BAB II
TINJAUAN PUSTAKA & DASAR TEORI

2.1 Tinjauan Pustaka

Penambangan teks (*text mining*) adalah proses mengubah data teks tidak terstruktur menjadi data terstruktur menggunakan NLP untuk mengekstraksi informasi berharga seperti kata kunci, pola, dan topik. Seiring berkembangnya *machine learning*, muncul teknik pemodelan topik, sebuah metode statistik *unsupervised* yang secara otomatis mengidentifikasi tema atau topik tersembunyi dalam kumpulan dokumen besar. Dengan mengelompokkan pola kata dan frasa serupa, pemodelan topik mengklasifikasikan dokumen berdasarkan topiknya, yang pada akhirnya sangat membantu pengguna dalam mengatur, memahami, serta mengoptimalkan proses pencarian dalam arsip elektronik yang besar (Goyal dan Kasyap, 2024).

Pendekatan yang umum digunakan untuk pemodelan topik meliputi *Latent Dirichlet Allocation* (LDA) dan *Non-negative Matrix Factorization* (NMF). Perkembangan lebih lanjut dalam metode ini mencakup *SeaNMF*, yang mengintegrasikan korelasi semantik, dan *KGNMF*, yang memanfaatkan pengetahuan eksternal untuk meningkatkan efisiensi serta interpretabilitas model. Selain itu, GSDMM (*Gibbs Sampling Dirichlet Mixture Model*) merupakan pengembangan dari LDA yang dirancang khusus untuk menangani tantangan teks pendek, seperti masalah *sparsity* (ketersebaran data) dan data berdimensi tinggi, dengan asumsi bahwa satu dokumen hanya memiliki satu topik tunggal (Udupa dkk., 2022).

Studi-studi terkini bahkan telah mengeksplorasi penerapan model BERT untuk pemodelan topik, seperti *BERTopic*, yang terbukti menunjukkan hasil menjanjikan, khususnya dalam analisis data media sosial (de Araújo dkk., 2024). Selain untuk mengidentifikasi tema tersembunyi seperti yang dilakukan oleh pemodelan topik, kemajuan pesat dalam NLP juga secara khusus berfokus pada analisis opini dan emosi. Bidang ini dikenal sebagai analisis sentimen. Analisis sentimen telah menjadi salah satu pendorong utama pesatnya perkembangan teknologi NLP belakangan ini. Evolusi metodenya terlihat jelas, bergerak dari

pendekatan standar menggunakan *machine learning* sederhana seperti *Naïve Bayes* atau SVM, menuju teknik *deep learning* berbasis *Recurrent Neural Networks* (RNN) dan *Convolutional Neural Network* (CNN). Puncaknya, metode terbaru seperti *transformer networks*, yang meliputi model BERT dan *Generative Pre-trained Transformer* (GPT), telah menunjukkan peningkatan performa yang substansial. Menurut Min dkk. (2021), kemampuan model-model mutakhir ini dalam memproses konteks secara mendalam telah mendorong lonjakan performa yang signifikan dalam menyelesaikan tugas-tugas NLP, termasuk analisis sentimen.

Penerapan NLP yang menggabungkan pemodelan topik dan analisis sentimen telah terbukti efektif dalam menggali wawasan dari ulasan pengguna, seperti yang dilakukan oleh Acheadeth dkk. (2022) pada domain *e-commerce* dengan *dataset* ulasan produk bayi. Dalam studi tersebut, peneliti mengembangkan metode hibrida yang memadukan *Latent Dirichlet Allocation* (LDA) dengan algoritma klasifikasi konvensional (SVM, *Logistic Regression*, dan MLP). Penelitian ini secara spesifik membuktikan bahwa kombinasi LDA dengan fitur TF-IDF jauh lebih unggul dibandingkan *Bag of Words* (BOW), di mana integrasi topik sukses meningkatkan akurasi dasar dari 0,61 menjadi 0,78. Di antara model yang diuji, algoritma MLP mencatatkan performa terbaik dengan akurasi 0,79 dan waktu eksekusi tercepat (5,56 detik). Namun, validitas studi ini terkendala oleh volume *dataset* yang sangat kecil (hanya 413 data anotasi manual) yang dianggap belum memenuhi standar akibat masalah ketidakseimbangan data. Merespons keterbatasan tersebut, penelitian ini mengusulkan penggunaan *dataset* yang jauh lebih besar (>10.000 data) dari media sosial X terkait kebijakan pemerintah, serta memperluas komparasi model topik tidak hanya terbatas pada LDA, melainkan juga mencakup GSDMM dan *BERTopic* untuk evaluasi yang lebih mendalam.

Dalam konteks domain kebijakan pemerintah di Indonesia, pendekatan *machine learning* juga telah diterapkan oleh Sukma dkk. (2020) untuk menganalisis respons publik terhadap kebijakan *Omnibus Law*. Menggunakan 8.344 *tweet* valid yang telah melalui lima tahapan prapemrosesan (termasuk *stopword removal* dan *stemming*) serta pembobotan TF-IDF, studi ini membandingkan performa *Naïve Bayes*, *Decision Tree*, dan SVM dengan skema pembagian data 70:30. Hasilnya

menunjukkan dominasi algoritma SVM yang mencapai akurasi 91,80%, presisi 89,81%, dan AUC 97,90%, mengungguli *Naïve Bayes* (89,75%) dan *Decision Tree* (73,60%). Hal ini mengindikasikan bahwa pada volume data yang cukup, algoritma konvensional mampu memberikan hasil yang baik.

Namun, tantangan signifikan ditemukan pada studi dengan volume data terbatas seperti yang dilakukan Azis dan Wahyudi (2024) mengenai pemindahan Ibu Kota Nusantara (IKN). Meskipun model *Naïve Bayes* yang dibangun menggunakan metodologi CRISP-DM pada 375 data bersih menghasilkan akurasi keseluruhan yang tinggi (92,82%), evaluasi mendalam menyingkap kelemahan fatal pada deteksi sentimen negatif. Model tersebut memiliki *recall* negatif yang sangat rendah (48,15%) dan presisi hanya 65,00%, yang berarti model gagal mengenali lebih dari separuh opini negatif yang ada. Selain itu, studi ini hanya menggunakan klasifikasi biner dan mengabaikan kelas netral. Berdasarkan celah tersebut, penelitian ini akan beralih dari metode tradisional ke pendekatan *deep learning* menggunakan model *transformer* (BERT) dengan skema tiga kelas (positif, negatif, netral), yang diharapkan mampu mengatasi masalah rendahnya sensitivitas model terhadap sentimen minoritas dan memberikan klasifikasi yang lebih presisi.

Selanjutnya, penelitian terkait dengan pemodelan topik mulai bergerak dari pendekatan statistik probabilistik menuju arsitektur berbasis *deep learning*. Penelitian Nanayakkara dan Thennakoon (2024) menjadi bukti empiris awal yang menyoroti transisi ini dalam konteks media sosial. Studi mereka pada data komentar YouTube mengungkap bahwa meskipun metode faktorisasi matriks seperti NMF mampu menangani data yang "kotor" lebih baik daripada *Latent Dirichlet Allocation* (LDA), pendekatan berbasis *transformer* (*BERTopic*) tetap mendominasi. Keunggulan ini didorong oleh pemanfaatan variasi *sentence embeddings* yang memungkinkan model menangkap konteks semantik laten dan inferensi bahasa alami, menghasilkan skor koherensi tertinggi (0,5056) yang sulit dicapai oleh model probabilistik standar. Temuan ini diperkuat secara signifikan oleh Lande, Pillay, dan Chandra (2023) yang mengembangkan kerangka kerja *BERT-TM* pada data Twitter selama pandemi. Hasil eksperimen mereka

menunjukkan disparitas performa yang tajam, di mana model berbasis BERT mencapai skor NPMI 0,69, jauh meninggalkan metode khusus teks pendek seperti GSDMM (0,39) dan LDA (0,34), membuktikan superioritas model *neural* dalam mengatasi masalah *sparsity* (data jarang) pada teks pendek.

