Analisis Komparatif Arsitektur RNN dan Transformer dengan Studi Ablasi pada Tokenisasi untuk Penerjemahan Inggris-Indonesia

Nasywa Kynda Sanina
442023618074
Teknik Informatika
Universitas Darussalam Gontor
nasywakyndasanina64@student.cs.unida.gontor.ac.id

Abstrak*—Penelitian ini bertujuan untuk mengevaluasi dan membandingkan kinerja dua arsitektur Neural Machine Translation (NMT) utama, yaitu Recurrent Neural Network (RNN) dengan mekanisme atensi dan Transformer. Evaluasi dilakukan pada tugas terjemahan dari Bahasa Inggris ke Bahasa Indonesia. Selain arsitektur model, studi ini juga menganalisis dampak dari dua strategi tokenisasi yang berbeda: word-level dan subword-level (menggunakan SentencePiece). Empat model eksperimental dilatih dan dievaluas: (1) RNN dengan tokenizer word-level sebagai baseline, (2) RNN dengan tokenizer subword, (3) Transformer dengan subword, dan (4) Transformer dengan tokenizer word-level. Hasil evaluasi menggunakan metrik SacreBLEU menunjukkan bahwa model Transformer yang dikombinasikan dengan tokenisasi subword mencapai kinerja tertinggi dengan skor 31.01, hasil ini menegaskan superioritas arsitektur Transformer dalam menangkap dependensi kontekstual dan efektivitas tokenisasi subword dalam mengangani kosakata yang beragam dan kata-kata langka.

Keywords*—Transformer, RNN, Tokenisasi Subword, Ablasi, Penerjemahan Inggris-Indonesia

I. PENDAHULUAN

Neural Machine Translation (NMT) telah menjadi standar de-facto dalam sistem terjemahan otomatis, menggantikan metode statistik tradisional karena kemampuannya menghasilkan terjemahan yang lebih flúen dan akurat. Perkembangan NMT didominasi oleh dua arsitektur utama: model sekuensial berbasis *Recurrent Neural Network* (RNN) dan arsitektur non-sekuensial berbasis *self-attention* yang dikenal sebagai Transformer.

Model RNN, khususnya dengan arsitektur *encoder-decoder* dan mekanisme atensi, memproses teks secara berurutan. Meskipun efektif, sifak sekuensial ini membatasi paralelisasi dan dapat kesulitan menangkap dependensi jarak jauh dalam teks. Di sisi lain, model Transformer, yang sepenuhnya mengandalkan mekanisme atensi, mampu memproses seluruh token dalam sebuah secara bersamaan, sehingga lebih efisien dalam pelatihan dan lebih unggul dalam memodelkan konteks global.

Selain arsitektur, metode tokenisasi—proses memecah teks menjadi unit-unit yang lebih kecil (token)—juga memainkan peran krusial. Tokenisasi *word-level* yang sederhana rentan terhadap masalah *out-of-volabulary* (OOV), di mana model tidak dapat menangani kata-kata yang tidak ada dalam data latih. Sebagai solusi, metode *subword-level* memecah kata menjadi unit yang lebih kecil dan bermakna, memungkinkan mode untuk menangani kata langka dan membangun representasi kata baru dari bagian-bagian yang sudah dikenal.

Penelitian ini bertujuan untuk melakukan analisis komparatif yang sistematis terhadap kombinasi arsitektur dan metode tokenisasi tersebut pada tugas terjemahan Bahasa Inggris ke Bahasa Indonesia.

II. AKTIVITAS TERKAIT

Penelitian mengenai *Neural Machine Translation* (NMT) untuk Bahasa Indonesia telah berkembang sejalan dengan tren global, dimulai dari eksplorasi arsitektur berbasis RNN. Pendekatan awal ini berfokus pada pemanfaatan varian RNN seperti *Long Short-Term Memory* (LSTM) yang dikombinasikan dengan mekanisme atensi untuk mengatasi masalah dependensi jarak jauh dalam kalimat. Studi oleh Huda (2020) secara spesifik mengembangkan model penerjemah Inggris-Indonesia menggunakan LSTM dan mekanisme atensi, menunjukkan bahwa arsitektur ini mampu menangani urutan data yang panjang dan meningkatkan akurasi terjemahan secara signifikan.

