

OLAH DATA EKUASI MENGGUNAKAN KOMBINASI GRAPH NEURAL NETWORK (GNN) DALAM MACHINE LEARNING

Adamsyach Prana Walidin, Fahra Pebiana Putri,
Muhammad Raffi Akbar, Khairul Fahmi Sagala, Fanny Ramadhani

Ilmu Komputer, Universitas Negeri Medan
Jl. William Iskandar Ps. V, Kenangan Baru, Kec. Percut Sei Tuan,
Kabupaten Deli Serdang, Sumatera Utara 20221, Indonesia
adamsyachpranawalidin@gmail.com

ABSTRAK

Pengolahan Data ini dikembangkan sebagai platform edukasi berbasis kecerdasan buatan yang membantu pengguna dalam menyelesaikan tugas akademik, seperti penyelesaian ekuasi matematis. Sistem ini memanfaatkan *Graph Neural Network* (GNN) untuk mengubah soal matematika menjadi representasi graf, memungkinkan model untuk memahami dan menyelesaikan persamaan secara lebih efektif. Implementasi awal telah mencapai *Minimum Viable Product* (MVP) dengan hasil yang menunjukkan bahwa pendekatan berbasis graf dapat diterapkan dalam konteks pendidikan. Menggunakan dataset soal ekuasi, dengan susunan graf yang didukung model *machine learning*, Ekuasi ditentukan menggunakan operator matematis yang sudah dideklarasikan pada model. Namun, masih terdapat beberapa tantangan, seperti validasi input dan optimasi model agar lebih akurat. Didapatkan hasil latih model dengan *epoch error loss* terbaik sebesar 0.374512 dalam pelatihan model. Model juga diukur menggunakan pertanyaan matematis dengan hasil yang akurat terhadap soal dengan tingkat akurasi sebesar 93,36%. Ke depannya, penelitian lebih lanjut dapat dilakukan dengan meningkatkan akurasi model, serta mengeksplorasi kombinasi GNN dengan *Large Language Models* (LLMs) untuk meningkatkan pemahaman terhadap soal yang lebih kompleks. Dengan pengembangan yang berkelanjutan, inovatif dalam membantu pelajar dan pengajar mengakses solusi berbasis AI yang lebih interaktif dan cerdas.

Kata kunci : *Graph Neural Network, olah data, edukasi, pemrosesan graf, pembelajaran mesin*

1. PENDAHULUAN

Perkembangan teknologi digital dalam bidang pendidikan telah mengalami kemajuan signifikan dalam beberapa tahun terakhir. Transformasi digital ini membuka peluang besar untuk menciptakan berbagai inovasi yang dapat menunjang proses belajar-mengajar, baik dalam bentuk sistem (*Learning management pembelajaran Management System*), platform berbagi materi, hingga aplikasi cerdas berbasis kecerdasan buatan (*Artificial Intelligence*). Salah satu tantangan yang masih dihadapi oleh banyak pelajar dan pengajar adalah keterbatasan waktu dan sumber daya dalam menyelesaikan tugas, memahami konsep rumit seperti penyelesaian persamaan matematika, debugging kode pemrograman, serta menyederhanakan atau memparafrasekan materi pelajaran. Tantangan ini menjadi pemicu lahirnya sebuah gagasan untuk mengembangkan platform berbasis web yang dapat memberikan bantuan dalam menyelesaikan persoalan tersebut secara otomatis dan efisien. Dalam era digital, pengolahan data berbasis kecerdasan buatan (AI) dan data mining telah menjadi elemen krusial dalam berbagai bidang, termasuk pendidikan. Penggunaan perangkat lunak dalam dunia akademik semakin berkembang, salah satunya dengan hadirnya sistem berbasis web yang dapat membantu siswa dalam menyelesaikan tugas akademik. Salah satu tantangan yang dihadapi adalah bagaimana menciptakan sistem yang tidak hanya memberikan jawaban tetapi juga membantu pengguna memahami konsep di balik jawaban tersebut.

