Multi-Sentence Contextual Modeling using Transformer Architecture for Arithmetic Operation Identification in Mathematical Word Problems

Cikal Dwidyanto [1)*)], Agung Prasetya [2)], Mohamad Khoirul Ansor [3)]

[1)2)3)]Program Studi Informatika, Fakultas Sains dan Teknologi, Universitas Bhinneka PGRI
[*)]**Correspondence author**: cikal.student@ubhi.ac.id, Tulungagung, Indonesia
DOI: https://doi.org/10.37012/jtik.v12i1.3431

Abstract

Mathematical word problems in Indonesian are generally presented as multi-sentence paragraphs, making the identification of arithmetic operations not solely dependent on recognizing numerical values, but also requiring an understanding of events and semantic relationships across sentences. This study formulates the task of arithmetic operation identification as a classification problem of a single primary operation into four classes: addition, subtraction, multiplication, and division. To capture contextual relationships across sentences, an encoder-based Transformer architecture is employed, which is capable of modeling long-range dependencies through a self-attention mechanism. The dataset consists of 900 elementary school-level mathematical word problems constructed in accordance with the Indonesian curriculum. Experimental results show that the model achieves an accuracy of 0.98 and an F1-score of 0.98. Per-class evaluation indicates high and consistent performance, although prediction errors are still observed in cases with ambiguous narrative patterns, particularly where addition is misclassified as multiplication or subtraction, and multiplication is misclassified as division. These findings demonstrate that the Transformer architecture is effective in leveraging multi-sentence context to improve the accuracy of arithmetic operation identification in mathematical word problems.

Keywords: *Transformer, BERT, NLP, Contextual Representation, Math Word Problems*

Abstrak

Soal cerita matematika berbahasa Indonesia umumnya disajikan dalam bentuk paragraf multi-kalimat, sehingga penentuan operasi aritmetika tidak cukup hanya dengan mengenali angka, tetapi juga memerlukan pemahaman terhadap peristiwa serta hubungan semantik antar kalimat. Penelitian ini memformulasikan tugas identifikasi operasi aritmetika sebagai masalah klasifikasi satu operasi utama ke dalam empat kelas, yaitu penjumlahan, pengurangan, perkalian, dan pembagian. Untuk menangkap relasi konteks antar kalimat, digunakan arsitektur *Transformer* berbasis *encoder* yang mampu memodelkan dependensi panjang melalui mekanisme *self-attention*. Dataset yang digunakan terdiri dari 900 soal cerita matematika tingkat sekolah dasar yang disusun sesuai dengan kurikulum di Indonesia. Hasil pengujian menunjukkan bahwa model mencapai akurasi sebesar 0,98 dan *F1-score* sebesar 0,98. Evaluasi per kelas menunjukkan performa yang tinggi dan konsisten, meskipun kesalahan prediksi masih ditemukan pada kasus dengan pola narasi yang ambigu, khususnya pada penjumlahan yang terklasifikasi sebagai perkalian atau pengurangan, serta perkalian yang terklasifikasi sebagai pembagian. Hasil ini menunjukkan bahwa arsitektur *Transformer* efektif dalam memanfaatkan konteks multi-kalimat untuk meningkatkan ketepatan identifikasi operasi aritmetika pada soal cerita matematika.

Kata Kunci: *Transformer, BERT, NLP,* Representasi Kontekstual*,* Soal Cerita Matematika

PENDAHULUAN

Soal cerita matematika penting untuk mengukur sejauh mana siswa mampu menerjemahkan masalah secara matematis. Salah satu kesulitan dalam pengerjaan soal matematika adalah menafsirkan jenis operasi dari makna cerita. Jenis operasi tidak selalu ditunjukkan secara eksplisit oleh kata-kata kunci tertentu. Bahkan, frase tertentu dalam soal cerita menunjuk lebih dari satu jenis operasi yang berbeda (Agusfianuddin et al., 2024; Hickendorff, 2021; Sarjana et al., 2020).

Sejumlah pendekatan telah dikembangkan untuk mengidentifikasi jenis operasi aritmetika. Pada (Mandal & Naskar, 2021), sejumlah kata kunci, berupa kata dan frase, secara literal dipetakan ke jenis operasi menggunakan sejumlah aturan. Karena adanya ambiguitas dan variasi bahasa, pendekatan-pendekatan berbasis neural memanfaatkan hasil pengkodean semantik kata (Liang et al., 2022). Sedangkan pada (Liang et al., 2023), masalah *out-of-vocabulary* diselesaikan dengan memanfaatkan *large language* model.

Namun demikian, pendekatan-pendekatan usulan tersebut belum mempertimbangkan konteks keseluruhan soal cerita. Konteks dalam soal cerita terjadi karena adanya hubungan makna antar kalimat di bagian narasi dan pertanyaan. Jenis operasi seringkali didapatkan dengan menghubungkan makna kalimat dari awal akhir. Masalah ini dapat menyebabkan kesalahan interpretasi sehingga akurasi menurun (W. Liu et al., 2025; Patel et al., 2021).

Untuk mengatasi permasalahan tersebut, penelitian ini mengusulkan *Transformer* (Zhang et al., 2019) dengan unit pemrosesan berupa kalimat, bukan kata. Setiap kalimat dalam soal cerita direpresentasikan sebagai satu vektor semantik kemudian dimodelkan dengan *self-attention* (Y. Liu & Lapata, 2019). Pendekatan usulan ini memungkinkan model menangkap hubungan makna antar kalimat dalam satu soal cerita, termasuk keterkaitan antara bagian narasi dan kalimat pertanyaan. Dengan memanfaatkan representasi tingkat kalimat, konteks soal cerita secara global dapat dipahami secara lebih utuh. Penentuan jenis operasi aritmetika tidak bergantung pada kata kunci lokal tetapi pada alur semantik dari awal hingga akhir soal. Pendekatan ini diharapkan mampu mengurangi ambiguitas interpretasi dan meningkatkan akurasi identifikasi jenis operasi.