Penelitian lebih lanjut pada arsitektur berbasis RNN juga dilakukan oleh Gunawan & Tursina (2021), yang membandingkan efektivitas mekanisme atensi Bahdanau dan Luong untuk penerjemahan Bahasa Indonesia ke Bahasa Melayu Ketapang. Studi ini menunjukkan bahwa eksplorasi pada komponen atensi merupakah salah satu fokus riset untuk meningkatkan kualitas mode sekuensial. Dalam penelitian tersebut, arsitektur Transformer juga telah disinggung sebagai pengembangan teknologi NMT yang lebih baru.

Seiring perkembangannya, fokus penelitian di Indonesia mulai bergeser ke arah perbandingan antara arsitektur RNN yang sudah ada dengan model Transformer yang lebih modern. Meskipun dalam tugas yang berbeda seperti peringkasan teks, penelitian oleh Ardyanti (2023) secara langsung membandingkan kinerja model LSTM dengan atensi melawan arsitektur Transformer. Hal ini menunjukkan bahwa perbandingan antara kedua arsitektur tersebut merupakan area riset yang aktif dan krusial untuk menentukan praktik terbaik dalam pemrosesan Bahasa Indonesia.

Di samping inovasi arsitektur, pemilihan metode tokenisasi juga menjadi faktor penentu performa model NLP, terutama untuk bahasa dengan morfologi yang kayak seperti Bahasa Indonesia. Sebagai solusinya, penelitian di Indonesia telah mengkaji penggunaan tokenisasi *subword*. Studi oleh Siahaan & Purwanti (2020) secara khusus membandingkan algoritma *subword* dalam *framework SentencePiece* untuk teks Bahasa Indonesia, menunjukkan potensinya sebagai metode tokenisasi *unsupervised* yang andal. Lebih lanjut, penelitian oleh Yovita & Suhartono (2022) membuktikan secara empiris bahwa implementasi *SentencePiece* mampu meningkatkan performa model klasifikasi berita Bahasa Indonesia dibandingkan dengan tokenisasi biasa, menegaskan bahwa pendekatan *subword* lebih efektif dalam merepresentasikan teks.

III. METODE

A. Model Arsitektur

a) RNN (Seq2Seq dengan Atensi): model ini terdiri dari *encoder* yang menggunakan GRU (*Gated Recurrent Unit*) bidirectional dengan dimensi embedding 256 dan hidden state 512. Encoder akan memproses sekuens input dan menghasilkan output states serta hidden state final. Mekanisme atensi yang digunakan adalah model aditif Bahdanau, yang diimplementasi dalam kelas BahdanauAttentionQKV. Sedangkan Decoder menggunakan GRU indirectional dengan dimensi embedding 256 dan hidden state 256. Prediksi final (logits) dihasilkan oleh lapisan linear yang menerima gabungan dari *output state* GRU, *context vector*, dan *word embedding*.

b) Transformer: model ini juga menggunakan arsitektur Transformer standar berbasis *self-attention* yang diimplementasikan melalui model nn.Transformer dari PyTorch. Setiap blok tersiri dari sub-lapisan *multi-head self-attention* dan *position-wise feed-forward network.* Informasi posisi token diinjeksikan menggunakan *sinusoidal Positional Encoding*. Arsitektur ini dapat dikonfigurasi melalui beberapa argumen, termauk d_model, nhead, num_encoder_layers, num_decoder_layers, dan dim_feedforward.

B. *Tokenization*

a) Word-level: Proses tokenisasi ini diatur oleh fungsi normalize_and_tokenize. Teks input pertama-tama melalui beberapa langkah normalisasi: konversi ke huruf kecil, normalisasi Unicode (NFKC), dan pembersihan karakter non-alfanumerik serta non-tanda baca menggunakan ekspresi reguler. Setelah itu, teks yang sudah bersih dibagi menjadi token berdasarkan spasi.

b) *Subword-level*: Metode ini diimplementasikan menggunakan library sentencepiece dengan algoritma *Byte Pair Encoding* (BPE) sebagai model_type default. Teks dipecah menjadi unit-unit linguistik yang lebih kecil dari kata (*subword*). Pendekatan ini secara efektif menangani masalah OOV karena kata-kata yang tidak dikenal dapat direpresentasikan sebagai sekuens dari subword yang ada dalam kosakata.

