202110370311193
Muhammad Bima Mauludin
Prodi Informatika

BAB II
TINJAUAN PUSTAKA

2.1. Transkripsi Video

Transkripsi video merupakan proses konversi sinyal audio dalam video menjadi bentuk teks menggunakan teknologi *Automatic Speech Recognition* (ASR) [5]. Proses ini memungkinkan konten audio yang bersifat lisan untuk dianalisis lebih lanjut menggunakan teknik pemrosesan bahasa alami. Dalam konteks penelitian ini, transkripsi video berperan sebagai sumber data utama yang akan diringkas secara otomatis.

Teks hasil transkripsi video memiliki karakteristik yang berbeda dibandingkan teks tertulis formal [13]. Transkripsi umumnya mengandung pengulangan kata, jeda pembicaraan, kalimat tidak lengkap, serta perpindahan topik yang tidak terstruktur. Selain itu, variasi jenis komunikasi seperti monolog, wawancara, dan podcast menyebabkan perbedaan pola bahasa yang signifikan [12]. Oleh karena itu, transkripsi video tergolong sebagai teks panjang dan kompleks yang membutuhkan pendekatan khusus dalam proses peringkasan.

2.2. Peringkasan Teks (*Text Summarization*)

Peringkasan teks (*text summarization*) merupakan salah satu bidang kajian penting dalam *Natural Language Processing* (NLP) yang bertujuan untuk menghasilkan ringkasan dari dokumen teks dengan tetap mempertahankan informasi utama dan makna esensial dari teks asli [14] [15]. Peringkasan teks banyak diterapkan pada dokumen berdurasi panjang seperti artikel berita, dokumen ilmiah, dan transkripsi video, guna membantu pengguna memahami inti pembahasan secara lebih cepat dan efisien [16]. Seiring meningkatnya volume data teks tidak terstruktur, kebutuhan terhadap metode peringkasan otomatis yang akurat dan efisien menjadi semakin signifikan [17]. Tantangan utama dalam peringkasan teks terletak pada kemampuan

model untuk mengidentifikasi informasi penting, mengurangi redundansi, serta menjaga koherensi dan keterbacaan ringkasan yang dihasilkan.

Secara umum, pendekatan peringkasan teks terbagi menjadi dua kategori utama, yaitu peringkasan ekstraktif dan peringkasan abstraktif [18]. Peringkasan ekstraktif bekerja dengan memilih kalimat-kalimat penting langsung dari teks sumber berdasarkan kriteria tertentu, sehingga relatif stabil dalam mempertahankan kesesuaian informasi dengan dokumen asli [19]. Namun, pendekatan ini sering menghasilkan ringkasan yang kurang mengalir karena hanya berupa kumpulan kalimat terpilih. Sebaliknya, peringkasan abstraktif berupaya menyusun kembali informasi utama dengan menghasilkan kalimat baru yang lebih ringkas dan natural, menyerupai cara manusia merangkum teks . Meskipun demikian, pendekatan abstraktif memiliki tantangan tersendiri, terutama ketika diterapkan pada teks panjang dan kompleks seperti transkripsi video, karena berisiko menghilangkan informasi penting atau menghasilkan ringkasan yang kurang akurat [20]. Oleh karena itu, banyak penelitian terkini mengembangkan pendekatan hibrida yang mengombinasikan keunggulan kedua metode tersebut untuk menghasilkan ringkasan yang lebih informatif dan koheren .

2.3. BERT untuk Representasi Kalimat

BERT (*Bidirectional Encoder Representations from Transformers*) merupakan model representasi bahasa berbasis arsitektur transformer yang dirancang untuk memahami konteks kata secara dua arah, yaitu dengan mempertimbangkan informasi dari kata-kata sebelum dan sesudahnya secara simultan . Pendekatan bidirectional ini memungkinkan BERT menghasilkan representasi semantik yang lebih kaya dibandingkan model bahasa sekuensial tradisional yang hanya memproses teks secara satu arah. Dengan mekanisme *self-attention*, BERT mampu mempelajari hubungan antar kata dalam suatu kalimat maupun antar kalimat dalam sebuah dokumen, sehingga sangat efektif dalam menangkap makna kontekstual teks. Sejak diperkenalkan, BERT

telah menjadi fondasi bagi berbagai tugas *Natural Language Processing* seperti klasifikasi teks, *question answering*, dan peringkasan teks.

