JITET (Jurnal Informatika dan Teknik Elektro Terapan)
Vol. 14 No. 3, pISSN: 2303-0577 eISSN: 2830-7062 http://dx.doi.org/10.23960/jitet.v14i3.10766

PENGEMBANGAN AI DALAM PENYAJIAN INFORMASI BERITA WILAYAH LHOKSEUMAWE MENGGUNAKAN NER DAN *WEB SCRAPING*

Rajaul Bani Safar[1*], Rizal Tjut Adek[2], Nunsina[3]

[1,2,3]Teknik Informatika Universitas Malikussaleh; Jl. Batam, Kampus Unimal Bukit Indah, Blang Pulo, Kec. Muara Satu, Kota Lhokseumawe, Aceh 24355

Keywords:
Artificial intelligence; *Local news summarization*; *Named entity recognition*; *Semantic similarity*; *Web scraping*.

Corespondent Email:
rajaul.200170153@mhs.unimal.ac.id

Abstrak. Kebutuhan informasi lokal di Kota Lhokseumawe belum terpenuhi secara optimal karena mesin pencari berita arus utama masih bersifat umum dan minim filter spasial tingkat kota. Penelitian ini bertujuan merancang sistem pencarian dan peringkasan berita lokal berbasis *Artificial Intelligence* (AI) untuk mengatasi masalah *information overload*. Metode yang diusulkan mengintegrasikan *web scraping* (HTTPX dan Selenium) untuk ekstraksi data dinamis, *Named Entity Recognition* (NER) untuk filter entitas lokasi geografis, serta *extractive summarization* dan *semantic similarity matching* menggunakan model *Sentence Transformer*. Hasil pengujian kuantitatif terhadap metode NER menunjukkan performa yang sangat baik dalam menyaring lokasi dengan nilai *Precision* 80,49%, *Recall* 91,67%, dan F1-Score 85,72%. Secara kualitatif, ringkasan yang dihasilkan mampu menyajikan informasi esensial secara relevan, mudah dibaca, dan koheren, dengan format keluaran yang adaptif (naratif atau daftar rekomendasi) menyesuaikan kueri pengguna. Kesimpulannya, integrasi metode pemrosesan bahasa alami ini terbukti efektif menyaring dan meringkas konten berita lokal secara spesifik. Pengembangan selanjutnya dapat mengeksplorasi metode peringkasan abstraktif.

Abstract. *The need for local information in Lhokseumawe City is not optimally fulfilled because mainstream news search engines remain generalized with minimal city-level spatial filters. This study aims to design an Artificial Intelligence (AI)-based local news search and summarization system to overcome the information overload problem. The proposed method integrates web scraping (HTTPX and Selenium) for dynamic data extraction, Named Entity Recognition (NER) for geographic location entity filtering, and extractive summarization and semantic similarity matching using the Sentence Transformer model. Quantitative testing results on the NER method showed excellent performance in spatial filtering, achieving 80.49% Precision, 91.67% Recall, and an 85.72% F1-Score. Qualitatively, the generated summaries effectively presented essential information that was highly relevant, readable, and coherent, with an adaptive output format (narrative or recommendation lists) tailored to user queries. In conclusion, the integration of these natural language processing methods is proven effective in specifically filtering and summarizing local news content. Future development may explore abstractive summarization methods*

1. PENDAHULUAN

Perkembangan teknologi informasi yang pesat telah mengubah cara masyarakat global dalam mengakses dan mengonsumsi berita. Digitalisasi memang mempercepat aliran data, namun di sisi lain memicu fenomena *information overload*, yakni kondisi di mana pengguna menerima volume informasi yang jauh melebihi kapasitas pemrosesan kognitif mereka [1], [2]. Dalam konteks wilayah daerah, khususnya di Kota Lhokseumawe, ketersediaan informasi lokal yang faktual sangat krusial. Sayangnya, distribusi berita lokal di internet saat ini masih tersebar secara acak. Masalah ini diperburuk oleh mesin pencari berita arus utama yang umumnya belum dilengkapi dengan mekanisme penyaringan presisi berdasarkan entitas geografis spesifik di tingkat kota [3]. Akibatnya, masyarakat kesulitan menyaring berita terkini yang benar-benar terfokus pada wilayah Lhokseumawe.

Menjawab tantangan tersebut (*gap analysis*), pemanfaatan *Artificial Intelligence* (AI) melalui cabang *Natural Language Processing* (NLP) menawarkan solusi automasi yang andal. Penggabungan *web scraping* dengan NLP terbukti sangat efektif untuk mengekstraksi informasi berskala besar dari portal berita secara dinamis [4], [5]. Untuk memastikan keakuratan penyaringan wilayah, metode *Named Entity Recognition* (NER) diterapkan guna mendeteksi secara otomatis keberadaan entitas lokasi pada teks berita [6], [7]. Lebih jauh, untuk meminimalisasi beban membaca, metode *extractive summarization* dan *semantic similarity matching* berbasis pemodelan *Sentence-Transformer* (seperti BERT) diaplikasikan [8], [9]. Integrasi model NLP ke dalam arsitektur aplikasi *web* (seperti Flask) telah terbukti andal dalam membangun sistem cerdas yang interaktif [10].