Penelitian ini berkontribusi dengan mengajukan pemodelan kontekstual multi-kalimat berbasis *Transformer* (Z. Wang et al., 2022) untuk identifikasi jenis operasi aritmetika. Pendekatan yang diusulkan diterapkan pada soal cerita matematika berbahasa Indonesia. Kinerja model kemudian dievaluasi dan dibandingkan dengan pendekatan baseline.

METODE PENELITIAN

Penelitian ini mengusulkan arsitektur *Transformer* berbasis *encoder (BERT)* untuk mengklasifikasi operasi aritmetika pada soal cerita matematika berbahasa Indonesia. Model *BERT* memanfaatkan mekanisme *attention* untuk memetakan hubungan konteks antar kalimat [12]. Model memprediksi urutan informasi dari soal tersebut ke dalam empat kelas operasi aritmetika, yaitu penjumlahan, pengurangan, perkalian, dan pembagian. Dengan pemetaan kelas:

$$\{0,1,2,3\} = \{[OP\_ADD], [OP\_SUB], [OP\_MUL], [OP\_DIV]\}$$

Setiap data terdiri dari pasangan x, y, Dimana x adalah teks soal cerita dan y adalah label operasi [4].

1. Sumber Data

Pada penelitian ini data diperoleh dari kumpulan soal cerita matematika berbahasa Indonesia tingkat sekolah dasar kelas 1-3 yang diperoleh dari buku pembelajaran digital, Data yang dikumpulkan sebanyak 900 soal cerita.

Tabel 1. Kategori Soal Cerita

Kategori		Label	Jumlah
Level 1	**Level 2**	**Operasi**	**Data**
Change	Result Unknwon (Plus)	OP_ADD	87
Change	Result Unknwon (Minus)	OP_SUB	191
Combine	Combine (Plus)	OP_ADD	130
Combine	Combine (Minus)	OP_SUB	13
Compare	Compare (Plus)	OP_ADD	3
Compare	Compare (Minus)	OP_SUB	29
Div-Mul	Multiplication	OP_MUL	225
Div-Mul	Division	OP_DIV	222

Perhatikan **Tabel 2**, Setiap soal memiliki satu label operasi. Dikelompokkan menjadi empat kelas, yaitu OP_ADD, OP_SUB, OP_MUL, OP_DIV. Data pelatihan berjumlah

600 soal, data validasi berjumlah 100 soal, dan data pengujian berjumlah 200 soal. Data pelatihan dan validasi memiliki distribusi operasi yang seimbang agar proses pelatihan menjadi lebih maksimal dan mengurangi bias pada data.

Tabel 2. Komposisi Data Train dan Validasi Tiap Kelas

Label Operasi	Jumlah Soal		
	Data Train	Data Validasi	Data Uji
OP_ADD	150	25	45
OP_SUB	150	25	58
OP_MUL	150	25	50
OP_DIV	150	25	47

2. Pra-Pemrosesan Data

Pada tahap ini, dataset disiapkan dalam format JSON, teks dinormalisasikan dengan merapikan karakter, spasi, dan tanda baca agar bentuk kalimat lebih stabil saat diproses. Setelah normalisasi awal, teks soal cerita dipisahkan menjadi urutan kalimat menggunakan *Stanza* untuk *sentence tokenizer* berbahasa Indonesia (Qi et al., 2020). Selanjutnya, angka pada soal dinormalisasikan menggunakan *numerical placeholder* untuk menekan bias model terhadap nilai numerik tertentu agar memfokuskan ke relasi semantik antar peristiwa, langkah ini didasarkan karena model *Natural Language Processing (NLP)* sering kali tidak konsisten dalam penalaran numerik pada soal cerita sederhana (W. Liu et al., 2025; Patel et al., 2021).

3. Embedding dan Arsitektur Model

Proses embedding dilakukan dengan mengubah setiap kalimat menjadi vektor semantik menggunakan *SBERT* (Reimers & Gurevych, 2019). Hasilnya berupa urutan vektor kalimat:

$$H = [h_1 h_2, \dots h_2] \in \mathbb{R}^{n \times d}. \quad (1)$$

Urutan vektor H ditambahkan *positional encoding* agar *Transformer* tetap mengenali informasi urutan kalimat:

$$X = H + PE, \quad (2)$$

dengan $PE \in \mathbb{R}^{n \times d}$. *Positional encoding* didefinisikan sebagai (Irani & Metsis, 2025; Vaswani et al., 2017):

$$PE(pos, 2i) = sin\left(\frac{pos}{10000^{2i/d}}\right), \quad (3)$$

$$PE(pos, 2i+1) = cos\left(\frac{pos}{10000^{2i/d}}\right). \quad (4)$$

Selanjutnya X diproses oleh *encoder Transformer* melalui mekanisme *self-attention* untuk menangkap hubungan antar kalimat (Faldu et al., 2022; Irani & Metsis, 2025; Vaswani et al., 2017; Xiong et al., 2020). Pada *encoder* dibentuk proyeksi *query, key,* dan *value* sebagai:

$$Q = XW^Q, \qquad K = XW^K, \qquad V = XW^V. \quad (5)$$

Self-attention dihitung menggunakan rumus *scaled dot-product attention* (Irani & Metsis, 2025; Jin et al., 2025; L. Liu et al., 2020; Vaswani et al., 2017; H. Wang et al., 2024; Xiong et al., 2020):

$$Attention\,(Q, K, V) = softmax\left(\frac{QK^T}{\sqrt{d_k}}\right)V. \quad (6)$$