C. Evaluasi Matrik

Kualitas terjemaan dievaluasi menggunakan SacreBLEU, sebuah implementasi standar dari emtrik BLEU (*Bilingual Evaluation Understudy.* Fungsi evaluate_sacrebleu dalam kode akan secara spesifik memanggil library sacrebleu untuk menghitung skor *corpus-level*. BLEU mengukur kesamaan antara terjemahan yang dihasilkan mesin dengan satu atau lebih terjemahan referensi berkualitas tinggi dengan menghitung presisi n-gram yang dimodifikasi. Skor berkisar antara 0 hingga 100, di mana skor yang lebih tinggi akan menunjukkan kualitas yang lebih baik.

IV. EKSPERIMEN

A. Set Eksperimen

Penelitian ini menggunakan dataset pasangan kalimat paralel Bahasa Inggris-Indonesia yang bersumber dari proyek Tatoeba dan didistribusikan melalui situs web ManyThings.org. Empat eksperimen dilakukan untuk membandingkan kombinasi model dan tokenizer yang berbeda. Seluruh model dilatih selama 20 epoch dalam platform CPU.

- Eksperimen 1: Baseline (RNN + *Word-level*)
- Eksperimen 2: RNN Ablation (RNN + *Subword*)
- Eksperimen 3: Transformer (Transformer + *Subword*)
- Eksperimen 4: Transformer Ablation (Transformer + *Word-level*)

Eksperimen 2 dan 4 secara spesifik dirancang sebagai studi ablasi. Tujuannya adalah untuk mengisolasi dan mengukur dampak dari komponen tokenisasi. Dengan membandingkan kinerja model yang menggunakan tokenisasi *subword* dengan versi yang menggunakan tokenisasi *word-level* yang lebih sederhana, kontribusi dari metode tokenisasi *subword* terhadap akurasi model dapat diukur secara kuantitatif.

B. Hiperparameter

Tabel berikut merangkum hiperparameter utama yang digunakan dalam setiap eksperimen

TABEL 1 Tabel Hiperparameter Dari Empat Eksperimen

Parameter	**Eks.1**	**Eks.2**	**Eks.3**	**Eks.4**
Model	RNN	RNN	Transformer	Transformer
Tokenizer	word	sp	sp	word
Epochs	20	20	20	20
Learning Rate	1e-4	5e-4	N/A (Warmup Scheduler)	N/A (Warmup Scheduler)
Dropout	0.3	0.2	N/A (Default)	N/A (Default)
Model Dim.	N/A	N/A	256	256
Num. Heads	N/A	N/A	8	8
Num. Layers	N/A	N/A	4 Enc/4 Dec	4 Enc/4 Dec
FF Dim.	N/A	N/A	1024	1024
Checkpoint	rnn_word.pt	rnn_subword.pt	transformer_subword.pt	transformer_word.pt
Output	rnn_word_history	rnn_subword_history	transformer_subword_history	transformer_word_history

V. HASIL ANALISIS DAN DISKUSI

A. Hasil Kuantitatif

Hasil evaluasi akhir pada *test set* disajikan dalam tabel 2. Metrik yang digunakan untuk menghitung nilai akhir performa model adalah *Loss*, *Perplexity* (PPL), dan SacreBLEU.