Berdasarkan celah identifikasi tersebut, penelitian ini bertujuan untuk merancang dan mengimplementasikan sebuah sistem cerdas berbasis *web* yang mengintegrasikan *web scraping*, filter lokasi otomatis menggunakan NER, serta peringkasan teks ekstraktif dalam satu alur pemrosesan (*pipeline*). Sistem ini difokuskan secara khusus untuk menyajikan pencarian berita di wilayah Kota Lhokseumawe.

2. TINJAUAN PUSTAKA

Web scraping merupakan teknik ekstraksi data terotomasi yang digunakan untuk mengambil informasi tidak terstruktur dari halaman HTML di internet, untuk kemudian diubah menjadi data terstruktur [11]. Dalam pemrosesan teks tingkat lanjut, *Named Entity Recognition* (NER) berperan sebagai algoritma untuk mengidentifikasi dan mengklasifikasikan kata ke dalam entitas spesifik seperti lokasi, nama tokoh, atau waktu, yang sangat krusial dalam menyaring korpus berita regional [12], [13].

Sementara itu, *extractive summarization* adalah pendekatan peringkasan yang mengekstrak kalimat-kalimat paling representatif dari dokumen asli tanpa mengubah susunan sintaksisnya. Pendekatan ini dapat dioptimalkan dengan menyusun struktur 5W1H sehingga membentuk *structured summary* yang lebih komprehensif [14]. Untuk mengevaluasi kemiripan makna kueri dengan ringkasan (*semantic similarity*), representasi *sentence embedding* menggunakan model *Sentence-Transformer* (seperti BERT) diterapkan karena kemampuannya dalam memahami konteks kalimat secara dua arah (*bidirectional*) [15], [16]. Perhitungan kemiripan ini secara matematis diformulasikan menggunakan persamaan *Cosine Similarity* sebagai berikut:

$$Cos \propto = \frac{A.B}{|A||B|} = \frac{\sum_{i=1}^{n} A_i \, x \, B_i}{\sqrt{\sum_{i=1}^{n} (A_i)^2} \; x \sqrt{\sum_{i=1}^{n} (B_i)^2}} \quad (1)$$

Integrasi dari berbagai metode di atas telah dieksplorasi dalam beberapa penelitian terdahulu yang menjadi landasan arsitektur sistem ini [4], [13]. Penelitian oleh Nurchim et al. (2023) menerapkan metode NER menggunakan *spaCy* untuk klasifikasi teks berita berbahasa Indonesia. Dengan mengekstraksi entitas tempat, tokoh, dan organisasi, pendekatan tersebut terbukti handal dengan perolehan *F1-score* sebesar 89,69%. Namun, penelitian tersebut berfokus pada klasifikasi statis, bukan pada penyaringan geolokasi dinamis berbasis *web scraping* dari mesin pencari.

Selanjutnya, penelitian Pichiyan et al. (2023) membuktikan efektivitas penggabungan *web scraping* otomasi dengan teknik NLP (seperti *tokenization* dan NER) untuk mengubah teks tak terstruktur dari berbagai

sumber di internet menjadi data analitis. Pendekatan tersebut efektif untuk menangani big data berita, namun belum menyentuh aspek kompresi informasi (*summarization*) bagi pengguna akhir[4].

Terkait kompresi teks, penelitian Nurma (2025) mengembangkan sistem peringkasan teks ekstraktif pada artikel berita berbahasa Indonesia menggunakan metode BERT dan *Cosine Similarity*. Model tersebut berhasil mengukur kedekatan antar-kalimat dan meminimalkan pengulangan informasi dengan ambang batas *alpha* 0,9, menghasilkan ringkasan dengan rasio kompresi 72,7%. Penelitian ini mengambil landasan algoritma tersebut dan mengembangkannya selangkah lebih maju dengan menambahkan struktur *Semantic Filtering* adaptif yang disesuaikan dengan jenis intensi kueri pengguna (interogatif 5W1H maupun rekomendatif)[13].

3. METODE PENELITIAN

Penelitian ini menggunakan pendekatan metode campuran (*mixed method*) yang mengintegrasikan pengembangan sistem perangkat lunak (*Software Development*) dengan pengujian kinerja secara kualitatif dan kuantitatif. Arsitektur sistem dibangun menggunakan bahasa pemrograman Python versi 3.12 dengan antarmuka berbasis web menggunakan *framework* Flask. Infrastruktur pemrosesan memanfaatkan berbagai pustaka Python, antara lain: HTTPX dan Selenium untuk automasi ekstraksi web, *spaCy* untuk deteksi entitas NER, serta *Sentence-Transformers* untuk komputasi semantik ruang vektor.