Dengan tujuan menambah efisiensi siswa dan sebagainya, model ini merupakan sebuah platform berbasis web yang bertujuan membantu pelajar dan pengajar dalam menyelesaikan tugas dengan memanfaatkan teknologi pembelajaran mesin (*Machine Learning*), khususnya pendekatan berbasis *Graph Neural Network* (GNN). Sistem ini dirancang untuk memberikan solusi otomatis terhadap tiga aspek utama dalam penyelesaian tugas: menyelesaikan persamaan matematika, melakukan debugging terhadap potongan kode pemrograman, dan melakukan parafrase terhadap kalimat atau paragraf. Latar belakang munculnya ide ini adalah berdasarkan pengamatan langsung terhadap kebutuhan pengguna di lingkungan pendidikan serta kurangnya solusi terintegrasi yang mampu memberikan hasil yang cepat dan akurat untuk berbagai jenis tugas tersebut.

2. TINJAUAN PUSTAKA

2.1. Penelitian Terdahulu

Penelitian "A Review on Linear Regression Comprehensive in Machine Learning" karya Dastan Hussen Maulud dan Adnan Mohsin Abdulazeez menyajikan tinjauan komprehensif terhadap berbagai pendekatan dan aplikasi regresi linear dalam konteks pembelajaran mesin selama lima tahun terakhir. Dalam penelitian ini, penulis membahas secara mendalam tiga jenis regresi utama, yaitu regresi linear sederhana (SLR), regresi linear berganda (MLR), dan regresi polinomial. Fokus utama diarahkan pada bagaimana model regresi digunakan untuk

memprediksi variabel dependen berdasarkan variabel independen, dengan mengandalkan teknik optimasi seperti least square method untuk meminimalkan kesalahan prediksi. Evaluasi performa dilakukan dengan menggunakan data simulasi dan data historis, yang menunjukkan bahwa regresi linear berganda menghasilkan akurasi tertinggi hingga mencapai 99.89%. Hal ini memperkuat posisi regresi linear sebagai metode prediktif yang kuat dalam skenario data yang memiliki hubungan linier yang jelas.

Meskipun fokus utama penelitian ini bukan pada model pembelajaran berbasis graf seperti Graph Neural Network (GNN), pemahaman tentang regresi linear tetap relevan, terutama dalam konteks output numerik dan evaluasi performa prediktif. Dalam konteks pemrosesan data matematis dan numerik seperti penyelesaian persamaan, prinsip dasar regresi linear sering kali digunakan untuk merepresentasikan hubungan antar elemen, seperti konstanta dan operator, secara kuantitatif. Namun demikian, studi ini memiliki beberapa keterbatasan, antara lain karena hanya menyoroti regresi linear tanpa eksplorasi mendalam mengenai kelemahan metode ini dalam menghadapi relasi non-linear atau struktur data kompleks seperti graf. Selain itu, pendekatan yang diambil cenderung teoritis dan tidak menyertakan perbandingan yang kuat dengan metode machine learning modern seperti neural network atau GNN. Akibatnya, jurnal ini kurang memberikan panduan praktis mengenai situasi-situasi di mana regresi linear lebih unggul atau justru sebaiknya dihindari dalam konteks aplikasi machine learning yang lebih kompleks.

2.2. Data Mining dan Teknik Frequent Pattern Mining

Data mining merupakan proses penggalian informasi berharga dan tersembunyi dari sekumpulan data yang besar dan tidak terstruktur. Proses ini terdiri dari beberapa tahap, yaitu: pemilihan data relevan dari database, praproses data seperti pengurangan noise dan redundansi, transformasi data menjadi format seragam (misalnya Boolean), pemilihan metode dan algoritma yang sesuai, pelaksanaan proses mining, evaluasi pola yang ditemukan, hingga penyajian informasi tersebut dalam bentuk yang mudah dipahami. Data mining memiliki lima metode utama: analisis klaster, klasifikasi dan prediksi, analisis outlier, serta analisis *association rules*. Konsep data mining tidak hanya terbatas pada penggalian informasi, tetapi juga mencakup proses penyusunan dan perumusan pengetahuan dari data. [2]

Frequent data mining berfokus pada pencarian *frequent itemsets*, yakni kombinasi item yang sering muncul bersama dalam suatu dataset. Tujuannya adalah untuk menemukan hubungan atau korelasi antar item melalui aturan asosiasi (*association rules*). Salah satu aplikasi populernya adalah *market basket analysis*, yang mengidentifikasi pola pembelian seperti "orang yang membeli popok juga cenderung

membeli bir". Ukuran utama dalam analisis ini adalah *support*, *confidence*, dan *lift*, yang masing-masing mengukur frekuensi kemunculan item bersama, kekuatan implikasi antar item, dan signifikansi dari hubungan antar item dibandingkan distribusi acaknya. [3]