Pada *multi-head attention*, perhatian dihitung pada beberapa *head* lalu digabungkan (Irani & Metsis, 2025; Jin et al., 2025; Vaswani et al., 2017; Xiong et al., 2020):

$$head_j = Attention\left(QW_j^Q, KW_j^K, VW_j^V\right), \quad (7)$$

$$MHA\,(X) = Concat(head_1, \ldots, head_m)\,W^O. \quad (8)$$

Keluaran setiap layer kemudian diproses oleh *feed-forward network (FFN)* (Vaswani et al., 2017; Xiong et al., 2020):

$$FFN(x) = \alpha(xW_1 + b_1)W_2 + b_2, \quad (9)$$

Dengan $\alpha(.)$ sebagai fungsi aktivasi (Vaswani et al., 2017). Distabilkan dengan *residual connection, layer normalization,* dan *dropout* untuk menjaga proses pelatihan tetap stabil dan mencegah *overfitting* (Ba et al., 2016; L. Liu et al., 2020; Vaswani et al., 2017; H. Wang et al., 2024; Xiong et al., 2020):

$$Z = LN\left(X + Dropout\left(MHA(X)\right)\right), \quad (10)$$

$$X' = LN\left(Z + Dropout\left(FFN(Z)\right)\right), \quad (11)$$

Self-attention memperhitungkan semua pasangan kalimat, informasi penting dapat tetap saling berinteraksi meskipun petunjuk operasi dan kuantitas muncul berjauhan. Representasi paragraf diperoleh dari keluaran token $[CLS]$ pada layer terakhir *encoder*

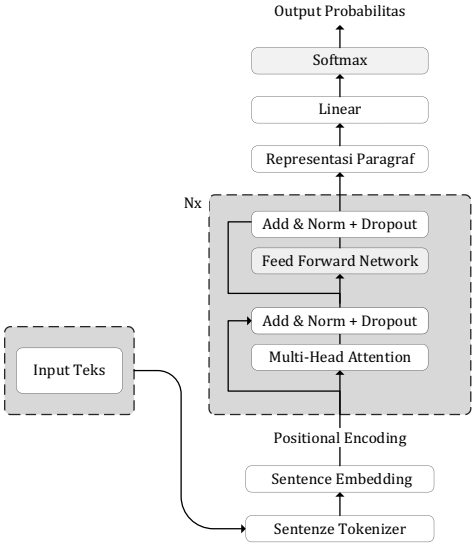
sebagai vektor $p \in \mathbb{R}^d$ (Devlin et al., 2019). Vektor paragraf (p) selanjutnya dipetakan oleh lapisan *linear* dan *softmax* (Jin et al., 2025; Xiong et al., 2020), untuk menghasilkan probabilitas empat kelas operasi (OP_ADD, OP_SUB, OP_MUL, OP_DIV), dengan:

$$z = Wp + b, \tag{12}$$

$$\hat{y} = softmax(z). \tag{13}$$

Fungsi kerugian menggunakan *cross-entropy* untuk klasifikasi multikelas (Grandini et al., 2020):

$$L = -\sum_{c=1}^{4} y_c \log(\hat{y}_c)\,. \tag{14}$$



Gambar 1. Arsitektur Model *Encorder Transformer*

4. Dataset dan Pelatihan

Dataset terdiri dari 900 soal cerita matematika kelas 1-3 sekolah dasar yang bersumber dari buku digital sesuai kurikulum di Indonesia. Data dibagi menjadi 600 (70%) data pelatihan, 100 (10%) data validasi, dan 200 (20%) data pengujian (Ayumi et al., 2025). Dilakukan proses augmentasi pada data pelatihan untuk memperkaya keragaman data tanpa mengubah makna matematis, *augmentasi* dilakukan melalui parafrase narasi soal, termasuk variasi kuantitatif maupun nilai bilangan, dengan tetap menjaga hubungan matematis tetap sama, serta melalui *expanding equation trees* untuk memperpanjang

langkah penyelesaian melalui penambahan peristiwa (Kim et al., 2023). Pengukuran kemiripan skor *BLEU* digunakan untuk mencegah duplikasi skor, dengan:

$$BLEU = BP \times \exp\left(\sum_{n=1}^{N} w_n log p_n\right). \quad (15)$$

Dimana p_n adalah *modified n-gram precision*, w_n adalah bobot, dan BP adalah *brevity penalty* (Papineni et al., 2002). Perhitungan *BLEU* skor memiliki statistik nilai mean 0,24 dan median 0,23, dengan rentang nilai antara 0 hingga 0,92 (Kim et al., 2023).

Dataset dilengkapi dengan anotasi dalam bentuk *expression tree postfix* menggunakan operator (OP_ADD, OP_SUB, OP_MUL, OP_DIV), dan pembentukan python *code* secara otomatis dari *postfix* tersebut. Proses anotasi menerapkan pendekatan *human-in-the-loop* dengan memastikan validitas anotasi melalui perbandingan hasil eksekusi dengan kunci jawaban (Kim et al., 2023).