Selain unggul dalam akurasi semantik, efektivitas model *neural* juga terbukti dari aspek efisiensi operasional dan stabilitas. Krishnan dan Kennedyraj (2023) serta Chen dkk. (2023) dalam studi terpisah pada domain ulasan aplikasi pemerintah dan berita keuangan menemukan benang merah yang sama: *BERTopic* meminimalkan beban pra pemrosesan. Berbeda dengan LDA yang sangat sensitif terhadap pembersihan data ekstensif (seperti *stemming* dan *lemmatization* yang agresif) untuk bekerja optimal, *BERTopic* mampu mempertahankan stabilitas dan kualitas topik dengan intervensi minimal. Chen dkk. (2023) bahkan mencatat bahwa pada korpus berita keuangan yang besar, *BERTopic* tidak hanya memberikan koherensi terbaik (0,823), tetapi juga waktu komputasi yang lebih wajar dibandingkan model *neural* lain seperti *Top2Vec* yang membutuhkan waktu pelatihan lama.

Namun demikian, penerapan model *transformer* bukanlah solusi universal tanpa celah. Mihajlov dkk. (2024) memberikan antitesis penting melalui studi *Computational Literary Studies*. Mereka menemukan bahwa pada teks sastra panjang tanpa mekanisme pemotongan (*chunking*), *BERTopic* justru gagal menangkap tema naratif yang mendalam dan cenderung hanya memunculkan nama karakter sebagai topik, tidak lebih baik dari metode tradisional NMF yang sukses mengekstraksi tema cerita. Temuan ini mengindikasikan bahwa arsitektur *BERTopic* memiliki batasan pada panjang dokumen, sebuah celah yang justru memvalidasi arah penelitian ini. Berangkat dari keterbatasan *BERTopic* pada teks panjang tersebut, penelitian ini secara spesifik akan memfokuskan penerapan model pada teks komentar media sosial yang memiliki karakteristik sangat pendek (rata-rata di bawah 100 karakter), di mana kemampuan kontekstual *BERTopic* diprediksi akan bekerja paling optimal dibandingkan pada dokumen panjang.

Merespons berbagai keterbatasan metodologis dari penelitian-penelitian terdahulu, penelitian ini hadir untuk mengisi beberapa kesenjangan krusial.

Pertama, mayoritas studi sebelumnya (seperti Nanayakkara, Chen, dan Krishnan) cenderung hanya mengevaluasi model menggunakan LDA sebagai *baseline* umum. Penelitian ini akan memperkaya komparasi dengan menghadirkan GSDMM (*Gibbs Sampling Dirichlet Multinomial Mixture*), sebuah model probabilistik yang memang dirancang khusus untuk teks pendek, sehingga perbandingan performa menjadi lebih adil (*apple-to-apple*) untuk data media sosial.

Kedua, kritik terhadap penggunaan metrik tunggal seperti koherensi topik atau NPMI (pada studi Lande dan Krishnan) akan dijawab dengan menerapkan evaluasi ganda yang mencakup *Topic Coherence* dan *Topic Diversity*, guna memastikan model tidak hanya menghasilkan topik yang koheren, tetapi juga beragam. Terakhir, penelitian ini tidak berhenti pada pemodelan topik semata, melainkan mengintegrasikannya dengan analisis sentimen berbasis BERT, memberikan lapisan analisis yang lebih dalam mengenai bagaimana sentimen publik terdistribusi ke dalam topik-topik kebijakan pemerintah yang terbentuk.

Perkembangan penelitian dalam pemrosesan bahasa alami semakin berfokus pada integrasi antara pemodelan topik dan analisis sentimen untuk mendapatkan wawasan yang lebih mendalam dari data teks. Efektivitas pendekatan modern dibandingkan metode konvensional telah dibuktikan secara signifikan oleh Asnawi dkk. (2024) dalam studi mereka terhadap korpus berita daring Indonesia. Dengan menerapkan model *Sentence Transformer* (spesifiknya *paraphrase-multilingual-MiniLM-L12-v2*) untuk *embedding*, serta *UMAP* dan *HDBSCAN* untuk klasterisasi, metode *BERTopic* mampu menghasilkan topik yang sangat koheren. Hal ini terlihat dari identifikasi yang jelas pada topik yang mencakup 4.587 dokumen terkait isu politik, yang jauh lebih terstruktur dibandingkan hasil *LDA*. Meskipun demikian, studi ini memiliki keterbatasan utama karena tidak menyertakan evaluasi matriks kuantitatif untuk memvalidasi kualitas topik tersebut secara objektif.

Keunggulan model berbasis *transformer* dalam menangani teks informal semakin dipertegas oleh penelitian Jefri, Fauzi, dan Fa'rifah (2025) mengenai kesehatan mental ibu di platform TikTok. Dengan memanfaatkan 10.000 data yang berisi bahasa gaul (*slang*), integrasi *IndoBERT* (*fine-tuned*) dan *BERTopic*

menunjukkan lonjakan performa yang drastis. Metode ini berhasil mencapai nilai *F1-score* dan akurasi sebesar 0,94, mengungguli secara signifikan metode *baseline* tradisional SVM (*Support Vector Machine*) yang hanya mampu mencapai angka 0,85. Kenaikan performa sebesar hampir 10% ini membuktikan bahwa arsitektur BERT jauh lebih andal dalam menangkap konteks semantik pada teks pendek media sosial.

Pentingnya peran pemodelan topik dalam meningkatkan akurasi analisis sentimen juga dibuktikan secara statistik oleh Araújo dkk. (2024). Melalui eksperimen pada *dataset* ulasan Spotify, mereka menemukan bahwa integrasi informasi topik menggunakan metode *Automatic Topic Consolidation* (ATC) mampu mendongkrak kinerja model. Model BERT yang diperkaya dengan informasi topik (*UCA + ATC*) berhasil mencatatkan *F1-score* sebesar 0,72, melampaui kinerja model BERT dasar (*baseline*) yang hanya mencapai 0,69. Angka ini menunjukkan bahwa pemodelan topik memberikan konteks tambahan yang krusial untuk klasifikasi sentimen yang lebih presisi.

Penerapan praktis dari kombinasi metode ini terlihat dalam studi kasus *e-wallet* XYZ oleh Susanto dan Mauritsius (2025). Dalam upaya mengatasi krisis reputasi aplikasi dengan *rating* rendah (2,7/5), peneliti membangun model sentimen menggunakan LSTM dan *Word2Vec* yang dilatih pada 1.741 dokumen. Model ini menghasilkan performa yang kuat dengan akurasi 91,40%, presisi 86%, dan *F1-score* 74,14%. Temuan kualitatif dari *BERTopic* pun sangat spesifik, mengidentifikasi topik "aplikasi_tidak_saldo_masuk" sebagai sumber utama sentimen negatif pada 98 ulasan baru. Namun, kelemahan mendasar dari penelitian ini adalah pendekatan *single-model*, di mana peneliti hanya menguji satu algoritma (LSTM dan *BERTopic*) tanpa melakukan komparasi dengan algoritma lain untuk mengukur efektivitas relatifnya.

