(2025), 3 (7): 46-54 **Scientica** 3021-8209

Jurnal Ilmiah Sain dan Teknologi

PERBANDINGAN TEKNIK DATA TEXT AUGMENTATION SYNONYM REPLACEMENT DAN KOMBINASI SYNONYM REPLACEMENT–STEMMING UNTUK MENINGKATKAN KINERJA NMT BAHASA INDONESIA–DAYAK BANJUR

Natalia Artika [#1], Herry Sujaini [*2], Rina Septiriana [#3]

[#]*Program Studi Informatika, Fakultas Teknik, Universitas Tanjungpura*
Jl. Prof. Dr.H. Hadari Nawawi, Pontianak, Kalimantan Barat 78115
[1]d1041211008@student.untan.ac.id

Abstrak (Indonesia)

Ketersediaan data paralel yang terbatas menjadi tantangan utama dalam pengembangan sistem *Neural Machine Translation* (NMT) untuk bahasa daerah, termasuk Bahasa Dayak Banjur. Penelitian ini berfokus pada evaluasi efektivitas dua teknik augmentasi data berbasis semantik, yaitu *synonym replacement* dan *stemming + synonym replacement*, dalam meningkatkan kinerja penerjemahan otomatis dari Bahasa Indonesia ke Bahasa Dayak Banjur. Dataset awal terdiri dari 5.000 pasangan kalimat paralel yang kemudian diperluas melalui masing-masing teknik augmentasi menjadi total 10.000 pasangan kalimat. Model NMT dikembangkan menggunakan arsitektur *encoder-decoder* berbasis *Recurrent Neural Network* (RNN) dengan mekanisme Bahdanau *Attention*. Pelatihan dilakukan secara terpisah pada masing-masing dataset dan dievaluasi menggunakan metrik BLEU. Hasil evaluasi menunjukkan bahwa *synonym replacement* memberikan dampak paling positif terhadap kualitas terjemahan, terbukti dari BLEU score tertinggi sebesar 48,19%. Skor ini lebih unggul dibanding *stemming + synonym replacement* yang hanya mencapai 46%, menunjukkan bahwa pelestarian bentuk kata asli lebih efektif dalam mempertahankan konteks kalimat. Adapun model tanpa augmentasi mencatatkan skor terendah, yaitu 42,17%. Temuan ini mengindikasikan bahwa augmentasi leksikal murni lebih efektif dibandingkan kombinasi dengan stemming, karena mampu memperkaya variasi kata tanpa mengorbankan keutuhan makna atau struktur kalimat. Sebaliknya, proses *stemming* dapat mengaburkan konteks dan mengurangi keterbacaan, sehingga berdampak pada penurunan akurasi terjemahan. Penelitian ini menegaskan pentingnya pemilihan teknik augmentasi yang tepat untuk mendorong generalisasi model NMT pada data sumber daya rendah.

Sejarah Artikel
Submitted: 3 November 2025
Accepted: 6 November 2025
Published: 7 November 2025

Kata Kunci
Bahasa Dayak Banjur, Bahdanau *Attention*, BLEU, Data *Augmentation, Neural Machine Translation, Stemming.*

I. PENDAHULUAN

Bahasa adalah sistem yang terdiri dari kata-kata dan aturan tata bahasa yang digunakan untuk membentuk kalimat yang bermakna, baik dalam bentuk lisan maupun tulisan. Bahasa digunakan sebagai alat komunikasi yang memungkinkan manusia menyampaikan informasi, mengekspresikan emosi, dan membangun hubungan sosial [1].

Indonesia memiliki bahasa daerah yang beragam, menurut Badan Pengembangan dan Pembinaan Bahasa terdapat 718 bahasa yang telah terdaftar pada situs peta bahasa di Indonesia [2]. Meskipun masih belum terdaftar pada situs peta bahasa di Indonesia, bahasa Dayak Banjur merupakan salah satu dari ratusan bahasa daerah yang ada di Indonesia. Bahasa Dayak Banjur adalah bahasa yang dipakai sebagian kecil masyarakat di Kalimantan Barat, secara khusus di beberapa daerah di kabupaten Sekadau dan kabupaten Sintang. Seiring perkembangan zaman jumlah penutur bahasa Banjur ini semakin sedikit, bahkan banyak generasi muda yang tidak lagi menggunakan bahasa Banjur.

