

Penerapan *Sentence BERT* Untuk Similaritas Kompetensi Pekerjaan dan Mata Kuliah

<http://dx.doi.org/10.28932/jutisi.vXiX.X>

Riwayat Artikel

Received: 12 Juli 2024 | Final Revision: 23 Juli 2024 | Accepted: 03 Agustus 2024

Creative Commons License 4.0 (CC BY – NC)



Kafka Febianto Agiharta^{✉#1}, Bernard Renaldy Suteja^{#2}, Mewati Ayub^{#3}

[#] Program Studi Magister Ilmu Komputer, Universitas Kristen Maranatha
Jl. Prof. drg. Surya Sumantri, M.P.H. No.65, Bandung, 40164, Indonesia

¹febianhdradeon5500@gmail.com

²bernard.rs@it.maranatha.edu

³mewati.ayub@it.maranatha.edu

[✉]Corresponding author: febianhdradeon5500@gmail.com

Abstrak — Penelitian ini berfokus pada penerapan model *Sentence BERT* (*S-BERT*), spesialisasi dari model *BERT* dan adaptasi dari arsitektur *Transformer* yang dirancang khusus untuk bahasa Indonesia, dalam mengeksplorasi konsep konsolidasi alih kredit mata kuliah sesuai dengan program Merdeka Belajar – Kampus Merdeka. Tujuan dari eksplorasi ini adalah mengembangkan sebuah model *S-BERT* berbahasa Indonesia dan mengaplikasikannya untuk melakukan pencarian serta analisis similaritas antar deskripsi sekuens aktivitas dengan Rencana Pembelajaran Semester (RPS) mata kuliah. Hasil dari analisis similaritas ini adalah identifikasi mata kuliah yang relevan berdasarkan *query* yang diberikan. Model yang dikembangkan berhasil menunjukkan kemampuan yang efektif dalam melakukan pencarian dan penentuan similaritas antar deskripsi sekuens aktivitas dan RPS mata kuliah dengan hasil evaluasi akurasi similaritas model menggunakan *cosine similarity* yaitu sekitar 37% pada $k=10$ dan pada similaritas menggunakan *dot product* yaitu sekitar 41% pada nilai k yang sama. Mata kuliah yang teridentifikasi sebagai relevan dengan *query* menunjukkan similaritas dan kecocokan yang tinggi, menunjukkan bahwa model *S-BERT* dapat diandalkan dalam proses konsolidasi alih kredit terhadap mata kuliah dalam konteks Merdeka Belajar – Kampus Merdeka.

Kata kunci— *BERT*; Kampus Merdeka; RPS; *S-BERT*; *Transformer*.

Application of Sentence BERT for Job Competency and Course Similarity

Abstract — This research focuses on the application of the *Sentence BERT* (*S-BERT*) model, a specialization of the *BERT* model and an adaptation of the *Transformer* architecture specifically designed for the Indonesian language, in exploring the concept of course credit transfer consolidation in accordance with the Merdeka Belajar – Kampus Merdeka program. The aim of this exploration is to develop an Indonesian-language *S-BERT* model and apply it to search and analyze the similarity between the descriptions of sequences of activities and course syllabi (Rencana Pembelajaran Semester or RPS). The result of this similarity analysis is the identification of relevant courses based on the given query. The developed model has demonstrated effective capabilities in searching and determining the similarity between the descriptions of sequences of activities and course syllabi, with evaluation results showing an accuracy of approximately 37% using *cosine similarity* at $k=10$ and around 41% using *dot product similarity* at the same k value. The courses identified as relevant to the query exhibit high similarity and compatibility, indicating that the *S-BERT* model can be relied upon in the process of course credit transfer consolidation within the context of the Merdeka Belajar – Kampus Merdeka program.

Keywords— *BERT*; Kampus Merdeka; RPS; *S-BERT*; *Transformer*.

I. PENDAHULUAN

Merdeka Belajar Kampus Merdeka (MBKM) adalah kebijakan yang diperkenalkan oleh Kementerian Pendidikan, Kebudayaan, Riset, dan Teknologi Republik Indonesia untuk memberikan kebebasan dan otonomi kepada lembaga pendidikan tinggi. Kebijakan ini mendorong mahasiswa mengembangkan potensi sesuai minat dan bakat melalui berbagai program seperti pertukaran pelajar, magang, penelitian, proyek desa, kewirausahaan, studi independen, dan proyek kemanusiaan [1].