2.3. Algoritma K-Means Untuk Klasterisasi Data

Clustering atau pengelompokan data merupakan teknik dalam *data mining* yang digunakan untuk mengelompokkan data berdasarkan kemiripan karakteristik. Algoritma K-Means adalah salah satu metode *clustering* yang umum digunakan dalam analisis data skala besar. Dalam konteks penelitian ini, algoritma K-Means dapat diterapkan untuk mengelompokkan jenis pertanyaan dan permasalahan yang sering dihadapi oleh pengguna *Assignment Assistant*, sehingga sistem dapat memberikan rekomendasi solusi yang lebih relevan (Apriyani et al., 2023). [4]

2.4. Graph Neural Network (GNN)

Graf adalah struktur yang digunakan untuk menunjukkan objek diskrit dan hubungan mereka satu sama lain. Graf biasanya digambarkan sebagai pasangan dua nilai atau tuple: $G = (V, E)$ di mana V adalah himpunan tidak kosong (vertices) dan E adalah himpunan sisi yang menghubungkan dua simpul. Graf terbagi menjadi beberapa jenis berdasarkan apakah mereka memiliki gelang atau sisi ganda. Yang pertama adalah graf sederhana, yang tidak memiliki sisi ganda, dan yang kedua adalah graf sisi ganda, yang merupakan kebalikannya, dengan sisi ganda. Selain itu, ada jenis khusus dari graf ganda, yaitu graf semu, yang memiliki sisi gelang. [5]

GNN adalah metode berbasis pembelajaran mendalam yang berjalan pada domain grafik. Karena kinerjanya yang luar biasa, GNN baru-baru ini menjadi metode analisis grafis yang banyak digunakan. Struktur data yang memodelkan sekumpulan objek (simpul) dan hubungannya (tepi) disebut grafik. Analisis grafik berkonsentrasi pada tugas-tugas seperti pengelompokan, prediksi tautan, dan klasifikasi simpul. Ini adalah jenis pembelajaran mesin yang unik dengan struktur data non-Euclidean. Metode berbasis grafik telah sangat populer dalam beberapa tahun terakhir dan terbukti menjadi cara yang efisien untuk penalaran relasi. CRF dan jaringan random walk diusulkan berdasarkan model grafik untuk segmentasi gambar yang efektif. Menurut cara seseorang mendefinisikan operator konvolusi grafik, GNN dapat dibagi menjadi dua jenis. Graph Neural Network juga disebut Graph Convolution Network (GCN) oleh banyak orang. Pelabelan peran semantik, klasifikasi relasi, dan terjemahan mesin adalah beberapa tugas pengolahan bahasa natural, di mana GCN digunakan untuk mengkodekan struktur sintaksis kalimat. Dalam metode ini, setiap simpul pusat dan simpul tetangga digabungkan dengan mendefinisikan fungsi agregasi. Metode spasial bertujuan untuk

menggabungkan setiap simpul pusat dan simpul tetangga dengan menentukan fungsi agregasi dari domain simpul. [6], [7]

2.5. Pytorch Geometry

PyTorch Geometric (PyG) adalah pustaka deep learning berbasis PyTorch yang dirancang untuk memproses data dengan struktur graf. A2 untuk menyelesaikan soal matematika berbasis teks. PyG menyediakan antarmuka modular dan efisien untuk membangun berbagai arsitektur Graph Neural Network (GNN), termasuk Graph Convolutional Networks (GCN) dan Graph Attention Networks (GAT). Dalam konteks proyek A2, PyG memungkinkan representasi formula matematika sebagai graf, di mana node merepresentasikan angka atau konstanta, dan edge merepresentasikan operasi matematika. Dengan fitur-fitur seperti propagasi pesan (message passing) dan dukungan untuk pelatihan batch pada graf yang berbeda, PyG memfasilitasi pelatihan model yang mampu menangkap hubungan kompleks antar elemen matematika. Studi oleh Rozemberczki et al. (2021) memperkenalkan PyTorch Geometric Temporal, sebuah ekstensi dari PyG yang mendukung pemrosesan sinyal spatiotemporal, menunjukkan fleksibilitas dan skalabilitas PyG dalam berbagai aplikasi, termasuk prediksi permintaan transportasi dan manajemen lalu lintas web, yang relevan dengan pengembangan sistem edukatif adaptif. [8]