Bahasa Dayak Banjur dalam pengucapan atau penuturannya relatif tidak terlalu sulit, serta

terdapat beberapa bahasa Dayak lain yang hampir mirip seperti Dayak Sebruang. Misalnya dalam kalimat bahasa Indonesia "Bagaimana cara mengerjakan ini?", dalam bahasa Dayak Banjur "Kati cara ngerja e tuk?" dan dalam bahasa Dayak Sebruang "Tipa cara ngerjakan e tuk?".

Sebagai upaya untuk tetap mempertahankan kelestarian bahasa Dayak Banjur, dibuat mesin penerjemah jaringan saraf tiruan bahasa Indonesia ke bahasa Dayak Banjur. Mesin penerjemah jaringan saraf tiruan merupakan mesin terjemahan yang terinspirasi dari cara kerja otak manusia untuk memproses dan menerjemahkan teks. *Neural Machine Translation* (NMT) adalah pendekatan baru dalam teknologi penerjemahan yang menggabungkan *encoder* dan *decoder*. *Encoder* berfungsi sebagai jaringan saraf berulang (RNN) yang mengonversi bahasa sumber menjadi vektor-vektor dengan panjang tetap, sedangkan *decoder* adalah jaringan saraf berulang lain yang menghasilkan terjemahan secara menyeluruh [5][11].

Untuk membangun model NMT diperlukan data dalam jumlah yang besar dan bervariasi agar model tidak terlalu hafal data latih yang membuat model jadi kurang bagus untuk data baru atau yang disebut dengan *overfitting*, sehingga dalam hal ini teknik *data augmentatio*n diperlukan. Augmentasi data adalah sebuah teknik yang efektif untuk menambah variasi data untuk model pembelajaran mesin. Dengan meningkatkan variasi pada data latih, augmentasi data membantu model untuk belajar pola yang lebih umum dan menjadi lebih *robust* [4]. Teknik augmentasi yang digunakan dalam penelitian ini, yang pertama *synonym replacement* yaitu mengganti kata-kata dalam kalimat dengan sinonimnya. Yang kedua, *stemming + synonym replacement* yaitu melakukan *stemming* atau pengubahan kata menjadi bentuk dasarnya terlebih dahulu sebelum melakukan penggantian kata dengan sinonimnya.

Berdasarkan uraian penjelasan yang telah dipaparkan, dapat disimpulkan bahwa dalam penelitian ini dilakukan empat teknik augmentasi data teks dan membangun sebuah model *Neural Machine Translation* (NMT) bahasa Indonesia ke bahasa Dayak Banjur dengan penerapan teknik *data augmentation* untuk menambah variasi data latih di dalamnya.

II. LANDASAN TEORI

A. *NMT*

Neural Machine Translation (NMT) adalah metode terjemahan mesin yang menggunakan jaringan saraf tiruan untuk menerjemahkan teks dari satu bahasa ke bahasa lain. Dalam NMT, ada dua komponen utama yaitu *encoder* dan *decoder*. *Encoder* mengubah input teks menjadi representasi vektor yang padat dengan memproses setiap kata dalam urutan secara berurutan, menghasilkan *hidden state* yang merangkum informasi penting dari input [10]. *Decode*r kemudian menggunakan representasi vektor ini untuk menghasilkan teks dalam bahasa target, satu kata pada satu waktu, sambil mempertimbangkan *context vector* yang dihasilkan oleh *encoder* [12][10]. Mekanisme *attention* sering digunakan untuk memungkinkan model fokus pada bagian spesifik dari *input* saat menghasilkan setiap kata *output*, meningkatkan akurasi terjemahan [9]. NMT unggul dalam menghasilkan terjemahan yang lebih alami dan sesuai konteks dibandingkan dengan metode sebelumnya, berkat pendekatan *end-to-end* yang memungkinkan pelatihan model secara menyeluruh dan efisien [5].