Rencana Pembelajaran Semester (RPS) adalah dokumen yang dibuat oleh dosen untuk memberikan panduan pembelajaran selama satu semester. Dokumen ini mencakup berbagai elemen penting seperti kompetensi dasar, tujuan pembelajaran, metode pembelajaran, dan evaluasi hasil belajar mahasiswa. RPS dirancang untuk memastikan proses pembelajaran berjalan sistematis dan terarah, serta dapat disesuaikan dengan perkembangan zaman, termasuk penerapan teknologi dan *e-learning* [2]. Dalam penyusunan RPS, dosen harus mempertimbangkan analisis kebutuhan, karakteristik mahasiswa, dan keterkaitan kompetensi dengan hasil belajar yang diharapkan. Pengembangan RPS yang efektif membutuhkan kolaborasi antara dosen, mahasiswa, dan pemangku kepentingan lainnya serta pendekatan berbasis riset [3].

Capaian Pembelajaran Mata Kuliah (CPMK), dan Sub-Capaian Pembelajaran Mata Kuliah (Sub-CPMK) adalah komponen penting dalam RPS yang memastikan proses pembelajaran sesuai standar kompetensi yang diharapkan. Setiap komponen ini dirancang untuk memberikan panduan yang jelas dalam proses pembelajaran, mulai dari hasil belajar yang diharapkan hingga metode pengajaran dan evaluasi yang digunakan [3]. Evaluasi terhadap CPMK dan Sub-CPMK sangat penting untuk memastikan bahwa komponen-komponen tersebut telah diimplementasikan dengan baik dan dapat mencapai hasil pembelajaran yang diharapkan [4].

Konsolidasi Alih Kredit Mata Kuliah dalam program Merdeka Belajar Kampus Merdeka (MBKM) memungkinkan mahasiswa untuk mengambil mata kuliah di luar program studi atau universitas mereka, namun tetap mendapatkan pengakuan kredit di program studi asal mereka. Program ini bertujuan untuk memperluas wawasan dan keterampilan mahasiswa dengan mengikuti berbagai kegiatan seperti magang, proyek desa, pertukaran mahasiswa, dan penelitian, yang diakui sebagai bagian dari kredit kuliah [5]. Dalam kegiatan proses konsolidasi alih kredit terhadap mata kuliah, kepala program studi bertugas mencocokkan kompetensi pekerjaan yang dibutuhkan perusahaan atau mitra dengan mata kuliah yang relevan, meliputi deskripsi, CPMK, dan Sub-CPMK. Deskripsi dan Sub-CPMK yang berasal dari CPMK sangat relevan dengan tujuan pembelajaran serta pengetahuan yang diharapkan bagi mahasiswa, sehingga memenuhi kompetensi yang dibutuhkan perusahaan. Namun, konversi ini masih dilakukan secara manual, berisiko kesalahan. Oleh karena itu, diperlukan pertimbangan cermat dari kepala program studi untuk menyesuaikan kompetensi yang dibutuhkan perusahaan dengan aspek-aspek dalam deskripsi, CPMK, dan Sub-CPMK.

Masalah ini bisa diatasi dengan pengembangan sebuah model kecerdasan buatan dalam bidang Pengolahan Bahasa Alami (*Natural Language Processing, NLP*). Model kecerdasan buatan ini dirancang untuk menginterpretasikan dan membandingkan teks, dengan tujuan menemukan mata kuliah yang sesuai dengan kompetensi yang dibutuhkan perusahaan. Ini dilakukan dengan mencocokkan dan membandingkan teks yang menjelaskan kompetensi yang dibutuhkan perusahaan dengan isi dokumen RPS dari setiap mata kuliah, termasuk deskripsi mata kuliah, CPMK, dan Sub-CPMK. Hasil akhirnya adalah daftar mata kuliah yang relevan dengan kompetensi yang dibutuhkan perusahaan tersebut.