Berdasarkan telaah literatur tersebut, teridentifikasi celah penelitian yang signifikan. Penelitian Asnawi dkk. (2024) unggul dalam metodologi, namun lemah dalam validasi metrik kuantitatif, sementara Susanto dan Mauritsius (2025) memiliki metrik evaluasi yang lengkap, namun tidak melakukan perbandingan algoritma. Oleh karena itu, penelitian ini dirancang untuk mengisi celah tersebut

dengan melakukan analisis komparatif antara tiga algoritma pemodelan topik: LDA, GSDMM (sebagai pembanding khusus teks pendek), dan *BERTopic*. Berbeda dengan penelitian sebelumnya, studi ini akan menerapkan matriks evaluasi kuantitatif yang terstandarisasi serta menggunakan model BERT untuk klasifikasi sentimen guna menghasilkan evaluasi yang tidak hanya tajam secara kualitatif, melainkan juga teruji secara statistik.

Penelitian terkait yang sudah dijelaskan pada paragraf – paragraf sebelumnya dapat dilihat pada Tabel 2.1.

Tabel 2.1 Penelitian terkait

No.	Peneliti	Metode	Domain dan dataset	Keterbatasan dan celah penelitian
1.	(Sukma, dkk, 2020)	SVM, Naïve Bayes Classifier, Decision Tree	10.200 tweet, terkait kebijakan pemerintah (Omnibus Law)	Komparasi model yang dilakukan pada penelitian ini hanya sebatas model *machine learning*
2.	(Azis & Wahyudi, 2024)	Naïve Bayes Classifier	560 tweet, dengan kata kunci "Ibu Kota Nusantara"	Penelitian hanya menggunakan klasifikasi biner, berdasarkan hasil penelitian, model tersebut sangat lemah dalam mengidentifikasi sentimen negatif (nilai *recall* hanya 48.15% dan *precision* negatif hanya 65.00%)
3.	(Susanto & Mauritsius, 2025)	*BERTopic* dan *LSTM*	2748 ulasan pengguna di Google Play Store	Hanya menggunakan algoritma *machine learning* (LSTM) dan BERTopic tanpa ada perbandingan dengan model lain

No.	Peneliti	Metode	Domain dan dataset	Keterbatasan dan celah penelitian
4.	(de Araújo, dkk, 2024)	*BERT base uncased*, *T5 Small*, dan *BERTopic*	Ulasan aplikasi pada Amazon App, Netflix App dan Spotify	Pemodelan topik yang dilakukan pada penelitian tersebut hanya menggunakan satu model saja, yakni BERTopic, tidak ada model alternatif yang digunakan sebagai pembanding
5.	(Chen, dkk, 2023)	LDA, Top2Vec, BERTopic, RoBERTa	38.240 artikel berita dari surat kabar Australian Financial Review (AFR)	Matriks evaluasi perbandingan model pemodelan topik hanya menggunakan *topic coherence* saja
6.	(Amaro & Bacao, 2024)	LDA, NMF, PVTM, Top2Vec, BERTopic	18.846 pesan dari platform Usenet yang mencakup 20 Subjek berbeda, 87.362 data dari Yahoo! Q&A dan 42.000 abstrak paten dari Big Patent	Meskipun dataset yang digunakan beragam, tidak menjamin hasil yang sama akan diperoleh pada konteks yang sedikit berbeda, tidak adanya pembanding GSDMM dan hanya fokus pada komparasi pemodelan topik saja
7.	(Asnawi, dkk, 2024)	LDA, BERTopic	Artikel berita dari kompas.com, viva.co.id, dan tribunnews.com	Kurangnya metriks evaluasi kuantitatif dan dataset yang sudah kadaluarsa, yang berasal dari bulan Desember 2015
8.	(Acheadeth, dkk, 2022)	LDA, SVM, LR, dan MLP	Dataset dengan anotasi manual sebanyak 413	Ukuran dataset pada penelitian ini sangat kecil untuk pelatihan dan

No.	Peneliti	Metode	Domain dan dataset	Keterbatasan dan celah penelitian
			baris data	pengujian menggunakan *machine learning*, akurasi model yang dicapai hanya 0.79 dan model klasifikasi yang masih menggunakan *machine learning*
9.	(Lande,dkk, 2023)	LDA, GSDMM, BERT - TM	Dataset dengan tweet berbahasa inggris di India sebanyak 519.000 cuitan	Penelitian hanya menggunakan skor NPMI sebagai metriks evaluasi, NPMI sendiri adalah ukuran perkiraan (*approximate measurement*) dan berisiko terhadap hasil yang bervariasi tergantung korpus yang digunakan
10.	(Nanayakkara & Thennakoon, 2024)	LDA, NMF, dan BERTopic	Komentar YouTube sebanyak 70.272 baris	Tidak membandingkan GSDMM yang merupakan metode standar *untuk short text topic modeling*, dan hanya melakukan pemodelan topik saja dalam penelitian tersebut
11.	(Krishnan & Kennedyraj, 2023)	LDA, LSA, NMF, PAM, Top2Vec, BERTopic	Ulasan aplikasi sebanyak 29.200 data, mengenai layanan pemerintah Uni Emirat Arab	Matriks evaluasi yang digunakan belum mempertimbangkan *Topic Diversity* yang berguna untuk memastikan model tidak menghasilkan topik yang berulang
12.	(Mihajlov, dkk, 2024)	LDA, NMF, BERTopic	100 Novel berbahasa Serbia, dengan rata - rata panjang novel	Penelitian ini menghasilkan model NMF lebih baik daripada BERTopic, dengan nilai *topic coherence* sebesar 0.568, dengan dataset *long text*,

No.	Peneliti	Metode	Domain dan dataset	Keterbatasan dan celah penelitian
			49.315 kata	BERTopic memiliki keterbatasan token, sehingga perlu dibuktikan untuk dataset berupa *short text*
13.	(Jefri, dkk, 2025)	BERTopic dan IndoBER	10.000 post pada TikTok (caption dan komentar) yang dikumpulkan antara tahun 2022 hingga 2024	Penelitian ini menggunakan dataset yang sudah diberikan label dalam menentukan topik (*supervised*), dan masih perlu adanya perbandingan alternatif model lainnya sebagai validasi model yang di usulkan

2.2 Dasar Teori

Pada subbab Dasar Teori, akan dijelaskan basis penelitian yang dilakukan terkait dengan pemodelan topik, analisis sentimen, matriks evaluasi pemodelan topik, dan penggunaan analisis sentimen berbasis aspek (*aspect-based sentiment analysis* atau ABSA).

2.2.1 Pemodelan Topik

Pendekatan Pendekatan text mining memegang peranan vital dalam proses ekstraksi fitur-fitur kunci dari data tekstual berskala besar. Di antara berbagai metode yang tersedia, pemodelan topik menjadi teknik yang paling sering diterapkan. Pemodelan topik merupakan pendekatan algoritmik yang umum digunakan dalam machine learning dan pemrosesan bahasa alami untuk menyingkap pola tematik yang tersembunyi di dalam sekumpulan teks. Penerapannya mencakup berbagai domain, termasuk ilmu sosial, di mana metode ini berperan penting dalam mengungkap preferensi laten konsumen, mengidentifikasi struktur semantik pada platform media sosial seperti Instagram, YouTube, Twitter (X), dan Facebook, serta meningkatkan kinerja sistem rekomendasi (Nanayakkara dan Thennakoon, 2024). Beberapa algoritma yang sering digunakan pada penelitian terdahulu antara lain adalah Latent Dirichlet

Allocation (LDA), Gibbs Sampling For Dirichlet Multinomial Mixture (GSDMM), dan penggunaan model neural seperti BERTopic.