Dalam konteks representasi kalimat, BERT digunakan untuk mengubah setiap kalimat menjadi vektor embedding berdimensi tetap yang merepresentasikan makna semantik kalimat tersebut. Representasi ini memungkinkan pengukuran kemiripan antar kalimat secara lebih akurat menggunakan metrik seperti *cosine similarity*. Pada penelitian peringkasan teks, khususnya peringkasan ekstraktif, embedding kalimat berbasis BERT berperan penting dalam mengidentifikasi kalimat-kalimat yang memiliki kemiripan makna maupun perbedaan topik. Dengan memanfaatkan embedding BERT sebagai dasar klasterisasi, proses seleksi kalimat representatif dapat dilakukan secara lebih objektif dan kontekstual, sehingga mampu mengurangi redundansi dan meningkatkan cakupan informasi dalam ringkasan akhir .

2.4. Algoritma Hierarchical Clustering

Hierarchical clustering merupakan salah satu metode pengelompokan data (unsupervised learning) yang membangun struktur klaster secara bertingkat dalam bentuk hierarki atau dendrogram [21]. Berbeda dengan metode partisi seperti K-Means, hierarchical clustering tidak memerlukan penentuan jumlah klaster di awal. Proses pengelompokan dilakukan secara bertahap dengan menggabungkan atau memisahkan data berdasarkan tingkat kemiripan tertentu hingga terbentuk struktur hierarki yang lengkap . Pada penelitian ini digunakan pendekatan *agglomerative hierarchical clustering*, yaitu metode bottom-up yang dimulai dengan menganggap setiap data sebagai satu klaster tersendiri, kemudian secara iteratif menggabungkan dua klaster terdekat hingga seluruh data berada dalam satu struktur hierarki.

Jarak antar klaster pada hierarchical clustering dapat dihitung menggunakan berbagai metode *linkage*, seperti *single linkage*, *complete linkage*, *average linkage*, atau *ward linkage*. Secara umum, proses penggabungan klaster didasarkan pada perhitungan jarak antar vektor data, yang dapat dirumuskan menggunakan jarak Euclidean sebagaimana ditunjukkan pada **Eq. 2.1** berikut.

$$d(x_i, x_j) = \sqrt{\sum_{k=1}^{n} (x_{ik} - x_{jk})^2}$$

Dengan x_i dan x_j merupakan dua vektor data, dan n menyatakan jumlah dimensi vektor. Nilai jarak ini digunakan sebagai dasar untuk menentukan pasangan klaster yang memiliki tingkat kemiripan tertinggi untuk digabungkan pada setiap iterasi.

Dalam konteks peringkasan teks, hierarchical clustering dimanfaatkan untuk mengelompokkan kalimat-kalimat berdasarkan kesamaan representasi semantik yang diperoleh dari embedding BERT. Struktur hierarki yang terbentuk memungkinkan pengelompokan topik secara lebih fleksibel dan adaptif terhadap kompleksitas teks. Dari setiap klaster, dipilih satu kalimat yang paling representatif sebagai kandidat ringkasan. Pendekatan ini efektif dalam mengurangi redundansi, menjaga keberagaman informasi, serta memastikan bahwa ringkasan mencerminkan struktur dan isi utama dokumen secara menyeluruh .

2.5. BART untuk Peringkasan Abstraktif

BART (*Bidirectional and Auto-Regressive Transformers*) merupakan model berbasis arsitektur *encoder–decoder* yang dikembangkan untuk menghasilkan ringkasan abstraktif dalam bentuk kalimat baru yang lebih natural dan mudah dipahami [22] [23] . Pada tahap *encoding*, model memproses seluruh teks masukan secara dua arah guna menangkap konteks global dari setiap token. Selanjutnya, *decoder* menghasilkan ringkasan secara bertahap melalui mekanisme autoregresif, dengan memanfaatkan representasi kontekstual yang dihasilkan oleh encoder. Secara umum, encoder bertugas mentransformasikan setiap token masukan menjadi representasi kontekstual yang kaya makna sebagai dasar proses generasi ringkasan.

Dari sisi arsitektur, BART mengadopsi konsep *denoising autoencoder*, di mana model dilatih untuk merekonstruksi teks asli dari masukan yang telah mengalami gangguan buatan. Gangguan tersebut diterapkan melalui berbagai strategi, seperti *token*

masking, *token deletion*, dan *sentence shuffling* . Pendekatan ini membuat BART lebih robust dalam memahami struktur bahasa serta lebih adaptif dalam menghasilkan ringkasan abstraktif [24]. Hal ini disebabkan oleh kemampuan model dalam menangkap konteks global melalui encoder, sekaligus menghasilkan teks secara bertahap dan autoregresif melalui mekanisme decoder [22].