Dalam proses konsolidasi alih kredit terhadap mata kuliah relevan, terdapat berbagai tantangan yang dihadapi oleh kepala program studi dalam mencocokkan kompetensi yang dibutuhkan perusahaan dengan mata kuliah yang relevan. Proses konversi manual yang saat ini diterapkan berisiko tinggi dalam hal kesalahan dan ketidakefisienan sehingga dapat menyebabkan proses konsolidasi alih kredit ini tidak obyektif antara kompetensi dengan mata kuliah. Oleh karena itu, penelitian ini bertujuan mengeksplorasi penggunaan kecerdasan. Model yang dikembangkan dan ditugaskan untuk mencari similaritas ini diharapkan dapat menginterpretasikan dan membandingkan teks untuk menemukan mata kuliah yang sesuai dengan kompetensi yang dibutuhkan perusahaan. Beberapa pertanyaan penelitian yang muncul adalah:

1. Bagaimana peran kecerdasan buatan, khususnya Pengolahan Bahasa Alami (*NLP*), dalam mengoptimalkan proses konsolidasi alih kredit terhadap mata kuliah yang relevan?
2. Dalam aspek teknis, bagaimana model kecerdasan buatan ini dapat dikembangkan untuk secara efektif membandingkan dan mencocokkan teks kompetensi yang dibutuhkan perusahaan dengan RPS dari setiap mata kuliah?

Penelitian ini berfokus pada pengembangan model bahasa alami berbasis arsitektur *Transformer* yaitu *Bidirectional Encoder Representations from Transformers (BERT)* yang khusus untuk tugas pemahaman semantik kata yaitu *Sentence BERT (S-BERT)*, dalam upaya meningkatkan proses konsolidasi alih kredit terhadap mata kuliah yang relevan. Penggunaan model kecerdasan *S-BERT* ini diharapkan dapat meningkatkan kemampuan sistem dalam menginterpretasikan teks dan konteks secara akurat, yang merupakan aspek krusial dalam analisis teks kompetensi yang dibutuhkan perusahaan dan RPS mata kuliah. Tujuan spesifik penelitian ini adalah:

1. Menguji dan mengevaluasi efektivitas model bahasa untuk mengidentifikasi kesesuaian antara kompetensi yang dibutuhkan perusahaan dan mata kuliah yang relevan, dengan fokus pada peningkatan akurasi dan relevansi.

2. Dalam aspek teknis, pengembangan model bahasa untuk membandingkan dan mencocokkan teks kompetensi yang dibutuhkan perusahaan dengan RPS dari setiap mata kuliah melibatkan beberapa langkah krusial yaitu pra-pemrosesan data, perancangan model, *fine-tuning* model, evaluasi model, dan pengujian model.

Penelitian ini terbatas pada program studi teknik informatika. Penelitian ini tidak membatasi hingga 20 SKS seperti MBKM pada dasarnya, karena tujuan dari penelitian tesis ini adalah untuk menguji kemampuan *downstream* task model S-BERT dalam melakukan pencocokan kompetensi yang dibutuhkan perusahaan dengan data mata kuliah yang relevan.

Transformer adalah arsitektur jaringan saraf yang sepenuhnya didasarkan pada mekanisme "*attention*", berbeda dengan arsitektur sebelumnya seperti *Recurrent Neural Network (RNN)* dan *Long Short Term Memory (LSTM)* yang memproses data secara sekuensial. Mekanisme *attention* memungkinkan arsitektur *Transformer* untuk fokus pada bagian-bagian tertentu dari data *input* secara paralel, yang membuatnya lebih efisien dalam menangani urutan data yang Panjang [6]. Sebelum munculnya arsitektur *Transformer*, *RNN* dan *LSTM* sering digunakan untuk tugas-tugas seperti terjemahan bahasa, tetapi memiliki keterbatasan dalam efisiensi dan kemampuan untuk menangani urutan data yang sangat panjang [7]. Arsitektur *Transformer* terdiri dari dua bagian utama: *encoder* dan *decoder*. *Encoder* bertugas memproses teks *input*, sementara *decoder* menghasilkan teks *output*. Setiap bagian terdiri dari beberapa lapisan yang mengandung mekanisme *attention* dan jaringan saraf yang terhubung penuh. Keunggulan utama dari arsitektur ini adalah kemampuannya untuk fokus pada bagian teks yang relevan tanpa terikat oleh urutan kata, yang memungkinkan efisiensi dan kecepatan yang lebih tinggi [8].