B. *Augmentasi Teks*

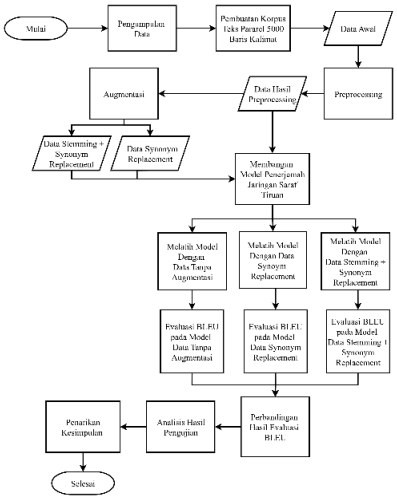
Augmentasi teks merupakan pembuatan data sintetis untuk meningkatkan keragaman data pelatihan, yang membantu meningkatkan kinerja model dan kemampuan generalisasi [14][15]. Dalam penelitian ini akan digunakan dua metode untuk augmentasi teks. Yang pertama adalah *Synonym replacement*, metode yang melibatkan penggantian kata-kata dalam teks dengan sinonimnya untuk menghasilkan data pelatihan baru [7]. Kemudian *stemming + synonym replacement*, metode ini secara teknis merupakan augmentasi *synonym replacement* yang sama dengan sebelumnya namun di langkah awal sebelum melakukan penggantian kata dengan sinonimnya dilakukan pengubahan semua kata dalam kalimat menjadi kata dasar.

C. *BLEU*

BLEU (*Bilingual Evaluation Understudy*) adalah metode evaluasi otomatis untuk sistem terjemahan mesin yang diperkenalkan oleh Papineni et al. pada tahun 2002 [6][13]. Metode ini mengukur kualitas terjemahan dengan membandingkan hasil terjemahan mesin terhadap satu atau lebih terjemahan referensi manusia menggunakan pendekatan berbasis *n-gram*. BLEU menghitung presisi *n-gram* antara hasil terjemahan dan referensi, lalu menggabungkannya menggunakan rata-rata geometrik, serta menerapkan penalti untuk terjemahan yang terlalu pendek melalui *brevity penalty*. Inovasi utama BLEU adalah kemampuannya mengevaluasi sistem secara efisien dan otomatis dalam skala besar, dengan korelasi yang cukup baik terhadap penilaian manusia. Sejak diperkenalkan, BLEU telah menjadi salah satu metrik standar dalam penelitian dan pengembangan sistem terjemahan mesin [6].

III. MOTODE PENELITIAN

Metode penelitian dimulai dengan tahap pengumpulan data sampai pada analisis hasil dan penarikan kesimpulan. Tahapan metode penelitian diilustrasikan pada Gambar. 1 sebagai berikut.



Gambar. 1 Metode Penelitian

A. *Pengumpulan Data*

Proses pengumpulan data dilakukan dengan menggabungkan beberapa metode. Pertama, menggunakan AI untuk menghasilkan kalimat dalam bahasa Indonesia. Selain itu, data juga dikumpulkan dari berbagai sumber daring, seperti cerita anak-anak yang terdapat di situs dongeng cerita rakyat serta kumpulan cerita lucu yang diperoleh melalui penelusuran di internet. Sebagian kalimat juga dibuat secara manual oleh peneliti untuk memperkaya variasi data. Setiap kalimat bahasa Indonesia yang dikumpulkan memiliki ketentuan panjang, yaitu minimal terdiri dari 10 kata dan maksimal 30 kata. Kalimat-kalimat tersebut kemudian diterjemahkan ke dalam bahasa Dayak Banjur oleh peneliti. Secara keseluruhan, data yang berhasil dikumpulkan terdiri dari 5.000 pasang kalimat untuk data awal sebelum augmentasi, 150 data awal untuk validasi dan 100 pasang kalimat untuk data uji.

B. *Pembuatan Korpus Teks Paralel*

Korpus adalah kumpulan data teks digital yang besar dan beragam, mencakup teks tertulis dan lisan [8]. Korpus paralel disusun secara manual dalam format CSV dengan dua kolom Bahasa Indonesia dan Bahasa Banjur. Korpus terdiri dari 5.000 pasang kalimat untuk data awal, 150 pasang untuk validasi, dan 100 pasang untuk pengujian. Adapun contoh korpus teks paralel dapat dilihat pada Tabel I.