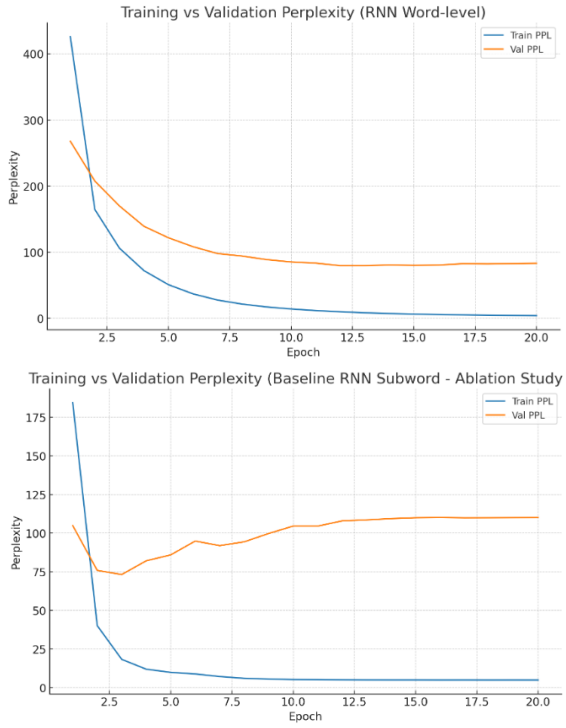
TABEL 2 Metrik Penilaian Test Akhir

Model Arsitektur	Tokenizer	Test Loss	Test PPL	Test SacreBLEU
Baseline RNN	Word-level	4.2894	72.92	12.34
Baseline RNN	Subword	4.6211	101.6	19.72
Transformer	**Subword**	**3.4886**	**32.74**	**31.01**
Transformer	Word-Level	3.9596	54.44	25.48

Berdasarkan hasil uji coba, model Transformer dengan tokenisasi *subword* secara meyakinkan memberikan kinerja terbaik, mengungguli semua model lainnya di seluruh metrik. Model ini mencapai skor SacreBLEU tertinggi sebesar 31.01, serta nilai *Loss* dan PPL terendah pada data tes. Analisis lebih lanjut menyoroti dua faktor utama. Pertama, dampak arsitektur, di mana Transformer secara lebih konsisten unggul daripada RNN; bahkan saat menggunakan tokenisasi *word-level* yang kurang optimal, Transformer (SacreBLEU 25.48) masih melampaui model RNN terbaik yang menggunakan *subword* (SacreBLEU 19.72).

Kedua, dampak tokenizer, di mana penggunaan tokenisasi *subword* memberikan peningkatan kinerja yang signifikan pada kedua arsitektur. Pada model RNN, skor BLEU meningkat dari 12.34 menjadi 19.72, sementara pada model Transformer peningkatannya bahkan lebih besar, dari 25.48 menjadi 31.01. ini menunjukkan bahwa tokenizer *subword* memberikan kontribusi sebesar 5.53 poin SacreBLEU pada model Transformer.