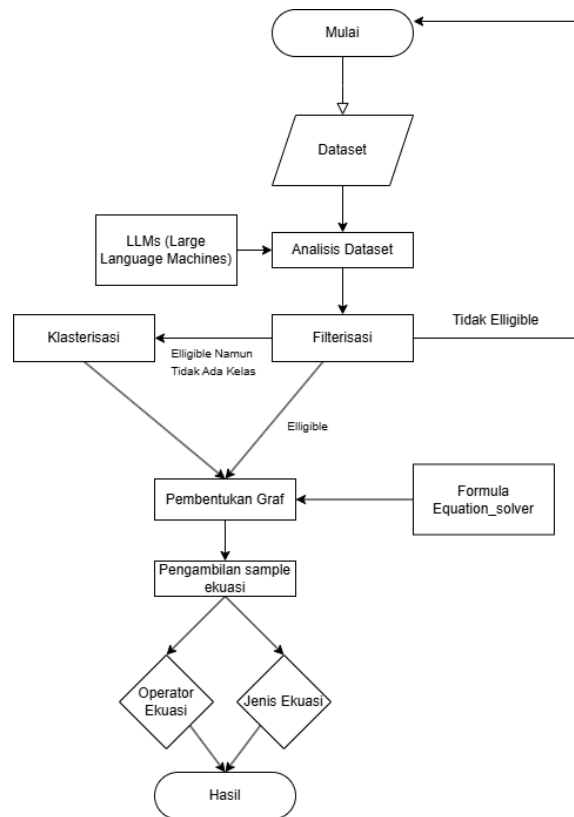
2.6. Large Language Machines (LLMs)

Large Language Machines (LLM), seperti GPT dan BERT, telah merevolusi bidang pemrosesan bahasa alami (Natural Language Processing/NLP) dengan kemampuannya memahami konteks dan menghasilkan teks secara manusiawi. Dalam konteks pengembangan proyek A2, LLM dapat dimanfaatkan untuk menginterpretasi soal matematika berbasis teks, menganalisis konteks linguistiknya, dan menerjemahkannya ke dalam formula matematis yang dapat diproses lebih lanjut oleh model Graph Neural Network (GNN). Integrasi LLM dengan modul reasoning dan symbolic solver dapat meningkatkan kemampuan sistem dalam menyelesaikan soal matematika kompleks yang memerlukan pemahaman semantik mendalam. Mereka mengusulkan framework "Think Like A Transformer" yang menggabungkan LLM dengan prosedur reasoning berbasis *chain-of-thought*, menghasilkan peningkatan signifikan dalam akurasi penyelesaian masalah matematika dan pemahaman logika simbolik. Temuan ini mendukung dalam menyediakan fungsi *natural language to formula parsing* dan reasoning lanjutan untuk mendukung pengguna dalam mengatasi hambatan pemahaman matematika secara intuitif dan efisien. [9], [10]

3. METODE PENELITIAN

Dengan menggunakan metode kualitatif disertai dengan *Research & develop (R&D)*, penelitian ini

didasari oleh observasi yang mendalam sehingga dapat mengungkap sekiranya bagaimana sebuah sistem dirancang agar dapat membantu user sesuai dengan tugas dan pekerjaannya.



Gambar 1. Flowchart Olah Data

3.1. Pengumpulan Data

Tahap pengumpulan data dan pemrosesan dataset dilakukan dengan menggunakan dataset MathQA yang tersedia dalam format .json dan .txt. Dataset ini digunakan khusus untuk fungsi penyelesaian soal matematika, karena menyediakan berbagai soal dengan anotasi formula yang sesuai untuk diubah menjadi struktur graf. Proses preprocessing data melibatkan pembersihan format formula, identifikasi token numerik dan operator, serta konversi menjadi struktur graf menggunakan PyTorch Geometric (PyG). Pada tahap ini juga dilakukan filtrasi data untuk memastikan hanya graf valid yang digunakan dalam pelatihan model.

