metrik precision, recall, dan F1-score terhadap ground truth. Hasil menunjukkan TextRank standar dengan threshold 7 memberikan performa paling seimbang (F1-score 71,05%), lebih stabil dibandingkan metode berbasis TF-IDF untuk data wawancara. Algoritma ini terbukti efektif membantu efisiensi proses identifikasi kebutuhan fungsional.

ABSTRACT

Functional requirement identification is a critical stage in information system development that determines system alignment with user needs. Interviewing is a dominant technique for eliciting requirements, but interview results in the form of long, unstructured text require significant analysis time and are prone to subjectivity if done manually. This research aims to apply the TextRank algorithm to automatically summarize interview results so that important information can be obtained more concisely. The research was conducted through text preprocessing stages, sentence similarity calculation (word overlap), graph construction, and sentence ranking using the PageRank algorithm. Important sentences were selected based on threshold variations ($k = 3, 5, 7, 9$). Three methods were compared: standard TextRank, TextRank with TF-IDF weighting, and TextRank with stemming preprocessing. Evaluation used precision, recall, and F1-score metrics against ground truth. Results show that standard TextRank with threshold 7 provided the most balanced performance (F1-score 71.05%), being more stable than TF-IDF based methods for interview data. This algorithm proved effective in improving the efficiency of the functional requirement identification process.

This is an open-access article under the [CC BY-SA](https://creativecommons.org/licenses/by-sa/4.0/) license



*Penulis Koresponden

Email: dika.rizky@polinema.ac.id

Cara sitasi IEEE::

R. D. Yunianto, A. R. Putri, V. N. Wijyaningrum, "Interview Result Extraction For Functional Requirements Identification Using TextRank Algorithm", *Journal of Artificial Intelligence and Software Engineering (J-AISE)*, vol. 6, no. 1, p. 69-75, Maret 2026. doi:10.30811/jaise.v6i1.8868

1. PENDAHULUAN

Perkembangan teknologi informasi yang pesat telah mendorong organisasi, perusahaan, maupun institusi pendidikan untuk mengadopsi sistem informasi berbasis website dalam mendukung kegiatan operasional dan pengambilan keputusan [1]. Kualitas sistem informasi sangat bergantung pada tahap awal pengembangannya, yaitu Requirements Engineering (RE) atau rekayasa kebutuhan. Kegagalan dalam mengidentifikasi kebutuhan secara akurat sering kali menjadi penyebab utama kegagalan proyek perangkat lunak [2][3]. Tanpa analisis kebutuhan yang tepat, sistem informasi yang dibangun berisiko tidak memenuhi harapan pengguna secara optimal [4]. Oleh karena itu, proses elisitasi kebutuhan menjadi krusial untuk memastikan sistem yang dibangun sesuai dengan ekspektasi pengguna [5].

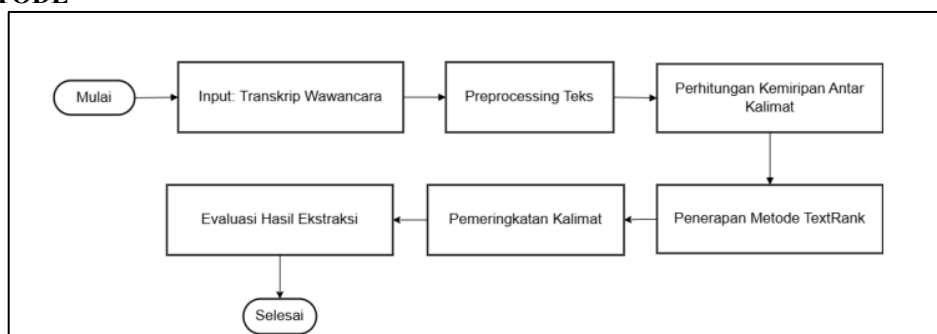
Dalam pengembangan sistem informasi, wawancara merupakan salah satu teknik elisitasi yang paling dominan digunakan karena kemampuannya menggali informasi mendalam mengenai masalah, harapan, dan proses bisnis pengguna [6][5][7]. Melalui wawancara, peneliti dapat memperoleh data kualitatif yang kaya. Namun, data wawancara sering kali bersifat heterogen, berupa teks naratif panjang, tidak terstruktur, dan mengandung beragam gaya bahasa serta informasi yang tidak relevan (noise) [8][4]. Analisis manual terhadap transkrip wawancara membutuhkan waktu yang lama (time-consuming) dan menjadi tantangan tersendiri, terutama ketika melibatkan banyak narasumber dengan volume data yang besar.