3.1. Kebutuhan Perangkat Sistem

Pengembangan sistem pemrosesan bahasa alami ini membutuhkan lingkungan kerja komputasi yang memadai. Perangkat keras (*hardware*) yang digunakan mencakup prosesor AMD A9, GPU AMD Radeon, RAM 8 GB, dan penyimpanan SSD 256 GB. Pada sisi perangkat lunak (*software*), arsitektur sistem dibangun di atas sistem operasi Windows 11 menggunakan bahasa pemrograman Python versi 3.12 dan Visual Studio Code. Sistem memanfaatkan berbagai pustaka *open-source*, antara lain: *Flask* untuk *routing backend*, *HTTPX* dan *Selenium* untuk automasi ekstraksi *web*, *spaCy* untuk deteksi entitas NER, serta *Sentence-Transformers* (arsitektur *all-MiniLM-L6-v2*) untuk komputasi semantik ruang vektor [15].

3.2. Alur Pemrosesan Sistem

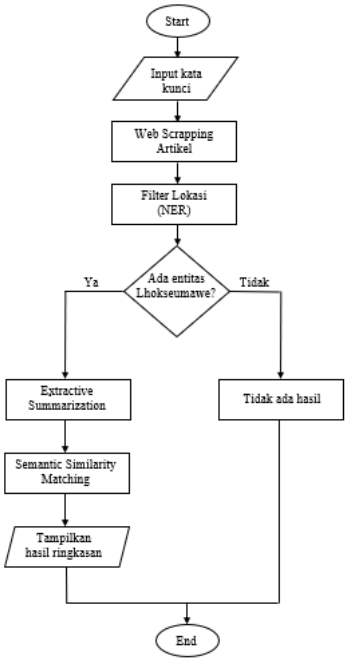
Tahap pertama adalah pengumpulan data menggunakan teknik *web scraping*. Sistem mengekstraksi konten teks dari mesin pencari menggunakan pustaka HTTPX secara *asynchronous* paralel. Apabila situs target memiliki proteksi pemuatan berbasis JavaScript atau memblokir *request* langsung, sistem secara otomatis beralih menggunakan kerangka kerja Selenium sebagai mekanisme *fallback* untuk merender halaman secara utuh.

Tahap kedua adalah penyaringan lokasi. Artikel yang berhasil diunduh divalidasi oleh modul NER menggunakan pustaka *spaCy*. Sistem hanya meloloskan dokumen yang secara spesifik memuat entitas "Lhokseumawe" dengan frekuensi kemunculan yang konsisten, guna menekan tingkat ketidakrelevanan geografis.

Tahap ketiga adalah *extractive summarization*. Sistem memilih kalimat esensial dengan mengukur tingkat kemiripan setiap kalimat terhadap representasi topik keseluruhan artikel (*anchor vector*) menggunakan komputasi *cosine similarity*. Parameter *Adaptive Top-N* diterapkan untuk mengalokasikan kuota ekstraksi kalimat secara dinamis menyesuaikan kompleksitas kueri dan volume teks.

Tahap keempat adalah *semantic similarity matching*. Vektor kalimat diukur tingkat kemiripan semantiknya secara spesifik dengan kueri pengguna menggunakan *Sentence-Transformer*. Kalimat dengan skor tertinggi dikelompokkan secara terstruktur dan disajikan dalam format adaptif: paragraf naratif untuk kueri informatif, dan daftar titik (*bullet points*) untuk kueri rekomendasi.

Kalimat dengan skor tertinggi disajikan dalam format adaptif: paragraf naratif untuk kueri informatif, dan *bullet points* untuk kueri rekomendasi. Secara visual, keseluruhan alur pemrosesan sistem ini dapat dilihat pada Gambar 1.



Gambar 1. Flowchart Alur Sistem

3.3. Skenario Evaluasi

Evaluasi kuantitatif mengukur kemampuan model NER menggunakan parameter matriks konfusi: *Precision*, *Recall*, dan *F1-Score*, yang secara berurutan dihitung menggunakan Persamaan (2), (3), dan (4) berikut:

$$Recall = \frac{True\ Positif}{True\ Positif + False\ Positif} \quad (2)$$

$$Precission = \frac{True\ Positif}{True\ Positif + False\ Positif} \quad (3)$$

$$F1 - score = 2.\frac{precision.recall}{precision + recall} \quad (4)$$

Setiap luaran artikel yang diekstraksi oleh sistem dianotasi secara manual dan diklasifikasikan ke dalam empat parameter matriks konfusi sebagai berikut: (1) True Positive (TP): Sistem meloloskan artikel dan artikel tersebut memang terverifikasi relevan membahas geolokasi "Kota Lhokseumawe" sebagai topik utama peristiwa; (2) False Positive (FP): Sistem meloloskan artikel, namun setelah diverifikasi, artikel tersebut tidak menjadikan Lhokseumawe sebagai fokus utama (misalnya hanya menyebutkan Lhokseumawe sebagai lokasi instansi penyalur bantuan, padahal bencana terjadi di wilayah lain); (3) True Negative (TN): Sistem menolak artikel dan dokumen tersebut memang tidak memiliki urgensi atau tidak relevan dengan geolokasi yang dituju; (4) False Negative (FN): Sistem menolak artikel karena gagal mencapai ambang batas frekuensi kemunculan entitas NER, namun secara manual teks tersebut dikonfirmasi relevan.