B. Analisis Kurva Pelatihan



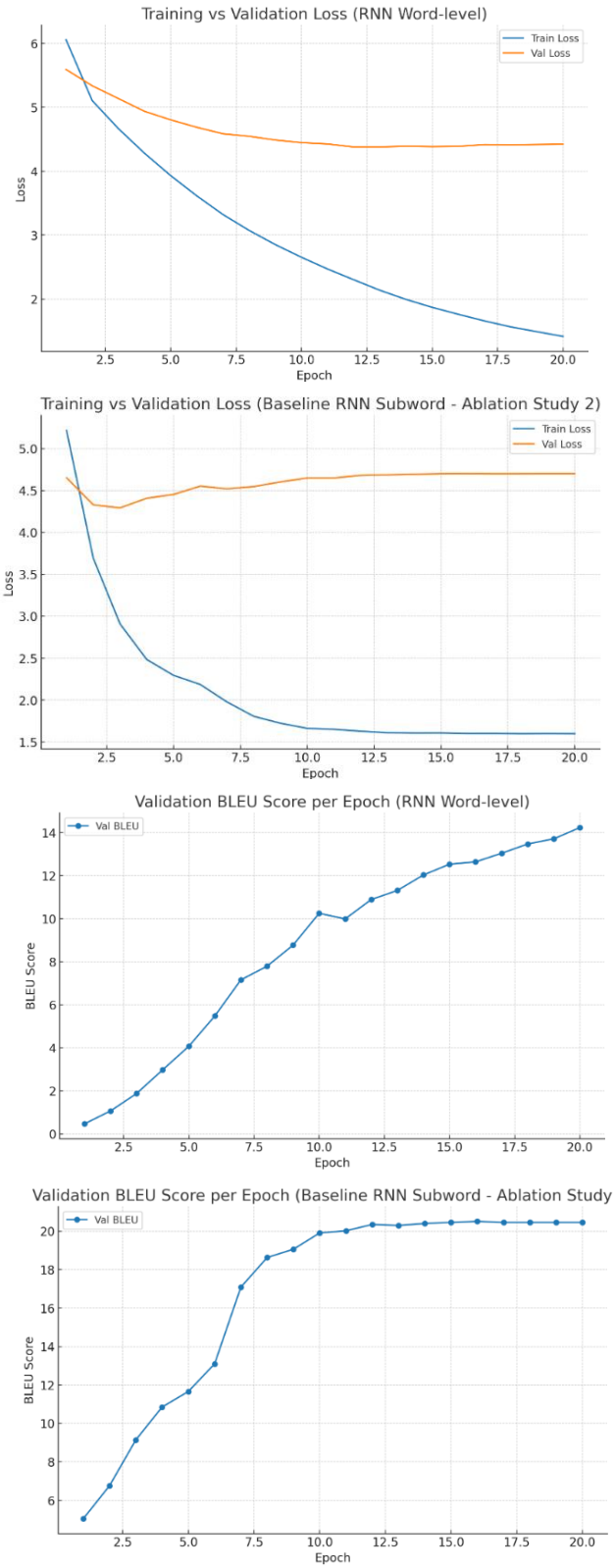
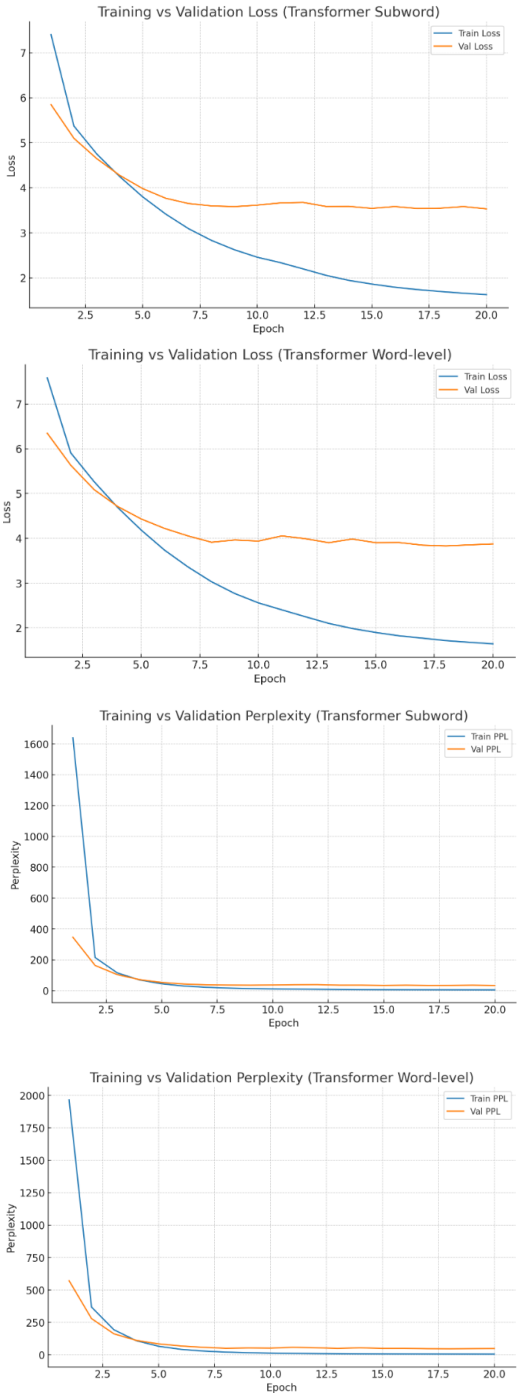


Fig 1. Grafik *loss,* PPL, dan SacreBLEU arsitektur RNN *word-level* dan *subword.*

Analisis kurva pelatihan memberikan wawasan lebih dalam mengenai proses belajar setiap model. Kedua model RNN, baik yang menggunakan *word-level* maupun *subword*, menunjukkan tanda-tanda *overfitting.* Grafik loss dari kedua model tersebut menunjukkan bahwa *validation loss* cenderung stagnan atau bahkan meningkat setelah beberapa epoch sementara *training loss* terus menurun, sebuah indikasi bahwa model mulai menghafal data latih.