Untuk mengatasi permasalahan tersebut, pendekatan Automatic Text Summarization (ATS) atau peringkasan teks otomatis dapat diterapkan [11][12]. Salah satu metode ATS yang efektif adalah TextRank, sebuah algoritma berbasis graf yang diadaptasi dari PageRank [13]. TextRank bekerja dengan merepresentasikan kalimat sebagai simpul (vertex) dalam graf dan menghitung tingkat kepentingan kalimat berdasarkan hubungannya dengan kalimat lain [14]. Keunggulan utama TextRank adalah sifatnya yang unsupervised, sehingga tidak memerlukan data latih (training data) yang besar, yang sering kali sulit diperoleh dalam konteks wawancara spesifik proyek yang datanya terbatas dan unik [15][16].

Penelitian sebelumnya telah menunjukkan keberhasilan TextRank dalam merangkum berbagai jenis teks. Penelitian oleh Sihombing et al. menerapkan algoritma TextRank untuk merangkum artikel ilmiah berbahasa Indonesia dan berhasil mencapai akurasi hingga 88,53% [17]. Penelitian lain menunjukkan bahwa TextRank mampu mengukur kesamaan antar segmen atau kalimat yang memiliki konten serupa [18]. Wibawa et al. menerapkan TextRank dengan Maximal Marginal Relevance untuk meningkatkan kualitas ringkasan. Penerapan TextRank dengan TF-IDF juga telah diteliti untuk meningkatkan performa peringkasan [19]. Sementara itu, studi oleh Novryan et al. berfokus pada analisis kebutuhan fungsional sistem informasi secara manual melalui wawancara dan observasi, namun menyimpulkan bahwa proses ini memerlukan waktu yang cukup lama dan sangat bergantung pada keahlian analis [8]. Beberapa penelitian lain mencoba memanfaatkan Natural Language Processing (NLP) untuk membantu proses analisis kebutuhan sistem [9][10] namun umumnya berfokus pada klasifikasi atau pemodelan, bukan peringkasan ekstraktif langsung dari transkrip.

Berdasarkan tinjauan literatur tersebut, pendekatan NLP terbukti mampu mendukung analisis kebutuhan, tetapi belum ada penelitian yang secara khusus menerapkan algoritma TextRank pada data hasil wawancara untuk mengidentifikasi kebutuhan fungsional sistem informasi. Oleh karena itu, penelitian ini mengusulkan penerapan algoritma TextRank dengan beberapa variasi metode, yaitu: TextRank standar, TextRank dengan pembobotan TF-IDF [17][19] serta TextRank dengan pendekatan preprocessing with stemming [20][21] untuk merangkum hasil wawancara pengguna. Melalui perbandingan ketiga metode tersebut dan evaluasi pada berbagai nilai threshold [18] penelitian ini bertujuan untuk menganalisis pengaruh metode dan konfigurasi threshold terhadap kualitas ringkasan hasil wawancara dalam mendukung proses analisis kebutuhan sistem yang lebih efisien dan terstruktur.

2. METODE



Gambar 1. Alur Tahapan

Penelitian dilakukan mengikuti alur sistematis seperti ditunjukkan pada Gambar 1. Proses dimulai dari pengumpulan data wawancara, dilanjutkan dengan tahap preprocessing untuk membersihkan data. Setelah data siap, dilakukan implementasi algoritma TextRank dengan variasi metode yang ditentukan. Hasil pemeringkatan kalimat kemudian dievaluasi menggunakan metrik precision, recall, dan F1-score dengan membandingkannya terhadap ground truth.