Selain itu, evaluasi kualitatif diterapkan untuk menilai mutu semantik ringkasan akhir melalui pengamatan deskriptif terhadap empat kriteria utama: tingkat relevansi (kesesuaian isi terhadap kueri), keterbacaan (kemudahan sintaksis), kelengkapan informasi esensial (cakupan 5W1H), dan koherensi antarkalimat pasca-ekstraksi.

4. HASIL DAN PEMBAHASAN

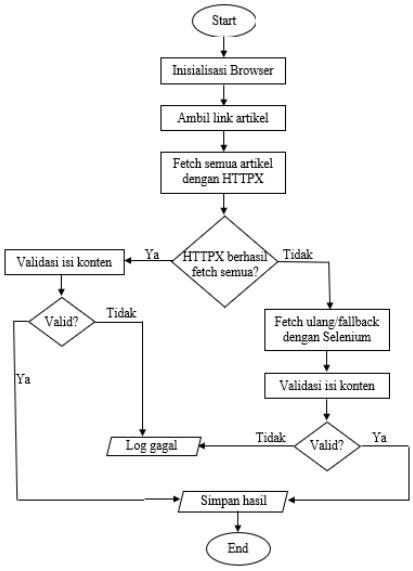
4.1. Implementasi Sistem dan Ekstraksi Data

4.1.1. Pengembangan Antarmuka dan Modul Web Scraping

Antarmuka pengguna (*frontend*) dirancang secara minimalis menyerupai aplikasi obrolan (*chat interface*) [17]. Saat pengguna mengakses sistem dan mengirimkan kata kunci (contoh: "kebakaran di lhokseumawe"), *backend* berbasis Flask menangkap masukan tersebut melalui *endpoint routing* HTTP POST [10]. Flask kemudian bertindak sebagai pengontrol utama yang memicu keseluruhan *pipeline* modul NLP.

Untuk mempersempit indeks pencarian Google pada tahap penarikan data mentah, kueri pengguna secara otomatis digabungkan dengan modifikasi *string* lokasi menjadi "kebakaran di lhokseumawe lhokseumawe". Pustaka HTTPX mengeksekusi penarikan struktur HTML dari puluhan URL secara *asynchronous* paralel untuk mempercepat *response time*. Sebagai pelapis operasional, kerangka kerja Selenium bertindak sebagai *fallback* ketika server artikel menerapkan proteksi *lazy-loading* berbasis *JavaScript* [11].

Secara prosedural, mekanisme penarikan dan validasi data mentah ini diatur melalui sebuah *pipeline* otomasi yang ketat. Gambar 3 mengilustrasikan diagram alir (*flowchart*) dari modul *web scraping* yang dikembangkan.



Gambar 2. *Flowchart* Modul *Web Scraping*

Berdasarkan Gambar 2, proses diinisialisasi dengan ekstraksi paralel menggunakan HTTPX. Setiap URL dievaluasi keberhasilannya; jika gagal akibat proteksi peladen (*server*), URL tersebut dilempar ke antrean (*queue*) Selenium untuk di-*fetch* ulang. Data HTML yang berhasil ditarik kemudian harus melewati tiga lapis validasi ketat: (1) Validasi Lokasi (menggunakan model NER), (2) Validasi Kelengkapan Isi (menolak teks yang terpotong), dan (3) Deteksi Kualitas Konten (membuang halaman *noise* seperti indeks direktori atau iklan). URL yang gagal pada salah satu validasi akan dicatat ke dalam fail *log* kegagalan berformat JSON, sedangkan teks yang lolos disimpan ke dalam struktur data terpusat untuk diproses pada modul berikutnya

4.1.2. Pengembangan Modul Filter Lokasi Geografis (NER)

Pustaka bahasa Indonesia bawaan dari *spaCy* memiliki keterbatasan dalam mengenali penamaan lokal yang tidak baku [6]. Oleh karena itu, modul filter spasial dikembangkan secara mandiri dengan merancang *blank model* ("id") yang diperkuat oleh komponen *EntityRuler*. Konfigurasi *patterns* ditambahkan secara manual ke dalam sistem untuk mengenali variasi leksikon wilayah secara spesifik. Aturan tersebut mencakup pola pengenalan teks seperti: {"label": "GPE", "pattern": "lhokseumawe"}, "Lhokseumawe", "lhok seumawe", hingga "KOTA LHOKSEUMAWE".