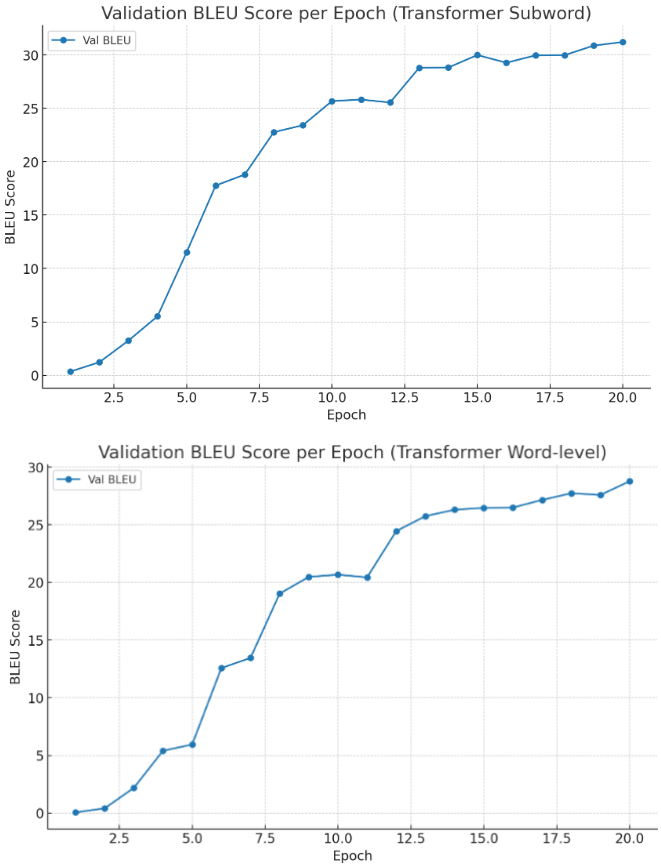


Fig 2. Grafik *loss,* PPL, dan SacreBLEU arsitektur Transformer *subword* dan *word-level.*

Sebaliknya, model Transformer menunjukkan kurva pelatihan yang lebih sehat. *Validation loss* pada kedua varian Transformer menurun secara konsisten bersamaan dengan *training loss,* dan jarak antara keduanya tidak terlalu besar yang menandakan kemampuan generalisasi yang lebih baik. Hal ini didukung oleh skor *validation* BLEU yang terus meningkat secara stabil sepanjang proses pelatihan hingga mencapai puncaknya pada epoch terakhir.

C. Analisis Kualitatif dan Kesalahan

Analisis contoh terjemahan dari *test set* menyoroti kekuatan dan kelemahan masing-masing model.

TABEL 3 Analisis Kualitatif dan Kesalahan Model

Model	SRC	TRG	PRED	Analisis
RNN + Word	i can't remember the melody of that song.	saya tidak bisa mengingat melodi lagu itu.	aku tidak bisa mengingat melodi itu. Itu.	pengulangan kata "itu" secara berulang
RNN + Subword	here is our answer to your fax message dated april 1st.	ini adalah jawab kami untuk pesan faks Anda pada tanggal 1 april.	adalah kita adalah Anda pada 28 denganuranila...	struktur rusak, meskipus tanpa <unk>, struktur kalimat dan pemilihan kata sangat buruk pada kalimat kompleks
Transformer + Word	that was a beautiful speech.	itu adalah pidato yang indah.	itu ubur-ubur.	OOV, model gagal menerjemahkan speech sehingga menghasilkan terjemahan yang sama sekali tidak relevan

Transformer + Word	tom is an <unk> <unk>	tom seorang fotografer <unk>	tom adalah seorang kawanku.	kesalahan konteks karena token <unk>, model menebak terjemahan yang salah secara kontekstual
Transformer + Subword	i'm not a doctor.	saya bukan dokter.	saya bukan dokter.	terjemahan sempurna identik dan akurat
Transformer + Subword	are you taking any medicine regularly?	apakah Anda mengonsumsi obat-obatan tertentu secara rutin?	apakah kamu berolahraga secara rutin?	terjemahan secara gramatikal benar, namun maknanya jadi bergeser dari "minum obat" jadi "berolahraga"

Analisis kualitatif dari contoh-contoh terjemahan mengkonfirmasi temuan kuantitatif dari metrik evaluasi. Terbukti bahwa model berbasis *wrod-level,* baik RNN maupun Transformer, sangat berpengaruh oleh token di luar kosakata <unk>, yang seringkali menghasilkan terjemahan yang tidak masuk akan atau salah secara kontekstual. Secara khusus, model RNN juga menunjukkan kesulitan dalam menjaga koherensi dan struktur kalimat yang lebih panjang, yang terlihat dari adanya pengulangan kata yang tidak perlu atau hasil yang tidak logis.