2.1. Data

Data penelitian dikumpulkan dari dua sumber utama. Data primer diperoleh melalui wawancara mendalam dengan stakeholder (Direktur dan Tim Teknis) di sebuah perusahaan startup teknologi mengenai pengembangan sistem CMS (Content Management System) baru. Data sekunder berupa dokumen Spesifikasi Kebutuhan Perangkat Lunak (SKPL) yang sudah ada, digunakan sebagai ground truth untuk validasi hasil. Dataset terdiri dari 21 transkrip wawancara dengan total 347 kalimat, di mana setiap transkrip memiliki rata-rata 23 kalimat. Contoh struktur data masukan dapat dilihat pada Tabel 1.

Tabel 1. Contoh Data Hasil Wawancara

No	Pertanyaan	Jawaban
1	Kenapa memutuskan menggunakan CMS baru?	Masalah utama dari sisi teknologi, cms lama develop pakai CI versi 3 udh tidak dikembangkan. Adanya versi 4 kesusahan upgrade.
2	Bagaimana kebutuhan fleksibilitasnya?	Dibutuhkan cms yang bisa punya banyak komponen yg bisa digunakan setiap saat oleh tim marketing jadi tidak perlu ke tim develop untuk ubah konten.

2.2. Preprocessing Data

Tahap preprocessing sangat krusial mengingat data wawancara bersifat lisan dan tidak formal [20][22][23]. Langkah-langkah yang dilakukan meliputi:

1. Case Folding: Mengubah seluruh huruf menjadi huruf kecil (lowercase) untuk menyeragamkan karakter [22].
2. Cleansing: Membersihkan teks dari elemen yang tidak diperlukan seperti angka, URL, simbol khusus, dan tanda baca yang berlebihan [20][23].
3. Normalisasi Kata: Mengubah kata tidak baku (slang) menjadi kata baku menggunakan kamus (contoh: "udh" menjadi "sudah", "bikin" menjadi "buat"). Ini penting untuk teks percakapan [18][24].
4. Stopword Removal: Menghapus kata-kata umum yang tidak memiliki makna signifikan (seperti "yang", "dan", "di") menggunakan pustaka Sastrawi [22][23].
5. Tokenizing & Sentence Segmentation: Memecah teks menjadi kalimat-kalimat terpisah sebagai unit analisis, kemudian memecahnya lagi menjadi token kata [25][23].

2.3. Text Rank

TextRank adalah algoritma berbasis graf untuk pemeringkatan kalimat yang diadaptasi dari konsep PageRank [14][13][26]. Prosesnya terdiri dari beberapa langkah utama:

1. Representasi Graf: Setiap kalimat hasil preprocessing direpresentasikan sebagai sebuah simpul (vertex) (V_i) dalam graf [14][27].
2. Perhitungan Similaritas: Hubungan antar kalimat dihitung menggunakan metode content overlap [14][15][22]. Semakin banyak kata yang sama antar dua kalimat, semakin kuat hubungannya. Rumus similaritas adalah:

$$Similarity(S_i, S_j) = \frac{|W_k|(W_k \in S_i \wedge W_k \in S_j)}{\log(|S_i|) + \log(|S_j|)} \quad (1)$$

Dimana ($|S_i|$) adalah jumlah kata dalam kalimat (i) [14][22].

3. **Pembangunan Graf:** Graf berbobot tidak berarah (undirected weighted graph) dibangun berdasarkan nilai similaritas tersebut [14][13][27]. Contoh graf dapat dilihat pada Gambar 2.
4. **Pemeringkatan (Ranking):** Skor kepentingan setiap kalimat (WS) dihitung secara iteratif menggunakan rumus PageRank [14][13]:

$$WS(V_i) = (1 - d) + d \times \sum_{V_j \in Adj(V_i)} \frac{w_{ji}}{\sum_{V_k \in Adj(V_j)} w_{jk}} WS(V_j) \quad (2)$$