2.2.1.1 *Latent Dirichlet Allocation* (LDA)

Latent Latent Dirichlet Allocation (LDA) didefinisikan sebagai model probabilistik generatif yang diterapkan pada koleksi data diskrit, khususnya korpus teks. Secara struktural, LDA merupakan model Bayesian hierarkis tiga tingkat di mana setiap dokumen dipandang sebagai campuran terbatas (*finite mixture*) dari berbagai topik yang mendasarinya. Setiap topik tersebut kemudian dimodelkan sebagai campuran distribusi probabilitas yang tak terbatas. Dalam konteks pemodelan teks, probabilitas topik ini berfungsi sebagai representasi eksplisit dari sebuah dokumen.

Penelitian terkait LDA memperkenalkan teknik inferensi pendekatan (*approximate inference*) yang efisien berbasis metode variasonal dan algoritma *Expectation-Maximization* (EM) untuk estimasi parameter Bayes empiris. Kinerja LDA telah dievaluasi dalam pemodelan dokumen, klasifikasi teks, dan *collaborative filtering*, serta menunjukkan keunggulan dibandingkan model *mixture of unigrams* dan *probabilistic Latent Semantic Indexing* (pLSI) (Blei dkk., 2003).

LDA telah lama menjadi standar dalam pemodelan topik berbasis probabilistik generatif, di mana pemahaman terhadap struktur topiknya sering dibantu oleh visualisasi *pyLDAvis* (Blair dkk., 2020). Namun, efektivitas LDA mulai dipertanyakan ketika dihadapkan pada karakteristik data media sosial. Menurut Sheldon dan Fesenmaier (2022), LDA menghadapi kendala signifikan saat memproses *dataset* yang pendek dan tidak terstruktur (*noisy*), yang mengakibatkan ketidakstabilan dalam pembelajaran statistik. Oleh karena itu, tren penelitian saat ini mulai beralih pada pengembangan dan penggunaan algoritma alternatif yang dirancang khusus untuk menangani karakteristik teks singkat (*short text*) secara lebih efektif (Egger dan Yu, 2022).

Latent Dirichlet Allocation (LDA) sendiri merupakan sebuah model probabilistik yang memandang dokumen sebagai campuran dari berbagai topik laten (tersembunyi), di mana setiap topik tersebut dikarakterisasi oleh distribusi

kata tertentu. Penerapan metode ini mengharuskan penentuan jumlah topik di awal serta penyetelan (*tuning*) parameter untuk mencapai hasil pemodelan yang optimal (Zankadi dkk., 2023). Model ini dikembangkan oleh Blei dkk. (2001) sebagai proses generatif yang mengasumsikan bahwa setiap dokumen dihasilkan sebagai campuran dari topik-topik yang mendasarinya. Proporsi campuran yang bernilai kontinu didistribusikan sebagai variabel acak Dirichlet laten. Suatu topik kemudian didefinisikan oleh distribusi atas semua kata dalam korpus. Untuk menghindari pengindeksan ganda pada dokumen, setiap dokumen dikaitkan dengan angka dari 1 hingga *D*. Secara teknis, bagaimana model ini bekerja adalah sebagai berikut :

1. Menentukan Definisi Topik (Distribusi Kata)
    a. Proses awal adalah menentukan berapa banyak distribusi topik yang dibutuhkan (K).
    b. Selanjutnya, Setiap topik (β_k) diambil dari distribusi *Dirichlet* dengan parameter (λ_β).
    c. Notasi, $\lambda_\beta = (\lambda_{\beta 1}, \dots, \lambda_{\beta V})$ merepresentasikan relevansi kata (menunjukkan seberapa penting atau frekuensi munculnya suatu kata) didalam suatu topik (k).
    d. Proses penentuan definisi topik ini nantinya akan menghasilkan daftar kata-kata apa saja yang mungkin muncul untuk setiap topik (contohnya adalah Topik 1 memiliki kecenderungan ke kata "ekonomi, uang, pasar", sedangkan Topik 2 cenderung ke "olahraga, bola, gol".
2. Menentukan Komposisi Topik per Dokumen
    a. Untuk sebuah dokumen spesifik $(d,)$, diambil distribusi topik (θ_d) dari distribusi Dirichlet dengan parameter (λ_α) .
    b. Notasi, $\lambda_\alpha = (\lambda_{\alpha 1}, \dots, \lambda_{\alpha K})$ merepresentasikan vektor relevansi topik untuk keseluruhan korpus (kumpulan dokumen).
    c. Proses penentuan komposisi topik ini nantinya menentukan prosentase isi dari dokumen (contohnya adalah Dokumen A terdiri dari 80% Topik Ekonomi dan 20% Topik Politik).
3. Menghasilkan Kata-Kata dalam Dokumen
    a. Langkah terakhir adalah proses ekstraksi kata-kata (N_d jumlah kata) ke

dalam dokumen tersebut seara berulang ($n = 1$ sampai N_d).

b. Memilih topik kata (Z_{nd}), untuk setiap posisi kata, algoritma akan memilih satu topik (Z_{nd}) berdasarkan proporsi topik dokumen tersebut (θ_d). Contohnya adalah, jika dokumen tersebut dominan Ekonomi, algoritma kemungkinan besar memilih topik "Ekonomi" untuk kata ini.

c. Memilih kata aktual (W_{nd}), setelah topik dipilih, algoritma memilih kata aktual (W_{nd}) berdasarkan distribusi kata yang berasosiasi dengan topik tersebut $(\beta_{Z_{nd}})$. (β_z) merupakan vektor probabilitas kemunculan kata $p\ (w|z)$ jika diketahui topiknya adalah z. Contohnya adalah apabila topik yang terpilih adalah "Ekonomi", algoritma kemudian memilih kata "Saham" dari daftar kata topik Ekonomi.

4. Dari uraian proses diatas, sebelum diterapkan pada model matematis *Latent Dirichlet Allocation*, perlu menentukan beberapa *hyperparameters* untuk menentukan bentuk distribusi yang ingin dihasilkan.

   a. (β_{kn}) Beta, Berisi semua probabilitas kata spesifik per topik, dimana merupakan peluang munculnya kata n jika diketahui topiknya adalah k, yang merupakan definisi dari isi topik itu sendiri.

   b. (θ_{dk}) Theta, Berisi semua probabilitas topik spesifik per dokumen, dimana merupakan peluang (proporsi) topik k ada di dalam dokumen d.

   c. Selanjutnya probabilitas topik (W_{nd}) yang muncul dihitung dengan menjumlahkan seluruh kemungkinan topik tersembunyi dengan persamaan berikut.

   $$p(w_{nd}|\theta_d, \beta) = \sum_{k=1}^{K} p(w_{nd}|z = k, \beta)\, p(z = k|\theta_d) \quad (2.1)$$

Untuk mengetahui peluang munculnya suatu kata tertentu, tidak dapat diketahui secara pasti topik mana yang dipilih (z), dengan persamaan tersebut, prosesnya adalah menjumlahkan peluang kata yang muncul pada Topik 1 dikali dengan bobot, proses ini berulang sampai dengan topik K (sesuai dengan yang diinginkan). Persamaan tersebut juga menjelaskan bahwa LDA merupakan konsep campuran (*Mixture Models*) dimana, $p(W_{nd}|z, \beta)$ merupakan distribusi kata (isi topik) sedangkan $p(z|\theta_d)$ merupakan seberapa banyak prosentase

topik tersebut dalam dokumen.

Selanjutnya, dari persamaan diatas diintegrasikan ke tingkat dokumen dengan persamaan berikut, untuk menghitung peluang satu dokumen utuh (d).

$$p(d|\lambda_\alpha, \beta) = \int p(\theta_d|\lambda_\alpha) \left(\prod_{n=1}^{N_d} \sum_{k=1}^{K} p(w_{nd}|z = k, \beta)\, p(z = k|\theta_d)\right) d\theta_d \quad (2.2)$$

Dikarenakan dokumen terdiri dari banyak kata, persamaan tersebut mengkalikan peluang semua kata yang ada di dalamnya. Karena proporsi topik θ_d itu sendiri adalah variabel acak, yang berasal dari distribusi *Dirichlet* (λ_α), persamaan tersebut melakukan integrasi terhadap θ_d. Artinya kita menghitung rata-rata peluang dokumen tersebut dengan memperhitungkan semua kemungkinan kombinasi proporsi topik (θ) yang mungkin terjadi.