Sebaliknya, model Transformer dengan tokenisasi *subword* secara konsisten menghasilkan terjemahan yang paling akurat dan flúen, bahkan mampu menerjemahkan beberapa kalimat dengan sempurna. Meskipun demikian, model ini terkadang masih melakukan kesalahan semantik minor, di mana struktur kalimat benar namun maknanya bergeser dari kalimat asli.

Berdasarkan temuan ini, langkah paling fundamental untuk meningkatkan model adalah pada kualitas dan diversitas korpus pelatihan. Penambahan pasangan kalimat dari domain yang lebih beragam (misalnya, teks formal, teknis, atau sastra) akan memperkaya pemahaman konteks model, membantu membedakan nuansa makna yang lebih subtil, dan pada akhirnya dapat mengurangi kesalahan semantik.

Selain itu, dari sisi teknis, implementasi strategi *decoding* yang lebih canggih seperti Beam Search pada saat inferensi dapat diterapkan, karena berpotensi menghasilkan hipotesis terjemahan yang lebih optimal dibandingkan pendekatan *greedy*. Sebagai langkah lebih lanjut, *fine-tuning* pada model bahasa pra-terlatih (*pre-trained models*) berskala besar yang telah memiliki pemahaman semantik mendalam, seperti mBART atau NLLB, dapat menjadi jalur yang menjanjikan untuk mencapai tingkat akurasi yang lebih tinggi.

VI. Conclusion

Eksperimen ini secara komprehensif membandingkan empat varian model NMT dan menghasilkan kesimpulan yang jelas. Pertama, arsitektur Transformer telah secara signifikan mengungguli arsitektur RNN dalam tugas terjemahan Inggris-Indonesia, kedua, tokenisasi *subword-level* secara konsisten memberikan peningkatan kualitas terjemahan yang substansial dibandingkan *word-level,* baik model RNN maupun Transformer.

Kombinasi arsitektur Transformer dengan tokenisasi *subword* terbukti menjadi yang paling efektif, menghasilkan skor SacreBLEU tertinggi (31.01) dan terjemahan kualitatif terbaik. Hasil ini menegaskan pentingnya pemilihan arsitektur yang mampu menangkap konteks global dan strategi tokenisasi yang kuat untuk membangun sistem NMT yang andal.

Untuk penelitian di masa depan, peningkatan dapat dieksplorasi melalui penggunaan dataset yang lebih besar, penyesuaian hiperparameter yang lebih mendalam, serta pemanfaatan model pra-terlatih (*pre-trained models*) yang lebih besar.

References

[1] Huda, A. (2020). *Neural machine translation with attention mechanism for English-Indonesian translation* [Skripsi tidak dipublikasikan]. Universitas Islam Indonesia.

[2] Gunawan, W., Sujaini, H., & Tursina, T. (2021). Analisis perbandingan nilai akurasi mekanisme attention Bahdanau dan Luong pada Neural Machine Translation Bahasa Indonesia ke Bahasa Melayu Ketapang dengan arsitektur Recurrent Neural Network. *Jurnal Edukasi dan Penelitian Informatika (JEPIN), 7*(3), 488-494.

[3] Ardyanti, C. P. R. (2023). Peringkasan teks berita berbahasa Indonesia menggunakan LSTM dan Transformer. *IJAI (Indonesian Journal of Applied Informatics), 7*(2), 145-154.

[4] Siahaan, F. R., & Purwarianti, A. (2020). Perbandingan algoritma Sentencepiece BPE dan Unigram pada tokenisasi artikel Bahasa Indonesia. *e-Proceeding of Engineering, 7*(2), 5396-5404.

[5] Yovita, L., & Suhartono, D. (2022). Implementasi algoritma Sentencepiece untuk meningkatkan performa Naive Bayes Classifier pada klasifikasi artikel berita. *Prosiding Konferensi Ilmiah Mahasiswa Unpar, 2*(1), 841-847.

[6] M. Young, The Technical Writer's Handbook. Mill Valley, CA: