BAB 2
LANDASAN TEORI

2.1 Natural Language Processing (NLP)

Natural Language Processing (NLP) merupakan bidang interdisipliner yang mengintegrasikan komputasi linguistik dan kecerdasan buatan untuk merancang sistem yang mampu menjalankan tugas-tugas fungsional berbasis bahasa manusia [21]. Secara operasional, NLP memungkinkan mesin untuk melakukan analisis komputasional terhadap data teks, sehingga informasi linguistik dapat diproses secara terstruktur [6]. Kompleksitas utama dalam analisis ini terletak pada domain *Natural Language Processing*, mengingat setiap bahasa memiliki keragaman aturan sintaksis, morfologi, dan semantik yang menyulitkan proses komputasi jika hanya mengandalkan algoritma matematis dasar [6]. Dalam perkembangannya, pemrosesan NLP difokuskan untuk membedah struktur hierarkis teks dan mengevaluasi kepatuhan suatu kalimat terhadap aturan tata bahasa baku [22]. Kemampuan komputasi linguistik pada tingkat sintaksis inilah yang menjadi fondasi utama sebelum sebuah sistem dapat mendeteksi, mengevaluasi, dan memproses struktur bahasa secara otomatis.

2.2 Struktur Kalimat Bahasa Indonesia

Struktur kalimat merupakan unsur pokok dalam tata bahasa yang menentukan kejelasan dan keefektifan penyampaian pesan [23]. Dalam ilmu kebahasaan, struktur kalimat merujuk pada susunan unsur-unsur pembentuknya, seperti subjek, predikat, objek, pelengkap, dan keterangan, yang diatur sedemikian rupa untuk menghasilkan konstruksi dengan makna yang utuh dan jelas [24].

Pola dasar struktur kalimat yang lazim digunakan dalam bahasa Indonesia mengikuti urutan Subjek-Predikat-Objek-Keterangan (SPOK).Meskipun memiliki beragam variasi bentuk, sebuah kalimat baku minimal harus mencakup dua unsur, yakni subjek dan predikat [24]. Subjek berkedudukan sebagai pelaku tindakan atau pokok pembicaraan, sedangkan predikat menjelaskan tindakan atau keadaan dari subjek terkait. Pemahaman terhadap fungsi masing-masing unsur tersebut diperlukan untuk menyusun kalimat yang sesuai dengan kaidah tata bahasa.

Dalam analisis deteksi kesalahan berbahasa, ketepatan struktur kalimat menjadi indikator krusial. Penyimpangan dari susunan struktur baku, seperti

ketidaksesuaian antara subjek dan predikat, penggunaan kata penghubung yang keliru, atau penempatan keterangan yang tidak tepat, dapat memicu ketidakjelasan makna hingga ambiguitas informasi [25]. Oleh karena itu, penguasaan terhadap kaidah struktur kalimat berfungsi sebagai landasan untuk memastikan bahwa kombinasi frasa dan kata telah sesuai dengan tata bahasa baku, sehingga kekeliruan penafsiran informasi dapat dihindari [23].

2.2.1 Kata

Dalam analisis sintaksis, kata merupakan satuan atau konstituen terkecil yang memiliki peran spesifik dalam membentuk kelompok kata atau frasa, klausa, maupun kalimat [9] [26]. Susunan kata di dalam sebuah konstruksi bahasa harus bersifat linier, tertib, dan bermakna agar dapat menyampaikan pesan yang logis [24]. Kata dikelompokkan ke dalam berbagai kategori leksikal, seperti nomina, verba, adjektiva, adverbia, numeralia, dan preposisi, yang masing-masing menduduki peran fungsional tertentu di dalam pembentukan struktur sintaksis yang lebih besar.