Lapis validasi kelengkapan isi juga diterapkan. Dokumen tidak serta-merta diloloskan walau terdapat entitas wilayah; sistem mengeksekusi logika *threshold* frekuensi, di mana artikel dianggap valid jika entitas Lhokseumawe muncul secara konsisten (minimal 3 kali) dan tidak didominasi penyebutan wilayah lain. Modul ini juga mendeteksi kualitas konten dengan membuang halaman web direktori bisnis atau iklan komersial yang jumlah karakternya tidak proporsional.

4.1.3. Pengembangan Modul Peringkasan dan Optimasi Semantik

Modul *summarizer* dikembangkan dengan mengklasifikasikan jenis kueri (*pre-processing*) menjadi kategori interogatif (Siapa, Kapan, Di mana, Mengapa, Bagaimana, Umum) dan kategori intensi (Rekomendasi, Umum). Sistem kemudian melakukan komputasi alokasi kuota ekstraksi kalimat (*Adaptive Top-N*). Parameter ini tidak bersifat kaku, melainkan dihitung secara dinamis berdasarkan interaksi empat faktor, sebagaimana dijabarkan pada Tabel 1.

Tabel 1. Parameter Perhitungan *Adaptive Top-N*

Faktor	Variabel	Cara Perhitungan	Nilai Batas
Jumlah artikel	*article_factor*	$\min(n\_artike$ $3, 1.5)$	1,0 – 1,5
Panjang kueri	*query_factor*	$\geq 6\ kata \rightarrow$ $1{,}2; \geq$ $4\ kata \rightarrow$ 1,1	1,0 – 1,2
Tipe kueri	*type_factor*	$HOW\ /$ $WH \rightarrow$ $1{,}3; lainnya \rightarrow$ 1,0	1,0 atau 1,3
Kepadatan teks	*density_factor*	> 5000 $karakter \rightarrow$ 1,2	0,9 – 1,2

Pada fase *Extractive Summarization*, representasi topik (*anchor vector*) dibentuk menggunakan model *Sentence-Transformer* [18]. Jarak kedekatan vektor setiap kalimat diukur menggunakan *Cosine Similarity* (Persamaan 1). Untuk meningkatkan presisi *Semantic Filtering*, sistem dirancang untuk

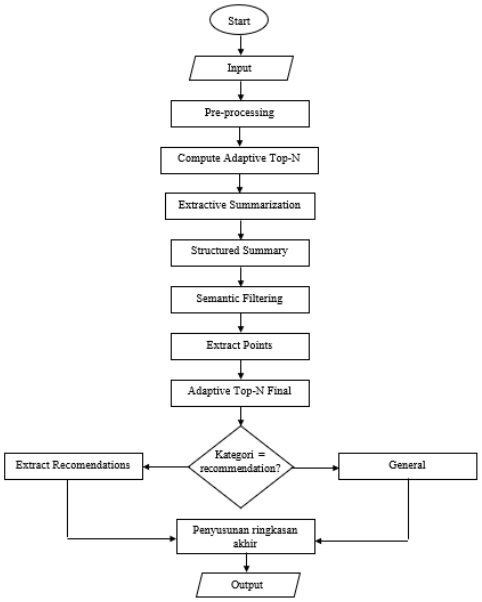
memberikan injeksi skor bonus heuristik pada kalimat yang terdeteksi mengandung jawaban langsung dari tipe kueri pengguna, sesuai dengan aturan pada Tabel 2.

Tabel 2. Aturan Bonus Skor *Semantic Filtering*

Tipe Kueri	Kondisi Kalimat	Bonus Skor
WHO	Mengandung leksikon jabatan (walikota, bupati) atau nama kapital	+0,10
WHEN	Mengandung pola tahun, hari, bulan, atau tanggal lengkap	+0,10
WHERE	Mengandung kata lokasi atau pola "Jalan / Jl."	+0,15 hingga +0,20
WHY/HOW	Mengandung kata kausalitas (akibat) atau prosedural (cara)	+0,15

Tahap terakhir dari *pipeline* ini adalah *semantic deduplication*. Mengingat sistem menarik data dari multi berita yang meliput peristiwa serupa, sistem melakukan komparasi antar-kalimat kandidat dengan ambang batas (*threshold*) kemiripan sebesar 0,85.

Kompleksitas tahapan kompresi teks ekstraktif hingga menjadi luaran yang siap dibaca oleh pengguna dipetakan secara terstruktur pada Gambar 3.