BERT merupakan model representasi bahasa yang revolusioner dalam pemrosesan bahasa alami (*NLP*) karena kemampuan untuk memahami konteks kata dari kedua sisi secara simultan. Model ini dilatih menggunakan teknik *pre-training* dua arah pada teks yang tidak dilabeli, memungkinkan *BERT* untuk memahami konteks kata dalam kalimat secara lebih mendalam [9]. Keunikan utama *BERT* terletak pada pendekatan *bidirectional* yang diterapkannya, berbeda dengan model sebelumnya seperti *RNN* dan *LSTM* yang hanya memproses teks dalam satu arah. *BERT* memanfaatkan arsitektur *Transformer* untuk memproses setiap kata dalam konteks kata-kata sebelum dan sesudahnya, memberikan pemahaman kontekstual yang lebih dalam [10].

IndoBERT, sebagai langkah maju dalam pengembangan model bahasa khusus untuk bahasa Indonesia, dilatih menggunakan korpus besar yang mencakup 74 juta kata dari berbagai sumber seperti media sosial, *blog*, berita, dan situs *web*. Model ini dirancang untuk menangani berbagai tugas pemahaman bahasa alami seperti klasifikasi teks, analisis sentimen, dan ekstraksi informasi, dan telah menunjukkan kinerja yang unggul dibandingkan model-model sebelumnya [11].

S-BERT adalah modifikasi dari *BERT* yang dioptimalkan untuk menghasilkan *embedding* kalimat berkualitas tinggi. *S-BERT* memperkenalkan pelatihan dengan *triplet loss* dan *fine-tuning* pada tugas-tugas spesifik yang memerlukan pemahaman kalimat secara keseluruhan, seperti pencocokan kalimat dan pengenalan teks semantik. *S-BERT* mampu mengukur kesamaan antara dua kalimat dengan lebih efisien dan akurat dibandingkan *BERT*, menjadikannya lebih cocok untuk aplikasi seperti pencarian informasi dan sistem rekomendasi [12].

Model *S-BERT* untuk bahasa Indonesia telah digunakan dalam berbagai tugas pemrosesan bahasa alami (*NLP*). Salah satu penelitian mengulas penggunaan model *BERT* dalam bahasa Indonesia, termasuk tujuh model *BERT* yang telah dilatih sebelumnya untuk bahasa Indonesia dan bahasa lokal Indonesia. Dari model-model ini, lima di antaranya adalah model *BERT* monolingual dalam bahasa Indonesia, satu model monolingual dalam bahasa Sunda, dan satu model multibahasa untuk bahasa Indonesia, Jawa, dan Sunda. Model-model ini digunakan dalam berbagai tugas seperti analisis sentimen, klasifikasi, dan rangkuman teks [13].

Penggunaan *S-BERT* dalam pencarian informasi telah menunjukkan hasil yang sangat baik dalam berbagai aplikasi. Model *S-BERT*, yang merupakan modifikasi dari *BERT*, dirancang untuk menghasilkan *embedding* kalimat yang bermakna secara semantik, yang sangat berguna dalam tugas-tugas pencarian informasi. Dalam konteks *document retrieval*, *S-BERT* digunakan untuk menghitung kesamaan antara kueri pengguna dan dokumen yang tersedia, membantu dalam menemukan dokumen yang paling relevan dengan lebih akurat dan efisien dibandingkan metode pencarian berbasis kata kunci tradisional [14]. Sebuah studi menunjukkan bahwa *S-BERT* dapat meningkatkan proses pencarian informasi dalam sistem pengetahuan desain *NASA*. Model yang di-*fine-tune* pada data spesifik ini menunjukkan peningkatan presisi pencarian yang signifikan. Selain itu, penerapan *S-BERT* dalam *crawling web* bertema juga telah terbukti efektif, membantu dalam mengidentifikasi dan mengindeks halaman *web* yang relevan berdasarkan kesamaan teks dengan kelas yang telah ditentukan sebelumnya [12].