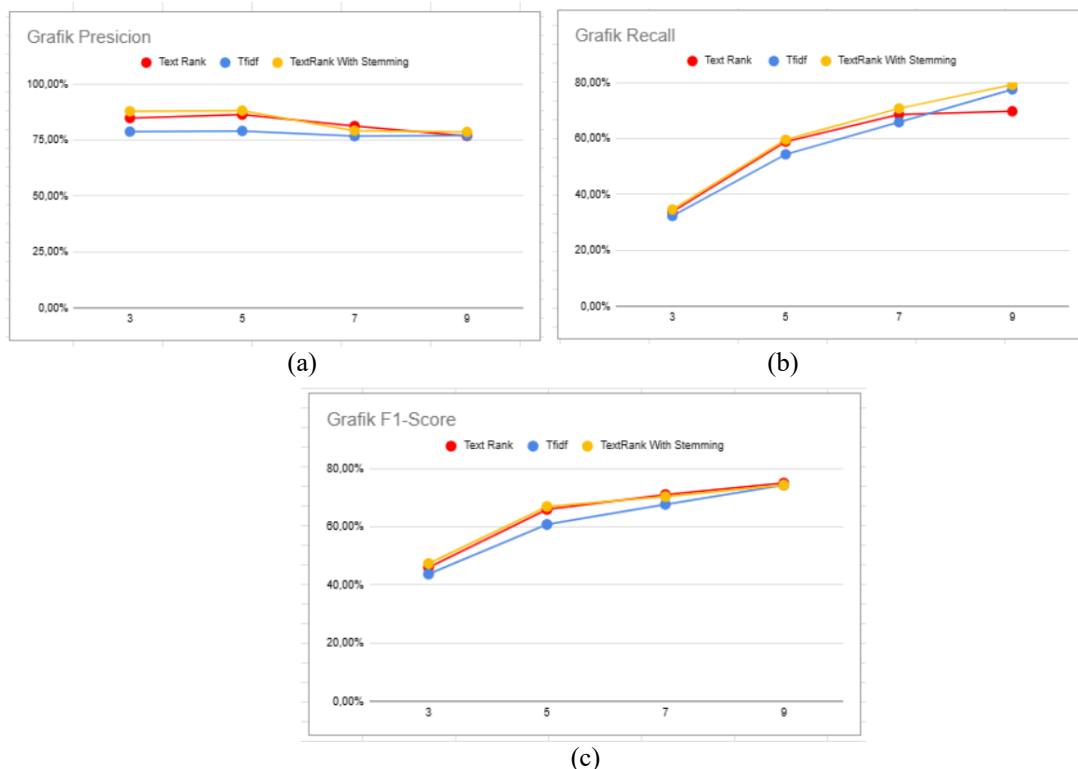
Dimana (d) adalah damping factor (biasanya 0.85) [14][16][13]. Iterasi dilakukan hingga konvergen

Tabel 2. Contoh Hasil Perangkingan Kalimat

Rank	Kalimat	Skor	Status
1	Kami membutuhkan CMS yang fleksibel sehingga tim marketing dapat mengubah konten tanpa bantuan developer.	0.0523	TP
2	Framework Laravel dipilih karena komunitas besar dan arsitektur yang lebih sederhana.	0.0509	FP
3	Sistem harus memiliki banyak komponen yang dapat digunakan kapan saja untuk menyusun halaman.	0.0498	TP

Tabel 3. Perbandingan Performa Berdasarkan Threshold

Threshold (k)	Precision (%)	Recall (%)	F1-Score (%)
3	77.27%	33.75%	46.03%
5	81.82%	58.88%	65.96%
7	79.45%	68.59%	71.05%
9	76.84%	69.74%	74.99%



Gambar 3. Grafik Perbandingan (a)Precision, (b)Recall, dan (c)F1-Score

Berdasarkan Tabel 3 dan Gambar 3, terlihat trade-off klasik antara Precision dan Recall, yang merupakan karakteristik umum dalam sistem ekstraksi informasi [28]. Pada threshold rendah ($k=3$), presisi cukup baik (77,27%) namun recall sangat rendah (33,75%), artinya banyak informasi penting yang hilang karena ringkasan terlalu singkat [11]. Pada threshold $k=5$, terjadi peningkatan recall yang signifikan (58,88%) dengan presisi yang justru meningkat (81,82%), menunjukkan bahwa kalimat-kalimat di posisi ini sangat relevan [18]. Threshold 7 memberikan keseimbangan terbaik (optimal) dengan F1-Score 71,05%, temuan ini sejalan dengan studi optimasi summarization yang menyarankan rasio kompresi moderat [16][18]. Meskipun presisi sedikit turun menjadi 79,45%, recall mencapai 68,59%. Pada titik ini, panjang ringkasan mencakup sekitar 20% dari teks asli, yang sesuai dengan standar rasio kompresi ideal (15-30%) dalam literatur peringkasan teks [11][12].