Pada penelitian ini, guna mempermudah proses perhitungan tersebut, digunakan bantuan pemrograman *Python* dengan menggunakan *library Gensim* (LDA) dan nantinya sebagai hasil akhir, visualisasi dilakukan menggunakan bantuan *library pyLDAvis* untuk melihat interaksi antar-topik dalam bentuk *bubble chart*.

Terkait proses *hyperparameter tuning* dan penentuan jumlah topik yang optimal, dilakukan simulasi pemodelan topik yang dimulai dari jumlah 5 topik sampai 30 topik dengan membandingkan metrik evaluasi koherensi (*coherence score*). Dengan demikian, jumlah topik yang optimal dapat ditemukan berdasarkan nilai evaluasi terbaik dari hasil simulasi tersebut.

2.2.1.2 *Gibbs Sampling For Dirichlet Multinomial Mixture* (GSDMM)

GSDMM pertama kali diperkenalkan oleh Yin dan Wang pada tahun 2014 dengan tujuan untuk pengelompokan teks-teks pendek. Untuk memahami bagaimana GSDMM menangani pengelompokan teks pendek, kita dapat membayangkan sebuah skenario di dalam kelas diskusi film. Bayangkan seorang profesor yang berencana membagi mahasiswanya ke dalam beberapa kelompok diskusi kecil. Agar diskusi berjalan hidup dan efektif, sang profesor menginginkan agar setiap kelompok diisi oleh mahasiswa yang memiliki selera atau pengalaman menonton yang serupa.

Prosesnya dimulai dengan profesor meminta setiap mahasiswa untuk

menuliskan daftar film yang pernah mereka tonton. Namun, ada batasan penting, yakni waktu yang diberikan sangat singkat. Akibatnya, mahasiswa tidak mungkin menuliskan semua film seumur hidup mereka, sehingga mereka hanya akan menuliskan segelintir judul yang paling baru ditonton atau yang paling berkesan bagi mereka. Daftar film yang ringkas ini secara akurat merepresentasikan karakteristik dokumen teks pendek (*short text*) yang memiliki jumlah kata (fitur) yang sangat terbatas dan spesifik.

Tugas algoritma GSDMM di sini bertindak selayaknya sang profesor tersebut. Algoritma ini berusaha mengidentifikasi pola dengan cara melihat daftar "film" (kata-kata) yang dimiliki oleh setiap "mahasiswa" (dokumen). Tujuannya adalah menempatkan mahasiswa dengan irisan daftar film yang sama ke dalam satu meja diskusi. Dengan demikian, terciptalah klaster-klaster di mana anggotanya berbagi minat yang seragam, memisahkan mereka dari kelompok lain yang memiliki topik diskusi berbeda. Inilah prinsip dasar yang dikenal sebagai *Movie Group Process* yang menjadi latar belakang GSDMM.

Dalam konteks pemrosesan teks pendek, GSDMM menawarkan pendekatan yang lebih unggul dibandingkan LDA. Jika LDA mengasumsikan dokumen memiliki campuran topik, GSDMM menyederhanakan asumsi tersebut dengan menetapkan satu topik untuk setiap dokumen, yang membuatnya sangat relevan untuk data singkat. Penelitian menunjukkan bahwa GSDMM mampu menangani masalah data *sparse* sekaligus menyajikan tema kata yang bermakna. Selain itu, algoritma ini unggul dalam efisiensi komputasi karena konvergensinya yang cepat dan kemampuannya mengidentifikasi jumlah klaster secara otomatis tanpa mengorbankan kualitas pengelompokan (Udupa dkk., 2022).

Proses ekstraksi topik selanjutnya adalah menggunakan algoritma GSDMM yang awalnya dirumuskan oleh Yin dan Wang pada tahun 2014. Metode ini merupakan modifikasi dari model LDA yang memanfaatkan teknik *Gibbs Sampler* pada kerangka kerja *Dirichlet Multinomial Mixture* (DMM) (Nigam, dkk, 2000). Adapun perbedaan mendasarnya adalah pada GSDMM, mengasumsikan bahwa setiap dokumen $(d = 1, \dots, D)$ dalam korpus dihasilkan oleh satu topik tunggal, berbeda dengan LDA yang berasumsi bahwa satu dokumen tersebut tersusun atas

campuran berbagai topik (Mazarura & De Waal, 2016). Sedangkan proses matematisnya dijelaskan sebagai berikut :

1. Tahap Inisialisasi Distribusi
    a. Sebelum dokumen dibuat, model menetapkan dua distribusi utama, pertama adalah distribusi kata per Topik (β_k) yang merupakan parameter untuk menentukan daftar kata apa saja yang relevan untuk setiap topik k yang diambil dari distribusi *Dirichlet* dengan parameter λ_β.
    b. Selanjutnya, menentukan distribusi topik untuk seluruh korpus (θ_c) dengan tujuan menentukan seberapa populer setiap topik dalam keseluruhan kumpulan data (korpus). Yang juga diambil dari distribusi *Dirichlet* dengan parameter λ_α.
2. Tahap Pembuatan Dokumen (Proses Generatif)
    a. Untuk setiap dokumen d dalam data $(d = 1, \dots, D)$ dilakukan langkah pemilihan topik (z_d) algoritma akan memilih satu topik tunggal (z_d) untuk dokumen tersebut berdasarkan distribusi topik korpus (θ_c), contoh dari proses ini misalnya adalah dokumen ini ditetapkan sepenuhnya sebagai topik "Olahraga".
    b. Langkah selanjutnya adalah memilih kata – kata (w_{nd}), setelah topik ditentukan, semua kata dalam dokumen tersebut dihasilkan berdasarkan probabilitas kata dari topik yang terpilih tadi β_{z_d}.
3. Perhitungan Probabilitas (Dasar Iterasi)
    a. Kemudian, dilakukan perhitungan peluang yang menjadi dasar dari algoritma Collapsed *Gibbs Sampler*, dimana peluang total dokumen $\big(p(d)\big)$ merupakan jumlah dari seluruh kemungkinan topik dengan persamaan sebagai berikut.

    $$\big(p(d)\big) = \sum_{k=1}^{K} p(d|z = k, \beta)\, p(z = k|\theta_c) \qquad (2.3)$$

    Dimana peluang dokumen d muncul adalah peluang dokumen jika topiknya adalah A dikali dengan popularitas Topik A ditambah dengan peluang dokumen jika topiknya B dikalikan dengan popularitas topik B dan seterusnya sesuai dengan jumlah topik yang diharapkan.

b. Setelah dilakukan perhitungan peluang dokumen, selanjutnya dilakukan perhitungan peluang kata untuk memastikan posisi kata tidak berpengaruh dengan kata lainnya (asumsi independensi) dengan persamaan dibawah ini.

$$p(d|z = k, \beta) = \prod_{w \in d} p(w|z = k, \beta) \quad (2.4)$$

Dimana posisi kata tidak berpengaruh (*independent*). Peluang dokumen *d* jika diketahui topiknya adalah *k*, dihitung dengan cara mengkalikan peluang kemunculan setiap kata yang ada di dalam dokumen tersebut pada topik *k*.