Salah satu penelitian menunjukkan bahwa *S-BERT* telah diadaptasi untuk pengambilan dokumen *ad-hoc* dengan cara mengevaluasi setiap kalimat secara individual dan menggabungkan skor kalimat tersebut untuk menghasilkan skor dokumen. Pendekatan ini terbukti efektif dalam pengujian pada koleksi mikroblog dan berita [15]. Integrasi model *BERT* ke dalam *toolkit Anserini* juga menunjukkan efektivitas pada koleksi dokumen besar, menyoroti tantangan teknis dalam menggabungkan kemampuan *NLP* dan *Information Retrieval (IR)* [16]. Sistem *Birch* yang mengintegrasikan *BERT* dengan *toolkit Anserini* berhasil menunjukkan kemampuan pencarian *end-to-end* yang efektif pada koleksi dokumen besar, seperti koleksi berita *TREC* dan media sosial. Sistem ini mengimplementasikan model ranking sederhana yang mencapai efektivitas *state-of-the-art*, menyoroti tantangan teknis dalam integrasi kemampuan *NLP* dan *IR* [16].

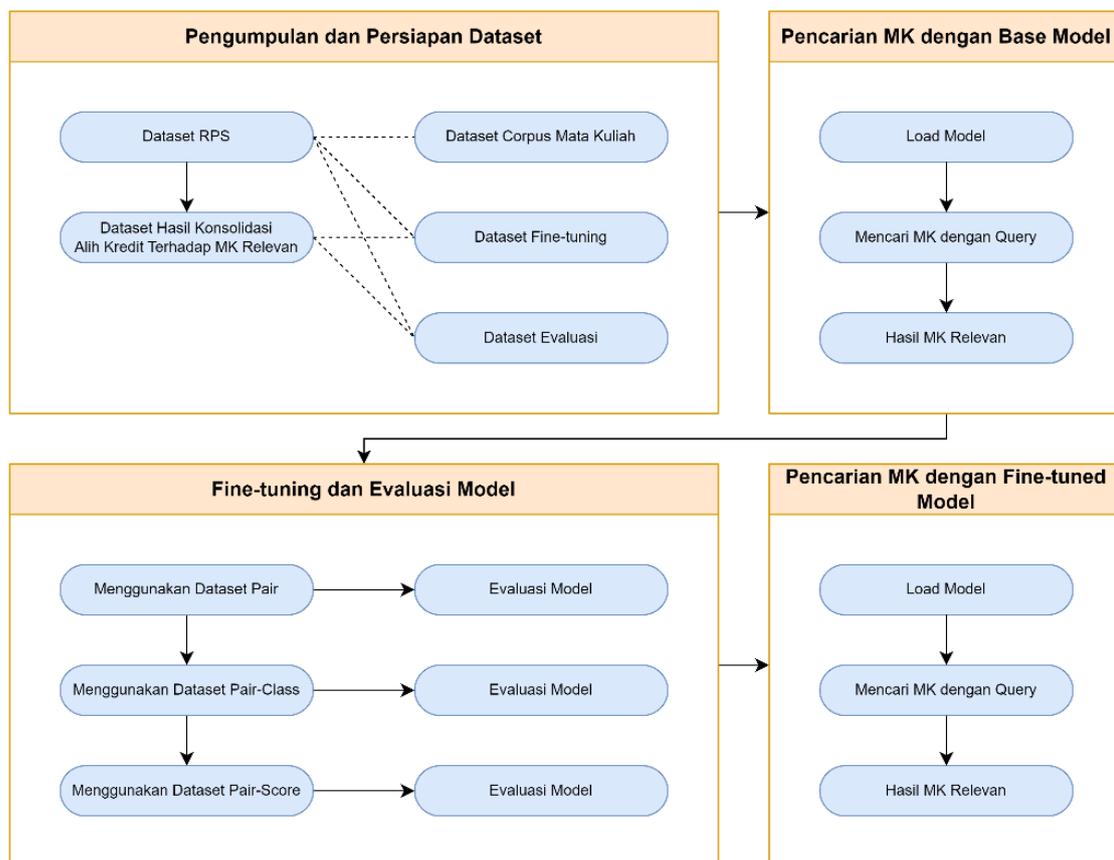
Penggunaan model *S-BERT* dalam pencarian pekerjaan telah membuktikan peningkatan yang signifikan dalam efisiensi dan akurasi pencocokan antara lowongan kerja dan pelamar. Model *S-BERT*, yang merupakan versi khusus dari *BERT*,

