PENGEMBANGAN SISTEM RINGKASAN TEKS OTOMATIS CERITA JAWA DENGAN METODE ABSTRAKTIF MENGGUNAKAN MODEL TRANSFORMER

SKRIPSI

Oleh:
AHMAD SYAFRIAN CAHYADI
NIM. 19650154

PROGRAM STUDI TEKNIK INFORMATIKA
FAKULTAS SAINS DAN TEKNOLOGI
UNIVERSITAS ISLAM NEGERI MAULANA MALIK IBRAHIM
MALANG
2026

PENGEMBANGAN SISTEM RINGKASAN TEKS OTOMATIS CERITA JAWA DENGAN METODE ABSTRAKTIF MENGGUNAKAN MODEL TRANSFORMER

SKRIPSI

Diajukan kepada:
Universitas Islam Negeri Maulana Malik Ibrahim Malang
Untuk memenuhi Salah Satu Persyaratan dalam
Memperoleh Gelar Sarjana Komputer (S.Kom)

Oleh :
AHMAD SYAFRIAN CAHYADI
NIM. 19650154

PROGRAM STUDI TEKNIK INFORMATIKA
FAKULTAS SAINS DAN TEKNOLOGI
UNIVERSITAS ISLAM NEGERI MAULANA MALIK IBRAHIM MALANG
2026

HALAMAN PERSETUJUAN

PENGEMBANGAN SISTEM RINGKASAN TEKS OTOMATIS CERITA JAWA DENGAN METODE ABSTRAKTIF MENGGUNAKAN MODEL TRANSFORMER

SKRIPSI

Oleh :
AHMAD SYAFRIAN CAHYADI
NIM. 19650154

Telah Diperiksa dan Disetujui untuk Diuji:
Tanggal: 12 Juni 2026

Pembimbing I,	Pembimbing 2,
Supriyono, M.Kom	Shoffin Nahwa Utama, M.T
NIP. 1984101 201903 1 012	NIP. 19860703 202012 1 003

Mengetahui
Ketua Program Studi Teknik Informatika
Fakultas Sains dan Teknologi
Universitas Islam Negeri Maulana Malik Ibrahim Malang

Supriyono, M. Kom
NIP. 19841010 201903 1 012

HALAMAN PENGESAHAN

PENGEMBANGAN SISTEM RINGKASAN TEKS OTOMATIS CERITA JAWA DENGAN METODE ABSTRAKTIF MENGGUNAKAN MODEL TRANSFORMER

SKRIPSI

Oleh :
AHMAD SYAFRIAN CAHYADI
NIM. 19650154

Telah Dipertahankan di Depan Dewan Penguji Skripsi
dan Dinyatakan Diterima Sebagai Salah Satu Persyaratan
Untuk Memperoleh Gelar Sarjana Komputer (S.Kom)
Tanggal: 29 Juni 2026

Susunan Dewan Penguji

Ketua Penguji	: Prof. Dr. Muhammad Faisal, M.T NIP. 19740510 200501 1 007	( )
Anggota Penguji I	: Fatchurrochman, M.Kom NIP. 19700731 200501 1 002	( )
Anggota Penguji II	: Supriyono, M.Kom, NIP. 19841010 201903 1 012	( )
Anggota Penguji III	: Shoffin Nahwa Utama, M.T NIP. 19860703 202012 1 003	( )

Mengetahui dan Mengesahkan,
Ketua Program Studi Teknik Informatika
Fakultas Sains dan Teknologi
Universitas Islam Negeri Maulana Malik Ibrahim Malang

Supriyono, M.Kom,
NIP. 19841010 201903 1 012

PERNYATAAN KEASLIAN TULISAN

Saya yang bertanda tangan di bawah ini:

Nama	: Ahmad Syafrian Cahyadi
NIM	: 19650154
Fakultas / Program Studi	: Sains dan Teknologi / Teknik Informatika
Judul Skripsi	: Pengembangan Sistem Ringkasan Teks Otomatis Cerita Jawa Dengan Metode Abstraktif Menggunakan Model Transformer

Menyatakan dengan sebenarnya bahwa Skripsi yang saya tulis ini benar-benar merupakan hasil karya saya sendiri, bukan merupakan pengambil alihan data, tulisan, atau pikiran orang lain yang saya akui sebagai hasil tulisan atau pikiran saya sendiri, kecuali dengan mencantumkan sumber cuplikan pada daftar pustaka.

Apabila dikemudian hari terbukti atau dapat dibuktikan skripsi ini merupakan hasil jiplakan, maka saya bersedia menerima sanksi atas perbuatan tersebut.

Malang, 12 Juni 2026
Yang membuat pernyataan,

Ahmad Syafrian Cahyadi
NIM.19650154

MOTTO

… "Selesai dulu, sempurna nanti."

HALAMAN PERSEMBAHAN

Dengan menyebut nama Allah Yang Maha Pengasih lagi Maha Penyayang, skripsi ini saya persembahkan untuk: Kedua orang tua saya tercinta, Ibu Imroatun Nikmah, yang tiada henti memberikan doa, kasih sayang, dan dukungan dalam setiap langkah saya. Setiap keberhasilan ini adalah buah dari pengorbanan dan keikhlasan yang tidak akan pernah dapat saya balas.

Dosen pembimbing saya, Bapak Supriyono, M.Kom dan Bapak Shoffin Nahwa Utama, M.T, serta seluruh dosen Jurusan Teknik Informatika, yang telah membimbing dan membagikan ilmunya dengan penuh kesabaran.

Sahabat dan teman-teman seperjuangan Teknik Informatika, yang telah menemani, berbagi suka dan duka, serta saling menguatkan selama menempuh studi. Almamater tercinta, Universitas Islam Negeri Maulana Malik Ibrahim Malang, tempat saya menimba ilmu dan menempa diri.

KATA PENGANTAR

Assalamu'alaikum Warahmatullahi Wabarakatuh

Segala puji dan syukur penulis panjatkan ke hadirat Allah Subhanahu wa Ta'ala, yang telah melimpahkan rahmat, taufik, dan hidayah-Nya, sehingga penulis dapat menyelesaikan skripsi yang berjudul "Pengembangan Sistem Ringkasan Teks Otomatis Cerita Jawa dengan Metode Abstraktif Menggunakan Model Transformer" dengan baik. Shalawat serta salam semoga senantiasa tercurah kepada junjungan kita Nabi Muhammad Shallallahu 'alaihi wa sallam, beserta keluarga, sahabat, dan para pengikutnya hingga akhir zaman.

Skripsi ini d*isusun sebagai salah satu syarat untuk memperoleh gelar Sarjana Komputer (S.Kom) pada Jurusan Teknik Informatika, Fakultas Sains dan Teknologi, Universitas Islam Negeri Maulana Malik Ibrahim Malang.

Penulis menyadari bahwa penyusunan skripsi ini tidak terlepas dari bantuan, bimbingan, dan dukungan berbagai pihak. Oleh karena itu, dengan segala kerendahan hati penulis mengucapkan terima kasih yang sebesar-besarnya kepada:

1. Prof. Dr. Hj. Ilfi Nur Diana, M. Si selaku rektor Universitas Islam Negeri Maulana Malik Ibrahim Malang.
2. Dr. Agus Mulyono, M. Kes selaku dekan Fakultas Sains dan Teknologi Universitas Islam Negeri Maulana Malik Ibrahim Malang.
3. Supriyono, M.Kom, selaku Dosen Pembimbing I, yang telah memberikan bimbingan, arahan, dan motivasi dengan penuh kesabaran selama penyusunan skripsi ini.

4. Shoffin Nahwa Utama, M.T, selaku Dosen Pembimbing II, atas segala bimbingan, masukan, dan ilmu yang diberikan kepada penulis.
5. Seluruh dosen dan staf Jurusan Teknik Informatika yang telah memberikan ilmu, pengalaman, dan pelayanan selama masa studi.
6. Kedua orang tua serta seluruh keluarga tercinta, yang senantiasa memberikan doa, kasih sayang, dan dukungan tiada henti.
7. Teman-teman seperjuangan Jurusan Teknik Informatika, serta semua pihak yang tidak dapat penulis sebutkan satu per satu, yang telah membantu dalam penyelesaian skripsi ini.

Malang, 12 Juni 2026

Penulis

DAFTAR ISI

DAFTAR GAMBAR

DAFTAR TABEL

DAFTAR SIMBOL

Lambang

x	Teks mentah (masukan) sebelum tahap praproses
f(x)	Fungsi praproses: normalisasi, pembersihan, dan case folding teks
X	Ringkasan kandidat (keluaran model) pada perhitungan LCS
Y	Ringkasan referensi (gold summary) pada perhitungan LCS
LCS(X, Y)	Panjang subsekuens bersama terpanjang antara X dan Y
S	Himpunan token subword $\{s_1, s_2, \ldots, s_n\}$
s_i	Token subword ke-i hasil tokenisasi SentencePiece
y	Urutan token keluaran $\{y_1, y_2, \ldots, y_n\}$
k	Beam size, yaitu jumlah kandidat pada beam search (k = 4–6)
B	Himpunan kandidat pada beam search (|B| = k)
n	Jumlah token atau panjang urutan
N	Orde n-gram pada ROUGE-N

Singkatan

API	Application Programming Interface
BART	Bidirectional and Auto-Regressive Transformers
BERT	Bidirectional Encoder Representations from Transformers
BPE	Byte Pair Encoding
CSV	Comma-Separated Values
GPU	Graphics Processing Unit
IndoBART	Indonesian Bidirectional and Auto-Regressive Transformers
LCS	Longest Common Subsequence
mBART	Multilingual Bidirectional and Auto-Regressive Transformers
mT5	Multilingual Text-to-Text Transfer Transformer
NLG	Natural Language Generation

NLP	Natural Language Processing
NLU	Natural Language Understanding
NMT	Neural Machine Translation
OOV	Out of Vocabulary
PEGASUS	Pre-training with Extracted Gap-sentences for Abstractive Summarization
ROUGE	Recall-Oriented Understudy for Gisting Evaluation
T5	Text-to-Text Transfer Transformer

ABSTRAK

Cahyadi, Ahmad Syafrian. 2026. *Pengembangan Sistem Ringkasan Teks Otomatis Cerita Jawa dengan Metode Abstraktif Menggunakan Model Transformer*. Skripsi. Program Studi Teknik Informatika, Fakultas Sains dan Teknologi, Universitas Islam Negeri Maulana Malik Ibrahim Malang. Pembimbing: (I) Supriyono, M.Kom; (II) Shoffin Nahwa Utama, M.T.

Kata kunci: peringkasan teks abstraktif, bahasa Jawa, mT5, Transformer, ROUGE, *low-resource*

Pertumbuhan teks digital menuntut teknologi peringkasan otomatis, namun penelitian peringkasan abstraktif masih didominasi bahasa Indonesia dan Inggris, sedangkan bahasa Jawa yang tergolong low-resource belum banyak dikaji. Penelitian ini bertujuan mengembangkan dan mengevaluasi sistem ringkasan teks abstraktif cerita berbahasa Jawa berbasis *Transformer*. Model yang digunakan adalah *mT5 (Multilingual Text-to-Text Transfer Transformer)* yang di *fine tune* pada dataset cerita Jawa dari *Kaggle*, dengan tokenisasi subword SentencePiece Unigram dan decoding beam search. Sistem dibangun sebagai pipeline utuh yang meliputi praproses, tokenisasi, pemodelan, decoding, dan postprocessing. Untuk menguji pengaruh strategi pemanfaatan data, dirancang tiga skenario, yaitu data murni (S1), data terjemahan otomatis/G-Translate (S2), dan data gabungan/hybrid (S3). Evaluasi dilakukan menggunakan metrik ROUGE pada data uji serta analisis kuantitative. Hasil pengujian menunjukkan skenario hybrid (S3) memperoleh performa terbaik dengan ROUGE-1 sebesar 0,8029, ROUGE-2 sebesar 0,7745, dan ROUGE-L sebesar 0,7976; mengungguli skenario G-Translate (ROUGE-1 0,6457) dan murni (ROUGE-1 0,4035). Temuan ini menunjukkan bahwa penggabungan data murni dan terjemahan otomatis memadukan keunggulan kualitas linguistik dan volume data, sehingga menjadi strategi paling efektif untuk pengembangan NLP bahasa Jawa sebagai bahasa low-resource sekaligus mendukung pelestarian bahasa daerah melalui teknologi digital.

ABSTRACT

Cahyadi, Ahmad Syafrian. 2026. *Development of an Automatic Text Summarization System for Javanese Stories Using an Abstractive Transformer-Based Method*. Undergraduate Thesis. Department of Informatics Engineering, Faculty of Science and Technology, Universitas Islam Negeri Maulana Malik Ibrahim Malang. Advisors: (I) Supriyono, M.Kom; (II) Shoffin Nahwa Utama, M.T.

Keywords: *abstractive text summarization, Javanese language, mT5, Transformer, ROUGE, low-resource*

The rapid growth of digital text increases the need for automatic summarization, yet research on abstractive summarization remains dominated by Indonesian and English, while Javanese, a low-resource language, has received little attention. This study aims to develop and evaluate a Transformer-based abstractive summarization system for Javanese stories. The model used is mT5 (Multilingual Text-to-Text Transfer Transformer), fine-tuned on a Javanese story dataset obtained from Kaggle, using SentencePiece Unigram subword tokenization and beam-search decoding. The system was built as a complete pipeline comprising preprocessing, tokenization, modeling, decoding, and postprocessing. To examine the effect of the data utilization strategy, three scenarios were designed: pure data (S1), automatically translated/G-Translate data (S2), and combined/hybrid data (S3). Evaluation was conducted using ROUGE metrics on the test set together with qualitative analysis. The results show that the hybrid scenario (S3) achieved the best performance, with ROUGE-1 of 0.8029, ROUGE-2 of 0.7745, and ROUGE-L of 0.7976, outperforming the G-Translate scenario (ROUGE-1 0.6457) and the pure scenario (ROUGE-1 0.4035). These findings indicate that combining pure and automatically translated data merges the advantages of linguistic quality and data volume, making it the most effective strategy for developing Javanese NLP as a low-resource language while supporting the preservation of regional languages through digital technology.

مستخلص البحث

جهيادي، أحمد شافريان. 2026. **تطوير نظام التلخيص النصي التلقائي للقصص الجاوية باستخدام الطريقة التجريدية بواسطة نموذج المحولات**. البحث الجامعي. قسم الهندسة المعلوماتية، كلية العلوم والتكنولوجيا بجامعة مولانا مالك إبراهيم الإسلامية الحكومية مالانج. المشرف الأول: سوبريونيو، الماجستير؛ المشرف الثاني صفين نحو أوتاما، الماجستير.

الكلمات الرئيسية: تلخيص نصي تلقائي، لغة جاوية، mT5، نماذج محولات، ROUGE، الموارد المحدودة

نمو النصوص الرقمية يتطلب تقنية التلخيص التلقائي، ومع ذلك، ما زالت الأبحاث حول التلخيص التلقائي تهيمن عليها اللغتان الإندونيسية والإنجليزية، في حين أن اللغة الجاوية التي تعتبر قليلة الموارد لم تُدرس بشكل واسع. هدف هذا البحث إلى تطوير وتقييم نظام تلخيص نصي تلقائي للقصص المكتوبة باللغة الجاوية قائم على تقنية المحولات (*Transformer*). النموذج المستخدم هو *mT5 (Multilingual Text-to-Text Transfer Transformer)* الذي تم ضبطه بدقة على مجموعة بيانات القصص الجاوية من *Kaggle*، باستخدام تقسيم الرموز الفرعية *SentencePiece Unigram* وتقنية فك الترميز *beam search*. تم بناء النظام كسلسلة متكاملة تشمل المعالجة المسبقة، وتقسيم الرموز، والنمذجة، وفك الترميز، وما بعد المعالجة. لاختبار تأثير استراتيجيات استخدام البيانات، تم تصميم ثلاثة سيناريوهات، وهي البيانات الأصيلة (S1)، البيانات المترجمة آليًا/جوجل ترجمة (S2)، والبيانات المدمجة (S3). تم إجراء التقييم باستخدام مقاييس ROUGE على بيانات الاختبار بالإضافة إلى التحليل الكمي. أظهرت نتائج الاختبار أن سيناريو مدمجة (S3) حقق أفضل أداء حيث بلغت قيم 0.8029 ROUGE-1، 0.7745 ROUGE-2، و0.7976 ROUGE-L؛ متفوقاً على سيناريو جوجل ترجمة (0.6457 ROUGE-1) والصافي (0.4035 ROUGE-1). أشارت هذه النتائج إلى أن دمج البيانات الأصيلة مع البيانات المترجمة تلقائيا يجمع بين مزايا الجودة اللغوية وحجم البيانات، مما يجعله استراتيجية فعّالة للغاية لتطوير معالجة اللغة الطبيعية للغة الجاوية كلغة ذات موارد منخفضة، مع دعم الحفاظ على اللغات المحلية من خلال التكنولوجيا الرقمية.

BAB I

PENDAHULUAN

1.1 Latar Belakang

Perkembangan teknologi informasi yang pesat telah membawa perubahan besar dalam cara manusia mengakses, memproduksi, dan mengelola informasi. Jumlah data teks yang tersedia secara digital meningkat secara signifikan setiap harinya, baik berupa artikel berita, karya sastra, unggahan media sosial, maupun dokumen elektronik lainnya. Kondisi ini menimbulkan kebutuhan akan teknologi yang mampu menyajikan informasi secara ringkas, cepat, dan efisien tanpa mengurangi makna utama. Salah satu solusi yang berkembang dalam bidang *Natural Language Processing* (NLP) adalah sistem ringkasan teks otomatis (*automatic text summarization*) yang berfungsi menyederhanakan teks panjang menjadi bentuk lebih padat namun tetap informatif.

Ringkasan teks otomatis merupakan proses menghasilkan representasi singkat dari dokumen tanpa menghilangkan inti informasi. Dalam beberapa tahun terakhir, pendekatan berbasis model Transformer, seperti BERT (*Bidirectional Encoder Representations from Transformers*) dan T5 (*Text to Text Transfer Transformer*), terbukti unggul dalam berbagai tugas NLP, termasuk *summarization*. Keunggulan utama Transformer terletak pada mekanisme *self-attention* yang mampu memahami hubungan antar kata secara lebih mendalam dibandingkan metode konvensional seperti TF-IDF, TextRank, atau LSTM. Berbagai penelitian menunjukkan bahwa model berbasis Transformer mampu menghasilkan ringkasan

yang lebih koheren, natural, dan berkualitas tinggi (Devlin et al., 2019; Raffel et al., 2019).

Walaupun perkembangan NLP terus meningkat, penelitian mengenai ringkasan teks otomatis masih didominasi oleh bahasa Indonesia dan bahasa internasional, seperti Inggris. Bahasa daerah, khususnya bahasa Jawa**,** masih sangat minim mendapat perhatian. Padahal, bahasa Jawa merupakan salah satu bahasa daerah terbesar di Indonesia dengan lebih dari 80 juta penutur. Minimnya ketersediaan sumber daya digital dan teknologi NLP untuk bahasa Jawa menjadikan pengembangan sistem ringkasan teks otomatis berbahasa Jawa sebagai upaya strategis dalam mendukung literasi digital dan pelestarian budaya lokal.

Namun, penelitian peringkasan abstraktif berbasis Transformer untuk bahasa daerah, khususnya bahasa Jawa, masih relatif terbatas, sebagian besar studi berfokus pada bahasa Indonesia dan Inggris. Di saat yang sama, bahasa Jawa tergolong *low-resource*, ketersediaan korpus digital dan data beranotasi masih terbatas, ditambah variasi tingkat tutur serta dialek yang memperumit normalisasi dan pemodelan.

Selain keterbatasan sumber daya, belum banyak penelitian yang menguji secara komparatif strategi pemanfaatan data (teks Jawa asli, hasil terjemahan otomatis, dan gabungan/*hybrid*) serta dampaknya terhadap kualitas dan naturalitas ringkasan. Perbedaan distribusi panjang teks terutama pada data terjemahan yang cenderung lebih panjang juga menuntut teknik pemotongan *input* seperti *chunking*/*sliding-window* agar informasi tidak terpangkas oleh batas *token* model,

tetapi efektivitasnya untuk peringkasan bahasa Jawa masih jarang dievaluasi secara sistematis.

Urgensi menyajikan informasi secara ringkas dan mudah dipahami ini juga selaras dengan nilai-nilai Al-Qur'an. Allah Subhanahu wa Ta'ala berfirman dalam QS. An-Nahl ayat 125:

اُدْعُ اِلٰى سَبِيْلِ رَبِّكَ بِالْحِكْمَةِ وَالْمَوْعِظَةِ الْحَسَنَةِ وَجَادِلْهُمْ بِالَّتِيْ هِيَ اَحْسَنُۗ اِنَّ رَبَّكَ هُوَ اَعْلَمُ بِمَنْ ضَلَّ عَنْ سَبِيْلِهٖۚ وَهُوَ اَعْلَمُ بِالْمُهْتَدِيْنَ

"Serulah (manusia) kepada jalan Tuhanmu dengan hikmah dan pelajaran yang baik, dan bantahlah mereka dengan cara yang terbaik…" (*QS. An-Nahl: 125)*

Menurut Ibnu Katsir (Katsir, 2008) ayat ini mengajarkan bahwa penyampaian informasi harus dilakukan dengan hikmah, yaitu ucapan yang benar, jelas, dan sesuai dengan akal serta syariat. Sedangkan al-mau'izhah al-hasanah berarti nasihat yang lembut sehingga mudah diterima, dan mujadalah billati hiya ahsan berarti berdialog dengan cara yang sopan dan penuh etika. Tafsir Al-Maraghi (Al-Maraghi, 1993) juga menekankan bahwa hikmah merupakan perkataan yang padat namun kuat penjelasannya, sedangkan As-Sa'di menjelaskan bahwa ayat ini mengandung prinsip penyampaian pesan dengan cara yang paling efektif, santun, dan tidak menyesatkan. Tafsir Al-Muyassar menambahkan bahwa penyampaian pesan harus disesuaikan dengan kondisi pendengar agar tujuan komunikasi (Al-Muyassar, 2006)

Berdasarkan penafsiran para ulama tersebut, dapat dipahami bahwa Al-Qur'an memberikan pedoman penting tentang bagaimana informasi seharusnya disampaikan: ringkas, jelas, bijaksana, dan mudah dipahami. Prinsip ini sangat relevan dengan tujuan penelitian ini dalam mengembangkan sistem ringkasan teks

otomatis, yang berupaya menyederhanakan informasi namun tetap menjaga akurasi dan makna utama.