Implementasi dilakukan menggunakan bahasa pemrograman python dengan dukungan kerangka kerja PyTorch dan HuggingFace *Transformer*. Optimisasi model menggunakan AdamW dengan *learning rate* (Loshchilov & Hutter, 2019). Menggunakan *early stopping* dan regulasi *dropout* untuk mengurangi *overfitting* (Srivastava et al., 2014). Pelatihan model dibatasi maksimal 30 epochs untuk mencapai konvergensi yang optimal.

5. Evaluasi Model

Evaluasi dilakukan pada data uji menggunakan metrik *Accuracy, Precision, Recall, F1-Score* (Chicco & Jurman, 2020; Grandini et al., 2020). Akurasi mengukur proporsi seluruh prediksi secara tepat terhadap total data pengujian:

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN}, \quad (16)$$

$$Precision = \frac{TP}{TP + FP}, \quad (17)$$

$$Recall = \frac{TP}{TP + FN}, \quad (18)$$

$$FI - Score = \frac{2PR}{P + R}. \quad (19)$$

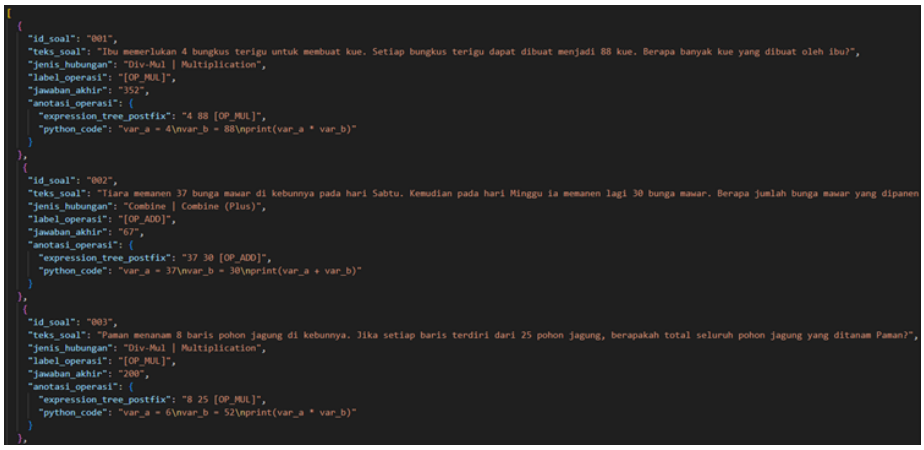
Nilai macro-average dihitung sebagai rata-rata metrik antar kelas agar evaluasi tetap seimbang pada klasifikasi multikelas. Selain itu, untuk melihat pola salah klasifikasi antar operasi dengan menggunakan *confusion matrix* (Opitz, 2024).

HASIL DAN PEMBAHASAN

Hasil Penelitian

1. Struktur Dataset

Dataset disimpan dalam format JSON untuk mempermudah pengelolaan dan pemanggilan data pada proses pelatihan berlangsung. Setiap entri memuat identitas soal, teks soal, label operasi, serta informasi pendukung.



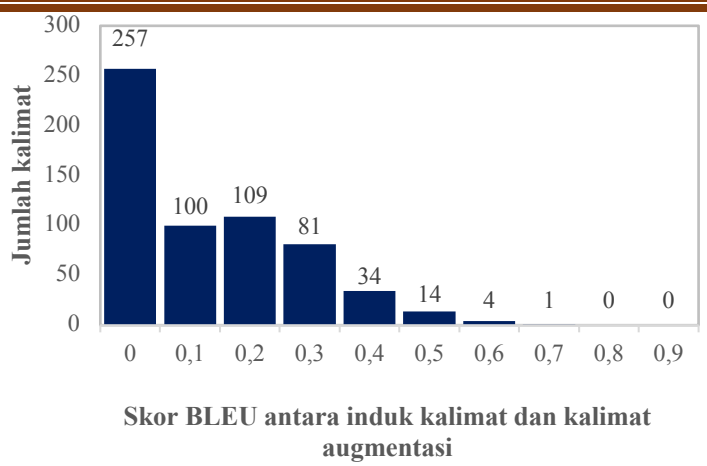
```
[
  {
    "id_soal": "001",
    "teks_soal": "Ibu memerlukan 4 bungkus terigu untuk membuat kue. Setiap bungkus terigu dapat dibuat menjadi 88 kue. Berapa banyak kue yang dibuat oleh ibu?",
    "jenis_hubungan": "Div-Mul | Multiplication",
    "label_operasi": "[OP_MUL]",
    "jawaban_akhir": "352",
    "anotasi_operasi": {
      "expression_tree_postfix": "4 88 [OP_MUL]",
      "python_code": "var_a = 4\nvar_b = 88\nprint(var_a * var_b)"
    }
  },
  {
    "id_soal": "002",
    "teks_soal": "Tiara memanen 37 bunga mawar di kebunnya pada hari Sabtu. Kemudian pada hari Minggu ia memanen lagi 30 bunga mawar. Berapa jumlah bunga mawar yang dipanen
    "jenis_hubungan": "Combine | Combine (Plus)",
    "label_operasi": "[OP_ADD]",
    "jawaban_akhir": "67",
    "anotasi_operasi": {
      "expression_tree_postfix": "37 30 [OP_ADD]",
      "python_code": "var_a = 37\nvar_b = 30\nprint(var_a + var_b)"
    }
  },
  {
    "id_soal": "003",
    "teks_soal": "Paman menanam 8 baris pohon jagung di kebunnya. Jika setiap baris terdiri dari 25 pohon jagung, berapakah total seluruh pohon jagung yang ditanam Paman?",
    "jenis_hubungan": "Div-Mul | Multiplication",
    "label_operasi": "[OP_MUL]",
    "jawaban_akhir": "200",
    "anotasi_operasi": {
      "expression_tree_postfix": "8 25 [OP_MUL]",
      "python_code": "var_a = 6\nvar_b = 52\nprint(var_a * var_b)"
    }
  },
```


Gambar 2. Struktur Dataset

Gambar 2, merupakan sample dataset soal, struktur tersebut dapat memastikan data konsisten dan mudah diproses pada tahap pra pemrosesan maupun pelatihan model.