2.2.2 Frasa

Frasa didefinisikan sebagai satuan gramatikal yang terdiri atas gabungan dua kata atau lebih yang sifatnya tidak predikatif dan tidak melampaui batas fungsi klausa [9]. Sebuah frasa selalu menempati dan mengisi salah satu fungsi di dalam struktur kalimat, baik menduduki posisi sebagai subjek, predikat, objek, pelengkap, maupun keterangan [9] [26]. Penggabungan kata dalam sebuah frasa merupakan perluasan yang tidak membentuk makna kata yang baru. Berdasarkan kategori kata yang menjadi unsur intinya, frasa diklasifikasikan menjadi frasa nomina, frasa verba, frasa adjektiva, frasa numeral, frasa adverbia, dan frasa preposisional.

2.2.3 Klausa

Klausa merupakan tataran gramatikal yang posisinya berada di atas frasa dan di bawah kalimat [9]. Klausa diartikan sebagai satuan bahasa berupa runtunan kata berkonstruksi predikatif [9] [26]. Di dalam konstruksi tersebut, kehadiran fungsi predikat beserta subjek bersifat wajib, sementara kehadiran unsur pendamping lain seperti objek, pelengkap, atau keterangan bersifat manasuka atau opsional [9]. Berdasarkan kemampuan distribusinya, klausa dibedakan menjadi klausa bebas

yang memiliki potensi untuk berdiri sendiri sebagai kalimat utuh, dan klausa terikat yang tidak mampu berdiri sendiri melainkan bergantung pada klausa utama. Kehadiran predikat mutlak diperlukan untuk membentuk klausa independen atau kalimat yang utuh. Ketiadaan predikat utama, misalnya akibat anak kalimat atau klausa subordinatif yang terlepas dari induk kalimatnya, akan menyebabkan susunan tersebut menjadi tidak gramatikal dan fungsi strukturalnya menggantung [24].

2.2.4 Kalimat

Kalimat merupakan satuan bahasa terkecil yang mengungkapkan pikiran atau pesan secara utuh [9]. Dalam wujud tulisan, struktur kalimat ditandai dengan awalan huruf kapital dan diakhiri dengan tanda baca final, seperti titik, tanda tanya, atau tanda seru [9]. Sebuah konstruksi gramatikal baru dapat disebut sebagai kalimat apabila sekurang-kurangnya memiliki konstituen dasar berupa subjek dan predikat [9].

A Unsur Kalimat

Sebuah kalimat secara internal dibangun oleh fungsi-fungsi sintaksis atau unsur-unsur yang memiliki peran tertentu. Secara umum, unsur-unsur pembentuk kalimat terdiri atas subjek (S), predikat (P), objek (O), pelengkap (Pel), dan keterangan (K):

1. Subjek (S)

   Bagian pokok dalam kalimat yang menandai apa atau siapa yang sedang dibicarakan. Posisinya umumnya berada di depan predikat dan sering kali diisi oleh kata atau frasa nominal [9] [24]. Secara gramatikal, fungsi subjek tidak boleh didahului oleh preposisi atau kata depan seperti di, dalam, bagi, atau untuk. Kehadiran preposisi di awal kalimat akan mengubah fungsi frasa nominal tersebut menjadi keterangan, sehingga mengakibatkan kalimat kehilangan subjeknya [24]. Selain itu, pada konstruksi kalimat majemuk yang memiliki subjek yang sama pada induk dan anak kalimat, subjek umumnya tidak perlu diulang agar struktur kalimat tetap hemat dan efektif [23][25].

2. Predikat (P)

Unsur utama yang berfungsi memberikan penjelasan mengenai subjek. Predikat dapat berupa kategori verbal, nominal, adjektival, maupun numeralia [9].

3. Objek (O)

   Unsur pendamping yang kehadirannya diwajibkan oleh predikat yang berstatus verba transitif aktif. Objek dapat beralih kedudukan menjadi subjek apabila kalimat diubah menjadi bentuk pasif [9] [24].

4. Pelengkap (Pel)

   Bagian kalimat yang berfungsi melengkapi makna predikat. Berbeda dengan objek, unsur pelengkap tidak dapat beralih kedudukan menjadi subjek dalam proses pemasifan kalimat [9] [24].

5. Keterangan (K)

   Unsur yang berfungsi memberikan informasi tambahan, seperti keterangan waktu, tempat, cara, atau tujuan. Posisi keterangan dalam kalimat umumnya lebih bebas dan dapat berpindah tempat [9] [24]. Meskipun posisinya relatif bebas, penempatan keterangan tidak boleh menyela atau memutus fungsi konstituen inti yang terikat erat, yakni antara predikat bervalensi transitif dan objek [23]. Selain itu, terdapat pula keterangan aposisi yang berfungsi sebagai penjelas nomina tambahan. Secara ortografis, keterangan aposisi di tengah kalimat wajib diapit oleh dua tanda koma secara lengkap untuk memisahkannya dari unsur utama kalimat [25].

B Jenis Kalimat

Berdasarkan jumlah klausa penyusunnya, struktur kalimat diklasifikasikan menjadi dua jenis utama [9] [24]:

1. Kalimat Tunggal

   Kalimat tunggal adalah kalimat yang hanya terdiri atas satu klausa. Artinya, kalimat ini pada dasarnya hanya memiliki satu pola hubungan tunggal antara subjek dan predikat, meskipun bisa saja diperluas dengan objek atau keterangan tambahan [24].

2. Kalimat Majemuk

Kalimat yang mengandung dua klausa atau lebih [9] [24]. Kalimat majemuk dibedakan menjadi kalimat majemuk setara yang antarklausanya dihubungkan secara koordinatif, dan kalimat majemuk bertingkat yang memiliki klausa utama serta klausa sematan atau subordinatif [24].

2.2.5 Konjungsi

Konjungsi atau kata penghubung merupakan kata tugas yang berfungsi menghubungkan kata, frasa, klausa, atau kalimat sehingga membentuk kepaduan makna dan kelogisan tata kalimat. Berdasarkan fungsinya dalam hierarki kalimat, konjungsi dibedakan menjadi[24]:

1. Konjungsi Intrakalimat
   Berfungsi menghubungkan unsur-unsur di dalam satu kalimat, baik antarklausa maupun antarunsur yang setara. Penggunaannya meliputi kata karena, sehingga, dan, atau, jika, sedangkan, dan tetapi [24].

2. Konjungsi Antarkalimat
   Berfungsi menghubungkan satu kalimat dengan kalimat lainnya untuk menciptakan kesinambungan gagasan di dalam suatu wacana atau paragraf. Contohnya meliputi frasa oleh karena itu, namun, dan kemudian [24].

Pemahaman klasifikasi dan fungsi konjungsi penting dalam analisis struktur kalimat karena kesalahan penggunaannya dapat menyebabkan ambiguitas dan kalimat menjadi tidak efektif [19]. Kesalahan yang sering terjadi meliputi penggunaan konjungsi ganda, tidak adanya konjungsi antarklausa, serta penempatan konjungsi intrakalimat di awal kalimat [23] [27].

2.3 Analisis Kesalahan Tata Bahasa

2.3.1 Grammatical Error Detection (GED)

Grammatical Error Detection (GED) didefinisikan sebagai tugas klasifikasi tingkat kata (token-level classification) yang bertujuan untuk memprediksi apakah suatu token dalam kalimat mengandung kesalahan gramatikal atau tidak [28]. Dalam kerangka kerja komputasional, GED diformulasikan sebagai masalah *sequence labeling*, di mana model memberikan label biner atau label multi-kelas yang merepresentasikan jenis kesalahan tertentu pada setiap kata input [28][29].

Deteksi yang akurat sangat bergantung pada kemampuan model untuk memahami konteks lokal di sekitar kata maupun konteks global kalimat [28].

2.3.2 Grammatical Error Correction (GEC)

Grammatical Error Correction (GEC) adalah tugas transformasi teks di mana kalimat input yang mengandung kesalahan dipetakan menjadi kalimat output yang benar secara tata bahasa dengan tetap mempertahankan makna semantik aslinya [28]. Pendekatan GEC dapat dikategorikan menjadi dua paradigma utama, yaitu pendekatan berbasis sequence generation yang mengadopsi arsitektur *Neural Machine Translation* (NMT), dan pendekatan berbasis *sequence tagging* yang memprediksi operasi edit seperti penyisipan, penghapusan, atau penggantian kata [30].

2.4 Context Free Grammar (CFG)

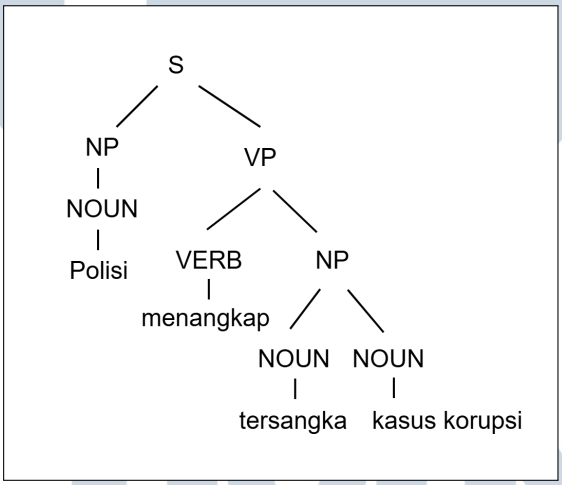
Context Free Grammar (CFG) merupakan sistem formal yang digunakan untuk mendefinisikan struktur sintaksis suatu bahasa melalui sekumpulan aturan produksi [15]. Dalam analisis *Natural Language Processing*, CFG berfungsi untuk memodelkan hubungan hierarkis antar konstituen kalimat secara eksplisit guna mendeteksi ketidaksesuaian struktur [15]. Persamaan umum dari *Context Free Grammar* didefinisikan sebagai empat tupel yang dapat dilihat pada Persamaan 2.1;

$$G = (N, \Sigma, R, S) \tag{2.1}$$

- N: Himpunan *non-terminals* atau variabel yang mewakili kategori sintaksis, seperti kalimat (S), frasa nomina (NP), atau frasa verba (VP).
- Σ: Himpunan *terminals* atau alfabet di dalam bahasa yang berupa kata-kata asli yang membentuk teks berita daring bahasa Indonesia.
- R: Himpunan *production rules* atau aturan produksi ($A \rightarrow \beta$; di mana $A \in N$ dan $\beta \in (\Sigma \cup N)^*$) yang mengatur transformasi simbol untuk mendeteksi struktur kalimat.
- S: *Start symbol* yang menandakan aturan produksi pertama yang dibaca oleh sistem untuk memulai analisis sintaksis pada kalimat.

CFG merupakan sistem formal yang bersifat universal karena hanya mendefinisikan kerangka matematis berupa himpunan simbol non-terminal, terminal, dan aturan produksi sebagaimana dirumuskan pada Persamaan 2.1, tanpa bergantung pada bahasa tertentu. Namun, penerapannya pada suatu bahasa memerlukan penyusunan aturan produksi R dan himpunan simbol terminal Σ yang disesuaikan dengan kaidah sintaksis bahasa tersebut [15].

Proses pembentukan kalimat dalam CFG dilakukan melalui mekanisme derivasi, yaitu serangkaian langkah penggantian simbol non-terminal secara bertahap hingga mencapai simbol terminal utuh. Hasil dari proses derivasi tersebut direpresentasikan secara visual melalui parse tree. Pohon urai menggambarkan hubungan struktural antar-kata, di mana setiap titik percabangan menunjukkan kategori sintaksis dan setiap daun pohon menunjukkan kata terminal [15].



Gambar 2.1. Contoh parse tree kalimat berdasarkan aturan produksi CFG

Dalam sistem deteksi kesalahan, pohon urai berperan sebagai instrumen untuk mengidentifikasi cacat sintaksis. Apabila sebuah urutan kata dalam kalimat tidak dapat membentuk struktur pohon yang valid berdasarkan himpunan aturan R, maka kalimat tersebut diidentifikasi memiliki kesalahan struktur [31].

2.4.1 Part-of-Speech (POS) Tagging

Part-of-Speech (POS) Tagging merupakan proses pelabelan kelas kata pada setiap token dalam kalimat berdasarkan fungsi gramatikalnya, seperti noun, verb, adjective, adverb, conjunction, dan preposition [32]. Dalam bidang *Natural Language Processing* (NLP), POS Tagging digunakan untuk mengidentifikasi

kategori sintaktis suatu kata sehingga dapat membantu berbagai tugas pemrosesan bahasa, seperti parsing, ekstraksi informasi, analisis sintaksis, serta machine translation [33].

Pada analisis bahasa alami, informasi kelas kata memiliki peran penting dalam memahami struktur kalimat dan hubungan antar unsur penyusunnya. Pelabelan POS memungkinkan sistem mengenali pola sintaktis tertentu, seperti hubungan antara subjek, predikat, objek, maupun keterangan dalam sebuah kalimat [33]. Selain itu, informasi POS sering digunakan sebagai tahap awal dalam berbagai metode parsing karena mampu menyediakan representasi linguistik yang lebih terstruktur dibandingkan token mentah [33].

2.4.2 Phrase Chunking

Phrase Chunking merupakan proses pengelompokan token yang telah diberi label *Part-of-Speech* (POS) ke dalam unit frasa sintaktis, seperti *Noun Phrase* (NP), *Verb Phrase* (VP), dan *Prepositional Phrase* (PP) [34]. Berbeda dengan full parsing, phrase chunking hanya mengidentifikasi konstituen utama tanpa membangun keseluruhan struktur sintaksis kalimat. Dalam NLP, metode ini digunakan untuk membantu analisis struktur bahasa dan berbagai tugas pemrosesan teks, seperti machine translation, information extraction, dan summarization [34], [35].

2.5 Text-to-Text Transfer Transformer (T5)

Text-to-Text Transfer Transformer (T5) merupakan model deep learning yang mengadopsi arsitektur *encoder-decoder Transformer* dalam kerangka kerja text-to-text yang seragam [36]. Paradigma ini memformulasikan setiap tugas kebahasaan sebagai proses transformasi urutan teks input menjadi urutan teks target. Komponen encoder berfungsi untuk mengidentifikasi konteks serta hubungan semantik dalam kalimat, sedangkan komponen decoder membangkitkan urutan teks target yang telah diperbaiki secara bertahap berdasarkan token-token yang telah dihasilkan sebelumnya [16].

Karakteristik utama T5 terletak pada mekanisme pre-training yang menggunakan objektif *span corruption*. Dalam proses ini, model dilatih untuk merekonstruksi potongan teks yang sengaja dihilangkan (masked) dalam sebuah kalimat. Melalui mekanisme tersebut, model mempelajari struktur sintaksis,

dependensi antarkata, serta kaidah kebahasaan secara implisit. Paradigma ini memberikan keunggulan pada tugas *Grammatical Error Correction* (GEC) karena model mampu melakukan penulisan ulang untuk memperbaiki kesalahan struktur kalimat tanpa menghilangkan makna semantik aslinya.

2.6 Evaluasi Model

Evaluasi model dilakukan untuk mengukur kinerja model dalam menyelesaikan tugas yang diberikan [37]. Hasil evaluasi digunakan untuk menilai tingkat ketepatan model dalam menghasilkan prediksi atau keluaran yang sesuai dengan data referensi [37].

2.6.1 Confusion Matrix

Confusion Matrix merupakan tabel evaluasi yang digunakan untuk membandingkan hasil prediksi model dengan label sebenarnya (ground truth) [37]. Matriks ini terdiri atas empat komponen utama, yaitu *True Positive* (TP), *True Negative* (TN), *False Positive* (FP), dan *False Negative* (FN). Struktur dasar *Confusion Matrix* ditunjukkan pada tabel 2.1. Berdasarkan nilai-nilai pada Confusion Matrix, beberapa metrik evaluasi dapat dihitung sebagai berikut [37].

Tabel 2.1. Confusion Matrix

	Actual Positive	*Actual Negative*
Predicted Positive	TP	FP
Predicted Negative	FN	TN

Berdasarkan nilai-nilai pada tabel di atas, kinerja model dihitung menggunakan empat metrik evaluasi utama sebagai berikut [37]:

1. Accuracy

   Rasio total prediksi benar terhadap keseluruhan data yang memberikan gambaran umum kinerja model.

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN} \tag{2.2}$$

2. Precision

Tingkat ketepatan prediksi positif model yang krusial untuk meminimalkan kesalahan tipe false positive.

$$Precision = \frac{TP}{TP + FP} \tag{2.3}$$

3. Recall

   Kemampuan model dalam menemukan kembali seluruh data positif guna meminimalkan kesalahan tipe false negative.

$$Recall = \frac{TP}{TP + FN} \tag{2.4}$$

4. F1-Score

   Rata-rata harmonis antara presisi dan recall yang menjadi acuan utama pada dataset tidak seimbang.

$$F1Score = 2 * \frac{Precision * Recall}{Precision + Recall} \tag{2.5}$$

2.6.2 Cross-Entropy Loss Function

Cross-Entropy Loss merupakan fungsi kerugian (loss function) yang digunakan untuk mengukur perbedaan antara distribusi probabilitas hasil prediksi model dan distribusi probabilitas target yang sebenarnya. Fungsi ini banyak digunakan dalam berbagai aplikasi machine learning dan deep learning karena efektif dalam mengoptimalkan proses pembelajaran melalui minimisasi kesalahan prediksi [38]. Secara matematis, Cross-Entropy Loss dirumuskan sebagai berikut:

$$L = -\sum_{i=1}^{C} y_i \log(p_i) \tag{2.6}$$

dengan:

- L = nilai *Cross-Entropy Loss*;
- C = jumlah kelas;

- y_i = nilai target aktual pada kelas ke-i;
- p_i = probabilitas prediksi model pada kelas ke-i.

Pada proses pelatihan model, fungsi *Cross-Entropy Loss* digunakan untuk mengevaluasi kesalahan prediksi pada setiap iterasi. Nilai *loss* yang lebih rendah menunjukkan bahwa distribusi probabilitas yang dihasilkan model semakin mendekati distribusi target yang sebenarnya.

2.6.3 Exact Match

Exact Match (EM) merupakan metrik evaluasi yang digunakan untuk mengukur tingkat kesesuaian sempurna antara kalimat hasil prediksi model dan kalimat referensi (ground truth) [39]. Suatu prediksi dianggap benar hanya apabila seluruh susunan kata, karakter, dan tanda baca identik dengan referensi. Apabila terdapat perbedaan sekecil apa pun, maka prediksi dianggap tidak cocok dan memperoleh nilai 0 [39]. Rumus Exact Match dapat dihitung sebagai berikut:

$$EM = \frac{\text{Jumlah Prediksi Benar}}{\text{Jumlah Seluruh Data}} \tag{2.7}$$

Nilai EM berada pada rentang 0 hingga 1, atau dapat dinyatakan dalam bentuk persentase dengan mengalikan hasilnya dengan 100

2.6.4 ROUGE-L

ROUGE-L (*Recall-Oriented Understudy for Gisting Evaluation*) merupakan metrik evaluasi yang mengukur tingkat kemiripan antara kalimat hasil prediksi dan kalimat referensi berdasarkan konsep *Longest Common Subsequence* (LCS) [40]. Semakin panjang subsekuens yang sama antara dua kalimat, semakin tinggi tingkat kemiripan keduanya [40]. Berbeda dengan ROUGE-N yang menggunakan pencocokan *n-gram*, ROUGE-L mempertimbangkan urutan kata dalam kalimat tanpa mengharuskan kata-kata tersebut bersebelahan.

Perhitungan ROUGE-L diawali dengan menghitung nilai *recall* dan *precision* berbasis LCS menggunakan Persamaan 2.8 dan Persamaan 2.9.

$$R_{LCS} = \frac{LCS(X,Y)}{m} \tag{2.8}$$

$$P_{LCS} = \frac{LCS(X,Y)}{n} \tag{2.9}$$

dengan:

- $LCS(X,Y)$ adalah panjang *Longest Common Subsequence* antara kalimat referensi dan kalimat prediksi.
- X adalah kalimat referensi (*reference sentence*).
- Y adalah kalimat hasil prediksi (*predicted sentence*).
- m adalah jumlah kata pada kalimat referensi.
- n adalah jumlah kata pada kalimat prediksi.

Selanjutnya nilai ROUGE-L dihitung menggunakan F-Measure:

$$ROUGE\text{-}L = \frac{(1+\beta^2)R_{LCS}P_{LCS}}{R_{LCS}+\beta^2 P_{LCS}} \tag{2.10}$$

di mana:

- R_{LCS} adalah nilai *recall* berbasis LCS.
- P_{LCS} adalah nilai *precision* berbasis LCS.
- β adalah parameter yang mengatur keseimbangan antara *precision* dan *recall*.

Nilai ROUGE-L berada pada rentang 0 hingga 1, di mana nilai yang lebih tinggi menunjukkan tingkat kesamaan yang lebih baik antara hasil prediksi dan referensi [40].

2.6.5 GLEU Score

GLEU (*Generalized Language Evaluation Understudy*) merupakan metrik evaluasi yang dirancang khusus untuk tugas *Grammatical Error Correction* (GEC) [41]. GLEU mengukur kesamaan n-gram antara hasil prediksi dan kalimat referensi, sekaligus memberikan penalti terhadap perubahan yang tidak diperlukan. Rumus dasar GLEU dapat dinyatakan sebagaimana pada Persamaan 2.11.

$$GLEU = \min\left(\frac{Count_{match}}{Count_{candidate}}, \frac{Count_{match}}{Count_{reference}}\right) \tag{2.11}$$

dengan:

- $Count_{match}$ adalah jumlah *n-gram* yang cocok antara kalimat prediksi dan kalimat referensi.
- $Count_{candidate}$ adalah jumlah *n-gram* pada kalimat hasil prediksi.
- $Count_{reference}$ adalah jumlah *n-gram* pada kalimat referensi.

Dalam implementasinya, GLEU biasanya menghitung rata-rata skor untuk beberapa ukuran n-gram (1-gram hingga 4-gram) sehingga diperoleh nilai evaluasi yang lebih stabil [41]. Secara umum, nilai GLEU dapat dirumuskan sebagai:.

$$GLEU = \exp\left(\frac{1}{N}\sum_{n=1}^{N}\log p_n\right) \tag{2.12}$$

di mana:

- N adalah jumlah ukuran *n-gram* yang digunakan.
- p_n adalah nilai *precision n-gram* yang telah mempertimbangkan penalti terhadap perubahan yang tidak diperlukan.

Semakin tinggi nilai GLEU, maka semakin baik kualitas hasil koreksi yang dihasilkan model terhadap kalimat referensi [41].