Pada penelitian ini, proses perhitungan dari algoritma GSDMM dilakukan dengan bantuan pemrograman *Python* menggunakan *library* GSDMM dari *rwalk* yang diambil dari *repository* GitHub. Adapun penentuan *hyperparameter* (*hyperparameter tuning*) dilakukan secara otomatis dengan mensimulasikan proses pemodelan melalui 15 kali iterasi untuk mendapatkan parameter terbaik apabila dibandingkan dengan metrik evaluasi yang dihasilkan.

2.2.1.3 BERTopic

Model topik tradisional sering kali memiliki keterbatasan dalam menangkap informasi kontekstual kata di dalam sebuah kalimat. Untuk mengatasi kekurangan tersebut, Maarten Grootendorst (2022) mengembangkan *BERTopic*, sebuah pendekatan pemodelan yang menghasilkan representasi topik secara berkelanjutan melalui teknik klasterisasi dan *class-based TF-IDF* (*c-TF-IDF*). Dengan memanfaatkan model BERT, data teks ditransformasikan menjadi *vektor embedding* berdimensi tinggi, di mana kemampuan BERT dalam memahami konteks memungkinkan pembentukan *embedding* yang kaya secara semantik (Bu dkk., 2023).

Dalam studi komparasi atau evaluasi, *BERTopic* mampu menangkap pola bahasa yang koheren pada teks pendek dan secara konsisten menunjukkan kinerja yang lebih unggul dibandingkan GSDMM (Udupa dkk., 2022). Studi lainnya pada domain keuangan dengan data sentimen berita keuangan mencoba mengkomparasi antara model probabilistik tradisional, yakni LDA, dengan model yang menggunakan *embedding* seperti *Top2Vec* dan *BERTopic*. Hasilnya, didapatkan

nilai koherensi topik tertinggi pada model *BERTopic* sebesar 0,823 (Chen dkk., 2023). Domain ulasan pelanggan pada *e-commerce* juga menyimpulkan hal yang sama; melalui studi komparatif yang dilakukan oleh Krishnan dan Kennedyraj pada tahun 2023 dengan membandingkan antara model *BERTopic*, *Top2Vec*, LDA, NMF, LSA, dan PAM, didapatkan hasil yang tidak jauh berbeda dengan penelitian yang dilakukan sebelumnya.

Adapun tahapan setiap proses yang dilakukan oleh model *BERTopic* adalah sebagai berikut :

1. Pembentukan *embedding* merupakan langkah awal yang melibatkan konversi dokumen teks menjadi vektor representasi numerik. Pada penelitian ini, secara spesifik digunakan model *pre-trained paraphrase-multilingual-MiniLM-L12-v2* yang dapat mendukung berbagai bahasa.
2. Reduksi dimensi, adapun vektor berdimensi tinggi yang dihasilkan kemudian diringkas menggunakan algoritma UMAP, yang bertujuan untuk memadatkan informasi agar struktur data lokal tetap terjaga namun lebih efisien untuk dikelompokkan.
3. Klasterisasi, dokumen yang telah direduksi dimensinya kemudian dikelompokkan menggunakan HDBSCAN, algoritma ini dipilih karena keunggulannya dalam nenangani klaster dengan densitas yang bervariasi serta kemampuannya mengisolasi outliers (data ekstrim).
4. Ekstraksi Kata Kunci, setiap klaster dokumen digabungkan menjadi satu dokumen besar (*bag-of-words*) untuk selanjutnya dianalisa frekuensi kata-kata yang muncul.
5. Pada tahap representasi topik, skor kepentingan kata dihitung dengan menggunakan algoritma *c-TF-IDF* (*class-based TF-IDF*). Metode ini pada dasarnya memodifikasi *TF-IDF* standar untuk bekerja pada tingkat klaster, bukan pada dokumen individu, guna menemukan kata-kata yang paling mewakili setiap topik yang dihasilkan.
6. Tahap penyempurnaan merupakan proses opsional. Setelah representasi topik dihasilkan, label tersebut selanjutnya dapat diperkaya dengan menggunakan model bahasa canggih seperti *GPT* atau *T5* untuk menghasilkan label topik

yang lebih presisi dan lebih mudah dipahami.

Dalam konteks bahasa Indonesia, studi komparatif juga dilakukan pada korpus berita bahasa Indonesia. Dengan membandingkan model *embedding BERTopic* dengan model probabilistik klasik LDA, disimpulkan bahwa *BERTopic* mampu menghasilkan topik yang lebih koheren dan mudah diinterpretasikan daripada LDA dalam konteks bahasa Indonesia (Asnawi dkk., 2024). Beberapa penelitian tersebut menjadi fundamental untuk validasi bahwa *BERTopic* memiliki kecenderungan menghasilkan koherensi topik terbaik apabila dibandingkan dengan LDA maupun GSDMM.

2.2.2 Matriks Evaluasi Pemodelan Topik

Dalam Dalam tahapan evaluasi model, penelitian ini berfokus pada penggunaan *Topic Coherence* (TC) sebagai indikator kinerja utama. Koherensi topik berfungsi sebagai alat ukur kuantitatif untuk menilai kepaduan topik yang dirancang untuk mensimulasikan penilaian manusia secara komputasional. Intuisi di balik metrik ini adalah melakukan komparasi antara kata-kata utama (*top words*) dari setiap topik terhadap dokumen asli di dalam korpus. Tujuannya adalah untuk menemukan seberapa koheren kedua himpunan tersebut, mengingat koherensi himpunan kata didefinisikan sebagai 'derajat dukungan satu himpunan bagian terhadap himpunan bagian lainnya' (Röder dkk., 2015).

Secara teknis, perhitungan TC dalam penelitian ini mengikuti kerangka kerja yang terdiri dari empat dimensi: *Segmentation*, *Probability Calculation*, *Confirmation Measure*, dan *Aggregation*. Kerangka ini memfasilitasi evaluasi yang jelas antar metode pengukuran. Penelitian ini menggunakan bantuan *library Gensim* untuk menjelaskan empat varian koherensi tersebut. Dalam pelaksanaannya, seluruh metrik evaluasi dikonfigurasi untuk memproses 10 kata teratas (*top 10 words*) yang merepresentasikan setiap topik.

Evaluasi pertama adalah menggunakan *UMASS Coherence*. Adapun konsep perhitungannya adalah sebagai berikut.

$$UMASS\ Coherence = \frac{2}{N \times (N-1)} \sum_{i=2}^{N} \sum_{j=1}^{i-1} \log \frac{P(wi,wj) + \varepsilon}{P(wj)} \quad (2.4)$$

Dimana, w_i dan w_j mewakili dua kata yang berbeda, dan *N* menunjukkan

ukuran total korpus. Variabel ε merupakan parameter yang ditambahkan untuk menghindari operasi logaritma terhadap nol, dengan nilai awal yang ditetapkan sebesar 1. Secara spesifik pada metrik ini, Segmentasi dilakukan secara berurutan (*sequential*) dengan membandingkan kata *w* dengan kata pada posisi selanjutnya. Untuk Estimasi Probabilitas, digunakan metode *Boolean document* guna menghitung probabilitas gabungan berdasarkan proporsi dokumen yang mengandung pasangan kata tersebut dalam korpus. Nilai koherensi kemudian dihitung menggunakan *log-conditional-probability* pada tahap Konfirmasi. Seluruh nilai ini kemudian diagregasi menggunakan *simple arithmetic mean* untuk menghasilkan satu nilai skala TC yang merepresentasikan kualitas topik.