3.2. Analisis Data

Pada Gambar 2, Dataset MathQA yang digunakan dapat dianalisis langsung oleh model setelah adanya modifikasi sesuai dari penyelesaian ekuasi. Pertanyaan yang diolah berupa pertanyaan berbentuk cerita, dengan opsi yang ditampilkan. Ada beberapa elemen ekuasi yang dapat dianalisis oleh model yakni *add*, *multiply*, *divide*, dst. Dan juga beberapa bentuk angka seperti konstanta hasil dari pengolahan ekuasi.

```
[Running] python -u "d:\Download\A2-project\backend\train.py"
Contoh data pertama: {'Problem': 'there are 1000 buildings
in a street . a sign - maker is contracted to number the
houses from 1 to 1000 . how many zeroes will he need ?',
'Rationale': 'divide as ( 1 - 100 ) ( 100 - 200 ) . . . .
( 900 - 1000 ) total 192 answer : c', 'options': 'a )
190 , b ) 191 , c ) 192 , d ) 193 , e ) 194', 'correct':
'c', 'annotated_formula': 'add(add(divide(1000, const_10),
multiply(subtract(const_10, 1), const_10)), const_2)',
'linear_formula': 'divide(n0,const_10)|subtract(const_10,
n1)|multiply(#1,const_10)|add(#0,#2)|add(#3,const_2)',
'category': 'general'}
```

Gambar 2 Analisis Dataset Dari Model PyG

3.3. Tahapan Pemodelan

Tahap pemodelan dan pelatihan model GNN merupakan inti dari penelitian ini. Model dikembangkan menggunakan framework PyTorch dan PyG. Model EquationSolverGNN dirancang sebagai jaringan GNN sederhana dengan beberapa lapisan konvolusi graf (GraphConv), ReLU sebagai fungsi aktivasi, serta lapisan linier untuk menghasilkan prediksi nilai numerik dari node dalam graf. Data graf yang sudah dikonversi sebelumnya digunakan sebagai masukan (fitur node dan struktur edge) untuk melatih model dengan target nilai numerik dari node. Fungsi loss yang digunakan adalah Mean Squared Error (MSE) atau L1 Loss, bergantung pada stabilitas dan kebutuhan eksperimen. Pelatihan dilakukan selama beberapa epoch dengan batch size tertentu untuk memantau penurunan nilai loss.

Setelah model selesai dilatih, tahap selanjutnya adalah penerapan model ke dalam sistem backend menggunakan FastAPI. API ini dibuat untuk menerima input dari pengguna berupa soal matematika dan formula anotasinya, lalu memproses input tersebut menjadi graf dan menjalankan inferensi melalui model yang telah dilatih.

Hasil prediksi akan dikembalikan ke frontend dalam format JSON. Endpoint utama yang dikembangkan adalah `/predict`, yang menerima permintaan POST dan merespons dengan hasil prediksi.

Tahap pengembangan frontend dilakukan dengan memodifikasi HTML, CSS, dan JavaScript dari sumber terbuka (open source). Antarmuka pengguna dibuat sesederhana mungkin dengan tampilan intuitif dan dominasi warna ungu sebagai warna utama. Di halaman penyelesaian soal matematika, tersedia kolom input untuk formula dan kolom hasil yang akan menampilkan prediksi model. Komunikasi frontend dan backend dilakukan menggunakan fetch API JavaScript yang mengirimkan data ke endpoint `/predict` dan menampilkan hasil respons.

Tahap terakhir adalah pengujian sistem secara menyeluruh. Pengujian dilakukan untuk memastikan bahwa frontend dapat berkomunikasi dengan backend secara lancar, dan bahwa model memberikan prediksi sesuai dengan input yang diberikan. Pengujian dilakukan dalam dua tahap: pertama, pengujian lokal dengan dataset terpisah untuk validasi model; kedua,

pengujian fungsionalitas sistem melalui interaksi antarmuka pengguna. Hasil pengujian menunjukkan bahwa sistem dapat memberikan respons prediktif dalam waktu singkat, meskipun akurasi model masih dapat ditingkatkan.