Gambar 3. Tampilan Hasil Peringkasan pada Antarmuka Sistem

Sebagaimana diilustrasikan pada Gambar 4, proses peringkasan diawali dengan fase *pre-processing* untuk mendeteksi tipe dan kategori kueri pengguna (misalnya klasifikasi pertanyaan *WHO*, *WHEN*, atau kueri bertipe *Recommendation*). Berdasarkan parameter ini, sistem menghitung kuota *Adaptive Top-N* dan melakukan *Extractive Summarization* berbasis *Cosine Similarity* terhadap *anchor vector* topik. Kalimat yang berhasil diekstraksi selanjutnya diklasifikasikan ke dalam struktur 5W1H (*Structured Summary*). Melalui *Semantic Filtering*, sistem mengeliminasi redudansi informasi dan menyaring kalimat yang memiliki skor kedekatan semantik tertinggi terhadap kueri asli. Pada tahap final (*Adaptive Top-N Final*), sistem mengeksekusi logika percabangan (*decision*); jika kueri terdeteksi sebagai rekomendasi, modul akan memicu jalur *Extract Recommendations* untuk menyusun luaran dalam format daftar titik (*bullet points*). Sebaliknya, untuk kueri umum, teks dijahit kembali menjadi paragraf naratif (*General Output*) yang koheren.

4.1.4. Simulasi Ekstraksi dan Pemeringkatan Kalimat (Studi Kasus)

Untuk mendemonstrasikan bagaimana parameter komputasi di atas bekerja, implementasi diuji menggunakan kueri "kebakaran di lhokseumawe". Kueri ini diklasifikasikan sebagai tipe UMUM (*General*). Dari hasil ekstraksi HTTPX, sistem memperoleh 3 artikel valid.

Sistem kemudian memecah ketiga artikel tersebut dan mengumpulkan 39 kalimat unik. Berdasarkan formulasi *Adaptive Top-N* (mengalikan faktor jumlah artikel, kueri, tipe, dan kepadatan teks bernilai 1.0), sistem menetapkan batas penyaringan semantik awal sebesar 10 kalimat (*semantic_top_n*). Setiap kalimat diukur *cosine similarity*-nya. Kalimat dengan skor tertinggi, misalnya *"Lhokseumawe – Sabtu, 27 Juni 2026 – BPBD Kota Lhokseumawe menangani kebakaran lahan di Desa Padang Sakti..."*, mendapat skor dasar sebesar 0,720238 tanpa bonus skor karena bertipe *General*.

Dari 10 kalimat teratas, algoritma *Semantic Deduplication* mendeteksi 1 kalimat dengan tingkat kemiripan 0,967 (di atas batas 0,85) yang merupakan versi identik dari portal berita lain. Kalimat duplikat ini dibuang, menyisakan

9 kalimat. Selanjutnya, sistem mengembalikan urutan 9 kalimat tersebut ke urutan kronologi penerbitan asli (*restore original order*). Karena nilai *Score cutoff* minimum ditetapkan 55% dari skor tertinggi ($0{,}603230 \times 0{,}55 = 0{,}331776$) dan kesembilan kalimat memiliki skor di atas 0,46, maka seluruh kalimat dipertahankan dan dijahit menjadi paragraf naratif utuh. Algoritma ini memastikan kompresi teks berjalan maksimal tanpa campur tangan (*hardcode*) manusia. Hasil akhir dari peringkasan berita yang ditampilkan kepada pengguna dapat dilihat pada Gambar 4.

Gambar 4. Tampilan Hasil Peringkasan pada Antarmuka Sistem

4.2. Analisis Kuantitatif Kinerja NER

Pengujian ditujukan untuk mengukur akurasi model NER dalam memilah artikel berdasarkan relevansi geografis Lhokseumawe. Eksperimen menggunakan korpus uji 56 artikel berita dari lima kueri dinamis. Berdasarkan anotasi manual, distribusi nilai klasifikasi yang diperoleh disajikan pada Tabel 3.

Tabel 3. Distribusi Nilai Klasifikasi Kinerja NER

Klasifikasi	Jumlah
True Positive (TP)	33
False Positive (FP)	8
True Negative (TN)	12
False Negative (FN)	3
Total Artikel Uji	56

Berdasarkan data tersebut, perhitungan performa sistem dirangkum pada Tabel 4.

Tabel 4. Metrik Evaluasi Kinerja NER

Metrik	Persentase
Precision	80,49%
Recall	91,67%
F1-Score	85,72%

Perhitungan performa sistem menghasilkan tingkat ketepatan (*Precision*) sebesar 80,49%, tingkat keberhasilan (*Recall*) sebesar 91,67%, dan akurasi harmonis keseluruhan (*F1-Score*) sebesar 85,72%. Galat *False Positive* mayoritas dipicu oleh berita yang memuat nama instansi sektoral berkedudukan di Lhokseumawe (misal: Bawaslu Kota Lhokseumawe) namun topik bencananya berfokus di kabupaten tetangga (Aceh Utara). Meski demikian, *F1-Score* 85,72% telah melampaui batas standar validitas 70%, membuktikan metode NER yang dikembangkan sangat andal untuk isolasi dokumen berbasis lokasi.

4.3. Analisis Kualitatif Kualitas Ringkasan

Hasil dan temuan dari evaluasi kualitatif pada 5 sampel kueri menunjukkan sistem memiliki adaptasi format luaran yang sangat baik. Rekapitulasi dari pengujian ini dapat dilihat pada Tabel 5.