TABEL I
CONTOH PASANGAN KALIMAT PARAREL

No	Bahasa Indonesia	Bahasa Banjur
1	Abangnya mungkin yang tadi kerumah Sarah, katanya dia mau menjemput Sarah tapi Sarah tidak ada dirumah dan sudah pergi bersama teman-temanya	Abangnya amang ti tadek kerumah Sarah, jakoke ya kak nyemput Sarah uleh Sarah ndai sek dirumah nyau angkat tamai kawan-kawane
2	Ada atau tidak ya Nata di rumah, kita harus segera selesaikan tugas kelompok ini dan semua bahan Nata yang bawa tadi	Adai atau ndai deh Nata di rumah, kitai alah segera nyelesaike tugas kelumpok tuk dan mhua bahan Nata ti ngemaike tadek
3	Sebenarnya, saudara-saudara saya sudah menikah, dan saya satu-satunya yang masih tinggal dengan orang tua saya.	Sebenare, menyadik-menyadikku udah nikah, dan ku sikuk-sikuke ti agik idup tamai apai inai ku

C. *Preprocessing*

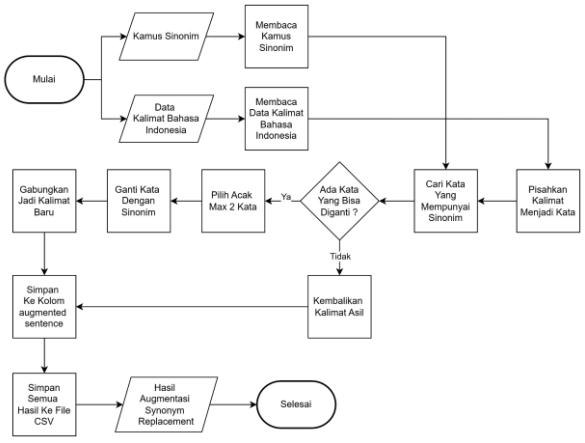
Data awal korpus paralel masih mengandung angka, tanda baca, dan huruf kapital sehingga perlu tahap *preprocessing* untuk pembersihan dan normalisasi teks. Proses ini meliputi konversi tipe data ke *string*, *case folding* ke huruf kecil, penghapusan angka, penggantian tanda hubung dengan spasi, penghapusan tanda baca, dan normalisasi spasi. Hasil *preprocessing* menghasilkan data yang lebih bersih dan siap untuk tahap augmentasi serta pelatihan model.

D. *Penerapan Augmentasi*

Augmentasi yang dilakukan untuk menambah variasi data menggunakan teknik *synonym replacement*, *stemming + synonym replacement*. Sebanyak 5.000 kalimat bahasa Indonesia dari data awal di augmentasi sehingga menghasilkan 5.000 kalimat baru.

1) *Synonym Replacement*

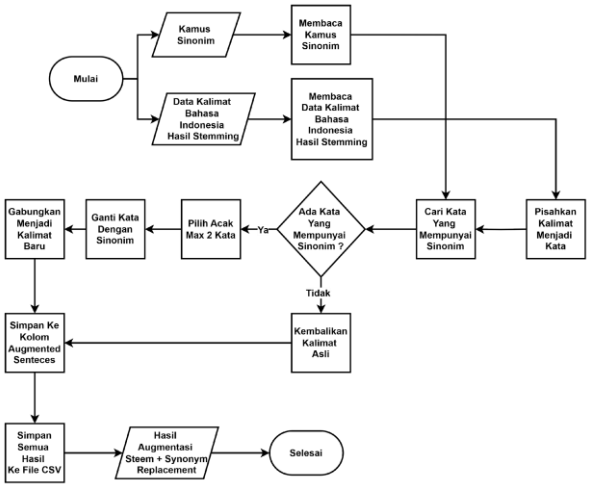
Alur *synonym replacement* diawali dengan membaca kamus sinonim dan data kalimat Bahasa Indonesia. Setiap kalimat dipecah menjadi kata, lalu dicari kata yang memiliki sinonim. Maksimal dua kata dipilih secara acak untuk diganti, kemudian disusun kembali menjadi kalimat baru. Jika tidak ada kata yang cocok, kalimat asli digunakan. Hasil augmentasi disimpan pada kolom *augmented sentence* dan diekspor ke file CSV. Adapun untuk skenario dari *synonym replacement* dapat dilihat pada Gambar 2.