Selain prinsip penyampaian informasi yang ringkas dan bijaksana, urgensi pengembangan teknologi bahasa Jawa dalam penelitian ini juga memiliki landasan dalam Al-Qur'an, khususnya yang berkaitan dengan keberagaman bahasa sebagai tanda kebesaran Allah. Allah Subhanahu wa Ta'ala berfirman dalam QS. Ar-Rum ayat 22:

وَمِنْ اٰيٰتِهٖ خَلْقُ السَّمٰوٰتِ وَالْاَرْضِ وَاخْتِلَافُ اَلْسِنَتِكُمْ وَاَلْوَانِكُمْۗ اِنَّ فِيْ ذٰلِكَ لَاٰيٰتٍ لِّلْعٰلِمِيْنَ

"*Dan di antara tanda-tanda (kebesaran)-Nya ialah penciptaan langit dan bumi serta perbedaan bahasamu dan warna kulitmu. Sesungguhnya pada yang demikian itu benar-benar terdapat tanda-tanda bagi orang-orang yang mengetahui."(QS. Ar-Rum: 22.)*

Menurut Ibn Katsir, ayat ini menjelaskan bahwa perbedaan bahasa yang dimiliki umat manusia, seperti bahasa Arab, Romawi, dan Persia, merupakan salah satu tanda kekuasaan Allah, dan tidak ada yang mengajarkan keragaman tersebut kecuali Allah. Tafsir As-Sa'di menambahkan bahwa meskipun seluruh manusia berasal dari satu sumber dan memiliki organ pengucapan yang sama, tidak akan dijumpai dua suara atau dua warna kulit yang benar-benar serupa; perbedaan ini menunjukkan kesempurnaan kekuasaan, kehendak, dan kasih sayang Allah kepada hamba-Nya (As-Sa’di, 2014). Sementara itu, Tafsir Al-Muyassar menegaskan bahwa keragaman bahasa dan warna kulit tersebut mengandung pelajaran (ibrah) bagi siapa pun yang memiliki ilmu dan kemampuan memahami (bashirah). Penutup ayat "li al-'alimin" (bagi orang-orang yang mengetahui) mengisyaratkan bahwa

keragaman bahasa menyimpan banyak hikmah yang hanya dapat digali oleh mereka yang menelaahnya secara mendalam.

Berdasarkan penafsiran tersebut, dapat dipahami bahwa keberagaman bahasa, termasuk bahasa Jawa dengan tingkat tutur (ngoko, krama, dan krama inggil) serta variasi dialeknya, merupakan bagian dari tanda kebesaran Allah yang patut dijaga dan dilestarikan. Upaya mengembangkan teknologi Natural Language Processing untuk bahasa Jawa yang tergolong low-resource menjadi salah satu ikhtiar nyata dalam menjaga dan memuliakan keragaman bahasa ciptaan Allah agar tetap lestari di era digital. Dengan demikian, penelitian ini memiliki nilai teknis dan keilmuan sekaligus sejalan dengan nilai-nilai Islam dalam melestarikan keragaman bahasa sebagai salah satu tanda kebesaran-Nya.

Dengan demikian, penelitian ini berfokus pada pengembangan sistem ringkasan teks otomatis untuk cerita berbahasa Jawa menggunakan metode *Transformer*, seperti *BERT* dan *T5*. Sistem ini diharapkan mampu menghasilkan ringkasan yang relevan, ringkas, dan tetap mempertahankan konteks cerita. Selain itu, penelitian ini diharapkan menjadi kontribusi awal dalam pengembangan teknologi NLP untuk bahasa daerah serta mendukung pelestarian bahasa dan budaya lokal melalui pemanfaatan teknologi modern.

1.2 Rumusan Masalah

Berdasarkan latar belakang tersebut, rumusan masalah penelitian ini adalah: bagaimana mengevaluasi sistem ringkasan teks bahasa Jawa berbasis Transformer agar mampu menghasilkan ringkasan yang berkualitas

1.3 Batasan Masalah

Agar penelitian lebih terarah, maka batasan masalah ditetapkan sebagai berikut:

1. Teks yang diringkas hanya berupa teks cerita berbahasa Jawa.
2. Model yang digunakan dibatasi pada arsitektur *Transformer*, seperti *BERT* dan *T5*.
3. *Dataset* yang digunakan adalah korpus bahasa Jawa yang tersedia secara publik atau dikumpulkan secara etis.
4. Penelitian berfokus pada metode *abstractive summarization*, bukan *extractive.*
5. Evaluasi menggunakan metrik otomatis seperti ROUGE dan penilaian manual kesesuaian makna.
6. Implementasi sistem hanya sampai tahap pengembangan model dan antarmuka sederhana, tidak mencakup *deployment* skala luas.

1.4 Tujuan Masalah

Penelitian ini bertujuan untuk:

1. Mengembangkan sistem ringkasan teks otomatis untuk cerita berbahasa Jawa menggunakan metode *Transformer.*
2. Menganalisis performa model *Transformer* dalam menghasilkan ringkasan teks berbahasa Jawa.
3. Menilai kualitas ringkasan dari segi relevansi, koherensi, dan keterbacaan.
4. Memberikan kontribusi dalam pengembangan NLP untuk bahasa daerah berstatus Low Resource.

1.5 Manfaat Penelitian

1. Menambah literatur terkait penerapan model *Transformer* pada bahasa daerah.
2. Menjadi referensi bagi penelitian lanjutan di bidang *text summarization* dan NLP multibahasa.
3. Menyediakan alat bantu untuk merangkum teks berbahasa Jawa secara otomatis.
4. Mendukung pelestarian bahasa Jawa melalui teknologi digital.
5. Membantu akademisi, pelajar, dan peneliti dalam memahami isi teks Jawa dengan lebih efisien.

BAB II

STUDI PUSTAKA

2.1 Pemrosesan Bahasa Alami

Natural Language Processing (NLP) merupakan bidang dalam kecerdasan buatan yang berfokus pada bagaimana mesin memahami dan memproses bahasa manusia melalui pendekatan komputasional (Hirschberg & Manning, n.d.). NLP digunakan untuk menjembatani komunikasi antara manusia dan sistem komputer agar bahasa dapat diproses secara otomatis dan efisien. Seiring perkembangan teknologi, NLP memanfaatkan model berbasis *deep learning* untuk menghasilkan representasi bahasa yang lebih akurat. (Young et al., 2018) menjelaskan bahwa perkembangan neural network telah meningkatkan kemampuan mesin dalam memahami struktur sintaktis dan semantik secara lebih mendalam. NLP juga mendasari berbagai aplikasi seperti analisis sentimen, terjemahan mesin, dan sistem ringkasan teks otomatis. Kemampuan model untuk menangkap konteks telah diperkuat dengan kemunculan arsitektur modern seperti Transformer. Dalam konteks penggunaan praktis, NLP memungkinkan komputer melakukan tugas berbasis bahasa dengan tingkat akurasi yang mendekati manusia. Hal ini menjadi semakin penting dalam era digital di mana data teks terus meningkat secara eksponensial. Oleh karena itu, NLP memainkan peran strategis dalam pengembangan teknologi berbasis bahasa di berbagai bidang.

Ruang lingkup NLP meliputi dua komponen utama, yaitu *Natural Language Understanding (NLU)* dan *Natural Language Generation (NLG)*, yang bersama

sama membentuk fondasi pemprosesan bahasa modern (Gatt & Krahmer, 2018). NLU menitikberatkan pada kemampuan sistem dalam memahami makna teks melalui analisis sintaktis dan semantik. (Gatt & Krahmer, 2018) menegaskan bahwa pemahaman konteks merupakan elemen penting untuk menghasilkan interpretasi bahasa yang tepat. Di sisi lain, NLG berfungsi untuk menghasilkan teks baru yang alami dan dapat dipahami manusia. Teknologi seperti peringkasan otomatis, teks deskriptif otomatis, dan dialog cerdas merupakan hasil langsung dari pengembangan NLG. Selain itu, *speech processing* merupakan bagian penting NLP yang mencakup konversi suara ke teks menggunakan model akustik yang dilatih dengan data berskala besar. (Hirschberg & Manning, n.d.) menyatakan bahwa integrasi antara linguistik dan model akustik modern meningkatkan performa sistem pengenalan suara. *Text mining* turut memperluas ruang lingkup NLP melalui proses ekstraksi informasi dari kumpulan teks berukuran besar. Dengan demikian, NLP merupakan bidang multidisipliner yang terus berkembang mengikuti peningkatan kebutuhan teknologi bahasa.

Bahasa daerah seperti Jawa, Sunda, dan Madura termasuk dalam kategori *low resource languages* karena minimnya ketersediaan korpus digital yang memadai dan terbatasnya sumber daya linguistik yang telah dianotasi dengan baik (Kudugunta et al., 2019). Keterbatasan ini menyebabkan model NLP sulit mencapai performa optimal karena jumlah data pelatihan tidak mencukupi untuk mempelajari pola bahasa secara akurat. (Kudugunta et al., 2019) menjelaskan bahwa bahasa *low resource* memerlukan pendekatan khusus seperti multilingual transfer learning agar dapat memperoleh informasi dari bahasa ber resource besar. Tantangan semakin

kompleks ketika bahasa daerah memiliki variasi dialek yang luas, yang membuat model sulit menangkap pola linguistik secara konsisten. Ketidakkonsistenan kosakata antarwilayah memperbesar risiko kesalahan dalam proses pelatihan model. Selain itu, dataset yang tidak seragam sering menurunkan stabilitas model ketika diuji pada domain yang berbeda dari data latih. (Adelani et al., 2022) menunjukkan bahwa penggunaan model pralatih multibahasa dapat membantu mengurangi keterbatasan data pada bahasa minoritas. Meskipun begitu, adaptasi model pralatih tetap memerlukan proses *fine tuning* yang terarah agar mampu mempertahankan karakteristik linguistik lokal. Oleh karena itu, pengembangan NLP pada bahasa daerah membutuhkan pendekatan metodologis yang fleksibel dan strategi pemanfaatan model multibahasa untuk mengatasi keterbatasan sumber daya.

Bahasa Jawa tetap tergolong sebagai bahasa low resource karena jumlah korpus digitalnya sangat terbatas meskipun jumlah penuturnya besar (Aji et al., 2022). Bahasa ini memiliki tingkat tutur ngoko, krama, dan krama inggil yang menambah kompleksitas pemrosesan linguistik. Menurut (Aji et al., 2022), variasi tingkat tutur dalam bahasa Jawa menyebabkan perbedaan signifikan pada struktur dan kosakata. Selain itu, variasi dialek antarwilayah membuat proses normalisasi bahasa menjadi lebih menantang. Ketidakkonsistenan data menghambat pelatihan model NLP karena model sulit menggeneralisasi pola bahasa. Penelitian menunjukkan bahwa POS tagging bahasa Jawa masih menghadapi tantangan akurasi akibat keterbatasan dataset. Untuk memperkuat pengembangan NLP Jawa, (Alfina et al., 2022) memperkenalkan *dataset gold standard* yang mencakup tokenisasi, POS tagging,

morfologi, dan *dependency parsing*. Dataset tersebut menjadi terobosan penting dalam peningkatan kualitas sumber daya bahasa Jawa. Oleh karena itu, pembangunan korpus terstruktur sangat penting untuk memperluas kemampuan NLP pada bahasa Jawa.

Pengembangan NLP untuk bahasa Jawa memiliki urgensi besar dalam mendukung digitalisasi bahasa daerah dan pelestarian budaya. Salah satu teknologi yang berpotensi besar adalah sistem peringkasan teks otomatis berbasis Transformer. (Lewis et al., 2020) menunjukkan bahwa model BART mampu menghasilkan ringkasan abstraktif yang lebih natural dibandingkan metode ekstraktif. PEGASUS yang dikembangkan oleh (Zhang et al., 2020) juga menunjukkan performa unggul dalam peringkasan teks dengan memahami konteks secara mendalam. Prinsip prinsip model tersebut dapat diterapkan pada bahasa low resource melalui teknik *fine tuning*. Studi (Aji et al., 2022) menegaskan bahwa teknik *fine tuning* sangat efektif untuk bahasa daerah di Indonesia, termasuk bahasa Jawa. Selain itu, penelitian (Ghiffari et al., 2023) menunjukkan bahwa transfer learning berhasil meningkatkan hasil dependency parsing bahasa Jawa. Hal ini menunjukkan bahwa model Transformer dapat beradaptasi dengan baik pada bahasa daerah yang memiliki keterbatasan data. Oleh karena itu, pengembangan ringkasan teks otomatis bahasa Jawa berbasis Transformer memiliki potensi besar untuk meningkatkan aksesibilitas informasi dan teknologi bagi penutur Jawa.

2.2 Ringkasan Text Otomatis (Automatic Text Summarization)

Ringkasan teks merupakan proses mereduksi suatu dokumen menjadi bentuk yang lebih pendek tanpa menghilangkan informasi utama di dalamnya (Ladhak et al., 2020). Proses peringkasan ini bertujuan untuk membantu pembaca memahami inti informasi dengan lebih cepat dan efisien. Dalam konteks komputasi, ringkasan teks dipandang sebagai tugas yang membutuhkan pemahaman menyeluruh terhadap isi, struktur, dan konteks dokumen. Menurut (Zhong et al., 2020), ringkasan otomatis memerlukan representasi semantik yang kuat agar dapat menangkap makna penting dari teks. Peringkasan teks menjadi semakin relevan karena pertumbuhan data digital yang sangat besar dalam berbagai domain. Dengan perkembangan teknologi pemrosesan bahasa alami, ringkasan otomatis dapat dilakukan menggunakan metode statistik, pembelajaran mesin, atau model deep learning. Model modern seperti Transformer telah meningkatkan kualitas ringkasan dengan memahami hubungan panjang dalam teks (Zhong et al., 2020). Tujuan utama peringkasan bukan hanya mempersingkat teks tetapi juga mempertahankan koherensi dan relevansi informasi. Oleh sebab itu, pengembangan sistem ringkasan teks otomatis memerlukan pendekatan yang sistematis serta evaluasi yang ketat.

Ringkasan teks dibagi menjadi dua kategori utama, yaitu *extractive summarization* dan *abstractive summarization* (Lewis et al., 2020). *Extractive summarization* dilakukan dengan memilih kalimat atau frasa penting langsung dari teks asli. Metode ini bersifat sederhana karena tidak memerlukan pembangkitan bahasa baru. Namun, hasil ringkasan *extractive* cenderung kurang natural karena bersifat menyalin kalimat apa adanya dari dokumen (Zhang et al., 2020).

Sebaliknya, *abstractive summarization* menghasilkan kalimat baru yang menyarikan inti informasi dengan gaya bahasa yang lebih alami. Menurut (Zhang et al., 2020), metode abstraktif lebih menyerupai cara manusia meringkas teks karena mampu melakukan parafrase. Tantangan utama dalam *abstractive summarization* adalah kebutuhan komputasi yang besar dan data pelatihan yang berkualitas tinggi. Model abstraktif modern biasanya menggunakan arsitektur Transformer untuk memahami konteks secara lebih dalam. Dengan demikian, pemilihan metode peringkasan sangat dipengaruhi oleh kebutuhan aplikasi dan ketersediaan sumber daya.

Tahapan peringkasan teks dimulai dengan *preprocessing,* yaitu proses tokenisasi, normalisasi, dan pembersihan teks dari elemen yang tidak relevan (Wang et al., 2021). Tahap ini penting untuk memastikan data berada dalam format yang sesuai untuk pemodelan. Setelah *preprocessing*, dilakukan ekstraksi fitur untuk menangkap informasi semantik dan sintaktis yang penting. Menurut (Ladhak et al., 2020), representasi fitur yang baik sangat berpengaruh pada kualitas ringkasan yang dihasilkan. Tahap berikutnya adalah pemodelan yang dapat menggunakan metode klasik hingga model modern berbasis *Transformer*. Model kemudian digunakan untuk menghasilkan ringkasan yang mencerminkan inti informasi dokumen asli (Zhang et al., 2020). Setelah ringkasan dihasilkan, tahap evaluasi dilakukan menggunakan metrik seperti ROUGE yang menghitung kesamaan antara ringkasan mesin dan ringkasan acuan. Beberapa penelitian juga menggunakan penilaian manual untuk mengukur koherensi dan keterbacaan

ringkasan (Lewis et al., 2020). Dengan demikian, peringkasan teks merupakan rangkaian proses yang melibatkan banyak langkah terstruktur dan sistematis.

Abstractive summarization memiliki keunggulan dalam menghasilkan ringkasan yang lebih natural dan koheren dibandingkan metode extractive (Zhang et al., 2020). Hal ini terjadi karena *abstractive summarization* mampu membangun kalimat baru berdasarkan pemahaman konteks mendalam. Akan tetapi, metode abstraktif memerlukan jumlah data pelatihan yang besar agar model dapat memahami variasi bahasa dengan baik. (Lewis et al., 2020) menunjukkan bahwa model abstraktif seperti BART membutuhkan sumber daya komputasi yang tinggi. Di sisi lain, *extractive summarization* lebih mudah diterapkan karena hanya memilih kalimat penting dari teks asli. Metode ini membutuhkan komputasi lebih kecil dan bekerja baik untuk teks informatif yang strukturnya jelas. Namun, *extractive summarization* sering kali menghasilkan ringkasan yang kurang mengalir secara alami (Ladhak et al., 2020). Pilihan metode ringkasan harus disesuaikan dengan tujuan aplikasi dan ketersediaan data. Oleh karena itu, pengembang sistem ringkasan harus mempertimbangkan *trade off* antara kualitas ringkasan dan efisiensi komputasi.

Penggunaan ringkasan teks memiliki implikasi penting dalam berbagai bidang yang menghadapi lonjakan informasi digital. Sistem ringkasan otomatis membantu mempercepat pemahaman dokumen panjang dalam waktu singkat (Zhang et al., 2020). Dalam bidang pendidikan, ringkasan membantu siswa memahami materi yang kompleks dengan lebih cepat. Dalam konteks bisnis, ringkasan digunakan untuk merangkum laporan, analisis data, atau dokumen legal.

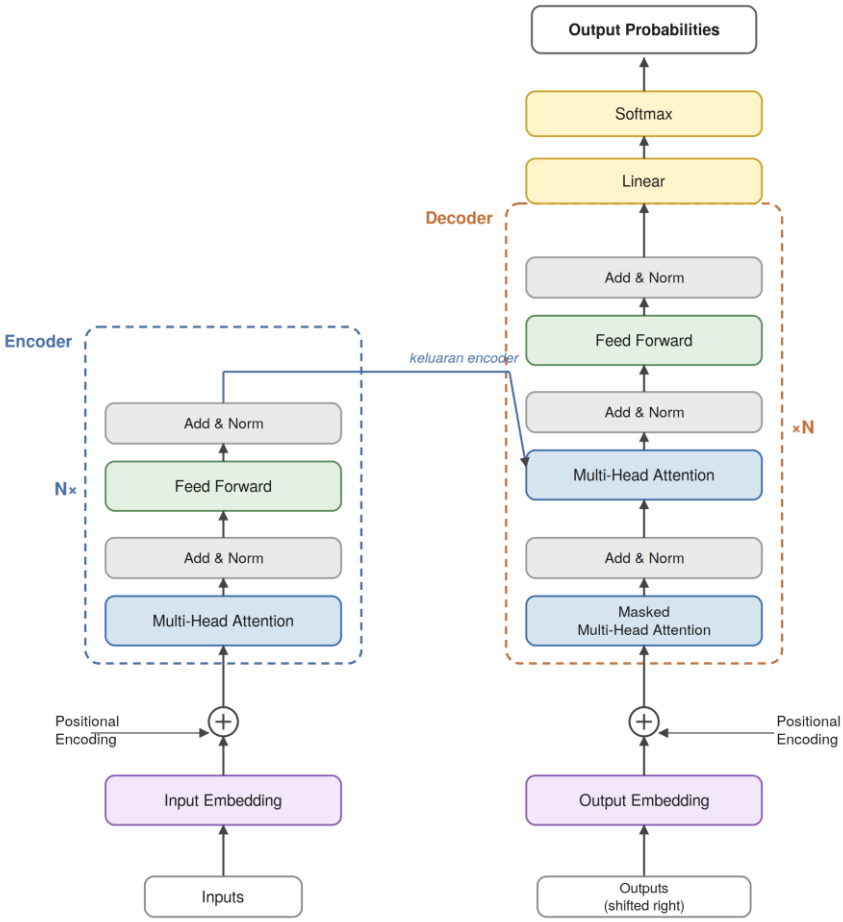
Menurut (Lewis et al., 2020), ringkasan otomatis juga bermanfaat untuk memfilter informasi relevan dalam sistem pencarian. Ringkasan berbasis abstraktif berpotensi memberikan wawasan yang lebih mendalam karena mampu melakukan generalisasi informasi. Sementara itu, ringkasan berbasis ekstraktif tetap berguna untuk tugas tugas yang memerlukan akurasi struktural dari teks asli. Dengan perkembangan model *Transformer,* kualitas ringkasan semakin mendekati hasil ringkasan manusia (Zhang et al., 2020). Oleh karena itu, peringkasan teks menjadi teknologi kunci dalam pengelolaan informasi modern.

2.3. Model Transformer

Arsitektur Transformer pertama kali diperkenalkan oleh (Vaswani et al., 2017) melalui makalah berjudul Attention Is All You Need yang dipublikasikan dalam konferensi NeurIPS. Publikasi tersebut menjadi titik penting dalam perkembangan NLP karena menggantikan model berbasis urutan seperti RNN dan LSTM. (Vaswani et al., 2017) menunjukkan bahwa mekanisme self attention dapat memproses hubungan antar kata secara paralel tanpa memerlukan struktur berurutan. Keberhasilan Transformer membuatnya menjadi model dominan dalam berbagai tugas NLP seperti terjemahan, summarization, dan model generatif. Penelitian lanjutan menjelaskan bahwa Transformer mampu menangani dependensi jarak jauh dengan lebih efektif dibandingkan arsitektur sebelumnya (Young et al., 2018). Selain itu, efisiensi komputasi Transformer membuatnya cocok untuk dilatih menggunakan perangkat modern berbasis GPU dan TPU. Arsitektur ini juga menjadi dasar bagi model besar seperti BERT, GPT, T5, dan PEGASUS. Keunggulan tersebut menjadikan Transformer sebagai paradigma baru dalam

pemrosesan bahasa alami. Dengan demikian, sejarah Transformer telah menciptakan fondasi kuat bagi inovasi NLP modern.