Pada threshold $k=9$, recall meningkat sedikit namun presisi mulai turun karena semakin banyak noise atau kalimat kurang relevan yang masuk ke dalam ringkasan [27][18]. Hasil perbandingan ketiga metode menunjukkan karakteristik yang berbeda, memperkuat studi sebelumnya mengenai dampak preprocessing terhadap TextRank [20][16]:

1. TextRank Standar: Paling stabil dan konsisten. Pada threshold optimal ($k=7$), menghasilkan F1-score 71,05%. Metode ini efektif menangkap hubungan antar kalimat berdasarkan pengulangan kata kunci teknis yang muncul alami dalam percakapan, mirip dengan temuan pada peringkasan teks berita [15][18].
2. TextRank + TF-IDF: Memberikan nilai precision yang cukup baik namun cenderung lebih rendah pada recall dibandingkan TextRank standar untuk data wawancara ini [17][19]. Pembobotan TF-IDF yang mengecilkan bobot kata berfrekuensi tinggi terkadang justru "menghukum" kata kunci penting yang sering diulang narasumber untuk penekanan. F1-score pada threshold 7 tercatat 67,69%.
3. TextRank + Stemming: Penerapan stemming meningkatkan recall karena menyamakan variasi kata (misal: "dikembangkan" dan "mengembangkan"), namun menurunkan precision [20][16]. Pada data percakapan yang banyak mengandung ambiguitas, stemming terkadang menghilangkan konteks spesifik. F1-score pada threshold 7 adalah 68,09%.

Secara keseluruhan, metode TextRank standar (tanpa stemming dan tanpa TF-IDF) justru memberikan kinerja terbaik untuk karakteristik data wawancara ini, karena mampu mempertahankan konteks frasa asli yang digunakan narasumber, konsisten dengan penelitian yang menekankan pentingnya fitur linguistik asli [18][24]. Analisis kualitatif menunjukkan TextRank berhasil menangkap kalimat dengan kata kunci dominan seperti "fitur", "sistem", "user", "data" [27][22]. Namun, sistem memiliki keterbatasan dalam menangkap kebutuhan implisit (tersirat) seperti yang sering ditemukan dalam elicitation [8][9]. Contohnya, pernyataan keluhan "sekarang prosesnya lambat sekali" yang menyiratkan kebutuhan "sistem harus cepat", seringkali tidak terpilih karena tidak mengandung kata kunci fungsional eksplisit [10]. Selain itu, sistem berbasis word overlap tidak dapat mengenali sinonim tanpa bantuan sumber daya semantik tambahan, yang menjadi batasan umum metode ekstraktif statistik [23].

4. KESIMPULAN

Penelitian ini berhasil menerapkan algoritma TextRank untuk mengekstraksi kebutuhan fungsional dari transkrip wawancara bahasa Indonesia yang bersifat semi-formal dan tidak terstruktur. Berdasarkan eksperimen, dapat disimpulkan bahwa TextRank efektif mengotomatisasi proses peringkasan hasil wawancara dengan F1-Score mencapai 71,05% pada konfigurasi optimal, menunjukkan metode berbasis graf dapat diadaptasi untuk domain rekayasa kebutuhan. Tahap preprocessing, khususnya normalisasi kata tidak baku (slang), memegang peranan vital dalam meningkatkan akurasi pembentukan graf pada data percakapan. Selain itu variasi threshold berpengaruh signifikan terhadap kualitas ringkasan, dengan $k=7$ terbukti sebagai konfigurasi paling optimal yang memberikan keseimbangan terbaik antara keringkasan (20% dari teks asli) dan kelengkapan informasi (recall 68,59%)

Perbandingan metode menunjukkan bahwa TextRank standar lebih cocok untuk data wawancara yang telah melalui preprocessing komprehensif dibandingkan variasi lainnya. Penambahan TF-IDF tidak selalu memberikan peningkatan signifikan untuk data wawancara yang naratif dan repetitif. Sedangkan preprocessing dengan stemming dapat meningkatkan recall namun berpotensi menurunkan presisi pada teks percakapan. Secara keseluruhan, penelitian ini memberikan kontribusi metodologis dalam penerapan NLP untuk mendukung fase elisitasi kebutuhan dalam Siklus Hidup Pengembangan Perangkat Lunak (SDLC) dengan harapan dapat meningkatkan efisiensi proses Requirements Engineering.