2. Analisis Augmentasi Soal dengan BLEU

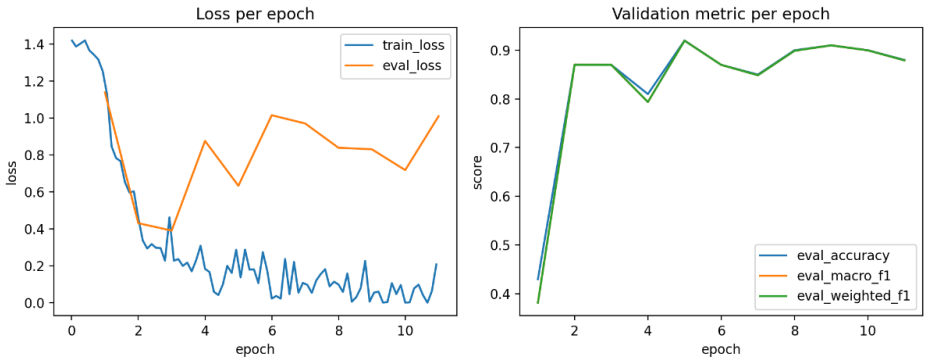
Setelah melalui proses *augmentasi*, kemudian dilakukan pengukuran tingkat kemiripan teks dilakukan dengan menggunakan *BLEU*.



Gambar 3. Grafik Kemiripan Kalimat Asli dan *Augmentasi*

Hasil pengukuran menunjukkan sebagian besar skor *BLEU* berada pada rentang 0.0 – 0.7. Kondisi tersebut menggambarkan bahwa hasil augmentasi menghasilkan variasi kalimat yang memadai dan tidak mengarah pada pengulangan teks secara berlebihan.

3. Pelatihan Model



Gambar 4. Kurva *Loss* dan Metrik Validasi

Hasil pelatihan menunjukkan bahwa *training loss* menurun secara tajam pada *epoch* awal hingga mendekati nol, menandakan model mampu mempelajari pola data dengan cepat. Sementara itu, *validation loss* sempat turun di awal, namun kemudian naik dan cenderung meningkat setelah beberapa *epoch*, yang mengindikasikan adanya gejala *overfitting*.

Dari sisi metrik evaluasi, nilai *accuracy* dan *F1-score* meningkat signifikan di awal pelatihan dan kemudian stabil pada kisaran tinggi tanpa peningkatan berarti. Hal ini menunjukkan bahwa model telah mencapai performa optimal sejak tahap awal. Secara

keseluruhan, model memiliki kemampuan generalisasi yang baik, dengan *epoch* terbaik berada pada rentang tengah pelatihan sebelum *overfitting* mulai terlihat.

4. Evaluasi Model

Evaluasi model menggunakan data pengujian soal cerita sebanyak 200 soal. Pada penelitian ini, metrik yang digunakan meliputi *accuracy, precision, recall,* dan *F1-score*. Serta metrik aggregate berupa *accuracy, macro average,* dan *weighted average.*

Tabel 3. Metrik Evaluasi Pengujian Model BERT per Kelas

Label	*Precision*	*Recall*	*F1-Score*	Jumlah
OP_ADD	1,00	0,93	0,96	45
OP_SUB	0,98	1,00	0,99	58
OP_MUL	0,96	0,98	0,97	50
OP_DIV	0,98	1,00	0,99	47

Sedangkan untuk performa *BERT* secara keseluruhan ditunjukkan pada Tabel 4.

Tabel 4. Metrik Evaluasi Model *BERT*

Metrik	Nilai
Accuracy	0,98
Macro avg Precision	0,98
Macro avg Recall	0,97
Macro avg F1-score	0,98
Weighted avg Precision	0,98
Weighted avg Recall	0,98
Weighted avg F1-score	0,98

Pada Tabel 4. Hasil *accuracy* sebesar 0,98 menunjukkan bahwa *encoder Transformer* pada *BERT* mampu mengklasifikasi benar sekitar 98% data pengujian. Dari 200 soal, model berhasil melakukan prediksi benar 196 soal dan prediksi salah 4 soal. Nilai *macro average* dan *weighted average* yang hampir sama menunjukkan performa model relatif merata pada semua label.

5. Kesalahan Prediksi Model

Berdasarkan hasil evaluasi model yang diperoleh, terdapat beberapa pola kesalahan pada soal data uji.