Metriks selanjutnya adalah menggunakan *UCI Coherence*, dimana persamaannya adalah sebagai berikut.

$$UCI\ Coherence = \frac{2}{N \times (N-1)} \sum_{i=1}^{N-1} \sum_{j=i+1}^{N} PMI\ (wi, wj) \quad (2.5)$$

Dimana $PMI\ (wi, wj) = \log \frac{P(wi,wj)+\varepsilon}{P(wi) \times P\ (wi)}$, w_i dan w_j merepresentasikan dua kata yang berbeda, dan *N* merupakan ukuran total korpus, adapun variabel ε merupakan parameter yang ditambahkan untuk mencegah terjadinya operasi logaritma terhadap nol, dengan nilai standar yang ditetapkan sebesar 1.

Mekanisme metriks tersebut dimulai dengan langkah segmentasi, di mana setiap kata dipasangkan dengan setiap kata lainnya dalam sebuah himpunan. Pada tahap konfirmasi, perhitungan dilakukan menggunakan rasio logaritmik (PMI), proses ini diakhiri dengan tahap agregasi menggunakan rata-rata aritmatika untuk menghasilkan satu nilai koherensi yang merepresentasikan kualitas topik yang dihasilkan.

Selanjutnya digunakan metriks *NPMI Coherence*, dimana persamaan matematisnya adalah sebagai berikut.

$$NPMI\ Coherence = \frac{2}{N \times (N-1)} \sum_{i=1}^{N-1} \sum_{j=i+1}^{N} NPMI\ (wi, wj) \quad (2.6)$$

Dimana, $NPMI\ (wi, wj) = \frac{PMI\ (wi,wj)}{-\log(P(wi,wj)+\varepsilon)}$; $PMI\ (wi, wj) = \log \frac{P(wi,wj)+\varepsilon}{P(wi) \times P\ (wi)}$ (2.7)

w_i dan w_j merepresentasikan dua kata yang berbeda, dan *N* merupakan ukuran total korpus, adapun variabel ε merupakan parameter yang ditambahkan

untuk mencegah terjadinya operasi logaritma terhadap nol, dengan nilai standar yang ditetapkan sebesar 1.

Selain dari metriks yang sudah dijelaskan pada paragraf sebelumnya, metriks evaluasi utama yang digunakan adalah *Context Vectors* atau C_v *Coherence*. Pemilihan C_v didasarkan pada akurasinya yang paling mendekati penilaian manusia dibandingkan metrik lain, berikut merupakan bagaimana proses perhitungan yang dilakukan yang merupakan penjabaran dari paragraf pembuka diatas.

1. *Segmentation* (S)

   Kata-kata dalam sebuah topik dipasangkan menggunakan teknik *one-set segmentation*. Jika kita memiliki set kata teratas *W* dalam sebuah topik, setiap kata w_i akan dipasangkan dengan seluruh kata dalam set tersebut, termasuk kata itu sendiri.

2. *Probability Calculation* (P)

   Probabilitas dihitung berdasarkan kemunculan kata dalam korpus referensi (korpus internal) menggunakan *sliding window* (dengan ukuran 110 kata). *Sliding window* sendiri bermakna sebagai cakupan sejumlah kata tertentu, jendela ini akan bergeser satu demi satu kata dari awal hingga akhir dokumen.

   - $P(w_i)$: Probabilitas kata w_i muncul dalam *window*
   - $P(w_i,w_j)$: Probabilitas kata w_i dan w_j muncul bersamaan dalam satu *window.*

3. *Direct Confirmation Measure* (NPMI)

   Sebelum membentuk vektor, selanjutnya dihitung kedekatan antar kata dengan menggunakan NPMI, adapun formulanya dapat dilihat pada persamaan (2.7).

4. *Context Vectors* (V)

   Selanjutnya, untuk setiap kata w_i , dibuatkan vektor konteks $v(w_i)$. Vektor tersebut berisi nilai NPMI antara kata w_i dengan semua kata lain w_j dalam topik tersebut :

   $$v(w_i) = [NPMI(w_i, w_1), NPMI(w_i, w_2), \dots, (w_i, w_n)] \quad (2.8)$$

   Jadi, apabila terdapat 10 kata teratas dalam satu topik, maka akan ada 10 vektor dimana masing-masing vektor memiliki 10 dimensi.

5. *Indirect Confirmation Measure*

   Setelah vektor terbentuk, selanjutnya dihitung kesamaan antara vektor kata w_i dengan vektor gabungan dari kata-kata lainnya menggunakan *Cosine Similarity*.

   $$sim(u, v) = \frac{u.v}{\|u\|\|v\|} \quad (2.9)$$

6. *Aggregation*

   Langkah terakhir adalah menghitung rata-rata dari seluruh nilai yang diperoleh, hasil akhir C_v pada dasarnya berada pada rentang 0 hingga 1.

   $$Cv = mean(\sum cosine\_sim) \quad (2.10)$$

Di sisi lain, aspek keberagaman topik diukur menggunakan *Topic Diversity*. Indikator ini menghitung persentase kata unik dalam 25 kata teratas dari setiap topik. Meskipun sederhana, *Topic Diversity* krusial untuk memvalidasi bahwa model mampu menghasilkan topik-topik yang berdiri sendiri dan dideskripsikan oleh kata-kata yang tidak tumpang tindih secara signifikan. Berikut merupakan formula bagaimana menghitung *Topic Diversity*.

$$TD = \frac{Jumlah\ Kata\ Unik\ dalam\ Top\ K\ di\ Semua\ Topik}{T \times K} \quad (2.11)$$

Dimana :

- T : Jumlah total topik yang dihasilkan oleh model.
- K : Jumlah kata teratas (*Top Words*) yang diambil dari setiap topik (K = 25).
- Kata Unik : Kumpulan kata yang hanya dihitung satu kali meskipun muncul di beberapa topik yang berbeda.

Seluruh matriks evaluasi tersebut nantinya diimplementasikan dengan menggunakan bantuan pemrograman *python* yakni dengan bantuan *library* Gensim, matriks tersebut juga diterapkan pada algoritma GSDMM dan juga model *BERTopic* yang bertujuan untuk mencari nilai koheren terbaik dari ketiga model pemodelan topik tersebut dengan parameter yang sesuai dengan model masing-masing.

2.2.3 Analisis Sentimen

Analisis Analisis sentimen bertujuan untuk mengelompokkan polaritas teks dalam suatu dokumen atau kalimat guna menentukan apakah opini tersebut masuk

dalam kategori positif, negatif, atau netral. Teknik ini sering disebut juga sebagai *opinion mining*. Metode ini mencakup proses pengumpulan dan evaluasi pandangan, perasaan, sikap, serta kesan masyarakat terhadap berbagai subjek, produk, maupun layanan. Seiring dengan pesatnya perkembangan aplikasi berbasis internet seperti blog, media sosial, dan situs web, masyarakat semakin aktif menghasilkan gagasan dan ulasan mengenai aktivitas harian serta produk yang mereka gunakan. Oleh karena itu, analisis sentimen menjadi instrumen berharga bagi pelaku bisnis, pemerintah, dan peneliti untuk menganalisis opini publik. Wawasan yang diperoleh dari analisis ini kemudian dapat digunakan sebagai tolok ukur strategis dalam proses pengambilan keputusan (Arvyantomo dkk., 2023).

Secara teknis, analisis sentimen mengategorikan teks menjadi beberapa klasifikasi dengan menggunakan beberapa pendekatan seperti pendekatan berbasis leksikon (kamus), *machine learning*, dan *deep learning*. Pada metode berbasis leksikon, klasifikasi dilakukan dengan mencocokkan kata dalam teks terhadap daftar istilah yang telah memiliki bobot sentimen (positif atau negatif). Sementara itu, pendekatan *machine learning* dan *deep learning* mengandalkan proses pelatihan model menggunakan data berlabel. Contoh algoritma yang sering digunakan dalam *machine learning* antara lain *Support Vector Machine* (SVM), *Naïve Bayes Classifier*, *Ensemble Stacking*, *Multi-Label Classification*, dan *Random Forest Classifier*. Di sisi lain, *deep learning* menerapkan arsitektur jaringan saraf seperti *Recurrent Neural Networks* (RNN) dan *Convolutional Neural Networks* (CNN) (Kharde dan Sonawane, 2016).