Tabel 5. Rekapitulasi Penilaian Kualitas Ringkasan

Kueri Uji	Relevansi	Keterbacaan	Kelengkapan Informasi	Koherensi
Banjir	Baik	Baik	Baik	Baik
Wali kota	Baik	Baik	Baik	Baik
Cafe hitz	Baik	Baik	Cukup	Baik
Kebakaran	Baik	Baik	Baik	Cukup
Razia	Baik	Baik	Baik	Cukup

Untuk kueri kronologis (seperti "banjir" atau "kebakaran"), sistem mengeksekusi format naratif yang mendapatkan penilaian "Baik" pada aspek relevansi, keterbacaan, dan kelengkapan informasi (berhasil menyarikan data waktu dan lokasi gampong secara utuh).

Untuk kueri intensi rekomendasi (seperti "cafe hits"), sistem beralih mengekstrak entitas lokasi menjadi format daftar titik (*bullet points*). Kategori ini mendapat nilai "Baik" untuk relevansi, namun "Cukup" pada kelengkapan informasi karena tidak memiliki narasi detail pendukung. Penurunan koherensi menjadi "Cukup" juga wajar terjadi akibat karakteristik alami *extractive summarization* yang menempelkan potongan kalimat asli dari media berbeda tanpa parafrase ulang. Namun secara garis besar (*what else*), implementasi arsitektur ini dapat secara praktis membantu jurnalis lokal maupun masyarakat menghindari paparan informasi yang berlebihan. Penurunan koherensi menjadi "Cukup" juga wajar terjadi akibat karakteristik alami *extractive summarization* yang menempelkan potongan

kalimat asli dari media berbeda tanpa parafrase ulang. Namun secara garis besar, implementasi arsitektur NLP yang dikembangkan ini terbukti sangat praktis dan esensial dalam membantu jurnalis lokal maupun masyarakat memfilter informasi secara cepat dan menghindari paparan *information overload.*

4.4. Keterbatasan Sistem

Meskipun arsitektur AI yang dibangun telah sanggup mengotomatisasi penyajian berita lokal, terdapat beberapa batasan (*limitations*) fungsional yang teridentifikasi selama masa pengujian. (1) Dukungan Monolingual: Modul pra-pemrosesan NLP dan model *Sentence-Transformer* (*all-MiniLM-L6-v2*) dirancang dan dilatih dominan pada *corpus* berbahasa Indonesia. Sistem belum mampu menangani ekstraksi atau peringkasan pada portal berita lokal yang menggunakan bahasa daerah (misalnya Bahasa Aceh) secara penuh. (2) Limitasi Pemrosesan Dokumen: Untuk memprioritaskan efisiensi waktu (*response time*), sistem menerapkan pemotongan (*break*) jika telah menemukan 3 artikel yang valid (*True Positive*) pada pengujian NER. Hal ini menjaga kinerja server, namun membatasi keluasan informasi jika sudut pandang (*cover-both-sides*) tersebar di artikel ke-empat dan seterusnya. (3) Ketergantungan Ekstraktif: Karena basis algoritma adalah *Extractive Summarization*, sistem hanya mengisolasi dan merangkai kalimat murni dari sumber. Akibatnya, pada berita dari sumber yang berbeda, transisi leksikal antar-kalimat terkadang terasa tidak mulus (koherensi kaku) jika dibandingkan dengan metode *Abstractive Summarization* yang menulis ulang sintaksis. (4) Verifikasi Fakta Tunggal: Sistem belum dilengkapi dengan arsitektur penganalisis kebenaran entitas silang (*cross-referencing fact checker*). Jika tiga portal berita menyajikan klaim jumlah korban yang saling bertentangan, sistem akan merangkum informasi tersebut apa adanya berdasarkan skor semantik tertinggi terhadap kueri.

5. KESIMPULAN

Berdasarkan hasil perancangan dan pengujian sistem, dapat ditarik beberapa kesimpulan sebagai berikut:

a. Integrasi *web scraping* dinamis (HTTPX dan Selenium) dengan modul klasifikasi *Named Entity Recognition* (NER) sangat andal dalam mengisolasi dokumen sesuai batasan geografis target (Kota Lhokseumawe).
b. Kinerja model NER terkonfirmasi memuaskan dengan nilai *Precision* 80,49%, *Recall* 91,67%, dan *F1-Score* 85,72%.
c. Metode *extractive summarization* dan *semantic similarity matching* terbukti efektif mengompresi volume teks.
d. Secara kualitatif, ringkasan memiliki tingkat relevansi dan keterbacaan yang tinggi, serta mampu beradaptasi antara format narasi dan daftar rekomendasi spasial.
e. Sebagai pengembangan ke depan, metode peringkasan dapat dieksplorasi menggunakan *abstractive summarization* untuk memperhalus transisi koherensi antarkalimat.