Tabel 5. Kesalahan Prediksi pada Soal

No	Teks Soal	Label Benar	Prediksi Model
1	Bu Jeni adalah penjual baju. Pada hari Senin bajunya terjual 28 buah. Kemudian pada hari Selasa bajunya terjual 38 buah. Berapa jumlah baju Bu Jeni yang terjual seluruhnya?	OP_ADD	OP_MUL
2	Toko kerudung Bu Fitri sangat ramai pengunjung. Kemarin ada 32 kerudungnya yang terjual. Kemudian hari ini ada 39 kerudung yang terjual. Berapa jumlah kerudung yang terjual?	OP_ADD	OP_MUL
3	Hari ini kantor pos menjual perangko sebanyak 1625 lembar. Kemarin menjual sebanyak 3247 lembar. Berapa jumlah perangko yang terjual?	OP_ADD	OP_SUB
4	Siswa di kelas Ika sedang belajar berbaris. Siswa dibagi dalam 3 regu. Setiap regu terdiri atas 8 siswa. Berapa orang siswa yang sedang berbaris?	OP_MUL	OP_DIV

Kesalahan prediksi Pada soal (1) dan (2), penjumlahan diprediksi sebagai perkalian karena pola kejadian berulang ditafsirkan sebagai repetisi, bukan akumulasi. Pada soal (3), model gagal mengenali arah hubungan kuantitas sehingga penjumlahan diprediksi sebagai pengurangan. Sementara pada soal (4), kata "dibagi" menyebabkan model memilih pembagian, meskipun struktur konteks menunjukkan perkalian. Hal ini menunjukkan keterbatasan model dalam memahami relasi makna pada konteks yang ambigu.

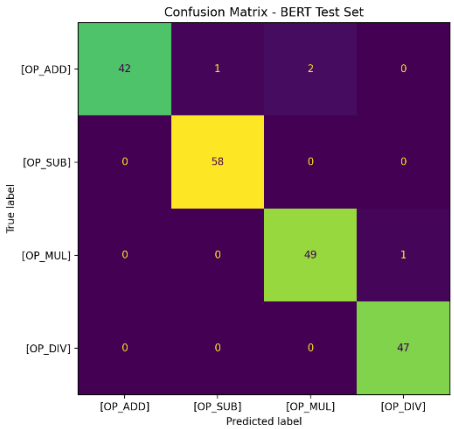
6. Perbandingan dengan Model Baseline

Perbandingan dilakukan untuk melihat efektivitas pendekatan *encoder Transformer* menggunakan *BERT* dibandingkan dengan pendekatan berbasis *BiLSTM* pada tugas klasifikasi operasi aritmetika.

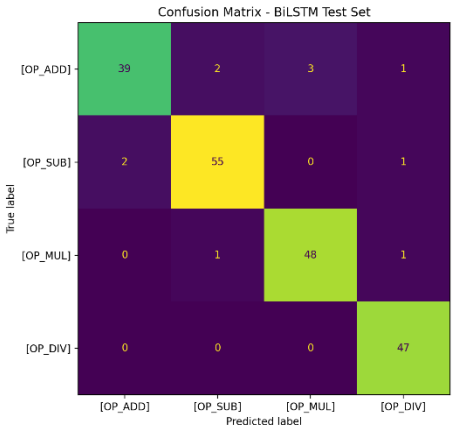
Tabel 6. Perbandingan Performa Model *BERT* dan *BiLSTM*

Model	*Accuracy*	*Macro Precision*	*Macro Recall*	*Macro F1*
BERT	0,98	0,98	0,97	0,98
BiLSTM	0,94	0,94	0,94	0,94

Berdasarkan Tabel 6. Model *BERT* menunjukkan performa yang lebih unggul dibandingkan model *BiLSTM* pada keseluruhan metrik. *Accuracy BERT* lebih tinggi sebesar 0,04 dan *macro F1* sebesar 0,04. Hasil ini menunjukkan bahwa *BERT* lebih efektif dalam menangkap konteks dan relasi antar kalimat pada soal cerita, sehingga kesalahan klasifikasi dapat ditekan. perbedaan *macro F1* juga memperlihatkan peningkatan performa model lintas label, bukan sekedar unggul pada satu kelas tertentu.



Gambar 1. *Confusion Matrix BERT*



Gambar 2. *Confusion Matrix BiLSTM*

Perhatikan Gambar 4 dan Gambar 5, dari *confusion matrix* tersebut dapat disimpulkan juga, jumlah kesalahan prediksi pada model *BERT* sebanyak 4 soal, sedangkan kesalahan prediksi pada model *BiLSTM* sebanyak 11 soal.

Tabel 7. Metrik Evaluasi per Kelas Model *BERT* dan *BiLSTM*

Label	*BERT*			*BiLSTM*		
	Precision	***Recall***	***F1-Score***	***Precision***	***Recall***	***F1-Score***
OP_ADD	0,97	0,93	0,95	0,90	0,88	0,89
OP_SUB	1,00	0,98	0,99	0,91	0,93	0,92
OP_MUL	0,90	1,00	0,95	0,95	0,94	0,94
OP_DIV	1,00	0,95	0,97	0,95	0,97	0,96

Perbedaan utama terlihat pada label OP_ADD dan OP_SUB, karena performa *BiLSTM* menurun pada *precision, recall,* dan *F1-score* sehingga lebih sering salah prediksi ke label lain. Sedangkan, operasi perkalian dan pembagian pada *BiLSTM* masih cukup baik hampir mendekati model *BERT*, yang menunjukkan pola kedua operasi ini lebih mudah dikenali pada data pengujian.

Pembahasan

Hasil penelitian menunjukkan bahwa model berbasis *encoder Transformer* mampu mencapai performa yang sangat tinggi dalam mengklasifikasikan operasi aritmetika, yang ditunjukkan oleh nilai akurasi dan *F1-score* masing-masing sebesar 0,98. Keseimbangan nilai *macro average* dan *weighted average* mengindikasikan bahwa kinerja model relatif merata pada seluruh kelas, sehingga dapat disimpulkan bahwa pendekatan *Transformer* efektif dalam mempelajari representasi semantik pada soal cerita matematika berbahasa Indonesia.