Dalam proses pelatihan, BART mengoptimalkan fungsi *loss* berbasis **Negative Log-Likelihood (NLL)** antara keluaran decoder dan token referensi. Fungsi *loss* ini bertujuan untuk memaksimalkan probabilitas model dalam menghasilkan urutan token yang sesuai dengan ringkasan referensi. Secara matematis, fungsi *loss* Negative Log-Likelihood dapat dirumuskan seperti **Eq.2.2** sebagai berikut:

$$\mathcal{L}_{NLL} = -\sum_{t=1}^{T} \log P\ (y_t \mid y_{<t}\ , x)$$

dengan y_t menyatakan token referensi pada langkah ke t, $y_{<t}$merupakan urutan token yang telah dihasilkan sebelumnya, x adalah teks input, dan T adalah panjang urutan token. Melalui optimasi fungsi *loss* ini, model BART dilatih untuk menghasilkan ringkasan yang semakin mendekati ringkasan referensi dengan meminimalkan perbedaan probabilistik antara prediksi model dan data target.

Selain itu, mekanisme perhatian (*self-attention* dan *cross-attention*) memegang peranan penting dalam arsitektur BART. Mekanisme ini memungkinkan model untuk memusatkan perhatian pada bagian-bagian teks yang paling relevan selama proses encoding maupun decoding [22] . *Self-attention* digunakan untuk memodelkan hubungan antar token dalam satu urutan teks, sedangkan *cross-attention* memungkinkan decoder memanfaatkan informasi dari representasi encoder. Dengan mekanisme ini, BART mampu menangkap dependensi jangka panjang serta hubungan kontekstual antar token secara lebih efektif, yang sangat penting dalam tugas peringkasan teks.

Mekanisme perhatian yang digunakan dalam BART adalah **scaled dot-product attention**, yang secara matematis dapat dirumuskan seperti **Eq.2.3** sebagai berikut:

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

dengan Q ***(query)***, K ***(key)***, dan V ***(value)*** merupakan matriks hasil transformasi embedding input, sedangkan d_k menyatakan dimensi vektor key. Faktor penskalaan **akar dari** d_k digunakan untuk menjaga stabilitas nilai gradien selama proses pelatihan. Melalui mekanisme ini, model dapat menentukan bobot perhatian terhadap setiap token berdasarkan tingkat relevansinya, sehingga menghasilkan representasi kontekstual yang lebih informatif.

Dengan kombinasi encoder bidirectional, decoder autoregresif, dan mekanisme attention, BART terbukti efektif dalam berbagai tugas peringkasan abstraktif dengan menghasilkan ringkasan yang lebih koheren, ringkas, dan mendekati gaya bahasa manusia .

2.6. Metrik Evaluasi ROUGE

Evaluasi terhadap hasil peringkasan otomatis umumnya dilakukan dengan memanfaatkan metrik **ROUGE** (*Recall-Oriented Understudy for Gisting Evaluation*) [26]. Metrik ini digunakan untuk mengukur tingkat kesesuaian antara ringkasan yang dihasilkan oleh sistem dan ringkasan acuan (*reference summary*) dengan melihat kemiripan pola n-gram, kesamaan urutan kata, serta tingkat tumpang tindih frasa yang muncul pada kedua ringkasan tersebut.

Beberapa varian metrik ROUGE yang umum digunakan dalam evaluasi peringkasan teks meliputi **ROUGE-1**, **ROUGE-2**, dan **ROUGE-L [ ]**. ROUGE-1 digunakan untuk mengukur tingkat kemiripan unigram atau kata tunggal antara ringkasan sistem dan ringkasan acuan, sementara ROUGE-2 berfokus pada kesesuaian bigram atau pasangan kata [23]. Adapun ROUGE-L mengevaluasi kesamaan ringkasan berdasarkan konsep *Longest Common Subsequence* (LCS), yang mempertimbangkan

keselarasan urutan kata tanpa harus berurutan secara ketat . Secara umum, nilai ROUGE dihitung menggunakan rumus yang ditunjukkan pada **Eq. 2.4** berikut.

$$ROUGE - N = \frac{\sum_{S \in (Reference\ Summaries)} \sum_{gram_n \in S} Count_{match}(gram_n)}{\sum_{S \in (Reference\ Summaries)} \sum_{gram_n \in S} Count(gram_n)}$$

Dengan N menyatakan panjang n-gram, $Count_{match}(gram_n)$menunjukkan jumlah n-gram yang sama antara ringkasan yang dihasilkan sistem dan ringkasan acuan, sedangkan $Count(gram_n)$merepresentasikan jumlah keseluruhan n-gram yang terdapat dalam ringkasan acuan.

Metrik ROUGE banyak dimanfaatkan dalam penelitian peringkasan teks karena mampu memberikan penilaian kuantitatif yang bersifat objektif terhadap kualitas ringkasan yang dihasilkan [23] . Pada penelitian ini, ROUGE digunakan untuk menilai tingkat kesesuaian antara ringkasan otomatis yang dihasilkan oleh metode ekstraktif dan ringkasan manual sebagai acuan.