UCAPAN TERIMA KASIH

Penulis mengucapkan terima kasih yang sebesar-besarnya kepada Program Studi Teknik Informatika, Fakultas Teknik, Universitas Malikussaleh atas penyediaan fasilitas akademik serta dukungan lingkungan riset yang kondusif selama proses penyelesaian penelitian ini.

DAFTAR PUSTAKA

[1] W. Kustiawan *et al.*, “Media Online Dan Perkembangannya,” *Maktab. J. Perpust. dan Inf.*, vol. 2, no. 1, p. 12, 2022.
[2] S. D. Tsabitah, D. Priharsari, and S. H. Wijoyo, “Analisis Kualitatif Implikasi Information Overload pada Pengguna Social Networking Sites (SNS),” *J. Pengemb. Teknol. Inf. dan Ilmu Komput.*, vol. 6, no. 2, pp. 802–808, 2022.
[3] R. Ramadhani and E. Setiawan, “Pengembangan Situs Web Untuk Promosi Warisan Budaya Lokal Dan Pariwisata Berbasis Berita,” *J. Teknol. Sist. Inf.*, vol. 5, no. 1, pp. 57–73, 2024.
[4] V. Pichiyan, S. Muthulingam, G. Sathar, S. Nalajala, A. Ch., and M. N. Das, “Web Scraping using Natural Language Processing: Exploiting Unstructured Text for Data Extraction and Analysis,” *Procedia Comput. Sci.*, vol. 230, pp. 193–202, 2023.
[5] E. H. Thullen, “From Information Overload to Knowledge Graphs: An Automatic Information Process Model,” Claremont Graduate University, 2023.

[6] A. H. Pradhana, E. Daniati, and M. N. Muzaki, "Penerapan Bi-LSTM Untuk Named Entity Recognition Pada Teks Bahasa Indonesia," *Indones. J. Comput. Sci. Res.*, vol. 4, no. 2, 2025.
[7] Z. Zainuddin, Mudassir, and Z. Tahir, "Entity Extraction in Indonesian Online News Using Named Entity Recognition (NER) with Hybrid Method Transformer, Word2Vec, Attention and Bi-LSTM," *Int. J. Informatics Vis.*, vol. 9, no. 3, pp. 964–973, 2025.
[8] M. R. Hadwirianto, F. Hamami, and O. N. Pratiwi, "Extractive Text Summarization Terhadap Artikel Berita Indonesia Berbasis Machine Learning," *e-Proceeding Eng.*, vol. 11, no. 4, pp. 3941–3946, 2024.
[9] N. R. Fadlilah, "Peringkasan Teks Ekstraktif Pada Artikel Berita Bahasa Indonesia Menggunakan Metode Bert dan Cosine Similarity," Universitas Islam Negeri Maulana Malik Ibrahim, Malang, 2025.
[10] F. G. Sembiring, V. F. Tambunan, S. A. Gulo, A. H. Sukma, and A. Perdana, "Pengembangan Aplikasi Pemeriksa Tata Bahasa Inggris Berbasis Web Menggunakan Flask dan Integrasi Artificial Intelligence," *JITET (Jurnal Inform. dan Tek. Elektro Ter.*, vol. 13, no. 3, pp. 387–394, 2025.
[11] C. Lotfi, S. Srinivasan, M. Ertz, and I. Latrous, "Web Scraping Techniques and Applications: A Literature Review," in *SCRS Conference Proceedings on Intelligent Systems*, 2022, pp. 381–394.
[12] S. Naseer, M. M. Ghafoor, S. B. K. Alvi, A. Kiran, S. U. Rahman, and G. Mustaza, "Named Entity Recognition (NER) in NLP Techniques, Tools Accuracy and Performance," *Pakistan J. Multidiscip. Res.*, vol. 2, no. 2, 2021.
[13] Nurchim, Nurmalitasari, and Z. A. Long, "Indonesian news classification application with named entity recognition approach," *J. INFOTEL*, vol. 15, no. 2, 2023.
[14] Q. Wang *et al.*, "SKGSum: Structured Knowledge-Guided Document Summarization," in *Findings of the Association for Computational Linguistics*, 2024, pp. 1857–1871.
[15] R. Saidi and F. Jarray, "Sentence Transformers and DistilBERT for Arabic Word Sense Induction," in *Proceedings of the International Conference on Agents and Artificial Intelligence (ICAART)*, 2023, pp. 1020–1027.
[16] A. R. P. Iwari *et al.*, "Implementasi Sentence Transformer Berbasis Bert Untuk Edukasi dan Informasi Tentang Hiv/Aids," in *Prosiding Seminar Nasional Sains dan Teknologi Seri III*, 2025, pp. 607–614.
[17] A. Septuaginta and B. J. Z. Abidin, "Perancangan User Interface Website Institut Manajemen Wiyata Indonesia," *Cakrawala -- Repos. IMWI*, vol. 3, no. 1, pp. 75–79, 2020.
[18] I. R. C. Tarumingkeng, *Natural Language Processing (NLP)*. RUDYCT e-PRESS, 2024.