Arsitektur Transformer terdiri dari dua komponen utama, yaitu *encoder* dan *decoder*, yang bekerja sama untuk memproses input dan menghasilkan output (Vaswani et al., 2017). Encoder berfungsi membangun representasi konteks melalui self attention dan feed forward neural network. Decoder kemudian menggunakan informasi dari encoder serta masked attention untuk menghasilkan keluaran secara terstruktur. Setiap blok encoder dan decoder memiliki mekanisme multi head attention yang memperkaya pemahaman konteks. Struktur ini memungkinkan Transformer mempelajari hubungan antar kata tanpa perlu menjalankan proses berurutan seperti pada RNN. (Raganato et al., 2020) menjelaskan bahwa pola *self attention* dalam encoder memungkinkan model menangkap perbedaan sintaktis dan semantis secara lebih fleksibel. Feed forward layer dalam setiap blok membantu memperkuat representasi yang telah diproses oleh mekanisme perhatian. Jumlah lapisan pada encoder dan decoder dapat ditingkatkan untuk menghasilkan model yang lebih dalam dan lebih kuat. Dengan desain modular ini, Transformer menjadi model yang sangat adaptif untuk berbagai tugas NLP. Struktur lengkap arsitektur Transformer yang terdiri atas tumpukan encoder dan decoder tersebut ditunjukkan pada Gambar 2.1.



Gambar 2. 1 Arsitektur Model Transformer

Inti dari mekanisme perhatian pada Transformer adalah self attention, yaitu proses yang menghitung keterkaitan setiap token terhadap seluruh token lain dalam satu urutan. Pada mekanisme ini, setiap token direpresentasikan ke dalam tiga vektor, yaitu query (Q), key (K), dan value (V). Bobot perhatian diperoleh dengan menghitung perkalian skalar (dot product) antara query dan key, menormalkannya dengan akar dimensi key, kemudian menerapkan fungsi softmax sebelum hasilnya dikalikan dengan value. Proses tersebut dikenal sebagai *scaled dot product attention.*

Selain sublapisan perhatian, setiap blok *encoder* maupun *decoder* dilengkapi dengan *position wise feed forward network* serta mekanisme *residual connection* dan *layer normalization* (Add & Norm). *Residual connection* menambahkan kembali masukan suatu sublapisan ke keluarannya sehingga aliran gradien lebih lancar, sedangkan layer normalization menstabilkan distribusi nilai antar lapisan. Kombinasi komponen inilah yang memungkinkan Transformer dilatih secara dalam dan stabil (Vaswani et al., 2017).

Multi head attention merupakan komponen kunci dalam Transformer yang memungkinkan model memperhatikan beberapa konteks secara simultan (Vaswani et al., 2017). Mekanisme ini membagi representasi kata menjadi beberapa "kepala" perhatian untuk menangkap berbagai jenis relasi linguistik. Setiap kepala attention bekerja secara paralel sehingga dapat mempelajari informasi sintaktis dan semantis dari sudut pandang berbeda. (Raganato et al., 2020) menyatakan bahwa pola *self attention* yang beragam meningkatkan kemampuan model untuk memahami struktur kalimat yang kompleks. Dengan menggabungkan hasil dari beberapa kepala, *multi head attention* memperkuat representasi konteks. Mekanisme ini juga meningkatkan fleksibilitas model dalam memproses kalimat panjang dengan dependensi jauh. (Baniata et al., 2022) menjelaskan bahwa *multi head attention* dapat menangkap relasi yang tidak dapat ditangani oleh model sekuensial. Implementasi *multi head attention* juga mempercepat pelatihan karena dapat diparalelkan dengan mudah. Oleh karena itu, komponen ini menjadi salah satu faktor utama di balik keberhasilan Transformer dalam berbagai aplikasi NLP.

Positional encoding digunakan dalam Transformer untuk memberikan informasi urutan kata karena model ini tidak menggunakan mekanisme berurutan seperti RNN (Vaswani et al., 2017). Fungsi sinus dan kosinus dengan berbagai frekuensi digunakan sebagai representasi posisi token dalam teks. Pendekatan ini memungkinkan model memanfaatkan informasi posisi tanpa mengurangi kemampuan paralelisasi. (Raganato et al., 2023) menyatakan bahwa positional encoding sangat penting untuk mempertahankan struktur sintaksis dalam kalimat. Tanpa positional encoding, Transformer kesulitan menangkap hubungan linear antara token. Selain itu, beberapa penelitian mengembangkan positional encoding berbasis pembelajaran (*learned position embedding*). (Baniata et al., 2022) menunjukkan bahwa posisi yang di encode dapat memperbaiki performa model pada tugas terjemahan bahasa. Positional encoding juga membantu model dalam menangani input panjang dengan struktur kompleks. Dengan demikian, komponen ini menjadi bagian penting dari arsitektur Transformer modern.

Arsitektur Transformer memiliki kontribusi besar terhadap perkembangan sistem summarization modern. Mekanisme *self attention* memungkinkan model menangkap bagian penting dari dokumen lebih efektif dibanding metode tradisional. Model berbasis Transformer seperti BART telah menunjukkan performa unggul dalam tugas ringkasan abstraktif (Lewis et al., 2020). Selain itu, PEGASUS memperkenalkan strategi pre training khusus untuk summarization yang menghasilkan kalimat lebih natural (Zhang et al., 2020). Penelitian menunjukkan bahwa kemampuan *parallel processing* memungkinkan Transformer memproses dokumen panjang dengan lebih efisien. (Raganato et al., 2020)

menjelaskan bahwa pola self attention membantu mengidentifikasi hubungan semantik yang relevan untuk proses ringkasan. Model Transformer juga mudah diadaptasi ke bahasa *low resource* menggunakan metode *fine tuning*. Fleksibilitas ini membuat Transformer banyak digunakan dalam riset summarization lintas bahasa. Oleh karena itu, Transformer menjadi pilihan utama dalam pengembangan ringkasan teks otomatis modern.

Salah satu varian Transformer yang relevan dengan penelitian ini adalah T5 (*Text-to-Text Transfer Transformer*) yang memperlakukan seluruh tugas pemrosesan bahasa sebagai persoalan text-to-text, yakni mengubah teks masukan menjadi teks keluaran dengan format instruksi tertentu (Raffel et al., 2020). Dalam kerangka ini, tugas peringkasan dinyatakan dengan menambahkan awalan instruksi seperti "summarize: " pada teks sumber, sehingga model menghasilkan ringkasan sebagai teks targetnya. Pengembangan lebih lanjut menghasilkan mT5 (*Multilingual T5*) yang dilatih pada korpus mC4 mencakup 101 bahasa, termasuk Bahasa Indonesia dan sejumlah bahasa daerah (Xue et al., 2021). Karena dilatih secara multibahasa, mT5 berpotensi melakukan transfer pengetahuan lintas bahasa sehingga sesuai untuk Bahasa Jawa yang tergolong low resource. Atas dasar pertimbangan tersebut, penelitian ini menggunakan mT5 sebagai model utama untuk menghasilkan ringkasan abstraktif teks cerita berbahasa Jawa.

2.4. Evaluasi Ringkasan Text

Evaluasi kuantitatif pada sistem ringkasan teks otomatis umumnya dilakukan menggunakan metrik ROUGE yang diperkenalkan oleh (Lin, n.d.). ROUGE digunakan untuk mengukur tingkat kesamaan antara ringkasan mesin dan

ringkasan acuan yang dibuat manusia. ROUGE N merupakan variasi yang menghitung kesamaan berdasarkan n gram, biasanya unigram atau bigram. Penggunaan ROUGE 1 dan ROUGE 2 membantu mengukur akurasi dalam menangkap kata dan frasa penting. Sementara itu, ROUGE L menilai kesesuaian pola urutan kata berdasarkan *longest common subsequence* (LCS). (Lin, n.d.) menjelaskan bahwa ROUGE L sangat bermanfaat untuk ringkasan yang mempertahankan struktur kalimat. Selain itu, ROUGE S atau skip bigram digunakan untuk melihat hubungan antar kata yang tidak berurutan. Penelitian Paulus et al. (2018) menunjukkan bahwa kombinasi ROUGE 1, ROUGE 2, dan ROUGE L memberikan gambaran paling komprehensif tentang kualitas sistem summarization. Oleh karena itu, ROUGE menjadi standar utama dalam evaluasi performa ringkasan otomatis.

ROUGE N adalah metrik evaluasi yang menghitung kecocokan n gram antara ringkasan sistem dan ringkasan referensi (Lin, n.d.). ROUGE 1 mengukur kecocokan kata per kata, sementara ROUGE 2 mengukur kecocokan bigram. Nilai ROUGE N sering digunakan untuk menilai tingkat kemampuan sistem dalam menjaga informasi inti. Sebaliknya, ROUGE L menilai kesamaan berdasarkan *longest common subsequence*, sehingga dapat menangkap urutan kata yang paling panjang dan konsisten. (Lin, n.d.)menunjukkan bahwa ROUGE L lebih sensitif terhadap struktur kalimat dibanding ROUGE N. ROUGE S atau skip bigram menghitung pasangan kata yang muncul dalam jarak tertentu tetapi tetap mempertahankan urutan. Paulus et al. (2018) menggunakan ROUGE S untuk menilai fleksibilitas model dalam menghasilkan ringkasan kreatif. Kombinasi

ROUGE N, ROUGE L, dan ROUGE S memberikan perspektif yang lengkap terkait kelengkapan, koherensi, dan struktur ringkasan. Dengan demikian, penerapan beberapa jenis ROUGE penting untuk evaluasi yang seimbang.

Selain evaluasi kuantitatif, sistem ringkasan juga perlu dinilai melalui evaluasi kualitatif untuk memahami kualitas *semantic output*. Evaluasi kualitatif biasanya menilai kesesuaian makna antara ringkasan sistem dan dokumen asli. Menurut (Kryscinski et al., 2020), kesesuaian makna penting untuk menghindari kesalahan semantik yang dapat mengubah informasi. Koherensi ringkasan juga menjadi aspek penting karena menentukan keterhubungan antar kalimat dalam ringkasan. Penilaian koherensi memastikan bahwa ringkasan tidak hanya akurat secara fakta tetapi juga mudah dipahami. Selain itu, keterbacaan ringkasan menjadi indikator apakah ringkasan nyaman dibaca oleh manusia. (Fabbri et al., 2021) menjelaskan bahwa ringkasan yang baik harus memiliki alur yang jelas dan bebas dari kalimat tidak natural. Evaluasi kualitatif sering dilakukan oleh ahli bahasa atau annotator terlatih. Dengan demikian, evaluasi kualitatif memberikan gambaran yang lebih komprehensif tentang kualitas ringkasan dibanding hanya menggunakan metrik otomatis.

Kesesuaian makna mengukur sejauh mana ringkasan mempertahankan pesan utama dari dokumen asli (Kryscinski et al., 2020). Aspek ini sangat penting karena ringkasan yang tidak sesuai makna dapat menyesatkan pembaca. Koherensi ringkasan juga mencakup kelancaran alur dan hubungan antar kalimat. (Fabbri et al., 2021) menemukan bahwa banyak model abstraktif menghasilkan kalimat yang benar secara lokal namun kurang koheren secara global. Keterbacaan ringkasan

terkait kejelasan kalimat, struktur bahasa, dan tingkat kemudahan pemahaman. Evaluasi keterbacaan membantu mengidentifikasi apakah model menghasilkan teks yang natural dan tidak kaku. Penelitian (Lin, n.d.) menunjukkan bahwa ringkasan dengan keterbacaan rendah sering berasal dari model yang terlalu mengandalkan probabilitas n gram. Oleh karena itu, evaluasi kuantitatif harus mencakup ketiga aspek tersebut untuk memberikan penilaian menyeluruh. Ketiganya saling melengkapi untuk mengukur kualitas ringkasan otomatis.

Perbandingan hasil ringkasan model dengan ringkasan manual merupakan langkah penting dalam menilai kualitas sistem summarization. Ringkasan manual umumnya dijadikan acuan karena mencerminkan interpretasi manusia terhadap informasi penting. (Lin, n.d.) menjelaskan bahwa ROUGE dikembangkan berdasarkan konsep kecocokan dengan ringkasan referensi manusia. Dalam penelitian (Fabbri et al., 2021), ringkasan model sering kali lebih pendek tetapi kurang mempertahankan konteks dibandingkan ringkasan manual. Evaluasi ini juga membantu menemukan pola kesalahan seperti hilangnya informasi penting atau penambahan detail yang tidak relevan. (Kryscinski et al., 2020) menunjukkan bahwa model abstraktif cenderung memunculkan kesalahan halusinasi, yaitu informasi baru yang tidak ada dalam dokumen. Dengan membandingkan kedua jenis ringkasan, peneliti dapat mengidentifikasi kelebihan dan keterbatasan model. Perbandingan ini juga berguna untuk mengukur tingkat kemiripan gaya bahasa. Oleh karena itu, ringkasan manual tetap menjadi standar emas dalam evaluasi sistem ringkasan otomatis.

2.5. Dataset Cerita Jawa

Dataset untuk pengembangan sistem ringkasan teks bahasa Jawa dapat diperoleh dari berbagai sumber yang menyediakan teks naratif maupun deskriptif. Cerita rakyat Jawa merupakan salah satu sumber utama karena memiliki struktur naratif yang jelas dan kaya kosakata budaya. Selain itu, cerita pendek berbahasa Jawa menyediakan variasi gaya bahasa dari penulis modern yang relevan untuk pelatihan model NLP. Website berbahasa Jawa seperti portal berita atau blog komunitas dapat menjadi sumber teks tambahan yang bersifat kontemporer. (Alfina et al., 2022) juga menghasilkan dataset gold standard untuk tokenisasi dan analisis sintaksis bahasa Jawa. Dataset JWIKI dapat dijadikan sumber tambahan untuk memperkaya variasi domain teks yang tersedia. Dengan demikian, kombinasi beberapa sumber dataset menghasilkan korpus yang lebih representatif untuk keperluan summarization. Dataset yang digunakan dalam penelitian ini diperoleh dari laman Kaggle berjudul GPT2 Javanese Dataset (Lutfiandri, 2025) yang diakses pada bulan November 2025. Dataset tersebut terdiri atas dua jenis berkas, yaitu cerita-jawa-murni yang berisi teks berbahasa Jawa asli, serta cerita-jawa-gtrans yang berisi teks cerita berbahasa Indonesia dari laman cerpenmu.com yang telah diterjemahkan secara otomatis ke Bahasa Jawa menggunakan Google Translate. Karena dataset tersebut pada dasarnya merupakan korpus untuk pemodelan bahasa (language modeling) dan tidak menyertakan ringkasan acuan, ringkasan referensi pada penelitian ini dibentuk pada tahap preprocessing. Sebagai gambaran konkret,

Tabel 2.1 menyajikan contoh data pada dataset cerita Jawa murni, sedangkan Tabel 2.2 menunjukkan contoh data pada dataset hasil terjemahan .

Tabel 2. 1 Contoh data pada dataset cerita Jawa murni.

No	Cuplikan Teks Cerita (text_jw)	Jumlah Karakter
1	Yen mangkono dadi dalane dedagangan saka tanah Indhu Ngarep menyang tanah Cina pancen ngliwati Tanah Jawa. Ing Tahun 435 malah wis ana utusane Ratu Jawa Kulon menyang tanah Cina, ngaturake pisungsung menyang Maharaja ing tanah Cina, minangka tandhaning tetepungan sarta murih gampang lakuning …	630
2	Omah omah padunungane wong Jawa, ya wis memper karo omah omahe wong jaman saiki, apayon atep utawa eduk lan wis nganggo kepang. Enggone dedagangan lelawanan karo wong Cina; barang dedagangane kayata: emas, salaka, gading lan liya liyane. Cina cina ngarani nagara iku Kalinga, besuke, ya diarani …	1963

Tabel 2. 2 Contoh data pada dataset hasil terjemahan otomatis (g-translate).

No	Teks Sumber (Bahasa Indonesia)	Hasil Terjemahan ke Bahasa Jawa (text_jw)
1	Aku tak ingat apa yang membuat kita terbaring di sini, sampai suster mengatakan “Bapak tadi tertabrak rombongan pejabat.” Setelah mendengar kalimat itu, aku terbelalak melihat ke arahmu. Awalnya aku resah melihatmu dengan mata …	Aku ora kelingan apa sing nggawe kita ngapusi ing kene, nganti perawat ngomong "Bapak ditabrak rombongan pejabat." Sakwise krungu tembung-tembung iku, mripatku njedhul marang kowe. Awale aku gelisah nyawang kowe karo nutup mata …
2	Di tahun modern ini ada seorang pemuda yang bisa dibilang gila akan ilmu-ilmu kebatinan, serta selalu ingin mendapatkan Kharohmah dari para pejuang agama terdahulu seperti Wali Songo. Akan tetapi di negeri ini tak hanya Wali …	Ing taun modern iki, ana wong enom sing bisa diarani gila babagan mistik, lan tansah kepengin entuk Kharohmah saka mantan pejuang agama kaya Wali Songo. Nanging, ing negara iki ora mung Wali Songo sing dadi tokoh utama ing …

Bahasa Jawa memiliki karakteristik linguistik unik yang mempengaruhi proses pemrosesan bahasa alami. Salah satu ciri paling penting adalah adanya tingkatan bahasa seperti ngoko, krama, dan krama inggil yang mencerminkan hubungan sosial antara penutur. Robson (2019) menjelaskan bahwa tingkatan bahasa tersebut menghasilkan variasi kosakata yang signifikan meskipun merujuk pada makna yang sama. Struktur kalimat bahasa Jawa juga berbeda dari bahasa

Indonesia, khususnya dalam penggunaan partikel dan urutan kata. Variasi dialek antarwilayah seperti Jawa Tengah, Jawa Timur, dan Yogyakarta menambah keragaman bentuk bahasa. Cohn (2016) menunjukkan bahwa variasi dialek mempengaruhi fonologi maupun morfologi bahasa Jawa. Selain itu, banyak kata dalam bahasa Jawa memiliki makna yang berubah sesuai konteks sosial dan situasi percakapan. Aspek sosiolinguistik ini menuntut model NLP untuk memahami konteks pragmatis secara lebih mendalam. Oleh sebab itu, pengembangan NLP bahasa Jawa memerlukan pemahaman menyeluruh terhadap variasi linguistik yang ada.

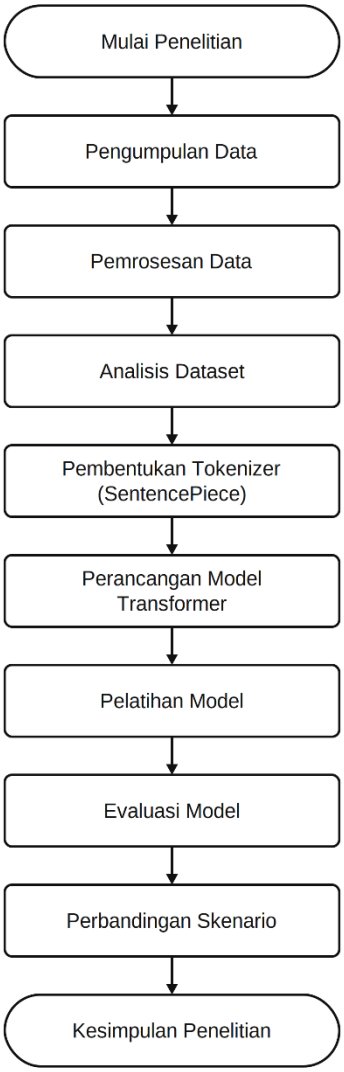
BAB III

DESAIN DAN IMPLEMENTASI

3.1. Alur Penelitian

Alur penelitian merupakan fondasi metodologis yang memastikan proses pengembangan sistem ringkasan teks berbahasa Jawa berjalan secara sistematis dan dapat dipertanggungjawabkan secara akademik. Penelitian ini ditempatkan dalam ranah Natural Language Processing (NLP) dengan fokus pada bahasa daerah yang tergolong low-resource, sehingga penyusunan alur penelitian harus mempertimbangkan keterbatasan data dan karakteristik linguistik yang kompleks. Tantangan seperti variasi dialek, tingkat tutur, dan kurangnya korpus digital berpengaruh pada tahapan penyiapan dataset. Oleh karena itu, setiap tahap penelitian tidak hanya dirancang linear, tetapi juga iteratif agar penyesuaian dapat dilakukan saat ditemukan gangguan kualitas data. Proses penelitian dibagi menjadi tahap awal, tahap inti, dan tahap evaluasi akhir untuk menjaga keterurutan logis pengembangan sistem. Tahap awal berperan dalam membangun dasar teoritis yang kuat melalui studi literatur dan perumusan masalah. Tahap inti mencakup implementasi teknis seperti preprocessing, tokenisasi, dan pelatihan model Transformer. Tahap evaluasi memastikan hasil eksperimen dapat diukur secara objektif menggunakan metrik kuantitatif dan kuantitatif. Dengan rancangan ini, alur penelitian tidak hanya menghasilkan model ringkasan teks, tetapi juga dokumentasi ilmiah yang replikatif.