dirancang untuk menghasilkan representasi teks yang lebih bermakna dalam menangani pasangan teks seperti *resume* dan deskripsi pekerjaan. Misalnya, penelitian oleh Lavi et al. (2021) mengembangkan *conSultantBERT*, sebuah model yang di-*fine-tune* menggunakan *dataset* besar yang berisi lebih dari 270.000 pasangan *resume*-lowongan, dan berhasil menunjukkan performa yang lebih baik dibandingkan dengan model *unsupervised* dan *supervised* lainnya [17]. Selain itu, *framework* berbasis *BERT* yang diusulkan oleh Abdollahnejad et al. (2021) dapat menyederhanakan proses penyaringan kandidat. *Framework* ini menggunakan data historis aplikasi pekerjaan yang berhasil dan gagal untuk mensimulasikan proses pengambilan keputusan oleh perekrut. Hasil penelitian menunjukkan bahwa *BERT* mengungguli berbagai model lainnya dengan margin yang signifikan, mengurangi beban kerja dan mempercepat proses penyaringan pelamar kerja [18]. Penelitian-penelitian ini menegaskan bahwa model *S-BERT* dan varian *BERT* lainnya dapat meningkatkan keakuratan dan efisiensi dalam pencocokan pekerjaan, memberikan manfaat besar bagi perekrut dan pencari kerja.

Hasil dari penelitian-penelitian ini menunjukkan bahwa *S-BERT* tidak hanya mampu memperbaiki efisiensi dalam pencarian informasi tetapi juga dapat diterapkan pada berbagai *domain* dan jenis data yang berbeda. Model ini juga mampu mengatasi masalah yang muncul pada pendekatan pencarian tradisional yang berbasis kata kunci dengan lebih baik, memberikan hasil yang lebih relevan dan akurat.

II. METODOLOGI PENELITIAN

Penelitian ini melibatkan serangkaian langkah yang sistematis, mencakup pengumpulan data, *pre-processing* data, *fine-tuning* model, evaluasi model, dan pengujian *fine-tuned* model. Alur penelitian ini ditampilkan pada Gambar 1.



Gambar 1. Alur Penelitian

Penelitian ini dimulai dengan tahap pengumpulan dan persiapan *dataset*. Pertama, dikumpulkan *dataset* RPS yang berisi informasi tentang struktur dan mata kuliah. Data hasil konsolidasi dari proses alih kredit digunakan untuk menentukan kompetensi pekerjaan yang relevan dengan mata kuliah tertentu. *Dataset corpus* mata kuliah dihasilkan dari *dataset* RPS kemudian *dataset fine-tuning* dihasilkan dari *dataset* RPS dan juga *dataset* hasil konsolidasi alih kredit dan yang terakhir *dataset* untuk evaluasi juga dihasilkan dari *dataset* RPS dan *dataset* hasil konsolidasi alih kredit.

Tahap berikutnya adalah pencarian mata kuliah menggunakan *base* model *S-BERT*. *Base* model ini pertama-tama dimuat. Kemudian, pencarian mata kuliah dilakukan dengan menggunakan *query*. Hasil pencarian dari *base* model ini memberikan daftar mata kuliah yang relevan berdasarkan *query* yang dimasukkan.

Setelah itu, dilakukan *fine-tuning* dan evaluasi model. Proses *fine-tuning* dilakukan dengan menggunakan *dataset* yang berpasangan untuk meningkatkan akurasi pencarian mata kuliah. Selain itu, digunakan juga *dataset* berpasangan dengan kelas tertentu untuk membantu model dalam mengenali dan mengklasifikasikan mata kuliah. *Dataset* berpasangan dengan skor tertentu juga digunakan untuk lebih mendetailkan kriteria relevansi dalam pencarian mata kuliah. Setelah model di-*fine-tune*, dilakukan evaluasi untuk memastikan peningkatan kinerja dan relevansi hasil pencarian.

Tahap terakhir adalah pencarian mata kuliah menggunakan model yang telah di-*fine-tune*. Model yang telah di-*fine-tune* ini dimuat, dan pencarian mata kuliah dilakukan kembali dengan menggunakan *query* tertentu. Hasil pencarian dari model yang telah di-*fine-tune* ini diharapkan memberikan daftar mata kuliah yang lebih relevan dan akurat dibandingkan dengan *base* model.