Adapun model *Long Short-Term Memory* (LSTM) dan transformator seperti BERT telah meningkatkan kemampuan sistem dalam memahami konteks teks yang cukup rumit. Meskipun begitu, efektivitas proses analisis sentimen masih terbatas oleh kompleksitas linguistik, di mana tantangan utama meliputi fenomena polisemi (*polysemy*) dan ambiguitas konteks yang maknanya sangat bergantung pada kalimat yang dianalisis. Lebih jauh lagi, deteksi sentimen pada ungkapan implisit seperti sarkasme, ironi, dan bahasa kiasan masih menjadi kendala karena adanya disparitas antara makna harfiah dan makna intensi (Minaee dkk., 2021). Selain itu, ketergantungan pada konteks domain juga krusial mengingat polaritas sentimen

suatu kata dapat berubah sesuai bidang pembahasannya.

2.2.4 Analisis Sentimen Berbasis Aspek

Analisis Analisis sentimen konvensional umumnya berfokus pada identifikasi sikap umum yang terkandung dalam sebuah kalimat. Namun, pendekatan ini memiliki keterbatasan karena belum mampu membuat sistem komputasi memahami fitur spesifik yang menjadi target sentimen tersebut (Nuha dan Lin, 2023). Menurut Alqaryouti (2020), analisis sentimen dapat dikategorikan ke dalam tiga tingkatan: level dokumen, level kalimat, dan level berbasis aspek.

Analisis Sentimen Berbasis Aspek (ABSA) menawarkan kedalaman analisis dengan mengungkap fitur spesifik yang dibahas, baik dari segi entitas maupun aspeknya. Sebagai ilustrasi, sebuah ulasan aplikasi seluler dapat mengandung polaritas positif pada aspek antarmuka (*display*), namun sekaligus menunjukkan sentimen negatif pada aspek aksesibilitas. Oleh karena itu, tujuan utama dari penelitian ABSA adalah untuk melakukan anotasi kalimat pada aspek tertentu dengan tingkat akurasi yang lebih tinggi (Liu dkk., 2020).

Evolusi metode ABSA ditandai dengan penerapan algoritma *machine learning* konvensional, seperti *Support Vector Machines* (SVM) dan *Naive Bayes*, yang memprediksi sentimen aspek melalui ekstraksi fitur tekstual berupa kata maupun *n-gram*. Di sisi lain, riset yang lebih mutakhir menunjukkan superioritas pendekatan *deep learning*. Model seperti *Recurrent Neural Networks* (RNN) dan *Long Short-Term Memory* (LSTM) terbukti lebih unggul dalam memahami dependensi atau hubungan jangka panjang di dalam struktur teks (Naim, 2021).

Meskipun pendekatan *deep learning* menawarkan prospek yang baik untuk ekstraksi dan klasifikasi sentimen, kedua tugas ini sering kali dijalankan secara terpisah sehingga belum menghasilkan kinerja yang optimal. Selain itu, metode khusus untuk pembuatan leksikon dinamis memiliki keterbatasan karena tidak mempertimbangkan *POS tagging* dan sulit melokalisasi aspek implisit. Oleh karena itu, diperlukan pendekatan terpadu untuk menghasilkan analisis yang lebih menyeluruh. Dalam konteks ini, model BERT telah digunakan untuk mentransformasi ABSA dari tugas klasifikasi kalimat tunggal menjadi tugas klasifikasi pasangan kalimat (*sentence-pair classification*).

Lebih lanjut, komponen *embedding* BERT dieksplorasi bersama berbagai model saraf untuk ABSA *end-to-end*, dan pendekatan seperti *Interactive Multi-task Learning Network* (IMN) diusulkan untuk mengekstraksi aspek dan opini secara bersamaan, serta penggunaan pendekatan pohon dependensi (*dependency tree*) untuk mengodekan informasi sintaksis yang komprehensif (Savanur dan Sumathi, 2022).

2.2.5 BERT (*Bidirectional Encoder Representations from Transformers*)

BERT merupakan inovasi dalam NLP yang diperkenalkan oleh Google pada tahun 2018. Berbeda dengan model pendahulunya yang memproses teks secara *linear* (satu arah), BERT menerapkan pendekatan bidireksional yang memungkinkan analisis konteks secara menyeluruh dari dua arah sekaligus. Secara arsitektur, BERT mengadopsi struktur *encoder* dari *Transformer*. Model ini memanfaatkan mekanisme atensi (*attention mechanism*) untuk memetakan korelasi antarkata secara simultan tanpa dibatasi oleh jarak posisi kata dalam kalimat.

Proses pembelajaran BERT didasarkan pada dua tugas fundamental: *Masked Language Modeling* (MLM), yang melatih model memprediksi kata yang ditutupi berdasarkan konteks sekitar, dan *Next Sentence Prediction* (NSP), yang bertujuan memahami koherensi dan urutan logis antarkalimat. Sinergi kedua metode ini menjadikan BERT sangat efektif dalam menangkap nuansa semantik dan hubungan kontekstual yang kompleks (Devlin dkk., 2019).

2.2.6 IndoBERT

IndoBERT merupakan model BERT *monolingual* pertama yang dikembangkan secara khusus untuk bahasa Indonesia. Model ini dilatih menggunakan kumpulan data yang sangat besar, yaitu lebih dari 220 juta kata dari berbagai sumber. Pengembangannya mengatasi hambatan dalam analisis teks berbahasa Indonesia, terutama karena algoritma klasifikasi lain seperti *Naïve Bayes Classifier, Multi-Label Classification*, dan *Random Forest Algorithm* tidak secara eksplisit dilatih untuk bahasa Indonesia. Keterbatasan model-model sebelumnya, yang mungkin tidak mampu mengidentifikasi kata *slang* atau sarkasme, dapat diatasi oleh *IndoBERT* (Mhafudiyah dan Alamsyah, 2023).

Sebagai konsekuensi dari adaptasi mendalam tersebut, *IndoBERT*

menunjukkan lonjakan performa yang signifikan ketika dikomparasikan dengan model-model pendahulunya. Berdasarkan penelitian Koto dkk. (2020), model ini berhasil mencapai skor *State-of-the-Art* (SOTA) dalam berbagai *benchmark* NLP bahasa Indonesia. Keunggulan utamanya terletak pada kapasitas model dalam menangkap nuansa kontekstual dan kedalaman semantik (makna) yang sering kali gagal dideteksi oleh metode konvensional. Keberhasilan ini tidak hanya membuktikan efektivitas *IndoBERT*, tetapi juga menyoroti potensi besarnya sebagai solusi *robust* dalam mengatasi berbagai tantangan komputasi linguistik yang kompleks bagi bahasa Indonesia.

Dalam implementasi praktis, *IndoBERT* telah menjadi tulang punggung bagi berbagai aplikasi hilir NLP. Pada tugas analisis sentimen, model ini mampu mengklasifikasikan polaritas opini baik positif, negatif, maupun netral dengan tingkat akurasi yang presisi. Kemampuannya juga meluas pada tugas *Named Entity Recognition* (NER), di mana *IndoBERT* secara efektif mengekstraksi dan mengategorikan entitas informasi krusial seperti nama tokoh, institusi organisasi, dan lokasi geografis. Selain itu, fleksibilitas model ini turut memfasilitasi tugas-tugas pemahaman bahasa lainnya, termasuk mendukung sistem penerjemahan mesin dua arah dan mekanisme tanya jawab otomatis.