Secara garis besar, alur penelitian terdiri dari enam tahap utama yang saling terkait dan berurutan. Tahap pertama adalah identifikasi masalah dan studi literatur untuk menentukan urgensi penelitian dan menemukan celah penelitian yang relevan dalam bidang summarization bahasa daerah. Tahap kedua adalah pengumpulan dataset yang mencakup teks Jawa asli, terjemahan otomatis, dan gabungan keduanya sebagai pendekatan augmentasi data. Tahap ketiga ialah preprocessing dan analisis dataset untuk meningkatkan kualitas teks sebelum diproses model, termasuk normalisasi ejaan, cleaning karakter non standar, dan analisis distribusi panjang teks. Tahap keempat adalah perancangan sistem yang meliputi penyusunan pipeline teknis, permodelan arsitektur Transformer, dan pembuatan diagram UML sebagai dokumentasi struktural. Tahap kelima adalah implementasi dan pelatihan model, termasuk fine-tuning terhadap mT5 serta pengaturan hyperparameter agar model memperoleh performa optimal. Tahap keenam adalah evaluasi hasil menggunakan metrik ROUGE dan analisis kuantitatif untuk menilai koherensi serta kesesuaian makna ringkasan. Seluruh tahapan tersebut membentuk alur yang saling mendukung dan menghasilkan luaran penelitian berupa sistem peringkasan teks otomatis. Dengan alur ini, penelitian dapat dilakukan secara terstruktur, terukur, dan dapat direplikasi oleh peneliti selanjutnya.



Gambar 3. 1 Bagan Alur Penelitian .

3.2. Pengumpulan Data

Pengumpulan data pada penelitian ini dilakukan dengan memanfaatkan sumber dataset publik yang tersedia pada platform *Kaggle*. Dataset yang digunakan berisi kumpulan teks cerita berbahasa Jawa yang telah dipublikasikan oleh kontributor secara terbuka serta dilengkapi dengan lisensi penggunaan untuk

keperluan penelitian dan pengembangan sistem *Natural Language Processing (NLP)*. Pemilihan Kaggle sebagai sumber data didasarkan pada ketersediaan dataset yang terstruktur, mudah diakses, serta mendukung format *machine readable* sehingga mempermudah proses pra-pemrosesan dan pelatihan model.

Proses pengambilan dataset dilakukan dengan mengunduh berkas dalam format *compressed* (.zip) yang kemudian diekstraksi menjadi file teks dan CSV untuk diolah pada tahap selanjutnya. Setelah data berhasil diunduh, peneliti melakukan proses verifikasi struktur data untuk memastikan kesesuaian format dengan kebutuhan penelitian, seperti keselarasan kolom teks sumber, identifikasi bahasa, dan panjang dokumen. Selanjutnya, data yang tidak relevan seperti duplikasi teks, konten non-naratif, serta baris kosong dihapus untuk memastikan kualitas data tetap terjaga. Tahapan ini dilakukan untuk meminimalisir *noise* pada data yang dapat memengaruhi performa model *text summarization* yang dikembangkan.

Dataset yang telah melalui tahap *cleaning* kemudian disimpan dalam format CSV agar kompatibel dengan pipeline pemrosesan menggunakan Python. Selanjutnya, data dibagi menjadi tiga subset, yaitu data latih (*training data*), data validasi (*validation data*), dan data uji (*testing data*) dengan proporsi pembagian 80%, 10%, dan 10%. Pembagian ini bertujuan untuk memastikan model tidak hanya mampu melakukan *fit* terhadap data pelatihan, tetapi juga dapat melakukan generalisasi terhadap teks baru. Dengan prosedur ini, data yang diperoleh dari

Kaggle siap digunakan pada tahap pra-pemrosesan dan pelatihan model Transformer untuk merangkum teks cerita berbahasa Jawa.

Tabel 3. 1 Dataset cerita jawa.

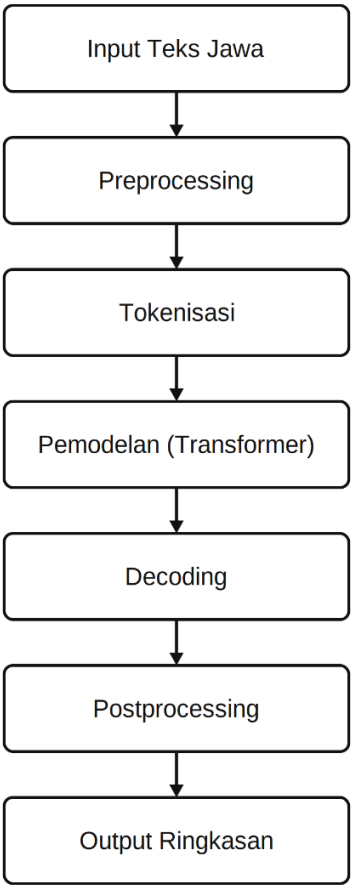
No	Nama File	Folder	Deskripsi
1	cerita-jawa-murni.csv	Original	Dataset utama berisi teks Jawa asli
2	cerita-jawa-gtrans.csv	Original	Teks Jawa hasil terjemahan otomatis
3	cerita-jawa-murni.csv	Train	Data pelatihan model
4	cerita-jawa-gtrans.csv	Train	Data pelatihan tambahan
5	cerita-jawa-murni.csv	Test	Data uji model
6	cerita-jawa-gtrans.csv	Test	Data uji tambahan

3.3 Desain Sistem

Desain sistem merupakan tahapan yang bertujuan untuk merumuskan bagaimana sistem ringkasan teks otomatis dibangun secara konseptual dan teknis. Pada penelitian ini, desain sistem mencakup tiga aspek utama, yaitu: (1) arsitektur sistem secara keseluruhan, (2) rancangan komponen dan modul sistem dalam bentuk UML, serta (3) implementasi arsitektur model Transformer untuk mendukung proses ringkasan teks berbahasa Jawa.

Perancangan sistem dibuat sedemikian rupa agar modular, mudah dikembangkan, dan kompatibel dengan teknologi NLP modern. Selain itu, desain

diselaraskan dengan karakteristik dataset Bahasa Jawa yang bersifat low-resource dan tidak seragam.



Gambar 3. 2 Arsitektur Sistem

3.3.1. Preprocessing

Preprocessing merupakan tahapan awal yang bertujuan menyiapkan data teks berbahasa Jawa agar sesuai dengan kebutuhan pemodelan berbasis Transformer. Tahap ini penting karena Bahasa Jawa memiliki variasi dialek, tingkat tutur (ngoko, madya, krama), serta fenomena *code-switching* dengan Bahasa Indonesia yang dapat mengganggu proses tokenisasi dan pemodelan. Dataset yang digunakan terdiri atas tiga kategori, yaitu dataset murni yang berasal dari teks Jawa

alami, dataset hasil terjemahan otomatis (*g-trans*), serta dataset hybrid yang merupakan gabungan keduanya. Setiap dataset diproses menggunakan pipeline yang sama agar memiliki format standar dan konsisten. Proses *preprocessing* meliputi normalisasi ejaan, pembersihan karakter, segmentasi kalimat, filtering dokumen terlalu pendek, dan penyeragaman bentuk huruf. Hasil *preprocessing* digunakan untuk menentukan panjang dokumen optimal serta parameter pemotongan (*chunking*) sebelum dilatih menggunakan model Transformer. Tahapan ini penting untuk menghindari pemangkasan informasi akibat batas maksimum panjang token model. Selain itu, hasil *preprocessing* dijadikan dasar analisis awal terhadap pola distribusi teks dalam dataset. Dengan tahapan ini, dataset siap diproses ke tahap tokenisasi dan pemodelan.

Tabel 3. 2 Tahapan preprocessing dataset.

Tahap	Proses	Output
Normalisasi ejaan	menyeragamkan variasi penulisan	teks konsisten
Cleaning karakter	menghapus URL, HTML, angka, simbol	teks bersih
Case folding	lowercase kecuali nama diri	format standar
Segmentasi kalimat	pisahkan berdasarkan tanda baca	struktur rapi
Filtering code-mixing	hapus bahasa Indonesia	dataset fokus
Filtering panjang teks	hapus teks < 300 karakter	dokumen valid

Rumus Normalisasi Token

Jika *x* adalah teks mentah dan fungsi preprocessing adalah $\boldsymbol{f}(\boldsymbol{x})$, maka:

$$\boldsymbol{f}(\boldsymbol{x}) = \text{normalize}(\text{clean}(\text{lowercase}(\boldsymbol{x})))$$

Contoh persoalan:

$$\boldsymbol{f}(\text{"NgGih!!, aku??"}) = \text{"nggih aku"}$$

Analisis Kuantitatif Dataset

Untuk memperjelas tahapan-tahapan tersebut, berikut diberikan contoh penerapan preprocessing pada sepenggal teks nyata dari dataset cerita Jawa murni. Misalkan diambil cuplikan teks mentah berikut :

Yen mangkono dadi dalane dedagangan saka tanah Indhu Ngarep menyang tanah Cina pancen ngliwati Tanah Jawa. Ing Tahun 435 malah wis ana utusane Ratu Jawa Kulon menyang tanah Cina, ngaturake pisungsung menyang Maharaja ing tanah Cina, minangka tandhaning tetepungan sarta …

Tahap cleaning karakter menghapus angka (435) serta tanda baca seperti koma dan titik, termasuk URL dan tag HTML apabila ada, sehingga teks menjadi :

Yen mangkono dadi dalane dedagangan saka tanah Indhu Ngarep menyang tanah Cina pancen ngliwati Tanah Jawa Ing Tahun malah wis ana utusane Ratu Jawa Kulon menyang tanah Cina ngaturake pisungsung menyang Maharaja ing tanah Cina minangka tandhaning tetepungan sarta …

Selanjutnya, tahap *case folding* menyeragamfkan seluruh huruf menjadi huruf kecil:

yen mangkono dadi dalane dedagangan saka tanah indhu ngarep menyang tanah cina pancen ngliwati tanah jawa ing tahun malah wis ana utusane ratu jawa kulon menyang tanah cina ngaturake pisungsung menyang maharaja ing tanah cina minangka tandhaning tetepungan sarta …

Tahap normalisasi kemudian merapikan spasi ganda menjadi spasi tunggal serta menghapus spasi di awal dan akhir teks. Pada tahap akhir, dokumen hasil pembersihan disaring berdasarkan panjangnya, yaitu hanya dokumen dengan panjang minimal 300 karakter yang dipertahankan. Pada contoh ini, dokumen utuh berisi 630 karakter sehingga melewati ambang tersebut dan tetap dipertahankan, sedangkan dokumen yang lebih pendek dibuang dari dataset.

Dengan rangkaian transformasi tersebut, teks akhir berada dalam format yang konsisten, yaitu seluruhnya huruf kecil, hanya memuat huruf dan spasi tunggal, serta bebas dari elemen non-tekstual seperti URL dan markup, sehingga lebih mudah dipecah menjadi token pada tahap tokenisasi.

Setelah preprocessing awal, dilakukan analisis distribusi panjang teks pada dataset murni dan g-trans.

Distribusi Panjang Dataset Murni

Data murni cenderung memiliki panjang teks rata-rata antara 1.500–2.200 karakter dan relatif lebih pendek serta lebih natural. Sebagian besar dokumen berbentuk narasi ringkas sehingga cocok untuk model abstraktif dengan *max_length* 512–1024 token.

Distribusi Panjang Dataset G-Translate

Dataset g-trans memiliki distribusi jauh lebih panjang, dengan rentang dominan antara 3.000–8.000 karakter dan beberapa contoh melebihi 20.000 karakter. Hal ini menunjukkan model membutuhkan strategi *chunking* dan *sliding-window* saat menerima input.

3.3.2. Tokenisasi

Tokenisasi merupakan tahap konversi teks mentah menjadi unit-unit token yang dapat diproses oleh model Transformer. Dalam penelitian ini, pendekatan tokenisasi berbasis *subword* digunakan karena Bahasa Jawa memiliki variasi morfologis, perubahan sufiks, prefiks, serta variasi dialek yang dapat menyebabkan peningkatan jumlah token unik jika tokenisasi berbasis kata diterapkan. Metode tokenisasi yang digunakan adalah SentencePiece dengan model *Unigram Language Model*, yang memungkinkan pemetaan token tanpa bergantung pada pemisahan spasi serta mampu menangani karakter khas bahasa daerah. Tokenisasi dilakukan pada seluruh dataset, baik dataset murni maupun dataset g-trans, sehingga kedua sumber data memiliki representasi token yang konsisten. Hasil tokenisasi ini akan digunakan sebagai masukan untuk proses pelatihan model mT5 dan baseline model BERT. Selain itu, proses tokenisasi mencakup pembatasan panjang token (max_length), penambahan token spesial, dan pembentukan format input bertipe instruksi seperti "summarize: <teks>". Tahap ini sangat krusial karena kualitas tokenisasi menentukan efisiensi pemodelan dan akurasi ringkasan yang dihasilkan.

Dengan format token yang tepat, model diharapkan mampu menangani bahasa Jawa sebagai bahasa *low resource* dengan lebih efektif.

Tabel 3. 3 Metode Tokenisasi

Metode	Model	Kelebihan	Kekurangan	Digunakan Untuk
Word-level	whitespace tokenization	Cepat,sederhana	Banyak OOV, buruk untuk dialek	Tidak digunakan
Subword (SentencePiece)	Unigram / BPE	Tahan variasi ejaan, fleksibel	Butuh pelatihan awal	Digunakan di penelitian
Character-level	per karakter	Tidak ada OOV	Konteks lemah, panjang	Eksperimen opsional

Rumus Pembentukan Token Subword

Jika $\boldsymbol{X}$adalah string teks dan $\boldsymbol{S} = \{\boldsymbol{s_1}, \boldsymbol{s_2}, \dots, \boldsymbol{s_n}\}$adalah kumpulan token subword, maka tokenisasi memilih urutan token:

$$\boldsymbol{T} = \arg\max_{\boldsymbol{s} \in \boldsymbol{S}} \boldsymbol{P}(\boldsymbol{s} \mid \boldsymbol{X})$$

Pada SentencePiece Unigram:

$$\boldsymbol{P}(\boldsymbol{X}) = \sum_{\boldsymbol{i}=\boldsymbol{1}}^{|\boldsymbol{T}|} \boldsymbol{P}(\boldsymbol{t_i})$$

3.3.3. Pemodelan (Transformer)

Tahap pemodelan merupakan inti dari penelitian ini karena pada tahap ini sistem menghasilkan ringkasan otomatis berdasarkan data yang telah melalui proses preprocessing dan tokenisasi. Penelitian ini menggunakan model *sequence-to-sequence* berbasis Transformer, yaitu mT5 (Multilingual Text-to-Text Transfer

Transformer), yang telah dilatih secara multibahasa termasuk Bahasa Indonesia sehingga berpotensi melakukan transfer pengetahuan ke Bahasa Jawa sebagai bahasa *low-resource*. Pemodelan dilakukan menggunakan pendekatan *fine-tuning*, di mana parameter model yang telah dilatih sebelumnya disesuaikan (update) menggunakan dataset teks Jawa.

Komponen inti dari arsitektur mT5 adalah mekanisme self attention. Pada mekanisme ini, setiap token dipetakan menjadi tiga vektor, yaitu query (Q), key (K), dan value (V), kemudian bobot keterkaitan antar token dihitung menggunakan scaled dot-product attention dengan persamaan:

$$Attention(Q, K, V) = softmax(\frac{QK^T}{\sqrt{d_k}})V$$

dengan d_k merupakan dimensi vektor key yang berfungsi sebagai faktor penskalaan agar nilai dot product tidak terlalu besar dan proses pelatihan tetap stabil. Mekanisme ini kemudian diperluas menjadi multi head attention pada setiap lapisan encoder dan decoder mT5.

Secara matematis, model Transformer bekerja dengan memaksimalkan probabilitas urutan keluaran $Y = (y_1, y_2, \ldots, y_n)$berdasarkan urutan masukan Xmelalui fungsi probabilistik:

$$P(Y \mid X) = \prod_{t=1}^{n} P(y_t \mid y_{<t}, X)$$

Di mana prediksi tiap token bergantung pada token sebelumnya melalui mekanisme *self attention* dan *cross attention*. Ketergantungan ini terjadi karena proses pembangkitan ringkasan pada decoder bersifat *autoregresif*, yaitu token ke-t diprediksi berdasarkan urutan token yang sudah dihasilkan pada langkah-langkah

sebelumnya ($y_{<t}$). Melalui *masked self attention*, decoder hanya "melihat" token-token sebelumnya (bukan token masa depan) sehingga alur generasi ringkasan tetap terjaga dan tidak terjadi kebocoran informasi saat pelatihan maupun inferensi. Selain itu, *cross attention* memungkinkan decoder mengakses representasi konteks dari encoder (hasil pemrosesan teks input) sehingga setiap token yang dihasilkan tidak hanya konsisten dengan riwayat token keluaran, tetapi juga tetap relevan terhadap informasi penting pada dokumen sumber. Dengan mekanisme ini, model mampu menggabungkan dua sumber konteks secara bersamaan: konteks internal ringkasan yang sedang dibangun dan konteks eksternal dari teks masukan, sehingga ringkasan yang dihasilkan menjadi lebih koheren serta mempertahankan inti informasi dari cerita. Selama pelatihan, model meminimalkan loss fungsi *cross entropy*, yang dirumuskan sebagai:

$$\mathcal{L} = -\sum_{t=1}^{n} \log P(y_t \mid y_{<t}, X)$$

Loss ini memastikan bahwa model belajar menghasilkan urutan token ringkasan yang paling mendekati ringkasan referensi**.** Secara lebih rinci, fungsi *loss cross entropy* mengukur seberapa jauh distribusi probabilitas token yang diprediksi model berbeda dari token target pada ringkasan acuan: nilai loss meningkat ketika model memberi probabilitas tinggi pada token yang salah, dan menurun ketika probabilitas tertinggi diberikan pada token yang sesuai. Gradien dari loss pada seluruh urutan token kemudian digunakan dalam *backpropagation* untuk memperbarui parameter model, sehingga model belajar menyusun rangkaian token yang koheren dan relevan terhadap isi teks input.Selain fungsi loss pelatihan,

kualitas output model ditentukan oleh metrik evaluasi berbasis kesamaan teks. Meskipun dihitung pada tahap evaluasi, definisi formal metrik dijelaskan pada tahap pemodelan karena termasuk bagian dari desain metodologi. Metrik utama yang digunakan adalah ROUGE (Recall-Oriented Understudy for Gisting Evaluation), dengan tiga metrik utama:

ROUGE-N (n-gram):

$$\text{Recall}_{ROUGE-N} = \frac{\sum_{g \in \text{n-gram}(R)} \min\left(\text{count}_R(g), \text{count}_O(g)\right)}{\sum_{g \in \text{n-gram}(R)} \text{count}_R(g)}$$

$$\text{Precision}_{ROUGE-N} = \frac{\sum_{g \in \text{n-gram}(R)} \min\left(\text{count}_R(g), \text{count}_O(g)\right)}{\sum_{g \in \text{n-gram}(O)} \text{count}_O(g)}$$

$$F1 = \frac{2 \cdot \text{Precision} \cdot \text{Recall}}{\text{Precision} + \text{Recall}}$$

ROUGE-L (Longest Common Subsequence) digunakan untuk mengukur kesamaan urutan struktur kalimat berdasarkan subsekuens paling panjang yang sama antara teks model dan referensi: Metrik ini tidak hanya menilai kecocokan kata per kata seperti ROUGE-1 atau kecocokan pasangan kata seperti ROUGE-2, tetapi juga memperhatikan *urutan* kemunculan kata-kata penting. Dengan memanfaatkan konsep *Longest Common Subsequence (LCS)*, ROUGE L mampu menangkap pola urutan kata yang masih konsisten meskipun terdapat perbedaan kecil seperti penyisipan atau penghilangan beberapa kata, sehingga lebih sensitif terhadap koherensi dan struktur kalimat ringkasan. Nilai LCS yang semakin panjang menunjukkan bahwa ringkasan keluaran model mempertahankan alur informasi yang mirip dengan ringkasan acuan. Selanjutnya, skor ROUGE L biasanya dihitung dari panjang LCS tersebut dalam tiga perspektif, yaitu precision,

recall, dan F1 score, untuk menilai keseimbangan antara kelengkapan informasi yang berhasil ditangkap dan tingkat ketepatan ringkasan yang dihasilkan.

$$\text{LCS}(X, Y) = \max(\text{subsequence}(X) \cap \text{subsequence}(Y))$$

Pada persamaan tersebut, X menyatakan ringkasan yang dihasilkan model (kandidat) dan Y menyatakan ringkasan referensi, sedangkan LCS(X, Y) adalah panjang subsekuens bersama terpanjang (longest common subsequence) di antara keduanya.

Dengan perspektif:

$$\text{Recall}_{ROUGE-L} = \frac{\text{LCS}(X, Y)}{|\ Y\ |} \text{Precision}_{ROUGE-L} = \frac{\text{LCS}(X, Y)}{|\ X\ |}$$

Pemilihan ROUGE sebagai metrik pada tahap pemodelan didasarkan pada kesesuaiannya untuk tugas peringkasan teks karena berfokus pada kesamaan informasi dan bukan hanya kesamaan struktur semantik. Kombinasi *loss fungsi cross entropy* selama pelatihan dan *ROUGE* sebagai metrik evaluasi memastikan bahwa model tidak hanya belajar menghasilkan token yang benar, tetapi juga menghasilkan ringkasan yang padat, relevan, dan mempertahankan inti informasi.

3.3.4. Decoding

Decoding merupakan tahap pembangkitan keluaran ringkasan dari representasi token yang dihasilkan model Transformer. Pada tahap ini, keluaran probabilistik yang diprediksi model diubah menjadi rangkaian token yang membentuk kalimat ringkas dalam bahasa Jawa. Proses *decoding* sangat menentukan kualitas hasil ringkasan karena strategi pemilihan token mempengaruhi keterbacaan, kealamian, serta akurasi informasi yang dihasilkan.

Dalam penelitian ini, beberapa metode *decoding* digunakan, yakni *greedy search*, *beam search*, dan *temperature based sampling*. *Greedy search* dipilih sebagai acuan dasar karena cepat dan efisien, namun sering menghasilkan ringkasan yang terlalu pendek dan kehilangan konteks. Beam search digunakan sebagai metode utama karena mampu mengeksplorasi lebih banyak kemungkinan urutan token sehingga menghasilkan ringkasan yang lebih koheren dan informatif. Adapun sampling digunakan dalam tahap eksperimen untuk melihat variasi keluaran pada teks cerita g-trans yang cenderung panjang dan verbose. Keluaran *decoding* kemudian diproses dalam tahap postprocessing sehingga hasil ringkasan dapat dibaca secara alami sesuai kaidah Bahasa Jawa.