Hasil performa model yang tinggi tidak terlepas dari kemampuannya dalam memanfaatkan konteks multi-kalimat. Melalui mekanisme *self-attention*, model mampu mengintegrasikan informasi yang tersebar di berbagai kalimat, seperti kondisi awal, perubahan, dan pertanyaan, sehingga menghasilkan pemahaman yang lebih menyeluruh dibandingkan pendekatan sekuensial yang bergantung pada urutan linear.

Hasil evaluasi juga menunjukkan adanya keterbatasan model dalam menghadapi ambiguitas linguistik. Pada beberapa kasus, model cenderung salah mengklasifikasikan operasi karena mengandalkan pola bahasa permukaan, misalnya menafsirkan pola kejadian berulang sebagai perkalian, atau terpengaruh oleh kata tertentu seperti “dibagi” yang mengarah pada prediksi pembagian, meskipun konteks sebenarnya menunjukkan operasi yang berbeda.

Perbandingan dengan model baseline memperlihatkan bahwa *Transformer* berbasis *BERT* memiliki keunggulan yang signifikan dibandingkan *BiLSTM* pada seluruh metrik evaluasi. Hal ini menunjukkan bahwa kemampuan *Transformer* dalam menangkap hubungan antar kalimat memberikan kontribusi penting terhadap peningkatan akurasi dan konsistensi prediksi.

Dengan demikian, dapat disimpulkan bahwa pemodelan kontekstual multi-kalimat berbasis *Transformer* merupakan pendekatan yang efektif untuk identifikasi operasi aritmetika. Meskipun demikian, peningkatan performa masih dapat dilakukan melalui pengembangan pemahaman relasi semantik yang lebih kompleks serta perluasan dataset.

KESIMPULAN DAN REKOMENDASI

Penelitian ini berhasil mengembangkan model identifikasi operasi aritmetika pada soal cerita matematika berbahasa Indonesia menggunakan pendekatan kontekstual multi-kalimat berbasis *Transformer*. Dengan memanfaatkan mekanisme *self-attention*, model mampu mengintegrasikan informasi antar kalimat sehingga penentuan operasi tidak bergantung pada kata kunci, melainkan pada relasi makna dalam keseluruhan konteks soal.

Hasil evaluasi menunjukkan bahwa model mencapai performa yang sangat baik dengan nilai akurasi sebesar 0,98. Kesalahan prediksi yang terjadi relatif sedikit dan umumnya disebabkan oleh kemiripan pola narasi antar operasi, khususnya pada penjumlahan yang diprediksi sebagai perkalian atau pengurangan, serta perkalian yang diprediksi sebagai pembagian.

Selain itu, hasil perbandingan dengan model baseline menunjukkan bahwa pendekatan *Transformer* lebih unggul dibandingkan *BiLSTM*, dengan selisih performa sebesar 0,04. Dengan demikian, dapat disimpulkan bahwa pemodelan kontekstual multi-kalimat berbasis *Transformer* efektif dalam meningkatkan ketepatan identifikasi operasi aritmetika pada soal cerita matematika.

REFERENSI

Agusfianuddin, Herman, T., & Turmudi. (2024). Investigation of Students' Difficulties in Mathematical Language : Problem-Solving in Mathematical Word Problems at Elementary Schools. *Jurnal Kependidikan: Jurnal Hasil Penelitian Dan Kajian Kepustakaan Di Bidang Pendidikan, Pengajaran Dan Pembelajaran*, *10*(2), 578–590. https://doi.org/10.33394/jk.v10i2.11257

Ayumi, V., Purba, M., & Rahman, A. (2025). Model Deep Learning Berbasis Word2Vec dan LSTM untuk Klasifikasi Umpan Balik Kompetensi Profesional Dosen. *JSAI : Journal Scientific and Applied Informatics*, *08*(2), 558–563. https://doi.org/10.36085/jsai.v8i2.8762

Ba, J. L., Kiros, J. R., & Hinton, G. E. (2016). Layer Normalization. *ArXiv*, 1–14. https://doi.org/10.48550/arXiv.1607.06450

Chicco, D., & Jurman, G. (2020). The advantages of the Matthews correlation coefficient (MCC) over F1 score and accuracy in binary classification evaluation. *BMC Genomics*, *21*(6), 1–13.

Devlin, J., Chang, M.-W., Lee, K., & Tautanova, K. (2019). BERT : Pre-training of Deep Bidirectional Transformers for Language Understanding. *In Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, 4171–4186. https://doi.org/10.18653/v1/N19-1423

Faldu, K., Sheth, A., Kikani, P., & Patel, D. (2022). MMTM: Multi-Tasking Multi-Decoder Transformer for Math Word Problems. *ArXiv*. https://doi.org/10.48550/arXiv.2206.01268

Grandini, M., Bagli, E., & Visani, G. (2020). METRICS FOR MULTI-CLASS CLASSIFICATION: AN OVERVIEW. *ArXiv*, 1–17. https://doi.org/10.48550/arXiv.2008.05756

Hickendorff, M. (2021). The Demands of Simple and Complex Arithmetic Word Problems on Language and Cognitive Resources. *Frontiers in Psychology*, *12*(1), 1–12. https://doi.org/10.3389/fpsyg.2021.727761

Irani, H., & Metsis, V. (2025). Positional Encoding in Transformer-Based Time Series Models: A Survey. *ArXiv*, 1–15. https://doi.org/10.48550/arXiv.2502.12370

Jin, P., Zhu, B., Yuan, L., & Yan, S. (2025). MoH: Multi-Head Attention as Mixture-of-Head Attention. *International Conference on Machine Learning*, 1–23. https://doi.org/10.48550/arXiv.2410.11842

Kim, J., Kim, Y., Baek, I., Bak, J., & Lee, J. (2023). It Ain ’ t Over : A Multi-aspect Diverse Math Word Problem Dataset. *Proceedings Ofthe 2023 Conference on Empirical Methods in Natural Language Processing*, 14984–15011. https://doi.org/10.18653/v1/2023.emnlp-main.927

Liang, Z., Yu, W., Rajpurohit, T., Clark, P., Zhang, X., & Kaylan, A. (2023). Let GPT be a Math Tutor: Teaching Math Word Problem Solvers with Customized Exercise Generation. *Association for Computational Linguistics*, 14384–14396. https://doi.org/10.18653/v1/2023.emnlp-main.889

Liang, Z., Zhang, J., Wang, L., Qin, W., Lan, Y., Shao, J., Zhang, X., & Dame, N. (2022). MWP-BERT : Numeracy-Augmented Pre-training for Math Word Problem Solving. *Association for Computational Linguistics*, 997–1009. https://doi.org/10.18653/v1/2022.findings-naacl.74

Liu, L., Liu, X., Gao, J., Chen, W., & Han, J. (2020). Understanding the Difficulty of Training Transformers. *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing*, 5747–5763. https://doi.org/10.48550/arXiv.2004.08249

Liu, W., Hu, H., Zhou, J. I. E., Ding, Y., Li, J., Zeng, J., He, M., & Chen, Q. (2025). Mathematical Language Models : A Survey. *Mathematical Language Models : A Survey*, *58*(6), 111:1-111:35. https://doi.org/10.1145/3773985

Liu, Y., & Lapata, M. (2019). Hierarchical Transformers for Multi-Document Summarization. *Association for Computational Linguistics*, 5070–5081. https://doi.org/10.18653/v1/P19-1500

Loshchilov, I., & Hutter, F. (2019). Decoupled Weight Decay Regularization. *International Conference on Learning Representations*, 1–8. https://doi.org/10.48550/arXiv.1711.05101

Mandal, S., & Naskar, S. K. (2021). Classifying and Solving Arithmetic Math Word Problems — An Intelligent Math Solver. *IEEE TRANSACTIONS ON LEARNING TECHNOLOGIES*, *14*(1), 28–41. https://doi.org/10.1109/TLT.2021.3057805

Opitz, J. (2024). A Closer Look at Classification Evaluation Metrics and a Critical Reflection of Common Evaluation Practice. *MIT Press*, *12*, 820–836. https://doi.org/10.1162/tacl_a_00675

Papineni, K., Roukos, S., Ward, T., & Zhu, W. (2002). B LEU : a Method for Automatic Evaluation of Machine Translation. *Proceedings of the 40th Annual Meeting of the Association for Computational Linguistics (ACL)*, *July*, 311–318. https://doi.org/10.3115/1073083.1073135

Patel, A., Bhattamishra, S., & Goyal, N. (2021). Are NLP Models really able to Solve Simple Math Word Problems? *Association for Computational Linguistics*, 2080–2094. https://doi.org/10.18653/v1/2021.naacl-main.168

Qi, P., Zhang, Y., Zhang, Y., Bolton, J., & Manning, C. D. (2020). Sta n z a : A Python Natural Language Processing Toolkit for Many Human Languages. *Proceedings Ofthe 58th Annual Meeting Ofthe Association for Computational Linguistics,* 101–108. https://doi.org/10.18653/v1/2020.acl-demos.14

Reimers, N., & Gurevych, I. (2019). Sentence-BERT: Sentence Embeddings using Siamese BERT-Networks. *Association for Computational Linguistics*, 3982–3992. https://doi.org/10.18653/v1/D19-1410

Sarjana, K., Hayati, L., & Wahidaturrahmi. (2020). Mathematical modelling and verbal abilities: How they determine students' ability to solve mathematical word problems? *Beta: Jurnal Tadris Matematika*, *13*(2), 117–129. https://doi.org/10.20414/betajtm.v13i2.390

Srivastava, N., Hinton, G., Krizhevsky, A., Sutskever, I., & Salakhutdinov, R. (2014). Dropout : A Simple Way to Prevent Neural Networks from Overfitting. *Journal of Machine Learning Research*, *15*(56), 1929–1958. https://doi.org/10.5555/2627435.2670313

Vaswani, A., Brain, G., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., Kaiser, Ł., & Polosukhin, I. (2017). Attention Is All You Need (Transformer Architecture). *Neural Information Processing Systems*, *Nips*, 1–15. https://doi.org/10.5555/3295222.3295349

Wang, H., Ma, S., Dong, L., Huang, S., Zhang, D., & Wei, F. (2024). DeepNet: Scaling Transformers to 1,000 Layers. *IEEE Computer Society*, *46*(10), 6761–6774. https://doi.org/10.1109/TPAMI.2024.3386927

Wang, Z., Gu, J., Kuen, J., Zhao, H., & Morariu, V. I. (2022). Learning Adaptive Axis Attentions in Fine-tuning : Beyond Fixed Sparse Attention Patterns. *Association for Computational Linguistics*, 916–925. https://doi.org/10.18653/v1/2022.findings-acl.74

Xiong, R., Yang, Y., He, D., Zheng, K., Zheng, S., Xing, C., & Zhang, H. (2020). On Layer Normalization in the Transformer Architecture. *Journal of Machine Learning Research*, 1–17. https://doi.org/10.5555/3524938.3525913

Zhang, X., Wei, F., & Zhou, M. (2019). HIBERT : Document Level Pre-training of Hierarchical Bidirectional Transformers for Document Summarization. *Association for Computational Linguistics*, 5059–5069. https://doi.org/10.18653/v1/P19-1499