4. HASIL DAN PEMBAHASAN

Penelitian ini berhasil mengembangkan prototipe Minimum Viable Product (MVP) yang berfungsi sebagai asisten penyelesaian tugas berbasis kecerdasan buatan. Sistem ini memungkinkan pengguna untuk memasukkan soal matematika dan mendapatkan penyelesaian dengan pendekatan berbasis Graph Neural Network (GNN).

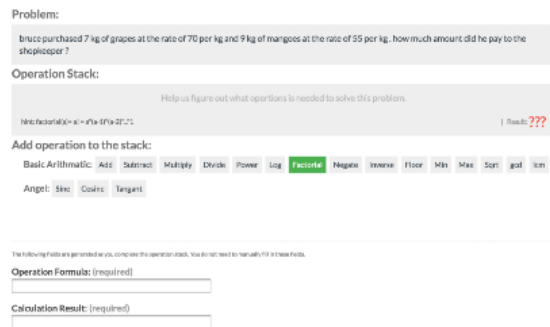
Setelah melatih model menggunakan GraphSAGE selama 20 epoch, diperoleh nilai loss terendah sebesar 0.374512. Nilai ini dihitung menggunakan Mean Squared Error (MSE), yang berfungsi untuk mengukur seberapa besar perbedaan antara prediksi rating dan rating aktual yang diberikan oleh pengguna. Semakin kecil nilai MSE, maka semakin akurat model dalam memprediksi rating yang sesuai dengan preferensi pengguna.

Selama proses pelatihan, grafik loss menunjukkan penurunan secara bertahap, yang mengindikasikan bahwa model mampu mempelajari pola dari data dengan baik tanpa menunjukkan gejala overfitting yang berarti.

Memulai training...		
Epoch	Loss	LR
1	0.896377	0.001000
2	0.849069	0.001000
3	0.809102	0.001000
4	0.767371	0.001000
5	0.729690	0.001000
6	0.689733	0.001000
7	0.673024	0.001000
8	0.635129	0.001000
9	0.607966	0.001000
10	0.580573	0.001000
11	0.558908	0.001000
12	0.523865	0.001000
13	0.501450	0.001000
14	0.485375	0.001000
15	0.458428	0.001000
16	0.444225	0.001000
17	0.423858	0.001000
18	0.405567	0.001000
19	0.388338	0.001000
20	0.374512	0.001000
Training selesai!		
Loss terbaik: 0.374512		

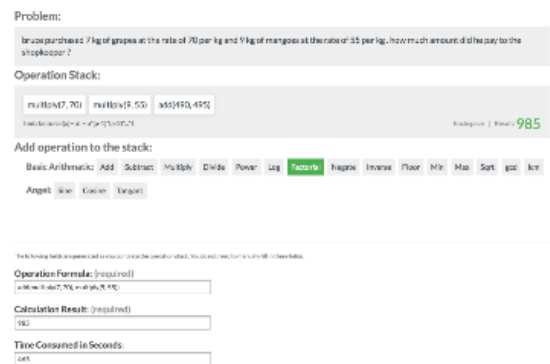
Gambar 3. Hasil Pelatihan Model

Untuk mengetahui apakah model yang dilatih ini sudah tepat persentase error dalam penyelesaian sebuah ekuasi, dilakukan tes serta Analisa pada sebuah interface yang didesain khusus untuk mengukur tingkat keakuratan perhitungan.



Gambar 4. Tes Ukur Akurasi Model

Dapat dilihat pada Gambar 4, Interface ini dirancang untuk dapat melakukan tes pada operator matematis seperti pengurangan, penjumlahan, perkalian, pembagian dan sebagainya. Disini digunakan soal cerita dengan menguraikan formula matematis pada kolom dibawahnya, lalu dilakukan pengujian satu persatu operator matematis dari formula tersebut.



Gambar 5. Hasil Tes Ukur Akurasi Model

Hasil dapat dilihat pada gambar 7, dimana sistem merekognisi uraian formula matematis yang dikerjakan oleh model dan mengeluarkan jawaban yang tepat dari soal yang sudah dimasukkan sebelumnya.

Hasil dari Langkah ini membuktikan bahwa pendekatan GNN dapat digunakan sebagai dasar untuk menyelesaikan soal matematika berbasis teks yang telah dianotasi dalam bentuk formula. Ini merupakan langkah maju dalam otomatisasi penyelesaian soal matematika berbasis representasi graf, yang selama ini didominasi oleh pendekatan NLP konvensional. Dalam aplikasi praktis, model ini dapat diterapkan dalam platform pembelajaran berbasis AI yang mampu memberikan bantuan otomatis dalam memecahkan soal matematika berbasis teks. Potensinya sangat besar, mulai dari penerapan dalam aplikasi latihan soal mandiri, hingga integrasi dalam sistem pembelajaran daring dan asisten pengajar digital.

5. KESIMPULAN DAN SARAN

Penelitian ini telah berhasil mencapai tahap Minimum Viable Product (MVP) dengan implementasi awal fitur penyelesaian soal berbasis Graph Neural Network (GNN). Sistem ini mampu mengubah soal matematika menjadi representasi graf dan memprosesnya untuk mendapatkan solusi. Meskipun demikian, masih terdapat beberapa tantangan teknis, seperti validasi input dan optimasi model agar lebih akurat. Dengan perbaikan lebih lanjut, proyek ini dapat berkembang menjadi platform edukasi berbasis kecerdasan buatan yang membantu siswa dalam menyelesaikan tugas secara efektif. Sebagai saran, penelitian lanjutan dapat difokuskan pada peningkatan akurasi model melalui transfer learning atau fine-tuning dengan dataset yang lebih besar dan kompleks. Selain itu, eksplorasi lebih lanjut terhadap transformer-based models seperti Graphormer atau kombinasi GNN dengan Large Language Models (LLMs) dapat memperkaya kemampuan sistem dalam memahami konteks soal. Pengembangan fitur tambahan, seperti debugging kode otomatis dan alat parafrase berbasis NLP, juga dapat menjadi arah penelitian yang menarik untuk meningkatkan fungsionalitas sebagai platform edukasi berbasis AI yang lebih komprehensif.

DAFTAR PUSTAKA

- [1] D. H. Maulud and A. M. Abdulazeez, "A Review on Linear Regression Comprehensive in Machine Learning," *Journal of Applied Science and Technology Trends*, vol. 01, no. 02, pp. 140–147, Dec. 2020, doi: 10.38094/jastt1457.
- [2] J. Han, M. Kamber, dan J. Pei, *Data Mining: Concepts and Techniques*, 3rd ed. Amsterdam: Morgan Kaufmann, 2011.
- [3] H. Xie, "Research and Case Analysis of Apriori Algorithm Based on Mining Frequent Item-Sets," *Open Journal of Social Sciences*, vol. 9, pp. 458–468, Apr. 2021, doi: 10.4236/jss.2021.94034.
- [4] P. Apriyani, A. H. Saputra, and R. P. Sari, "Penerapan Algoritma K-Means dalam Klasterisasi Kasus Stunting Balita Desa Tegalwangi," *J. Ilmu Komput.*, vol. 2, no. 1, pp. 20–33, 2023.
- [5] M. G. A. Giffera, "Penerapan Graph Neural Network dalam Pembangunan Sistem Rekomendasi," Makalah IF2120 Matematika Diskrit, Institut Teknologi Bandung, 2021.
- [6] R. Li et al., "A Graph Neural Network Model for the Diagnosis of Lung Adenocarcinoma Based on Multimodal Features and an Edge-Generation Network," *Quantitative Imaging in Medicine and Surgery*, vol. 13, no. 8, pp. 5333–5348, 2023.
- [7] Y. Chen et al., "Graph-Based Global Reasoning Networks," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2021.
- [8] B. Rozemberczki et al., "PyTorch Geometric Temporal: Spatiotemporal Signal Processing with Neural Machine Learning Models," *arXiv*

- preprint arXiv:2104.07788*, 2021. [Online]. Available: <https://arxiv.org/abs/2104.07788>
- [9] Y. Wang, X. Li, S. Liu, H. Zhang, and W. Liu, "Think Like A Transformer: Boosting Symbolic Mathematical Reasoning with Chain-of-Thought LLMs," *arXiv preprint arXiv:2306.00931*, Jun. 2023. [Online]. Available: <https://arxiv.org/abs/2306.00931>
- [10] S. Imani, L. Du, and H. Shrivastava, "MathPrompter: Mathematical Reasoning using Large Language Models," *arXiv preprint arXiv:2303.05398*, Mar. 2023. [Online]. Available: <https://arxiv.org/abs/2303.05398>