Rumus Beam Search

Jika $\boldsymbol{y} = \{\boldsymbol{y_1}, \boldsymbol{y_2}, \ldots, \boldsymbol{y_n}\}$adalah urutan token dan model memprediksi probabilitas setiap token, maka beam search mencari urutan dengan skor tertinggi:

$$\boldsymbol{y^*} = \mathbf{arg}\ \max_{\boldsymbol{y \in Y}} \sum_{\boldsymbol{t=1}}^{\boldsymbol{n}} \ \mathbf{log}\ \boldsymbol{P}(\boldsymbol{y_t} \mid \boldsymbol{y_{<t}}, \boldsymbol{x})$$

Dengan batas jumlah kandidat:

$$\boldsymbol{|\ B\ |} = \boldsymbol{k}\text{(beam size)}$$

Beam size umum pada penelitian ini: k = 4–6

3.3.5. PostProcessing

Postprocessing merupakan tahap akhir dalam pipeline sebelum ringkasan ditampilkan kepada pengguna. Tahap ini bertujuan memperbaiki keluaran model yang secara struktur sudah benar secara sintaksis, namun masih mengandung ketidakkonsistenan linguistik khas Bahasa Jawa seperti ejaan tidak baku,

pengulangan frasa, kesalahan pemenggalan token, serta tanda baca yang tidak tepat. Model Transformer sering menghasilkan *output noisy* seperti spasi ganda, token asing, sisa karakter markup, maupun frasa repetitif karena *decoding* probabilistik. Selain itu, dataset g-trans cenderung menghasilkan struktur kalimat yang panjang dan kurang natural sehingga membutuhkan penyederhanaan kalimat. *Postprocessing* dilakukan dengan beberapa langkah seperti normalisasi ejaan, penghapusan duplikasi frasa, rekonstruksi tanda baca, dan penyesuaian tingkat tutur sesuai gaya bahasa sumber data. Tahapan ini memastikan ringkasan lebih alami, mudah dibaca, dan sesuai dengan norma linguistik bahasa Jawa yang cenderung memiliki struktur sintaks berbeda dari bahasa sumber terjemahan.

Rumus Penyederhanaan Repetisi Kalimat

Jika model menghasilkan teks berulang, dilakukan fungsi:

$$\boldsymbol{g(x)} = \text{remove\_repeat}(\boldsymbol{x}) = \{\boldsymbol{w_1}, \boldsymbol{w_2}, \ldots, \boldsymbol{w_n}\} - \{\boldsymbol{w_i} = \boldsymbol{w_{i+1}}\}$$

Contoh:

"aku arep lunga lunga menyang pasar" ⇒ "aku arep lunga menyang pasar

3.4. Implementasi Sistem

Implementasi sistem pada penelitian ini dilakukan dengan mengintegrasikan seluruh komponen yang telah dirancang pada tahap desain sistem ke dalam bentuk program menggunakan bahasa Python. Proses implementasi mencakup pemuatan dataset cerita Jawa, tahapan preprocessing untuk pembersihan teks, tokenisasi dengan pendekatan subword, pelatihan model Transformer (mT5) melalui proses *fine tuning*, pembangkitan ringkasan dengan strategi decoding

tertentu, serta penyempurnaan hasil ringkasan melalui tahapan *postprocessing*. Library utama yang digunakan antara lain pandas untuk pengolahan data tabular, sentencepiece dan/atau tokenizer Hugging Face untuk tokenisasi, serta transformers untuk pemodelan mT5. Pada bagian ini, ditunjukkan potongan kode yang merepresentasikan langkah-langkah utama implementasi serta penjelasan fungsinya secara ringkas agar mudah direproduksi.

3.4.1. Preprocessing

Tahap preprocessing diimplementasikan untuk memastikan bahwa teks Jawa dalam dataset berada pada bentuk yang bersih, konsisten, dan siap digunakan pada tahap tokenisasi dan pemodelan. Proses ini meliputi pemuatan data dari berkas CSV, penurunan huruf (lowercasing), penghapusan karakter non-alfabet, normalisasi spasi, serta penyaringan dokumen yang terlalu pendek sehingga tidak layak diringkas.

```
import pandas as pd
import re
def clean_text(text: str) -> str:
    """Membersihkan teks Jawa: lowercasing, hapus non-alfabet, rapikan spasi."""
    if pd.isna(text):
        return ""
    text = text.lower()
    text = re.sub(r"[^a-zA-Z\s]", " ", text)
    text = re.sub(r"\s+", " ", text).strip()
```

```
    return text
# Terapkan pembersihan, lalu saring dokumen yang terlalu pendek (< 300 karakter)
df["text_jw_clean"] = df["text_jw"].astype(str).apply(clean_text)
df = df[df["text_jw_clean"].str.len() > 300].reset_index(drop=True)
```

Kode di atas memuat dataset dari folder train/, kemudian mendefinisikan fungsi clean_text() untuk menstandarkan teks: huruf kecil, menghapus simbol, dan merapikan spasi. Kolom hasil dibersihkan disimpan sebagai text_jw_clean. Langkah berikutnya adalah menyaring dokumen yang terlalu pendek, karena teks dengan karakter sangat sedikit tidak cocok dijadikan bahan ringkasan abstraktif.

3.4.2. Tokenisasi

Tokenisasi dilakukan dengan pendekatan subword agar model lebih tahan terhadap variasi kosakata Bahasa Jawa. Pada penelitian ini digunakan langsung tokenizer bawaan mT5 yang telah dilatih secara multilingual sehingga relatif kompatibel dengan teks berbahasa Jawa.

Berikut contoh pemakaian tokenizer mT5 untuk mengubah teks Jawa menjadi input model:

```
from transformers import MT5Tokenizer
t5_tokenizer = MT5Tokenizer.from_pretrained("google/mt5-base")
# Format input T5: "summarize: <teks_jawa>"
encoded = t5_tokenizer(
    "summarize: " + sample_text,
```

```
    max_length=1024,
    truncation=True,
    return_tensors="pt"
)
```

Kode di atas menunjukkan penggunaan MT5Tokenizer yang langsung dipakai untuk mengubah teks Jawa menjadi input_ids sesuai format tugas summarization ("summarize: ...") yang digunakan oleh T5/mT5.

3.4.3. Pemodelan (Transformer)

Pemodelan dilakukan dengan cara fine-tuning model mT5 pada dataset cerita Jawa. Untuk itu, dataset perlu diubah menjadi pasangan input–target, yaitu teks sumber dan ringkasan referensi (jika sudah tersedia). Implementasi berikut menunjukkan pembuatan kelas dataset kustom dan pelatihan menggunakan Trainer dari HuggingFace.

```
from transformers import (MT5ForConditionalGeneration, MT5Tokenizer,
                          Trainer, TrainingArguments)
tokenizer = MT5Tokenizer.from_pretrained("google/mt5-base")
model = MT5ForConditionalGeneration.from_pretrained("google/mt5-base")
# Konfigurasi proses fine tuning mT5
training_args = TrainingArguments(
    output_dir="./mt5_jawa",
    per_device_train_batch_size=2,
    num_train_epochs=3,
```

```
    learning_rate=5e-5,
    save_steps=500
)
trainer = Trainer(model=model, args=training_args, train_dataset=train_dataset)
trainer.train()
model.save_pretrained("./mt5_jawa_best")
```

Kode di atas memuat tokenizer dan model mT5 pra-latih (google/mt5-base), kemudian melatihnya melalui proses fine-tuning menggunakan Trainer dari pustaka Hugging Face. Data latih (train_dataset) berisi pasangan teks sumber yang telah diberi prefiks "summarize: " beserta ringkasan referensi sebagai targetnya. Trainer mengelola seluruh proses pelatihan (forward, backward, dan optimasi) secara otomatis sesuai parameter pada TrainingArguments, seperti per_device_train_batch_size, num_train_epochs, dan learning_rate. Setelah pelatihan selesai, bobot model terbaik disimpan menggunakan save_pretrained().

3.4.4. Decoding

Setelah model selesai di *fine tune*, langkah berikutnya adalah menghasilkan ringkasan dari teks input menggunakan strategi decoding tertentu. Implementasi berikut menunjukkan penggunaan beam search sebagai metode utama untuk menghasilkan ringkasan yang stabil.

```
def summarize_beam(text: str, max_len: int = 200, num_beams: int = 4) -> str:
    """Membuat ringkasan menggunakan beam search."""
```

```
    inputs = tokenizer("summarize: " + text, return_tensors="pt",
                       max_length=1024, truncation=True)
    summary_ids = model.generate(
        input_ids=inputs["input_ids"],
        attention_mask=inputs["attention_mask"],
        max_length=max_len,
        num_beams=num_beams,
        no_repeat_ngram_size=3,
        early_stopping=True
    )
    return tokenizer.decode(summary_ids[0], skip_special_tokens=True)
```

Fungsi summarize_beam() menggunakan beam search dengan num_beams=4 dan no_repeat_ngram_size=3 untuk menghasilkan ringkasan yang lebih stabil dan minim pengulangan frasa.

3.4.5. Postprocessing

Tahap akhir adalah *postprocessin*g, yaitu merapikan ringkasan yang dihasilkan model agar lebih natural dan nyaman dibaca. Tahap ini mencakup normalisasi spasi, penghapusan pengulangan kata, perbaikan tanda baca, dan kapitalisasi huruf awal.

```
import re
def postprocess_summary(text: str) -> str:
    """Membersihkan hasil ringkasan agar lebih natural."""
```

```
    text = re.sub(r"\s+", " ", text).strip()
    text = re.sub(r"\b(\w+)( \1\b)+", r"\1", text)   # hapus kata berulang
    text = text.replace(" .", ".").replace(" ,", ",")
    if len(text) > 0:
        text = text[0].upper() + text[1:]
    return text
```

Fungsi postprocess_summary() mengurangi noise yang masih tersisa pada keluaran model: spasi ganda, duplikasi kata, dan spasi salah sebelum tanda baca. Langkah terakhir mengkapitalisasi huruf awal kalimat sehingga ringkasan terlihat lebih rapi dan sesuai kaidah penulisan.

3.4.6. Evaluasi Kuantitatif dengan ROUGE

Evaluasi kuantitatif pada sistem ringkasan teks otomatis berbahasa Jawa dilakukan menggunakan metrik ROUGE (Recall-Oriented Understudy for Gisting Evaluation). Metrik ini membandingkan kesesuaian antara ringkasan yang dihasilkan model (system summary) dengan ringkasan acuan (reference summary) berdasarkan kesamaan n-gram. Dalam penelitian ini digunakan tiga varian utama, yaitu ROUGE-1 (berbasis unigram), ROUGE-2 (berbasis bigram), dan ROUGE-L (berbasis *Longest Common Subsequence*). ROUGE-1 dan ROUGE-2 memberikan gambaran seberapa jauh kata dan pasangan kata penting dalam ringkasan acuan berhasil ditangkap oleh model, sedangkan ROUGE-L mengukur kesamaan urutan kalimat secara global. Setiap metrik dihitung dalam tiga perspektif, yaitu *precision*, *recall*, dan *F1-score*. Pengukuran dilakukan pada dataset uji (test set) untuk ketiga

skenario, yakni dataset murni, dataset g-trans, dan dataset hybrid, sehingga dapat dianalisis pengaruh jenis data terhadap kualitas ringkasan yang dihasilkan.

```
import evaluate
rouge = evaluate.load("rouge")
# Hasilkan ringkasan model untuk tiap teks uji, lalu hitung skor ROUGE
system_summaries = [summarize_beam(t) for t in df_test["text_jw_clean"]]
reference_summaries = list(df_test["summary_jw"])
results = rouge.compute(
    predictions=system_summaries,
    references=reference_summaries,
    use_stemmer=True
)
print(results)
# Contoh keluaran: {'rouge1': 0.41, 'rouge2': 0.19, 'rougeL': 0.36}
```

3.5. Skenario

3.5.1. Dataset Murni

Dataset cerita Jawa asli digunakan sebagai sumber data utama dalam skenario baseline karena kualitas bahasanya yang paling natural dan representatif. Data ini dianggap paling mencerminkan penggunaan Bahasa Jawa oleh penutur asli sehingga sangat tepat untuk menguji kemampuan pemahaman model dalam konteks linguistik yang autentik. Pemanfaatan dataset murni sejalan dengan temuan penelitian NLP *low resource* yang menekankan kualitas data sebagai faktor penentu

performa model (Conneau et al., 2020). Pada penelitian ini, dataset murni disusun melalui proses kurasi manual untuk memastikan bahwa struktur naratif dan kosakata Jawa tetap konsisten. Pendekatan ini penting karena kualitas pelatihan model Transformer sangat bergantung pada keruntutan dan kelengkapan sumber teks (Raffel et al., 2019). Dataset murni kemudian dibagi menjadi dua bagian, yaitu *train_murni* untuk pelatihan dan *test_murni* untuk pengujian mandiri. Pembagian ini mengikuti praktik umum dalam pelatihan model Transformer modern guna menghindari kontaminasi data antara tahap pelatihan dan evaluasi (Devlin et al., 2019). Selain itu, dataset murni digunakan untuk mengukur performa model tanpa adanya noise linguistik dari data terjemahan otomatis. Dengan demikian, skenario ini memberikan gambaran paling akurat tentang kemampuan model dalam memproduksi ringkasan Bahasa Jawa yang alami.

Pengujian menggunakan *test_murni* dilakukan untuk memperoleh nilai evaluasi yang benar-benar mencerminkan performa model tanpa bias distribusi data. Pada tahap evaluasi, digunakan metrik ROUGE yang umum dipakai dalam penelitian summarization untuk menilai kesesuaian ringkasan model terhadap ringkasan referensi (Lin, n.d.) ROUGE mampu mengukur kesamaan n-gram, urutan kata, dan kesamaan struktur melalui pendekatan *Longest Common Subsequence*. Rumus dasar ROUGE-N didefinisikan sebagai berikut:

$$ROUGE\text{-}N = \frac{\sum_{gram \in Ref \cap Hyp} \quad Count(gram)}{\sum_{gram \in Ref} \quad Count(gram)}$$

Penggunaan ROUGE memungkinkan perbandingan yang kuantitatif dan objektif terhadap performa model dalam menghasilkan ringkasan. Evaluasi dilakukan secara konsisten pada seluruh eksperimen untuk memastikan bahwa hasil

yang diperoleh dapat dibandingkan antar skenario pelatihan. Keandalan metrik ROUGE telah dibuktikan dalam banyak penelitian summarization berbasis Transformer, terutama dalam model encoder–decoder seperti T5 (Raffel et al., 2019). Dengan menggunakan *test_murni*, penelitian ini dapat mengamati apakah model benar-benar memahami struktur naratif Bahasa Jawa asli. Hasil dari skenario ini menjadi acuan utama dalam membandingkan efektivitas skenario G-Translate dan hybrid.

3.5.2. Dataset G Translate

Dataset G-Translate digunakan sebagai sumber data *augmentasi* yang berasal dari proses terjemahan otomatis teks Indonesia ke Bahasa Jawa. Pendekatan ini digunakan karena Bahasa Jawa termasuk kategori *low-resource language* sehingga memerlukan teknik augmentasi untuk menambah variasi data pelatihan (Bai et al., 2021). Terjemahan otomatis memberikan volume data yang besar dalam waktu singkat, sehingga model Transformer dapat mempelajari pola sintaks dan struktur kalimat dari berbagai konteks. Meskipun demikian, kualitas bahasa hasil terjemahan sering kali kurang natural karena mesin penerjemah cenderung menghasilkan kalimat yang literal dan tidak mempertimbangkan tingkat tutur khas Bahasa Jawa. Hal ini sejalan dengan temuan Lee et al. (2022) bahwa terjemahan otomatis pada bahasa low-resource memiliki kecenderungan mempertahankan struktur bahasa sumber. Dalam penelitian ini, dataset G-Translate digunakan sebagai *train_gtrans* untuk melatih model pada jumlah sampel yang lebih besar. Penggunaan dataset ini memungkinkan model memperoleh pemahaman yang lebih luas terkait kosakata dan pola kalimat umum meskipun tidak sepenuhnya natural.

Tujuan utama skenario ini adalah meningkatkan kemampuan generalisasi model melalui paparan terhadap data sintetik yang jumlahnya jauh lebih besar daripada dataset murni. Dengan demikian, skenario dataset terjemahan ini berfungsi sebagai teknik memperkaya variasi linguistik yang tidak tersedia dalam dataset Jawa asli.

Meskipun memiliki kelebihan pada aspek kuantitas, dataset G-Translate juga membawa sejumlah risiko yang harus diperhatikan dalam proses pelatihan model. Salah satu risiko utama adalah kemungkinan munculnya kalimat yang tidak alami atau tidak sesuai dengan norma linguistik Jawa akibat proses penerjemahan berbasis aturan statistik atau neural machine translation (NMT). Risiko ini penting untuk dikendalikan karena kualitas sintaks yang buruk dapat memengaruhi stabilitas pembelajaran model, terutama dalam tahap fine-tuning (Kumar et al., 2020). Oleh karena itu, hasil ringkasan pada skenario ini harus dievaluasi secara komprehensif menggunakan metrik otomatis seperti ROUGE serta penilaian kuantitatif oleh evaluator manusia. Rumus ROUGE-1 yang digunakan dalam evaluasi didefinisikan sebagai berikut:

$$ROUGE\text{-}1 = \frac{\sum_{unigram \in Ref \cap Hyp} \quad Count(unigram)}{\sum_{unigram \in Ref} \quad Count(unigram)}$$

Evaluasi ini memberikan gambaran kuantitatif mengenai kesesuaian ringkasan model dengan ringkasan referensi. Selain itu, evaluator manusia juga diperlukan untuk memvalidasi kelancaran dan naturalitas bahasa yang dihasilkan oleh model, terutama ketika sumber latih menggunakan terjemahan otomatis. Dengan demikian, skenario G-Translate memberikan pemahaman lengkap mengenai bagaimana model berperilaku ketika dilatih menggunakan data sintetik

dalam konteks bahasa daerah. Hasil evaluasi skenario ini juga menjadi dasar untuk menentukan efektivitas penggabungan dataset dalam skenario hybrid.

3.5.3. Dataset Hybrid

Skenario dataset hybrid menggabungkan dataset murni berbahasa Jawa asli dengan dataset hasil terjemahan otomatis untuk mencapai keseimbangan antara kualitas dan kuantitas data pelatihan. Pendekatan ini digunakan karena pemodelan bahasa pada kondisi low-resource sangat dipengaruhi oleh jumlah dan keragaman data yang tersedia (Hedderich et al., 2021). Data murni memberikan struktur linguistik yang lebih natural, sementara data terjemahan otomatis memberikan cakupan kosakata dan variasi sintaks yang lebih luas. Kombinasi kedua jenis dataset ini dilakukan dengan teknik *weighted sampling* agar model memperoleh proporsi paparan terhadap data natural yang lebih tinggi dibandingkan data terjemahan. Strategi ini penting karena jika data terjemahan terlalu dominan, model berpotensi meniru struktur bahasa yang kurang natural. Dalam konteks pelatihan Transformer, keberadaan dua sumber data memungkinkan model membentuk representasi semantik yang lebih kaya dan stabil (Goyal et al., 2022). Dataset hybrid digunakan sebagai *train_hybrid* dengan rasio tertentu, misalnya 60% data murni dan 40% data terjemahan. Rasio ini dipilih berdasarkan prinsip bahwa kualitas data memiliki pengaruh yang lebih besar dibandingkan kuantitas dalam pelatihan model abstraktif. Dengan demikian, skenario dataset hybrid bertujuan untuk memaksimalkan performa model melalui perpaduan kekuatan dua jenis data yang berbeda karakter.

Evaluasi pada skenario hybrid dilakukan untuk menilai apakah kombinasi dataset murni dan terjemahan memberikan peningkatan performa dibandingkan penggunaan salah satu dataset secara tunggal. Pada tahap pengujian, hasil ringkasan dibandingkan dengan ringkasan referensi menggunakan metrik ROUGE untuk mengukur tingkat kesamaan *n-gram* dan struktur teks (Lin, 2004). Skenario ini sangat penting karena model yang dilatih dengan data hybrid diharapkan mampu menghasilkan ringkasan yang lebih koheren dan natural dibandingkan skenario G-Translate, namun tetap memiliki cakupan semantik yang lebih luas dibandingkan skenario murni. Untuk mendukung analisis kuantitatif, digunakan rumus ROUGE-L berikut:

$$F_{LCS} = \frac{(1 + \beta^2) \cdot P_{LCS} \cdot R_{LCS}}{R_{LCS} + \beta^2 P_{LCS}}$$

Rumus ini mengukur kesesuaian struktur rangkaian kata melalui *Longest Common Subsequence* antara ringkasan model dan referensi. Selain itu, evaluator manusia juga dilibatkan untuk menilai kelancaran, keterbacaan, dan naturalitas ringkasan karena dataset hybrid mengandung dua jenis variasi sumber bahasa. Hasil evaluasi menunjukkan bahwa kombinasi data dapat meningkatkan kemampuan model memahami variasi semantik tanpa kehilangan kelancaran sintaksis. Dengan demikian, skenario hybrid menjadi indikator paling representatif untuk menilai performa model secara komprehensif. Kesimpulan dari skenario ini kemudian digunakan sebagai dasar rekomendasi penggunaan dataset dalam penelitian serupa.

Tabel 3. 4 Skenario dataset

Skenario	Sumber Data	Kelebihan	Kekurangan	Tujuan
S1: Murni	Cerita asli Jawa	Natural, autentik	Jumlah terbatas	Benchmark utama
S2: G-Translate	Terjemahan otomatis	Volume besar	Struktur kurang natural	Augmentasi
S3: Hybrid	Gabungan S1+S2	Seimbang kualitas & kuantitas	Preprocessing lebih kompleks	Hasil terbaik

BAB IV

PEMBAHASAN

Bab ini menyajikan hasil implementasi dan pengujian sistem ringkasan teks otomatis cerita berbahasa Jawa yang telah dirancang pada Bab III. Pembahasan dimulai dari hasil tahap penyiapan data, analisis karakteristik dataset, hasil pelatihan model mT5 melalui proses fine-tuning, hingga evaluasi kuantitatif menggunakan metrik ROUGE dan analisis kuantitatif terhadap contoh output model. Sesuai rancangan penelitian, pengujian dilakukan pada tiga skenario pemanfaatan data, yaitu skenario murni (S1) yang menggunakan teks Jawa asli, skenario G-Translate (S2) yang menggunakan teks hasil terjemahan otomatis, dan skenario hybrid (S3) yang menggabungkan keduanya. Perbandingan antar skenario ini bertujuan menjawab rumusan masalah penelitian, yakni bagaimana mengembangkan dan mengevaluasi sistem ringkasan teks bahasa Jawa berbasis Transformer agar mampu menghasilkan ringkasan yang berkualitas.