Gambar 2. Skenario *Synonym Replacement*

2) *Stemming + Synonym Replacement*

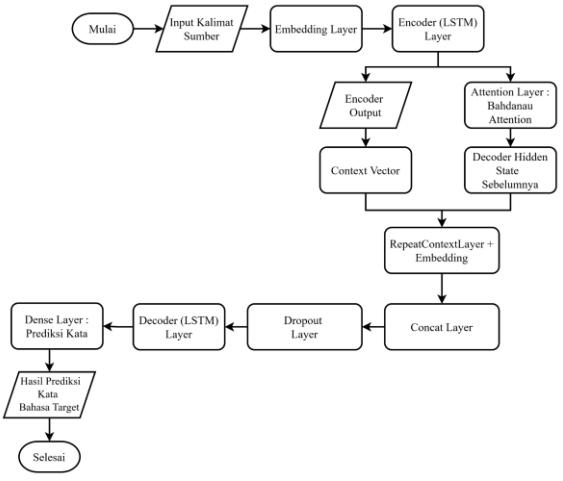
Pada tahap *stemming + synonym replacement*, proses diawali dengan *stemming* menggunakan *library* Sastrawi untuk mengubah kata berimbuhan menjadi bentuk dasar. Data teks bahasa Indonesia dibaca menggunakan pandas, kemudian setiap kalimat di-*stemming* dan hasilnya disimpan dalam kolom baru, Kolom tersebut kemudian diekspor ke file csv. Proses ini bertujuan untuk menyederhanakan kata sebelum tahap penggantian kata dengan sinonimnya. Langkah setelah melakukan *stemming* dan setelah didapatkan data hasil *stemming* dalam format csv dilanjutkan dengan langkah-langkah yang sama pada proses augmentasi *synonym replacement* pada pembahasan di poin 1. Adapun untuk skenario dari *stemmig + synonym replacement* dapat dilihat pada Gambar 3.



Gambar 3. Skenario *Stemming + Syonym Replacement*

E. Membangun Model NMT

Penelitian ini menggunakan platform Kaggle untuk mengimplementasikan kode program dalam membangun model *Neural Machine Translation* (NMT). Arsitektur yang digunakan berbasis *Encoder-Decoder* dengan Bahdanau *Attention* dan LSTM. Proses dimulai dari tokenisasi kalimat sumber yang kemudian diubah menjadi vektor melalui *Embedding Layer* dan diproses oleh *Encoder* LSTM untuk menghasilkan *hidden states*. Bahdanau *Attention* digunakan untuk menghitung *context vector* dengan membandingkan *hidden state* dari decoder dengan seluruh *output* dari *encoder*. *Context vector* tersebut digabungkan dengan *embedding* dari *input decoder*, lalu diproses oleh *decoder* LSTM. Hasilnya diteruskan ke *Dropout* dan *TimeDistributed Dense* layer dengan aktivasi *softmax* untuk menghasilkan prediksi kata hingga token <endseq> tercapai. Adapun arsitektur model NMT dalam penelitian ini dapat dilihat pada Gambar 7.



Gambar 4. Arsitektur Model NMT

Penjelasan dari tahapan-tahapan arsitektur model NMT pada Gambar 7 adalah sebagai berikut:

1) *Input* Kalimat Sumber: Kalimat sumber yang adalah Bahasa Indonesia digunakan sebagai input untuk terjemahan.
2) *Embedding Layer*: Kalimat sumber yang sudah di-tokenisasi diubah menjadi vektor representasi berdimensi tetap.
3) *Encoder* (LSTM) *Layer*: *Embedding* vektor dari kalimat sumber diproses oleh LSTM *encoder*. Hasilnya berupa urutan *hidden states* yang menyimpan informasi dari seluruh kalimat [9].
4) *Encoder Output*: *Hidden states* dari LSTM *encoder* disimpan sebagai *encoder output*, yang digunakan dalam perhitungan *attention*.
5) *Attention Layer*: Bahdanau *Attention* menghitung *context vector* dengan membandingkan *hidden state* sebelumnya dari *decoder* dengan seluruh *encoder output*. Tujuannya adalah untuk menentukan bagian kalimat sumber mana yang paling relevan untuk menghasilkan kata target berikutnya [3].
6) *Decoder Hidden State* Sebelumnya: Digunakan sebagai *input* dalam mekanisme *attention*, untuk menentukan konteks yang relevan pada setiap langkah *decoding*.
7) *Context Vector*: Hasil dari *attention*, berupa ringkasan terfokus dari kalimat sumber yang relevan dengan kata target yang sedang dihasilkan.
8) RepeatContextLayer + *Embedding*: *Context vector* direplikasi sebanyak panjang *input decoder* dan digabungkan dengan *embedding* dari token target sebelumnya.
9) *Concat Layer*: *Layer* ini menggabungkan *embedding input* target dan *context vector* agar bisa diproses bersama oleh *decoder*.
10) *Decoder* (LSTM) *Layer*: Gabungan dari *context* dan *embedding* diproses oleh LSTM *decoder* untuk menghasilkan *hidden state* baru.
11) *Dropout Layer*: Digunakan sebagai teknik regularisasi untuk menghindari *overfitting*, dengan cara mengabaikan sebagian neuron selama pelatihan.
12) *Dense Layer*: *Output* dari *decoder* diproses oleh *Dense layer* dengan aktivasi *softmax* untuk menghasilkan probabilitas setiap kata dalam kosakata target.
13) Hasil Prediksi Kata Bahasa Target: Kata dengan probabilitas tertinggi dipilih sebagai prediksi *output*, yaitu kata dalam bahasa target (Dayak Banjur).

F. Training Model

Setelah proses pembangunan model selesai, dilakukan pelatihan secara terpisah pada tiga model berbeda, yang masing-masing menggunakan dataset yang berbeda pula. tiga dataset tersebut terdiri dari: data tanpa augmentasi, data dengan teknik *synonym replacement*, dan data dengan teknik *stemming + synonym replacement*. Dalam penelitian ini, peneliti menerapkan rasio pembagian data sebesar 80:20 untuk data latih dan data validasi.

Untuk model tanpa augmentasi, data awal yang terdiri atas 5000 pasang kalimat ditambahkan dengan 150 pasang kalimat dari data evaluasi awal, sehingga total menjadi 5150 pasang kalimat. Dari jumlah tersebut, sebanyak 4120 pasang kalimat (80%) digunakan sebagai data latih, sedangkan 1030 pasang kalimat sisanya (20%) digunakan sebagai data validasi.

Sementara itu, untuk dua model lainnya yang menggunakan data hasil augmentasi, masing-masing dataset terdiri dari 10.000 pasang kalimat hasil augmentasi, kemudian ditambahkan 150 pasang kalimat dari data evaluasi awal. Dengan demikian, jumlah total menjadi 10.150 pasang kalimat. Dari jumlah ini, sebanyak 8120 pasang kalimat (80%) dialokasikan sebagai data latih, dan sisanya sebanyak 2030 pasang kalimat (20%) digunakan sebagai data validasi.

Model yang telah dirancang sesuai dengan arsitektur pada Gambar 7 kemudian direplikasi menjadi lima model. Setiap model tersebut dilatih secara terpisah menggunakan masing-masing lima dataset berbeda agar dapat dilakukan analisis perbandingan kinerja model terhadap berbagai teknik augmentasi data yang diterapkan.

G. Evaluasi Model Dengan BLEU

Evaluasi dilakukan untuk memperoleh nilai akurasi hasil terjemahan dari model *Neural Machine Translation* (NMT) setelah dilatih menggunakan data tanpa augmentasi maupun data hasil lima teknik augmentasi yang diterapkan. Nilai akurasi ini diperoleh melalui pengukuran menggunakan skor BLEU (*Bilingual Evaluation Understudy*) sebagai indikator kinerja model.

H. Penarikan Kesimpulan

Kesimpulan dalam penelitian ini akan ditarik berdasarkan hasil pengujian yang diperoleh, guna menilai sejauh mana penelitian mampu menghasilkan analisis perbandingan yang relevan terhadap permasalahan yang telah diidentifikasi.

IV. HASIL DAN PEMBAHASAN

Data awal yang digunakan untuk proses augmentasi terdiri dari 5000 baris kalimat pada kolom Bahasa Indonesia saja, sedangkan kolom Bahasa Dayak Banjur tetap dipertahankan tanpa perubahan.

1) Hasil Augmentasi

Contoh hasil dari masing-masing teknik augmentasi *synonym replacement* dan *stemming + synonym replacement* dapat dilihat pada Tabel II dan Tabel III.

TABEL II

CONTOH HASIL AUGMENTASI *SYNONYM REPLACEMENT*

No	Sebelum *Synonym Replacement*	Sesudah *Synonym Replacement*
1	abangnya mungkin yang tadi kerumah sarah katanya dia mau menjemput sarah tapi sarah tidak ada dirumah dan sudah pergi bersama teman temanya	abangnya mungkin yang mulanya kerumah sarah katanya beliau mau menjemput sarah tapi sarah tidak ada dirumah dan sudah pergi bersama teman temanya
2	ada atau tidak ya nata di rumah kita harus segera selesaikan tugas kelompok ini dan semua bahan nata yang bawa tadi	ada atau tidak ya nata di rumah kita harus segera selesaikan darma kelompok ini bersama semua bahan nata yang bawa tadi

TABEL III

CONTOH HASIL AUGMENTASI *STEMMING + SYNONYM REPLACEMENT*

No	Sebelum Augmentasi stemming + *Synonym Replacement*	Setelah Augmentasi stemming + *Synonym Replacement*
1	abangnya mungkin yang tadi kerumah sarah katanya dia mau menjemput sarah tapi sarah tidak ada dirumah dan sudah pergi bersama teman temanya	abang mungkin yang mulanya rumah sarah kata dia akan jemput sarah tapi sarah tidak ada rumah dan sudah pergi sama teman teman
2	ada atau tidak ya nata di rumah kita harus segera selesaikan tugas kelompok ini dan semua bahan nata yang bawa tadi	ada atau tidak ya nata di bait kita harus segera selesai tugas kelompok ini bersama semua bahan nata yang bawa tadi

2) Hasil Evaluasi

Data hasil augmentasi digunakan untuk melatih model yang telah direplikasi, kemudian dilakukan evaluasi pada setiap model yang telah dilatih dengan 3 data berbeda. Hasil Evaluasi BLEU pada setiap model dapat dilihat pada Tabel IV.

TABEL IV
HASIL EVALUASI BLEU

Model	**BLEU Score**
Tanpa Augmentasi	42,17 %
Synonym Replacement	48,19%
Stemming + Synonym Replacement	46%

Berdasarkan hasil evaluasi BLEU pada Tabel IV, baik *synonym replacement* maupun *stemming + synonym replacement* mampu meningkatkan performa model dibandingkan tanpa augmentasi. Teknik *synonym replacement* menghasilkan BLEU score tertinggi sebesar 48,19%, sedangkan kombinasi *stemming + synonym replacement* menghasilkan skor sedikit lebih rendah, yaitu 46%. Perbedaan ini menunjukkan bahwa penambahan proses *stemming* sebelum penggantian sinonim tidak selalu memberikan peningkatan yang lebih besar. Hal ini kemungkinan disebabkan oleh hilangnya nuansa kata akibat proses *stemming*, sehingga konteks kalimat menjadi kurang tepat saat dilakukan penggantian sinonim. Dengan demikian, penggunaan *synonym replacement* tanpa *stemming* dinilai lebih efektif dalam mempertahankan makna dan meningkatkan kualitas terjemahan.

V. KESIMPULAN

Dari dua teknik augmentasi yang diuji, synonym replacement menghasilkan BLEU score tertinggi sebesar 48,19%, sedangkan *stemming + synonym replacement* sedikit lebih rendah, yaitu 46%. Hasil ini menunjukkan bahwa *synonym replacement* lebih efektif dalam meningkatkan performa model. Penambahan proses *stemming* sebelum penggantian sinonim tidak memberikan keuntungan tambahan, bahkan dapat mengurangi kualitas terjemahan karena stemming berpotensi menghilangkan bentuk kata yang sesuai konteks, sehingga pemilihan sinonim menjadi kurang tepat. Dengan demikian, augmentasi menggunakan *synonym replacement* tanpa stemming lebih disarankan karena mampu mempertahankan makna dan struktur kalimat secara lebih baik.

REFERENSI

[1] Mailani, O., Nuraeni, I., Syakila, S. A., & Lazuardi, J. (2022). Bahasa sebagai alat komunikasi dalam kehidupan manusia. *Kampret Journal*, *1*(2), 1-10.

[2] Badan Pengembangan dan Pembinaan Bahasa. (n.d.). *Data Bahasa – Peta Bahasa*. Kementerian Pendidikan dan Kebudayaan Republik Indonesia. https://petabahasa.kemdikbud.go.id/databahasa.php

[3] Bahdanau, D. (2014). Neural machine translation by jointly learning to align and translate. *arXiv preprint arXiv:1409.0473*.

[4] Shorten, C., Khoshgoftaar, T. M., & Furht, B. (2021). Text data augmentation for deep learning. *Journal of big Data*, *8*(1), 101.

[5] Wang, H., Wu, H., He, Z., Huang, L., & Church, K. W. (2022). Progress in machine translation. Engineering, 18, 143–153. https://doi.org/10.1016/j.eng.2021.03.023

[6] Papineni, K., Roukos, S., Ward, T., & Zhu, W. J. (2002). *BLEU: A method for automatic evaluation of machine translation*. In *Proceedings of the 40th Annual Meeting of the*

Association for Computational Linguistics (pp. 311–318). Association for Computational Linguistics. https://aclanthology.org/P02-1040

[7] Wei, J., & Zou, K. (2019). EDA: Easy data augmentation techniques for boosting performance on text classification tasks. *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing*, 6382–6388. https://doi.org/10.18653/v1/D19-1670

[8] Almos, R., Pramono, P., Seswita, S., Asma, R. A., & Putri, N. O. (2023). Linguistik Korpus: Sarana dan Media Pembelajaran pada Mata Kuliah Leksikologi dan Leksikografi di Perguruan Tinggi. Lectura: Jurnal Pendidikan, 14(1), 45-59.

[9] Chauhan, M. (2021, January 30). A simple overview of RNN, LSTM and Attention Mechanism. Retrieved from https://medium.com/swlh/a-simple-overview-of-rnn-lstm-and-attention-mechanism-9e844763d07b

[10] GeeksforGeeks. (2024, January 10). Self-attention in NLP. Retrieved from https://www.geeksforgeeks.org/self-attention-in-nlp/

[11] GeeksforGeeks. (2020, October 25). Understanding of OpenSeq2Seq. Retrieved from https://www.geeksforgeeks.org/understanding-of-openseq2seq/

[12] Prijono, B. (2018, April 4). Pengenalan Recurrent Neural Network (RNN) – Bagian 1. Retrieved from https://indoml.com/2018/04/04/pengenalan-rnn-bag-1/

[13] Nagar, P. (2024, October 6). BLEU: A method for automatic evaluation of machine translation. Retrieved from https://medium.com/understanding-research-papers/bleu-a-method-for-automatic-evaluation-of-machine-translation-bcf0f6d3a881

[14] Jin, C., Qiu, S., Xiao, N., & Jia, H. (2022). AdMix: A mixed sample data augmentation method for neural machine translation. *arXiv preprint arXiv:2205.04686*.

[15] Wang, X., Pham, H., Dai, Z., & Neubig, G. (2018). SwitchOut: an efficient data augmentation algorithm for neural machine translation. *arXiv preprint arXiv:1808.07512*.