Untuk pengembangan penelitian selanjutnya, dapat disarankan menggunakan metode pembobotan berbasis semantik seperti Word Embedding (Word2Vec, BERT) agar sistem dapat mengenali sinonim dan konteks yang lebih kompleks. Selain itu dapat mengintegrasikan teknik klasifikasi teks (seperti SVM atau Deep Learning) sebagai tahap post-processing untuk memfilter kalimat yang benar-benar merepresentasikan kebutuhan fungsional vs non-fungsional serta menambahkan pencocokan pola berbasis aturan (rule-based pattern matching) untuk mendeteksi frasa indikator kebutuhan seperti "sistem harus" atau "pengguna dapat" guna meningkatkan presisi dan mengeksplorasi peringkasan multi-dokumen untuk menganalisis hasil wawancara dari banyak narasumber sekaligus.

REFERENSI

- [1] L. P. A. S. Tjahyanti and G. R. Utama, "Peran Analisis Kebutuhan Dalam Menciptakan Sistem Informasi Yang Responsif Dan Berkelanjutan," *Jurnal Komputer Dan Teknologi Sains (KOMTEKS)*, vol. 3, no. 2, pp. 1–11, 2024.
- [2] I. Sommerville, *Software Engineering*, 10th ed. Boston: Pearson, 2016.
- [3] R. S. Pressman and B. R. Maxim, *Software Engineering: A Practitioner's Approach*, 8th ed. New York: McGraw-Hill Education,

- 2015.
- [4] L. Setiyani and A. Gintings, "Analisis Kebutuhan Fungsional Sistem Informasi Pengelolaan Skripsi," in *Symposium Nasional Ilmiah*, 2019, pp. 405–414.
 - [5] D. Zowghi and C. Coulin, "Requirements Elicitation: A Survey of Techniques, Approaches, and Tools," in *Engineering and Managing Software Requirements*, A. Aurum and C. Wohlin, Eds., Springer, 2005, pp. 19–46.
 - [6] K. Wiegers and J. Beatty, *Software Requirements*, 3rd ed. Redmond: Microsoft Press, 2013.
 - [7] A. P. Purfini, *Analisis Sistem Informasi-Teknik Wawancara*. Bandung: Universitas Komputer Indonesia, 2010.
 - [8] M. R. Novryan, A. Suraya, and T. Sumarni, "Analisis Kebutuhan Fungsional Sistem Informasi Manajemen Berbasis Web (CV PT Meskom Bengkalis)," *Jurnal Kajian Ilmiah Interdisipliner*, vol. 8, no. 6, 2024.
 - [9] S. Nasim, Z. Zahid, and N. Sabahat, "Requirement Elicitation using Natural Language Processing," *Lahore Garrison University Research Journal of Computer Science and Information Technology*, vol. 7, no. 1, pp. 1–6, 2023.
 - [10] D. Y. Bernanda, D. N. A. Jawawi, S. A. Halim, and F. Adikara, "Natural Language Processing For Requirement Elicitation In University Using Kmeans And Meanshift Algorithm," *Baghdad Science Journal*, vol. 21, no. 2, pp. 561–567, 2024.
 - [11] M. Gambhir and V. Gupta, "Recent automatic text summarization techniques: a survey," *Artificial Intelligence Review*, vol. 47, no. 1, pp. 1–66, 2017.
 - [12] M. Allahyari et al., "Text Summarization Techniques: A Brief Survey," *International Journal of Advanced Computer Science and Applications*, vol. 8, no. 10, 2017.
 - [13] S. Brin and L. Page, "The anatomy of a large-scale hypertextual Web search engine," *Computer Networks and ISDN Systems*, vol. 30, no. 1-7, pp. 107–117, 1998.
 - [14] R. Mihalcea and P. Tarau, "TextRank: Bringing Order into Texts," in *Proceedings of the 2004 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, Barcelona, Spain, 2004, pp. 404–411.
 - [15] A. Y. S. A. Setiawan and E. Alexander, "Penguujian Algoritma TextRank Dalam Merangkum Teks," in *Seminar Nasional AMIKOM Surakarta (SEMNAS)*, 2023.
 - [16] M. Zieve et al., "Systematic TextRank Optimization in Extractive Summarization," in *International Conference Recent Advances in Natural Language Processing (RANLP)*, 2023, pp. 1274–1281.
 - [17] J. J. Sihombing, Arnita, S. I. Al Idrus, and D. Y. Niska, "Implementation of text summarization on Indonesian scientific articles using TextRank algorithm with TF-IDF web-based," *Journal of Soft Computing Exploration*, vol. 5, no. 3, pp. 310–319, 2024.
 - [18] D. Gunawan, D. Witarasyah, D. Syamsuar, A. Amalia, and R. F. Rahmat, "Observing the Performance of the TextRank Algorithm on Automatic Text Summarization for Bahasa Indonesia," *International Journal on Advanced Science, Engineering and Information Technology*, vol. 13, no. 3, 2023.
 - [19] Y. C. Gurning, S. C. Saragih, Y. Y. Lase, and J. Julham, "Perbandingan TextRank Berbasis TF-IDF dan Word2Vec dalam Peringkasan Teks Berita Bahasa Indonesia," *Jurnal Komputer Teknologi Informasi Sistem Informasi (JUKTISI)*, vol. 4, no. 2, pp. 922–928, 2025.
 - [20] A. Setiawan, Z. Abidin, and M. Imamudin, "Impact of Preprocessing on Indonesian Extractive Summarization Using LexRank, TextRank, DivRank, and Cosine Similarity," *G-Tech: Jurnal Teknologi Terapan*, vol. 9, no. 4, pp. 2311–2321, 2025.
 - [21] I. M. Suwija Putra, Y. Adiwinata, D. P. Singgih Putri, and N. P. Sutramiani, "Extractive Text Summarization of Student Essay Assignment Using Sentence Weight Features and Fuzzy C-Means," *International Journal of Artificial Intelligence Research*, vol. 5, no. 1, 2021.
 - [22] N. M. A. J. Astari, D. G. H. Divayana, and G. Indrawan, "Analisis Sentimen Dokumen Twitter Mengenai Dampak Virus Corona Menggunakan Metode Naive Bayes Classifier," *Jurnal Sistem Dan Informatika (JSI)*, vol. 15, no. 1, pp. 27–29, 2020.
 - [23] D. Jurafsky and J. H. Martin, *Speech and Language Processing*, 3rd ed. draft, 2023.
 - [24] M. Adha and M. Nasucha, "Peringkasan Teks Putusan Pengadilan Berbahasa Indonesia Menggunakan Algoritma TextRank," *SINTECH Journal*, vol. 12, 2025.
 - [25] M. Jelita, "Text Mining dengan Topic Modelling LDA dari Pertanyaan Gelar Wicara Literasi Perpustakaan Nasional RI," *Media Pustakawan*, vol. 31, no. 3, pp. 253–265, 2023.
 - [26] G. Erkan and D. R. Radev, "LexRank: Graph-based lexical centrality as salience in text summarization," *Journal of Artificial Intelligence Research*, vol. 22, pp. 457–479, 2004.
 - [27] M. R. Hadwirianto, F. Hamami, and O. N. Pratiwi, "Extractive Text Summarization Terhadap Artikel Berita Indonesia Berbasis Machine Learning," *E-Proceeding of Engineering*, vol. 11, 2024.
 - [28] D. M. W. Powers, "Evaluation: From Precision, Recall and F-Measure to ROC, Informedness, Markedness & Correlation," *Journal of Machine Learning Technologies*, vol. 2, no. 1, pp. 37–63, 2011.
 - [29] C.-Y. Lin and E. Hovy, "Automatic Evaluation of Summaries Using N-gram Co-Occurrence Statistics," in *Proceedings of HLT-NAACL 2003*, 2003.