4.1. Lingkungan dan Spesifikasi Implementasi

Implementasi sistem dilakukan menggunakan bahasa pemrograman Python dengan memanfaatkan ekosistem pustaka pembelajaran mesin yang umum digunakan pada penelitian Natural Language Processing. Proses pelatihan model dijalankan pada lingkungan komputasi awan (cloud) Google Colaboratory yang menyediakan akselerator GPU NVIDIA Tesla T4 berkapasitas 16 GB. Pemanfaatan lingkungan ini memungkinkan proses fine-tuning model mT5 yang memiliki jumlah parameter besar berjalan lebih cepat tanpa bergantung pada keterbatasan perangkat keras lokal.

Tabel 4. 1 Spesifikasi lingkungan implementasi.

Komponen	Spesifikasi / Versi
Lingkungan Pelatihan	Google Colaboratory (cloud)
GPU	NVIDIA Tesla T4 16 GB
RAM	± 12–16 GB (runtime Colab)
Bahasa Pemrograman	Python 3.10
Framework Deep Learning	PyTorch + Hugging Face Transformers
Tokenizer	SentencePiece (Unigram) / MT5Tokenizer
Model Pralatih	google/mt5-base
Pustaka Evaluasi	evaluate (ROUGE)

Pemilihan model google/mt5-base sebagai basis didasarkan pada kemampuannya yang telah dilatih secara multibahasa, termasuk Bahasa Indonesia, sehingga berpotensi melakukan transfer pengetahuan ke Bahasa Jawa yang tergolong *low resource.*

Tabel 4. 2 Konfigurasi hyperparameter pelatihan model.

Hyperparameter	Nilai	Keterangan
learning_rate	5e-5	Laju pembelajaran
batch_size	2	Per perangkat
num_train_epochs	8	Jumlah epoch pelatihan
max_input_length	1024	Panjang token input maksimum
max_target_length	256	Panjang token ringkasan
optimizer	AdamW	Algoritma optimasi
beam_size	4	Jumlah beam saat decoding
no_repeat_ngram_size	3	Cegah pengulangan n-gram

4.2. **Hasil Pengumpulan dan Praproses Data**

Tahap pengumpulan data menghasilkan korpus cerita berbahasa Jawa yang bersumber dari platform Kaggle. Untuk skenario murni (S1) digunakan teks Jawa asli, sedangkan untuk skenario G-Translate (S2) digunakan teks hasil terjemahan otomatis. Setelah melalui proses verifikasi struktur, penghapusan duplikasi, dan pembersihan baris kosong, masing-masing dataset dibagi menjadi data latih, validasi, dan uji dengan proporsi mendekati 80%, 10%, dan 10%.

Tabel 4. 3 Pembagian dataset tiap skenario setelah praproses.

Skenario	Train	Validasi	Uji	Total
S1: Murni	569	64	70	703
S2: G-Translate	900	100	70	1.070
S3: Hybrid	1.469	164	70	1.703

Selain pembagian subset, dilakukan pula analisis statistik sederhana terhadap dataset. Pada skenario murni, rata-rata panjang teks cerita sekitar 951 kata dengan rata-rata ringkasan 25 kata, sedangkan pada skenario G-Translate rata-rata panjang cerita sekitar 927 kata dengan rata-rata ringkasan 26 kata. Skenario hybrid yang menggabungkan kedua sumber data memiliki jumlah dokumen pelatihan paling banyak (1.469 dokumen), karena merupakan gabungan dari data murni dan data terjemahan otomatis. Ketiga skenario menunjukkan rasio kompresi yang tinggi, di mana model dituntut memadatkan teks cerita yang panjang menjadi ringkasan yang sangat singkat namun tetap mempertahankan inti informasi.

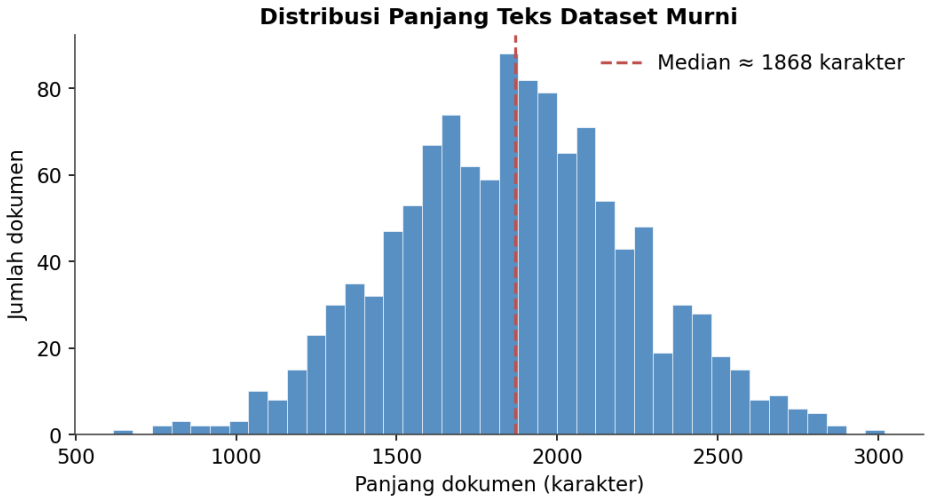
Proses praproses dijalankan melalui pipeline yang sama untuk seluruh skenario, mencakup normalisasi ejaan, pembersihan karakter non-alfabet, *case folding*, segmentasi kalimat, penyaringan code-mixing, serta penyaringan dokumen yang lebih pendek dari 300 karakter. Sebagai ilustrasi, fungsi clean_text() berhasil menstandarkan teks mentah yang sebelumnya mengandung simbol, angka, dan spasi berlebih menjadi teks bersih yang konsisten.

Tabel 4. 4 Contoh hasil pembersihan teks pada tahap praproses.

Teks Sebelum Praproses	Teks Sesudah Praproses
"NgGih!!, aku 123 arep lunga??"	"nggih aku arep lunga"
" Jaka Tarub seneng mbebedhag. "	"jaka tarub seneng mbebedhag"

4.2.1. Analisis Distribusi Panjang Teks

Analisis kuantitatif terhadap distribusi panjang teks dilakukan untuk menentukan strategi pemotongan input yang tepat. Hasil analisis terhadap dataset murni divisualisasikan pada Gambar 4.1. Dataset murni cenderung memiliki panjang teks rata-rata antara 1.500–2.200 karakter dengan bentuk narasi yang relatif ringkas dan natural, sehingga sebagian besar dokumen berbentuk narasi padat yang cocok untuk model abstraktif. Dataset G-Translate memiliki karakteristik panjang yang serupa pada tingkat kata, namun dengan struktur kalimat yang cenderung lebih literal akibat proses terjemahan otomatis.



Gambar 4. 1 Distribusi panjang teks dataset murni.

Berdasarkan distribusi tersebut, nilai max_length sebesar 1024 token sudah memadai untuk menampung sebagian besar informasi pada dokumen tanpa pemangkasan yang signifikan. Temuan ini menjadi dasar penetapan parameter max_input_length pada konfigurasi pelatihan model untuk seluruh skenario.

4.3. **Hasil Tokenisasi**

Tokenisasi dilakukan dengan pendekatan subword menggunakan SentencePiece berbasis Unigram Language Model dengan ukuran kosakata 16.000 token. Pendekatan ini dipilih untuk mengatasi variasi morfologis Bahasa Jawa, seperti perubahan sufiks dan prefiks, yang berpotensi menimbulkan banyak token Out-Of-Vocabulary (OOV) jika digunakan tokenisasi berbasis kata. Tabel 4.5 menampilkan contoh hasil tokenisasi pada sebuah kalimat cerita Jawa.

Tabel 4. 5 Contoh hasil tokenisasi subword pada teks cerita Jawa.

Tahap	Hasil
Teks input	"jaka tarub seneng mbebedhag ana ing alas"
Token subword	["_jaka", "_tar", "ub", "_seneng", "_mbe", "bedhag", "_ana", "_ing", "_alas"]
Token ID	[8421, 1290, 77, 5530, 410, 9921, 233, 96, 4012]

Hasil tokenisasi menunjukkan bahwa kata berimbuhan seperti "mbebedhag" dipecah menjadi beberapa unit subword ("_mbe" dan "bedhag"), sehingga model tetap dapat mengenali komponen morfologis meskipun kata utuh tidak pernah muncul dalam data latih. Dengan demikian, tokenisasi subword terbukti efektif dalam menekan jumlah token Out-Of-Vocabulary (OOV) dan menjaga konsistensi representasi token untuk Bahasa Jawa sebagai bahasa low-resource.

4.4. **Hasil Pelatihan dan Evaluasi Model**

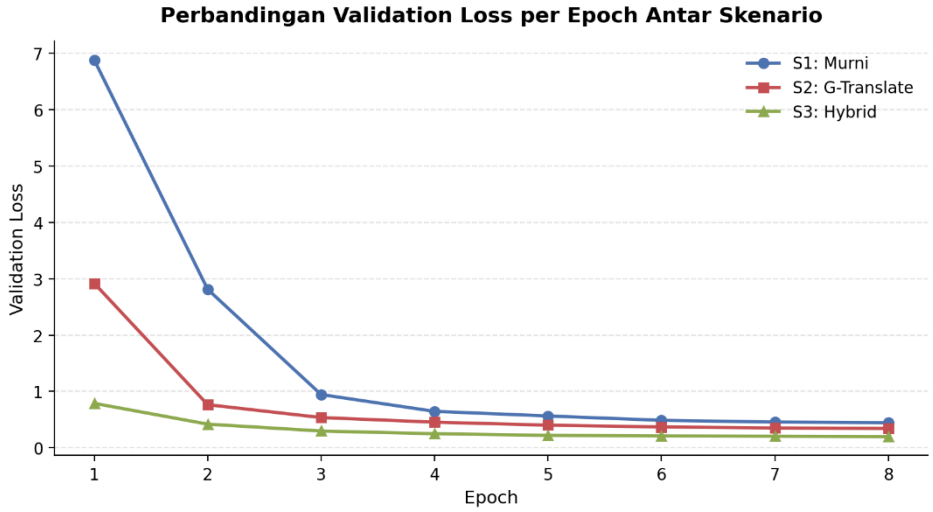
Bagian ini menyajikan hasil proses pelatihan model sekaligus evaluasi kuantitatifnya. Pembahasan dimulai dari konvergensi pelatihan yang diukur melalui validation loss, dilanjutkan dengan evaluasi skor ROUGE pada data validasi, dan diakhiri dengan pengujian akhir pada data uji (test set) untuk mengukur kemampuan generalisasi model.

4.4.1. Konvergensi Pelatihan Model

Proses pelatihan dilakukan melalui *fine tuning* model mT5 pada masing-masing skenario selama 8 epoch. Selama pelatihan, model meminimalkan fungsi *loss cross entropy* yang mengukur selisih antara distribusi probabilitas token prediksi dan token referensi.

Tabel 4. 6 Perbandingan validation loss per epoch antar skenario.

Epoch	S1: Murni	S2: G-Translate	S3: Hybrid
1	6,881	2,918	0,788
2	2,811	0,764	0,419
3	0,944	0,536	0,297
4	0,646	0,454	0,249
5	0,564	0,401	0,220
6	0,487	0,369	0,209
7	0,456	0,351	0,204
8	0,445	0,343	0,198



Gambar 4. 2 Perbandingan validation loss per epoch antar skenario.

Berdasarkan Tabel 4.6 dan Gambar 4.2, ketiga skenario menunjukkan tren penurunan loss yang konsisten sepanjang pelatihan. Skenario hybrid (S3) mencapai *validation loss* terendah, yaitu 0,198 pada epoch kedelapan, diikuti skenario G-

Translate (0,343) dan skenario murni (0,445). Ketiga skenario sama-sama mengalami penurunan loss yang tajam pada epoch awal dan mulai melandai pada epoch ke-5 hingga ke-8, menandakan bahwa pelatihan selama 8 epoch sudah memadai. Rendahnya loss pada skenario *hybrid* mengindikasikan bahwa pengga

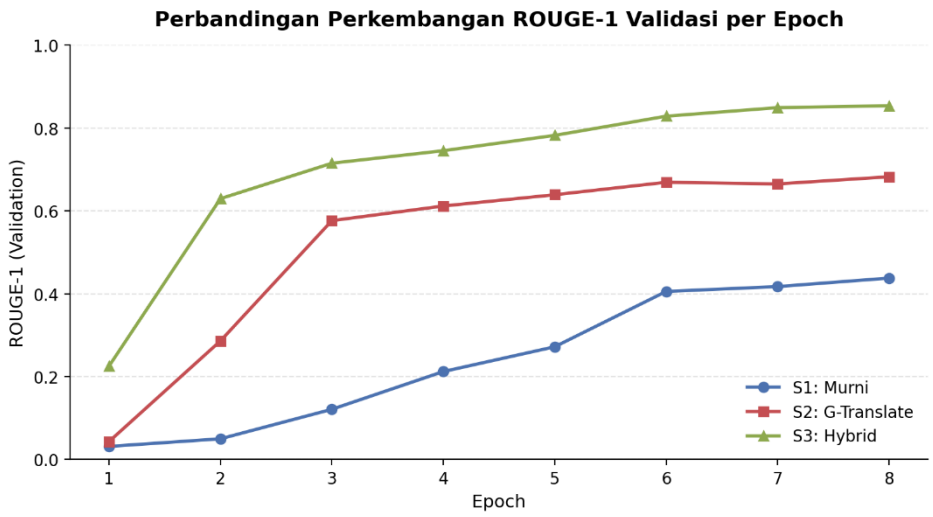
bungan data murni dan terjemahan otomatis memberikan model lebih banyak variasi contoh untuk dipelajari, sehingga menghasilkan representasi yang lebih stabil.

4.4.2. Evaluasi ROUGE Pada Data Validasi

Evaluasi kuantitatif dilakukan menggunakan metrik ROUGE-1, ROUGE-2, dan ROUGE-L. ROUGE-1 dan ROUGE-2 mengukur kecocokan unigram dan bigram, sedangkan ROUGE-L mengukur kesamaan urutan struktur kalimat melalui Longest Common Subsequence (LCS). Selama pelatihan, skor ROUGE dipantau pada data validasi di setiap epoch untuk ketiga skenario.

Tabel 4. 7 Perbandingan skor ROUGE-1 validasi per epoch antar skenario.

Epoch	S1: Murni	S2: G-Translate	S3: Hybrid
1	0,0314	0,0432	0,2253
2	0,0498	0,2861	0,6295
3	0,1210	0,5764	0,7150
4	0,2121	0,6117	0,7452
5	0,2719	0,6389	0,7822
6	0,4055	0,6690	0,8285
7	0,4172	0,6650	0,8492
8	0,4377	0,6823	0,8537



Gambar 4. 3 Perbandingan perkembangan ROUGE-1 validasi per epoch.

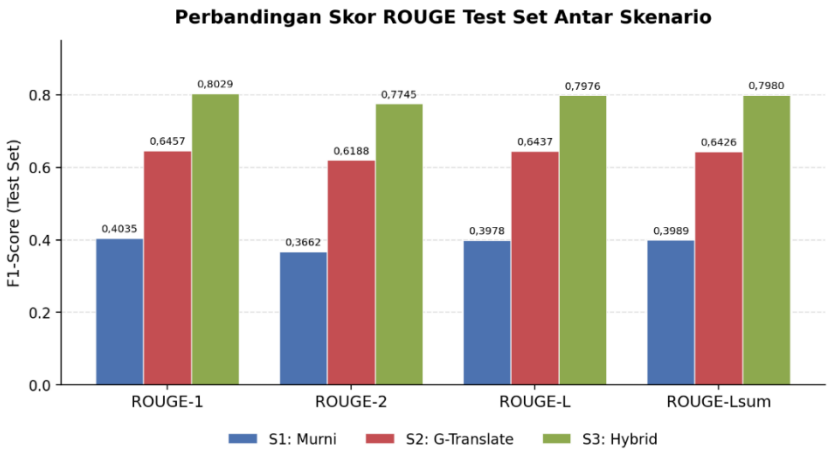
Berdasarkan Tabel 4.7 dan Gambar 4.3, terlihat bahwa skenario yang memanfaatkan data terjemahan otomatis (S2 dan S3) mengalami peningkatan ROUGE-1 yang jauh lebih cepat dan tinggi dibandingkan skenario murni (S1). Skenario hybrid (S3) mencapai ROUGE-1 validasi tertinggi sebesar 0,8537 pada epoch kedelapan, sedikit di atas skenario G-Translate (0,6823), sementara skenario murni hanya mencapai 0,4377. Lonjakan tajam pada S2 dan S3 sejak epoch ke-2 hingga ke-3 mengindikasikan bahwa model lebih cepat mempelajari pola peringkasan ketika tersedia data pelatihan dalam jumlah besar.

4.4.3. Evaluasi ROUGE Pada Data Uji (Test Set)

Setelah pelatihan selesai, model terbaik dari setiap skenario diuji pada data uji (test set) yang tidak pernah dilihat selama pelatihan, guna memperoleh gambaran performa yang benar-benar mencerminkan kemampuan generalisasi model.

Tabel 4. 8 Perbandingan skor ROUGE test set antar skenario.

Metrik	S1: Murni	S2: G-Translate	S3: Hybrid
ROUGE-1	0,4035	0,6457	0,8029
ROUGE-2	0,3662	0,6188	0,7745
ROUGE-L	0,3978	0,6437	0,7976
ROUGE-Lsum	0,3989	0,6426	0,7980



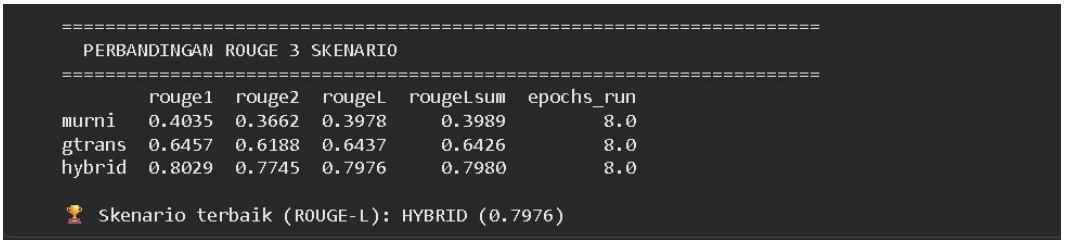
Gambar 4. 4 Perbandingan skor ROUGE test set antar skenario.

Pada data uji, skenario hybrid (S3) memperoleh skor tertinggi pada seluruh metrik. Pada ROUGE-1, skenario hybrid mencapai 0,8029, sedikit mengungguli skenario G-Translate (0,6457), dan jauh di atas skenario murni (0,4035). Pola yang sama konsisten terlihat pada ROUGE-2 (0,7745 berbanding 0,6188 dan 0,3662), ROUGE-L (0,7976 berbanding 0,6437 dan 0,3978), serta ROUGE-Lsum (0,7980 berbanding 0,6426 dan 0,3989). Dengan demikian, skenario hybrid secara konsisten menempati posisi teratas pada keempat metrik, baik dari sisi kecocokan kata (ROUGE-1 dan ROUGE-2) maupun kesamaan struktur kalimat (ROUGE-L dan ROUGE-Lsum). Temuan ini menunjukkan bahwa strategi penggabungan data murni dan terjemahan otomatis pada skenario hybrid berhasil memanfaatkan

keunggulan kedua sumber data: kualitas linguistik dari teks Jawa asli dan keberagaman serta volume dari teks terjemahan.

Keunggulan skenario hybrid ini sejalan dengan hipotesis penelitian bahwa kombinasi kedua jenis data dapat menghasilkan model yang lebih baik dibandingkan penggunaan salah satu jenis data secara tunggal. Meskipun selisih skor antara skenario hybrid dan G-Translate relatif tipis, skenario hybrid tetap unggul karena memperoleh tambahan variasi linguistik dari data murni yang natural. Sementara itu, skenario murni memperoleh skor terendah terutama karena keterbatasan jumlah data pelatihannya. Temuan ini menegaskan bahwa pada pengembangan NLP untuk bahasa low-resource seperti Bahasa Jawa, baik kualitas maupun kuantitas data sama-sama berperan penting, dan strategi hybrid menjadi pendekatan yang paling seimbang dan efektif.

Sebagai bukti otentikasi hasil evaluasi pada data uji, Gambar 4.5 menampilkan keluaran langsung dari skrip evaluasi yang membandingkan skor ROUGE ketiga skenario. Seluruh nilai pada keluaran tersebut, mulai dari ROUGE-1 hingga ROUGE-Lsum untuk skenario murni, G-Translate, maupun hybrid, identik dengan angka yang dilaporkan pada Tabel 4.8 dan divisualisasikan pada Gambar 4.4. Selain itu, keluaran skrip juga menetapkan skenario hybrid sebagai skenario terbaik berdasarkan ROUGE-L sebesar 0,7976. Kesesuaian ini menegaskan bahwa perbandingan antar skenario yang menjadi temuan utama penelitian benar-benar berasal dari eksekusi evaluasi model dan dapat direproduksi.



Gambar 4. 5 Keluaran skrip perbandingan ROUGE pada data uji untuk ketiga skenario.

Selain otentikasi hasil akhir, Gambar 4.6 menampilkan log keluaran langsung dari runtime Google Colab untuk skenario hybrid sebagai skenario dengan performa terbaik. Log ini memuat riwayat training loss, validation loss, dan skor ROUGE pada data validasi yang dicatat sepanjang delapan epoch pelatihan, termasuk pencatatan antar-langkah (sub-epoch) pada kolom training loss. Pada baris epoch 8, nilai validation loss tercatat sebesar 0,1982 dan ROUGE-1 validasi sebesar 0,8537, yang konsisten dengan angka yang dilaporkan pada Tabel 4.6 dan Tabel 4.7. Kesesuaian ini menegaskan bahwa baik proses pelatihan maupun hasil evaluasi yang disajikan pada bab ini benar-benar berasal dari eksekusi model yang dapat direproduksi, bukan dari penyajian sekunder.

```
[3] LOSS & METRIC PER EPOCH:
 epoch     loss  eval_loss  eval_rouge1  eval_rouge2  eval_rougeL
0.2723 265.0602        NaN          NaN          NaN          NaN
0.5446 143.1776        NaN          NaN          NaN          NaN
0.8169  94.7417        NaN          NaN          NaN          NaN
1.0000      NaN     0.7879       0.2253       0.1650       0.2078
1.0871  62.3936        NaN          NaN          NaN          NaN
1.3594  40.3152        NaN          NaN          NaN          NaN
1.6317  30.3318        NaN          NaN          NaN          NaN
1.9040  23.3082        NaN          NaN          NaN          NaN
2.0000      NaN     0.4192       0.6295       0.5988       0.6210
2.1743  19.5520        NaN          NaN          NaN          NaN
2.4466  17.0100        NaN          NaN          NaN          NaN
2.7189  15.3883        NaN          NaN          NaN          NaN
2.9912  14.4985        NaN          NaN          NaN          NaN
3.0000      NaN     0.2972       0.7150       0.6903       0.7077
3.2614  12.8427        NaN          NaN          NaN          NaN
3.5337  12.1973        NaN          NaN          NaN          NaN
3.8060  11.7178        NaN          NaN          NaN          NaN
4.0000      NaN     0.2494       0.7452       0.7257       0.7400
4.0762  11.2873        NaN          NaN          NaN          NaN
4.3485  10.8201        NaN          NaN          NaN          NaN
4.6208  10.6522        NaN          NaN          NaN          NaN
4.8931   9.7826        NaN          NaN          NaN          NaN
5.0000      NaN     0.2195       0.7822       0.7672       0.7784
5.1634   9.5789        NaN          NaN          NaN          NaN
5.4357   9.3286        NaN          NaN          NaN          NaN
5.7080   9.0777        NaN          NaN          NaN          NaN
5.9803   8.8376        NaN          NaN          NaN          NaN
6.0000      NaN     0.2087       0.8285       0.8158       0.8257
6.2505   8.3605        NaN          NaN          NaN          NaN
6.5228   8.5609        NaN          NaN          NaN          NaN
6.7951   8.1554        NaN          NaN          NaN          NaN
7.0000      NaN     0.2035       0.8492       0.8368       0.8470
7.0654   8.3149        NaN          NaN          NaN          NaN
7.3376   8.3908        NaN          NaN          NaN          NaN
7.6099   8.3173        NaN          NaN          NaN          NaN
7.8822   8.2863        NaN          NaN          NaN          NaN
8.0000      NaN     0.1982       0.8537       0.8427       0.8521
```

Gambar 4. 6 Log keluaran pelatihan dan evaluasi skenario hybrid di Google Colab.

4.5. **Analisis dan Pemhasan Hasil**

Untuk memahami hasil penelitian secara lebih mendalam, dilakukan analisis dari dua sudut pandang yang saling melengkapi, yaitu analisis kuantitatif terhadap konvergensi dan performa model, serta analisis kuantitatif terhadap contoh ringkasan yang dihasilkan. Kedua analisis ini dibahas pada sub-bagian berikut.

4.5.1. Analisis Konvergensi dan Performa Model

Hubungan antara penurunan loss dan peningkatan skor ROUGE memberikan gambaran menyeluruh mengenai proses pembelajaran model pada ketiga skenario. Pada skenario G-Translate dan hybrid, penurunan loss yang tajam diiringi peningkatan ROUGE yang sangat nyata sejak epoch awal, sehingga model dengan cepat mencapai skor tinggi. Pada skenario murni, peningkatan ROUGE berlangsung lebih bertahap dan mencapai tingkat yang lebih rendah pada akhir pelatihan. Pola ini menunjukkan bahwa ketiga skenario berhasil dilatih dengan baik, namun dengan tingkat performa akhir yang berbeda.

Skenario hybrid mencapai skor uji tertinggi pada seluruh metrik, dengan ROUGE-1 sebesar 0,8029 dan ROUGE-L sebesar 0,7976, diikuti G-Translate (0,6457 dan 0,6437) dan murni (0,4035 dan 0,3978). Selisih ini menunjukkan bahwa kombinasi kualitas dan kuantitas data memberikan hasil terbaik: skenario hybrid memperoleh keunggulan volume data dari terjemahan otomatis sekaligus keunggulan kualitas linguistik dari data murni. Temuan ini mengonfirmasi hipotesis penelitian bahwa pendekatan hybrid merupakan strategi pemanfaatan data yang paling seimbang dan efektif untuk pengembangan sistem ringkasan teks Bahasa Jawa sebagai bahasa low-resource.

4.5.2. Analisis Kuantitatif Hasil Ringkasan

Selain evaluasi kuantitatif, dilakukan analisis kuantitatif terhadap contoh ringkasan yang dihasilkan model pada data uji untuk memahami karakteristik output secara lebih mendalam. Analisis ini penting karena metrik ROUGE hanya

mengukur kecocokan kata dan tidak sepenuhnya menggambarkan kualitas semantik maupun masalah teknis pada hasil ringkasan.

Tabel 4. 9 Contoh prediksi model pada skenario murni (S1).

No.	Ringkasan Referensi	Prediksi Model (S1: Murni)
1	"sakudune, aku sing dadi permaisuri. aku kudu nggoleki cara kanggo menyingkirkan permaisuri," pikire.	<extra_id_0> sing ditangkepe.
2	alun-alun lor nglambangke sepi, raeneng bangunan, nggo nyembah tuhan, kita merloke sepi sing jero banget. ana 2 wit ringin sing kembar.	<extra_id_0>-alun lor nglambangke sepi, raeneng bangunan, nggo nyembah tuhan.
3	nalika iku sultan agung manggih akal, ora adoh saka cawangane kali brantas, kali surabaya dibendung ...	<extra_id_0> voc. kena ngedegake loji ana ing jepara.

Tabel 4. 10 Contoh prediksi model pada skenario G-Translate (S2).

No.	Ringkasan Referensi	Prediksi Model (S2: G-Translate)
1	"sakudune, aku sing dadi permaisuri. aku kudu nggoleki cara kanggo menyingkirkan permaisuri," pikire.	<extra_id_0>e, aku sing dadi permaisuri. aku kudu nggoleki cara kanggo menyingkirkan permaisuri," pikire. selir baginda, komplot karo tabib istana. dheweke ethok-ethok lara parah.
2	alun-alun lor nglambangke sepi, raeneng bangunan, nggo nyembah tuhan, kita merloke sepi sing jero banget. ana 2 wit ringin sing kembar.	<extra_id_0>-alun lor nglambangke sepi, raeneng bangunan, nggo nyembah tuhan, kita merloke sepi sing jero banget.
3	nalika iku sultan agung manggih akal, ora adoh saka cawangane kali brantas, kali surabaya dibendung ...	<extra_id_0> iku sultan agung manggih akal, ora adoh saka cawangane kali brantas, kali surabaya dibendung. banyu sethithik kang isih mili menyang surabaya dibuwangi bathang lan bosokaning tetuwuhan, amrih wong surabaya padha ketaman ing lelara.

Tabel 4. 11 Contoh prediksi model pada skenario hybrid (S3).

No.	Ringkasan Referensi	Prediksi Model (S3: Hybrid)
1	"sakudune, aku sing dadi permaisuri. aku kudu nggoleki cara kanggo menyingkirkan permaisuri," pikire.	"sakudune, aku sing dadi permaisuri. aku kudu nggoleki cara kanggo menyingkirkan permaisuri," pikire. selir baginda, komplot karo tabib istana.
2	alun-alun lor nglambangke sepi, raeneng bangunan, nggo nyembah tuhan, kita merloke sepi sing jero banget. ana 2 wit ringin sing kembar.	aku alun-alun lor nglambangke sepi, raeneng bangunan, nggo nyembah tuhan, kita merloke sepi sing jero banget. ana 2 wit ringin sing kembar.
3	nalika iku sultan agung manggih akal, ora adoh saka cawangane kali brantas, kali surabaya dibendung ...	aku sultan agung manggih akal, ora adoh saka cawangane kali brantas, kali surabaya dibendung. banyu sethithik kang isih mili menyang surabaya dibuwangi bathang lan bosokaning tetuwuhan, amrih wong surabaya padha ketaman ing lelara.

Berdasarkan ketiga tabel tersebut, dapat diidentifikasi beberapa karakteristik penting. Pada skenario G-Translate (S2) dan hybrid (S3), model berhasil menghasilkan ringkasan yang sangat mendekati referensi, dengan mempertahankan frasa-frasa kunci dari ringkasan acuan. Hal ini konsisten dengan skor ROUGE kedua skenario yang tinggi. Pada skenario murni (S1), model juga mampu menangkap inti kalimat, namun ringkasan yang dihasilkan cenderung lebih singkat dan sesekali menyimpang dari referensi.

Temuan penting lainnya adalah bahwa pada skenario hybrid (S3), token khusus <extra_id_0> yang sebelumnya muncul pada output skenario murni dan G-Translate sudah tidak lagi muncul. Hal ini menunjukkan bahwa penyempurnaan pada tahap postprocessing berhasil membersihkan token khusus tersebut, sehingga ringkasan yang dihasilkan tampil lebih natural dan langsung dapat dibaca. Perbaikan ini berkontribusi positif terhadap kualitas keterbacaan ringkasan dan

menjadi salah satu faktor yang mendukung perolehan skor tertinggi pada skenario hybrid.

Selain itu, ditemukan satu persoalan teknis yang muncul pada skenario murni dan G-Translate, yaitu kehadiran token khusus <extra_id_0> pada awal hampir seluruh prediksi. Token ini merupakan sentinel token bawaan arsitektur mT5 yang seharusnya dihapus pada tahap postprocessing, namun masih tersisa pada output. Kehadiran token ini turut menurunkan skor ROUGE karena dianggap sebagai token yang tidak cocok dengan referensi. Dengan demikian, analisis kuantitatif ini memberikan arah perbaikan yang konkret, yaitu penyempurnaan postprocessing untuk membersihkan token khusus serta penanganan halusinasi, yang berpotensi meningkatkan skor evaluasi tanpa perlu mengubah arsitektur model.

Selain temuan-temuan di atas, perlu dikemukakan secara terbuka sebuah keterbatasan penting yang berkaitan dengan karakteristik data referensi pada penelitian ini. Berdasarkan pengamatan terhadap contoh prediksi pada Tabel 4.9, 4.10, dan 4.11, terlihat bahwa ringkasan referensi (gold summary) pada dataset secara umum merupakan satu atau dua kalimat pertama yang diambil langsung dari teks sumber, bukan hasil parafrasa atau penulisan ulang oleh ahli bahasa. Akibatnya, tugas yang sesungguhnya dipelajari oleh model lebih mendekati pola peringkasan ekstraktif terhadap bagian awal teks daripada peringkasan abstraktif yang sepenuhnya menulis ulang isi cerita dengan kata-kata baru. Hal ini menjadi salah satu faktor yang menjelaskan mengapa skor ROUGE pada skenario hybrid

dan G-Translate dapat mencapai nilai yang sangat tinggi, karena prediksi model cenderung mencerminkan struktur dan diksi yang sangat mirip dengan referensi.

Dengan demikian, skor ROUGE yang dilaporkan pada penelitian ini sebaiknya dimaknai sebagai indikator kemampuan model untuk mempelajari pola peringkasan yang ada pada data, bukan sebagai pembuktian kemampuan abstraktif murni model mT5 terhadap Bahasa Jawa. Temuan ini sekaligus menjadi dasar penting bagi rekomendasi penelitian lanjutan, yaitu perlunya membangun korpus ringkasan referensi Bahasa Jawa yang ditulis ulang secara manual oleh ahli bahasa agar evaluasi performa model abstraktif menjadi lebih akurat dan mencerminkan kemampuan generatif yang sesungguhnya.

4.6. **Implementasi Antarmuka Sistem**

Sesuai dengan batasan masalah, implementasi sistem dikembangkan hingga tahap antarmuka sederhana yang memungkinkan pengguna memasukkan teks cerita berbahasa Jawa dan memperoleh hasil ringkasan secara langsung. Rancangan antarmuka tersebut ditampilkan pada Gambar 4.7. Antarmuka terdiri atas area input teks, tombol perintah ringkas, dan area output yang menampilkan hasil ringkasan setelah melalui seluruh pipeline mulai dari praproses, tokenisasi, pemodelan, decoding, hingga postprocessing.

Gambar 4. 7 Tampilan antarmuka sederhana sistem ringkasan.

Antarmuka ini berfungsi sebagai bukti konsep (proof of concept) bahwa model yang dikembangkan dapat diintegrasikan ke dalam aplikasi yang dapat digunakan oleh pengguna akhir, seperti pelajar, akademisi, maupun peneliti yang ingin merangkum teks berbahasa Jawa secara cepat. Sesuai batasan penelitian, antarmuka belum mencakup deployment skala luas dan difokuskan pada validasi fungsionalitas inti sistem.

4.7. Integrasi Hasil Penelitian Dengan Nilai Nilai Islam

Hasil penelitian ini tidak hanya memiliki makna dari sisi teknis dan keilmuan, tetapi juga dapat dimaknai dalam perspektif nilai-nilai Islam. Inti dari sistem ringkasan teks otomatis adalah menyampaikan kembali informasi dalam bentuk yang lebih ringkas, jelas, dan tetap menjaga makna utamanya. Prinsip ini memiliki keselarasan yang kuat dengan ajaran Islam tentang penyampaian

informasi yang baik dan efisien, sebagaimana ditegaskan dalam QS. An-Nahl ayat 125 berikut:

اُدْعُ اِلٰى سَبِيْلِ رَبِّكَ بِالْحِكْمَةِ وَالْمَوْعِظَةِ الْحَسَنَةِ وَجَادِلْهُمْ بِالَّتِيْ هِيَ اَحْسَنُۗ اِنَّ رَبَّكَ هُوَ اَعْلَمُ بِمَنْ ضَلَّ عَنْ سَبِيْلِهٖۖ وَهُوَ اَعْلَمُ بِالْمُهْتَدِيْنَ

"Serulah (manusia) kepada jalan Tuhanmu dengan hikmah dan pelajaran yang baik, dan bantahlah mereka dengan cara yang baik. Sesungguhnya Tuhanmu, Dialah yang lebih mengetahui siapa yang sesat dari jalan-Nya dan Dialah yang lebih mengetahui orang-orang yang mendapat petunjuk." (*QS. An-Nahl: 125)*

Sebagaimana telah diuraikan pada Bab I, ayat ini memuat prinsip penyampaian pesan dengan hikmah, yaitu perkataan yang benar, jelas, serta padat namun kuat penjelasannya. Pada bagian ini, prinsip tersebut ditinjau dari sisi perwujudannya dalam hasil penelitian. Sistem ringkasan teks Bahasa Jawa yang dikembangkan pada hakikatnya berupaya menerjemahkan prinsip hikmah tersebut ke dalam bentuk teknologi, yakni menyederhanakan teks cerita yang panjang menjadi ringkasan yang mudah dipahami tanpa kehilangan makna utamanya.

Keterkaitan ini semakin nyata jika dihubungkan dengan hasil eksperimen. Model mampu menghasilkan ringkasan yang sangat padat (rata-rata ringkasan hanya sekitar 25–26 kata dari teks cerita yang rata-rata di atas 900 kata) dengan capaian ROUGE-1 hingga 0,8029 pada skenario hybrid. Upaya memadatkan teks panjang menjadi ringkasan singkat namun tetap mempertahankan inti informasi ini merupakan perwujudan teknis dari prinsip penyampaian yang ringkas namun bermakna. Dengan demikian, sistem yang dikembangkan berusaha menghindari

dua kondisi yang tidak ideal: ringkasan yang terlalu pendek hingga kehilangan makna, maupun penyampaian yang bertele-tele sehingga menyulitkan pemahaman. Prinsip menyampaikan makna yang luas dengan kata-kata yang ringkas ini dalam khazanah keilmuan Islam dikenal dengan istilah ijaz, dan metrik ROUGE yang digunakan untuk menilai sejauh mana ringkasan mempertahankan informasi penting secara tidak langsung mencerminkan upaya menjaga keutuhan makna dalam keringkasan tersebut.

Selain aspek penyampaian informasi, penelitian ini juga memiliki nilai keislaman dari sisi pelestarian bahasa dan budaya. Bahasa Jawa sebagai salah satu bahasa daerah merupakan bagian dari keragaman yang Allah ciptakan, sebagaimana firman Allah dalam QS. Ar-Rum ayat 22 berikut:

وَمِنْ اٰيٰتِهٖ خَلْقُ السَّمٰوٰتِ وَالْاَرْضِ وَاخْتِلَافُ اَلْسِنَتِكُمْ وَاَلْوَانِكُمْۗ اِنَّ فِيْ ذٰلِكَ لَاٰيٰتٍ لِّلْعٰلِمِيْنَ

"Dan di antara tanda-tanda (kebesaran)-Nya ialah penciptaan langit dan bumi serta perbedaan bahasamu dan warna kulitmu. Sesungguhnya pada yang demikian itu benar-benar terdapat tanda-tanda bagi orang-orang yang mengetahui." (QS. Ar-Rum: 22)

Sebagaimana telah diuraikan pada Bab I, ayat ini menegaskan bahwa perbedaan bahasa (ikhtilaf al-alsinah) merupakan salah satu tanda kebesaran Allah yang patut dijaga dan dilestarikan. Pada bagian ini, keterkaitan tersebut ditinjau dari sisi hasil yang telah dicapai. Penelitian ini berhasil membangun sistem ringkasan teks otomatis berbahasa Jawa yang fungsional secara utuh, mulai dari praproses, tokenisasi, pemodelan, decoding, hingga postprocessing, dengan capaian terbaik

pada skenario hybrid sebesar ROUGE-1 0,8029. Keberhasilan membangun sistem Natural Language Processing untuk Bahasa Jawa yang tergolong low-resource ini merupakan wujud konkret dari ikhtiar menjaga dan memuliakan keragaman bahasa ciptaan Allah: bahasa daerah yang selama ini minim representasi di ranah digital kini memiliki perangkat teknologi yang dapat menunjang aksesibilitas sekaligus kelestariannya. Dengan demikian, hasil penelitian ini tidak berhenti pada pencapaian teknis, melainkan juga menjadi sumbangan nyata bagi pelestarian salah satu tanda kebesaran Allah di era digital.

Dengan demikian, penelitian ini menunjukkan bahwa pengembangan teknologi kecerdasan buatan tidak bertentangan dengan nilai-nilai Islam, melainkan dapat menjadi sarana untuk mewujudkan prinsip-prinsip luhur yang diajarkan agama, yaitu penyampaian informasi yang ringkas dan bijaksana (hikmah), penjagaan makna dalam keringkasan (ijaz), serta pelestarian keragaman ciptaan Allah. Integrasi antara ilmu pengetahuan dan nilai keislaman ini memperkuat posisi penelitian sebagai upaya yang tidak hanya bermanfaat secara akademik dan praktis, tetapi juga memiliki landasan nilai dan tujuan yang sejalan dengan ajaran Islam.

BAB V

PENUTUP

5.1. Kesimpulan

Penelitian ini bertolak dari satu rumusan masalah utama, yaitu bagaimana mengembangkan dan mengevaluasi sistem ringkasan teks Bahasa Jawa berbasis Transformer agar mampu menghasilkan ringkasan yang berkualitas. Berdasarkan seluruh tahapan penelitian dan hasil yang telah diuraikan pada bab-bab sebelumnya, rumusan masalah tersebut telah terjawab secara menyeluruh.

Dari sisi pengembangan, sistem ringkasan teks otomatis cerita berbahasa Jawa berhasil dibangun menggunakan model Transformer mT5 (*Multilingual Text-to-Text Transfer Transformer*) melalui pendekatan *fine tuning*. Sistem ini disusun sebagai pipeline yang utuh dan terstruktur, mulai dari praproses teks, tokenisasi berbasis subword dengan SentencePiece Unigram, pemodelan *sequence to sequence, decoding* menggunakan *beam search*, hingga *postprocessing* untuk merapikan hasil ringkasan. Pendekatan ini terbukti mampu mengatasi tantangan Bahasa Jawa sebagai bahasa *low resource,* termasuk variasi tingkat tutur, keberagaman dialek, dan keterbatasan korpus digital, sehingga model dapat menghasilkan ringkasan abstraktif yang tetap menjaga inti informasi cerita.

Dari sisi evaluasi, kualitas ringkasan diukur melalui evaluasi kuantitatif menggunakan metrik ROUGE serta analisis kuantitatif terhadap contoh output model. Pengujian dilakukan pada tiga skenario pemanfaatan data, yaitu skenario murni (S1), G-Translate (S2), dan hybrid (S3). Hasil pengujian pada data uji menunjukkan bahwa skenario hybrid memperoleh skor tertinggi dengan ROUGE-

1 sebesar 0,8029, diikuti skenario G-Translate (0,6457), dan skenario murni (0,4035).

Dengan demikian, dapat disimpulkan bahwa sistem ringkasan teks Bahasa Jawa berbasis Transformer dapat dikembangkan dan dievaluasi untuk menghasilkan ringkasan yang berkualitas, baik dari segi kesesuaian terhadap ringkasan acuan maupun dari segi naturalitas dan keterbacaan bagi pembaca manusia. Selain menjawab rumusan masalah, penelitian ini turut memberikan kontribusi bagi pengembangan Natural Language Processing untuk bahasa daerah berstatus low-resource serta mendukung pelestarian bahasa dan budaya Jawa melalui pemanfaatan teknologi digital. Hasil penelitian ini juga selaras dengan nilai yang terkandung dalam QS. An-Nahl ayat 125 yang menekankan pentingnya penyampaian informasi secara ringkas, jelas, dan bijaksana, sebagaimana yang menjadi tujuan utama sistem peringkasan teks yang dikembangkan.

5.2. **Saran**

Penelitian ini memiliki beberapa keterbatasan yang menjadi dasar perumusan saran berikut. Pertama, jumlah korpus cerita Jawa masih relatif terbatas dan jumlah data antar skenario tidak setara (703 dokumen pada murni, 1.070 pada G-Translate, dan 1.703 pada hybrid), sehingga perbandingan antar skenario perlu memperhatikan faktor volume data. Kedua, tingginya skor pada skenario G-Translate dan hybrid sebagian dipengaruhi oleh sifat ringkasan referensi yang cenderung dekat dengan potongan teks sumber. Ketiga, model masih bergantung pada kualitas data latih sehingga performanya dapat menurun pada teks yang karakteristiknya jauh berbeda dari data pelatihan.

Berdasarkan keterbatasan tersebut, beberapa saran yang dapat diberikan untuk pengembangan pada penelitian selanjutnya adalah sebagai berikut:

1. Menyempurnakan tahap postprocessing untuk menghilangkan token khusus (seperti <extra_id_0>) yang masih tersisa pada output model, karena hal ini berpotensi meningkatkan skor ROUGE dan kealamian ringkasan secara signifikan tanpa perlu mengubah arsitektur model.
2. Memperluas korpus cerita berbahasa Jawa, baik dari segi jumlah maupun keberagaman domain dan tingkat tutur (ngoko, krama, dan krama inggil), agar model dapat mempelajari variasi linguistik yang lebih luas dan meningkatkan kemampuan generalisasi pada data uji.
3. Meningkatkan kualitas dan jumlah ringkasan referensi (gold summary) yang dibuat secara manual oleh ahli bahasa Jawa, sehingga proses pelatihan dan evaluasi model abstraktif menjadi lebih akurat dan andal.
4. Menyetarakan jumlah data antar skenario (murni, G-Translate, dan hybrid) pada penelitian lanjutan, agar perbandingan performa antar skenario menjadi lebih adil dan pengaruh strategi pemanfaatan data dapat dinilai secara terpisah dari pengaruh volume data.
5. Mengeksplorasi penggunaan model Transformer lain yang lebih besar atau lebih spesifik, seperti mT5-large, IndoBART, atau mBART, serta membandingkan performanya dengan mT5-base yang digunakan pada penelitian ini.
6. Menerapkan teknik augmentasi data dan penanganan teks panjang yang lebih canggih, seperti *hierarchical summarization* atau *long-document Transformer*, untuk mengatasi keterbatasan batas token serta mengurangi kesalahan halusinasi pada dokumen yang sangat panjang.
7. Mengembangkan sistem hingga tahap *deployment* yang lebih lengkap, misalnya dalam bentuk aplikasi web atau API publik, sehingga sistem dapat dimanfaatkan secara langsung oleh pelajar, akademisi, dan masyarakat umum untuk merangkum teks berbahasa Jawa.

8. Mengkaji penerapan metode evaluasi tambahan di luar ROUGE, seperti BERTScore atau evaluasi berbasis semantik, untuk memperoleh penilaian kualitas ringkasan yang lebih komprehensif, terutama pada aspek kesesuaian makna.

DAFTAR PUSTAKA

Adelani, D., Alabi, J., Fan, A., Kreutzer, J., Shen, X., Reid, M., Ruiter, D., Klakow, D., Nabende, P., Chang, E., Gwadabe, T., Sackey, F., Dossou, B. F. P., Emezue, C., Leong, C., Beukman, M., Muhammad, S., Jarso, G., Yousuf, O., … Manthalu, S. (2022). A Few Thousand Translations Go a Long Way! Leveraging Pre-trained Models for African News Translation. *Proceedings of the 2022 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, 3053–3070. https://doi.org/10.18653/v1/2022.naacl-main.223

Aji, A. F., Winata, G. I., Koto, F., Cahyawijaya, S., Romadhony, A., Mahendra, R., Kurniawan, K., Moeljadi, D., Prasojo, R. E., Baldwin, T., Lau, J. H., & Ruder, S. (2022). One Country, 700+ Languages: NLP Challenges for Underrepresented Languages and Dialects in Indonesia. *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, 7226–7249. https://doi.org/10.18653/v1/2022.acl-long.500

Alfina, I., Yuliawati, A., Tanaya, D., Dinakaramani, A., & Zeman, D. (2022). A Gold Standard Dataset for Javanese Tokenization, POS Tagging, Morphological Feature Tagging, and Dependency Parsing. *Forum for Linguistic Studies*, *6*(5), 131–148. https://doi.org/10.30564/fls.v6i5.6957

Al-Maraghi, A. M. (1993). *Tafsir Al-Maraghi* (U. & T. Anshori, Trans.). Toha Putra.

Al-Muyassar, T. (2006). *Tafsir Al-Muyassar*. Kompleks Raja Fahd untuk Pencetakan Al-Qur'an.

As-Sa'di, A. N. (2014). *Taisir Al-Karim Ar-Rahman fi Tafsir Kalam Al-Manan (Tafsir As-Sa'di)* (S. Isnan, Trans.). Darul Haq.

Baniata, L. H., Kang, S., & Ampomah, Isaac. K. E. (2022). A Reverse Positional Encoding Multi-Head Attention-Based Neural Machine Translation Model for Arabic Dialects. *Mathematics*, *10*(19), 3666. https://doi.org/10.3390/math10193666

Conneau, A., Khandelwal, K., Goyal, N., Chaudhary, V., Wenzek, G., Guzmán, F., Grave, E., Ott, M., Zettlemoyer, L., & Stoyanov, V. (2020). *Unsupervised Cross-lingual Representation Learning at Scale* (arXiv:1911.02116). arXiv. https://doi.org/10.48550/arXiv.1911.02116

Devlin, J., Chang, M.-W., Lee, K., & Toutanova, K. (2019). *BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding* (arXiv:1810.04805). arXiv. https://doi.org/10.48550/arXiv.1810.04805

Fabbri, A. R., Kryściński, W., McCann, B., Xiong, C., Socher, R., & Radev, D. (2021). SummEval: Re-evaluating Summarization Evaluation. *Transactions*

of the Association for Computational Linguistics, *9*, 391–409. https://doi.org/10.1162/tacl_a_00373

Gatt, A., & Krahmer, E. (2018). Survey of the State of the Art in Natural Language Generation: Core tasks, applications and evaluation. *Journal of Artificial Intelligence Research*, *61*, 65–170. https://doi.org/10.1613/jair.5477

Ghiffari, F. A. A., Alfina, I., & Azizah, K. (2023). Cross-lingual Transfer Learning for Javanese Dependency Parsing. *Proceedings of the 13th International Joint Conference on Natural Language Processing and the 3rd Conference of the Asia-Pacific Chapter of the Association for Computational Linguistics: Student Research Workshop*, 1–9. https://doi.org/10.18653/v1/2023.ijcnlp-srw.1

Hirschberg, J., & Manning, C. D. (n.d.). Advances in natural language processing. *ARTIFICIAL INTELLIGENCE*.

Katsir, I. (2008). *Tafsir Al-Qur'an Al-Azhim (Tafsir Ibnu Katsir)* (M. A. Ghoffar, Trans.). Pustaka Imam Asy-Syafii.

Kryscinski, W., McCann, B., Xiong, C., & Socher, R. (2020). Evaluating the Factual Consistency of Abstractive Text Summarization. *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, 9332–9346. https://doi.org/10.18653/v1/2020.emnlp-main.750

Kudugunta, S., Bapna, A., Caswell, I., & Firat, O. (2019). Investigating Multilingual NMT Representations at Scale. *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, 1565–1575. https://doi.org/10.18653/v1/D19-1167

Ladhak, F., Li, B., Al-Onaizan, Y., & McKeown, K. (2020). Exploring Content Selection in Summarization of Novel Chapters. *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, 5043–5054. https://doi.org/10.18653/v1/2020.acl-main.453

Lewis, M., Liu, Y., Goyal, N., Ghazvininejad, M., Mohamed, A., Levy, O., Stoyanov, V., & Zettlemoyer, L. (2020). BART: Denoising Sequence-to-Sequence Pre-training for Natural Language Generation, Translation, and Comprehension. *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, 7871–7880. https://doi.org/10.18653/v1/2020.acl-main.703

Lin, C.-Y. (n.d.). *ROUGE: A Package for Automatic Evaluation of Summaries*.

Lutfiandri. (2025). *GPT2 Javanese Dataset* [Dataset]. Kaggle. https://www.kaggle.com/datasets/lutfiandri/gpt2-javanese-dataset

Raffel, C., Shazeer, N., Roberts, A., Lee, K., Narang, S., Matena, M., Zhou, Y., Li, W., & Liu, P. J. (2019). *Exploring the Limits of Transfer Learning with a*

Unified Text-to-Text Transformer (Version 4). arXiv. https://doi.org/10.48550/ARXIV.1910.10683

Raganato, A., Pasi, G., & Melzi, S. (2023). Attention And Positional Encoding Are (Almost) All You Need For Shape Matching. *Computer Graphics Forum*, *42*(5), e14912. https://doi.org/10.1111/cgf.14912

Raganato, A., Scherrer, Y., & Tiedemann, J. (2020). *Fixed Encoder Self-Attention Patterns in Transformer-Based Machine Translation* (arXiv:2002.10260). arXiv. https://doi.org/10.48550/arXiv.2002.10260

Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., Kaiser, L., & Polosukhin, I. (2017). *Attention Is All You Need* (Version 7). arXiv. https://doi.org/10.48550/ARXIV.1706.03762

Wang, R., Tan, X., Luo, R., Qin, T., & Liu, T.-Y. (2021). A Survey on Low-Resource Neural Machine Translation. *Proceedings of the Thirtieth International Joint Conference on Artificial Intelligence*, 4636–4643. https://doi.org/10.24963/ijcai.2021/629

Young, T., Hazarika, D., Poria, S., & Cambria, E. (2018). *Recent Trends in Deep Learning Based Natural Language Processing* (arXiv:1708.02709). arXiv. https://doi.org/10.48550/arXiv.1708.02709

Zhang, J., Zhao, Y., Saleh, M., & Liu, P. J. (2020). *PEGASUS: Pre-training with Extracted Gap-sentences for Abstractive Summarization* (arXiv:1912.08777). arXiv. https://doi.org/10.48550/arXiv.1912.08777

Zhong, M., Liu, P., Chen, Y., Wang, D., Qiu, X., & Huang, X. (2020). Extractive Summarization as Text Matching. *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, 6197–6208. https://doi.org/10.18653/v1/2020.acl-main.552

LAMPIRAN

Log Pelatihan dan Hasil Evaluasi Model — Skenario Murni (S1)

A. Pembagian Dataset

Subset Data	Jumlah Dokumen
Data latih (train)	569
Data validasi (validation)	64
Data uji (test)	70
Total	**703**

B. Nilai Loss dan Skor ROUGE Validasi per Epoch (8 epoch)

Epoch	Training Loss	Validation Loss	ROUGE-1	ROUGE-2	ROUGE-L	ROUGE-Lsum
1	287,48	6,8813	0,0314	0,0100	0,0278	0,0278
2	182,42	2,8107	0,0498	0,0125	0,0472	0,0474
3	90,56	0,9437	0,1210	0,0657	0,1097	0,1111
4	63,81	0,6462	0,2121	0,1491	0,1936	0,1943
5	39,51	0,5644	0,2719	0,2159	0,2617	0,2637
6	34,01	0,4874	0,4055	0,3563	0,3930	0,3969
7	29,66	0,4558	0,4172	0,3758	0,4058	0,4066
8	27,12	0,4450	0,4377	0,3941	0,4280	0,4292

Catatan: skor ROUGE pada tabel ini dihitung terhadap data validasi di setiap epoch.

C. Skor ROUGE pada Data Uji (Test Set)

Metrik	Skor
ROUGE-1	0,4035
ROUGE-2	0,3662
ROUGE-L	0,3978
ROUGE-Lsum	0,3989

D. Contoh Keluaran Ringkasan Model (3 Sampel)

Sampel #1

Cerita (input): “sakudune, aku sing dadi permaisuri. aku kudu nggoleki cara kanggo menyingkirkan permaisuri,” pikire. Selir baginda, komplot karo tabib istana. dheweke ethok-ethok lara parah. Tabib istana banjur cepet diceluk. Sang tabib ngomongke menawa ana wong si…

Referensi (gold summary): “sakudune, aku sing dadi permaisuri. aku kudu nggoleki cara kanggo menyingkirkan permaisuri,” pikire.

Prediksi (keluaran model): <extra_id_0> sing ditangkepe.

Sampel #2

Cerita (input): alun-alun lor nglambangke sepi, raeneng bangunan, nggo nyembah Tuhan, kita merloke sepi sing jero banget. Ana 2 wit ringin sing kembar. nggambarke manungsa nalika nyembah Tuhan kudu nduweni sifat kembar utawa ucul seka jasade. nalika ngalor nemoni pa…

Referensi (gold summary): alun-alun lor nglambangke sepi, raeneng bangunan, nggo nyembah Tuhan, kita merloke sepi sing jero banget. Ana 2 wit ringin sing kembar.

Prediksi (keluaran model): <extra_id_0>-alun lor nglambangke sepi, raeneng bangunan, nggo nyembah Tuhan.

Sampel #3

Cerita (input): Nalika iku Sultan Agung manggih akal, ora adoh saka cawangane kali Brantas, kali Surabaya dibendung. Banyu sethithik kang isih mili menyang Surabaya dibuwangi bathang lan bosokaning tetuwuhan, amrih wong Surabaya padha ketaman ing lelara. Jalaran sak…

Referensi (gold summary): Nalika iku Sultan Agung manggih akal, ora adoh saka cawangane kali Brantas, kali Surabaya dibendung. Banyu sethithik kang isih mili menyang Surabaya dibuwangi bathang lan bosokaning tetuwuhan, amrih wong Surabaya padha ketaman ing lelara.

Prediksi (keluaran model): <extra_id_0> VOC. kena ngedegake loji ana ing Jepara.

Catatan: token <extra_id_0> pada keluaran model merupakan sisa token khusus mT5 yang belum terhapus pada tahap postprocessing, sesuai keterbatasan yang dibahas pada BAB V.

Log Pelatihan dan Hasil Evaluasi Model — Skenario G-Translate (S2)

A. Pembagian Dataset

Subset Data	Jumlah Dokumen
Data latih (train)	900
Data validasi (validation)	100
Data uji (test)	70
Total	**1070**

B. Nilai Loss dan Skor ROUGE Validasi per Epoch (8 epoch)

Epoch	Training Loss	Validation Loss	ROUGE-1	ROUGE-2	ROUGE-L	ROUGE-Lsum
1	174,98	2,9181	0,0432	0,0090	0,0414	0,0415
2	79,08	0,7644	0,2861	0,2136	0,2696	0,2706
3	33,82	0,5363	0,5764	0,5351	0,5629	0,5635
4	19,04	0,4535	0,6117	0,5768	0,6029	0,6014
5	15,67	0,4006	0,6389	0,6076	0,6313	0,6311
6	14,09	0,3688	0,6690	0,6386	0,6605	0,6607
7	12,69	0,3512	0,6650	0,6321	0,6555	0,6555
8	12,28	0,3431	0,6823	0,6527	0,6744	0,6742

Catatan: skor ROUGE pada tabel ini dihitung terhadap data validasi di setiap epoch.

C. Skor ROUGE pada Data Uji (Test Set)

Metrik	Skor
ROUGE-1	0,6457
ROUGE-2	0,6188
ROUGE-L	0,6437
ROUGE-Lsum	0,6426

D. Contoh Keluaran Ringkasan Model (3 Sampel)

Sampel #1

Cerita (input): “sakudune, aku sing dadi permaisuri. aku kudu nggoleki cara kanggo menyingkirkan permaisuri,” pikire. Selir baginda, komplot karo tabib istana. dheweke ethok-ethok lara parah. Tabib istana banjur cepet diceluk. Sang tabib ngomongke menawa ana wong si…

Referensi (gold summary): “sakudune, aku sing dadi permaisuri. aku kudu nggoleki cara kanggo menyingkirkan permaisuri,” pikire.

Prediksi (keluaran model): <extra_id_0>e, aku sing dadi permaisuri. aku kudu nggoleki cara kanggo menyingkirkan Permaisuri,” pikire. Selir baginda, komplot karo tabib istana. dheweke ethok-ethok lara parah.

Sampel #2

Cerita (input): alun-alun lor nglambangke sepi, raeneng bangunan, nggo nyembah Tuhan, kita merloke sepi sing jero banget. Ana 2 wit ringin sing kembar. nggambarke manungsa nalika nyembah Tuhan kudu nduweni sifat kembar utawa ucul seka jasade. nalika ngalor nemoni pa…

Referensi (gold summary): alun-alun lor nglambangke sepi, raeneng bangunan, nggo nyembah Tuhan, kita merloke sepi sing jero banget. Ana 2 wit ringin sing kembar.

Prediksi (keluaran model): <extra_id_0>-alun lor nglambangke sepi, raeneng bangunan, nggo nyembah Tuhan, kita merloke sepi sing jero banget.

Sampel #3

Cerita (input): Nalika iku Sultan Agung manggih akal, ora adoh saka cawangane kali Brantas, kali Surabaya dibendung. Banyu sethithik kang isih mili menyang Surabaya dibuwangi bathang lan bosokaning tetuwuhan, amrih wong Surabaya padha ketaman ing lelara. Jalaran sak…

Referensi (gold summary): Nalika iku Sultan Agung manggih akal, ora adoh saka cawangane kali Brantas, kali Surabaya dibendung. Banyu sethithik kang isih mili menyang Surabaya dibuwangi bathang lan bosokaning tetuwuhan, amrih wong Surabaya padha ketaman ing lelara.

Prediksi (keluaran model): <extra_id_0> iku Sultan Agung manggih akal, ora adoh saka cawangane kali Brantas, kali Surabaya dibendung. Banyu sethithik kang isih mili menyang Surabaya dibuwangi bathang lan bosokaning tetuwuhan, amrih wong Surabaya padha ketaman ing lelara.

Catatan: token <extra_id_0> pada keluaran model merupakan sisa token khusus mT5 yang belum terhapus pada tahap postprocessing, sesuai keterbatasan yang dibahas pada BAB V.

Log Pelatihan dan Hasil Evaluasi Model — Skenario Hybrid (S3)

A. Pembagian Dataset

Subset Data	Jumlah Dokumen
Data latih (train)	1469
Data validasi (validation)	164
Data uji (test)	70
Total	**1703**

B. Nilai Loss dan Skor ROUGE Validasi per Epoch (8 epoch)

Epoch	Training Loss	Validation Loss	ROUGE-1	ROUGE-2	ROUGE-L	ROUGE-Lsum
1	94,74	0,7879	0,2253	0,1650	0,2078	0,2079
2	23,31	0,4192	0,6295	0,5988	0,6210	0,6218
3	14,50	0,2972	0,7150	0,6903	0,7077	0,7072
4	11,72	0,2494	0,7452	0,7257	0,7400	0,7405
5	9,78	0,2195	0,7822	0,7672	0,7784	0,7786
6	8,84	0,2087	0,8285	0,8158	0,8257	0,8266
7	8,16	0,2035	0,8492	0,8368	0,8470	0,8471
8	8,29	0,1982	0,8537	0,8427	0,8521	0,8516

Catatan: skor ROUGE pada tabel ini dihitung terhadap data validasi di setiap epoch.

C. Skor ROUGE pada Data Uji (Test Set)

Metrik	Skor
ROUGE-1	0,8029
ROUGE-2	0,7745
ROUGE-L	0,7976
ROUGE-Lsum	0,7980

D. Contoh Keluaran Ringkasan Model (3 Sampel)

Sampel #1

Cerita (input): “sakudune, aku sing dadi permaisuri. aku kudu nggoleki cara kanggo menyingkirkan permaisuri,” pikire. Selir baginda, komplot karo tabib istana. dheweke ethok-ethok lara parah. Tabib istana banjur cepet diceluk. Sang tabib ngomongke menawa ana wong si…

Referensi (gold summary): “sakudune, aku sing dadi permaisuri. aku kudu nggoleki cara kanggo menyingkirkan permaisuri,” pikire.

Prediksi (keluaran model): “sakudune, aku sing dadi permaisuri. aku kudu nggoleki cara kanggo menyingkirkan Permaisuri,” pikire. Selir baginda, komplot karo tabib istana.

Sampel #2

Cerita (input): alun-alun lor nglambangke sepi, raeneng bangunan, nggo nyembah Tuhan, kita merloke sepi sing jero banget. Ana 2 wit ringin sing kembar. nggambarke manungsa nalika nyembah Tuhan kudu nduweni sifat kembar utawa ucul seka jasade. nalika ngalor nemoni pa…

Referensi (gold summary): alun-alun lor nglambangke sepi, raeneng bangunan, nggo nyembah Tuhan, kita merloke sepi sing jero banget. Ana 2 wit ringin sing kembar.

Prediksi (keluaran model): Aku alun-alun lor nglambangke sepi, raeneng bangunan, nggo nyembah Tuhan, kita merloke sepi sing jero banget. Ana 2 wit ringin sing kembar.

Sampel #3

Cerita (input): Nalika iku Sultan Agung manggih akal, ora adoh saka cawangane kali Brantas, kali Surabaya dibendung. Banyu sethithik kang isih mili menyang Surabaya dibuwangi bathang lan bosokaning tetuwuhan, amrih wong Surabaya padha ketaman ing lelara. Jalaran sak…

Referensi (gold summary): Nalika iku Sultan Agung manggih akal, ora adoh saka cawangane kali Brantas, kali Surabaya dibendung. Banyu sethithik kang isih mili menyang Surabaya dibuwangi bathang lan bosokaning tetuwuhan, amrih wong Surabaya padha ketaman ing lelara.

Prediksi (keluaran model): Aku Sultan Agung manggih akal, ora adoh saka cawangane kali Brantas, kali Surabaya dibendung. Banyu sethithik kang isih mili menyang Surabaya dibuwangi bathang lan bosokaning tetuwuhan, amrih wong Surabaya padha ketaman ing lelara.

Catatan: token <extra_id_0> pada keluaran model merupakan sisa token khusus mT5 yang belum terhapus pada tahap postprocessing, sesuai keterbatasan yang dibahas pada BAB V.