A. Pengumpulan Data

Dataset RPS mencakup berbagai mata kuliah pada program studi Teknik Informatika yang terdiri dari beberapa atribut penting seperti ID mata kuliah, kode, nama, jumlah SKS, semester, dan deskripsi mata kuliah. Setiap mata kuliah dalam RPS juga memiliki beberapa CPMK dan Sub-CPMK yang merupakan rincian capaian pembelajaran yang diharapkan dari mata kuliah tersebut. Sub-CPMK ini kemudian digabungkan menjadi satu *string* panjang dan di-*append* ke deskripsi mata kuliah untuk menghasilkan deskripsi yang lebih lengkap dan komprehensif. Data mata kuliah terdiri dari 29 mata kuliah dari program studi Teknik Informatika yang mencakup seluruh semester. Contoh isi data mata kuliah dapat dilihat pada Tabel 1.

TABEL 1
CONTOH ISI *DATASET* RPS MATA KULIAH

Kode	Nama	SKS	Semester	Deskripsi
IN213	Bahasa Inggris	2	1	Mata kuliah ini mengajarkan kemampuan berbahasa Inggris...
IN221	Arsitektur & Keamanan Jaringan	3	2	Mata kuliah ini bertujuan untuk memberikan kemampuan...
IN224	Desain Basis Data	2	2	Mata kuliah ini disusun agar mahasiswa memahami konsep...

Data Sub-CPMK terdiri dari 350 data deskripsi dari mata kuliah program studi Teknik Informatika. Masing-masing deskripsi terhubung pada satu mata kuliah. Contoh isi data Sub-CPMK mata kuliah dapat dilihat pada Tabel 2.

TABEL 2
CONTOH ISI *DATASET* RPS SUB-CPMK

Kode	Deskripsi
IN212	Mahasiswa mampu membuat halaman <i>web</i> dengan menggunakan minimal 80% <i>text format</i> .
IN212	Mahasiswa mampu memahami tentang <i>JavaScript Lanjut</i> .
IN216	Mahasiswa mampu menjelaskan konsep <i>CT</i> , soal <i>HOTS</i> dan implementasinya minimal 80% benar melalui kuis.

Dataset hasil konsolidasi alih kredit terhadap mata kuliah relevan terdiri dari deskripsi pekerjaan yang telah dikonsolidasikan dan relevan dengan mata kuliah tertentu. *Dataset* ini dihasilkan melalui proses alih kredit di mana deskripsi pekerjaan dikaitkan dengan mata kuliah yang sesuai. Informasi yang terkandung dalam *dataset* ini meliputi deskripsi pekerjaan yang relevan dengan capaian pembelajaran dari berbagai mata kuliah. *Dataset* hasil konsolidasi alih kredit terdiri dari deskripsi sekuens, kode mata kuliah relevan dan nama mata kuliah relevan. Contoh isi data hasil konsolidasi alih kredit dapat dilihat pada Tabel 3.

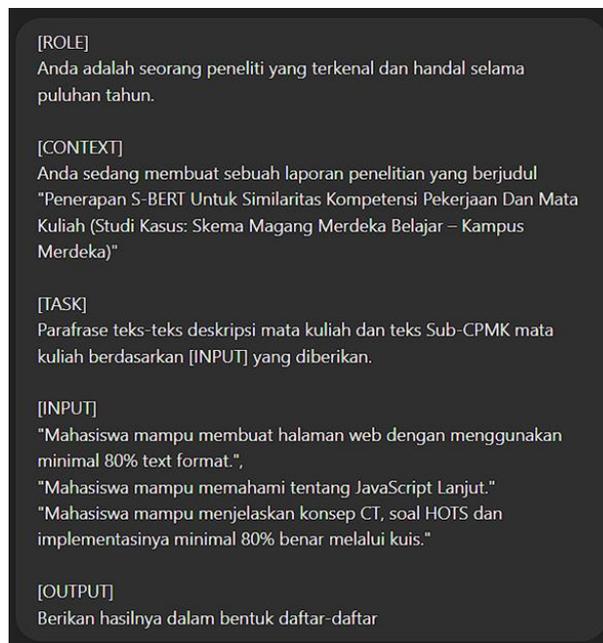
TABEL 3
CONTOH DATA HASIL KONSOLIDASI ALIH KREDIT
TERHADAP MATA KULIAH RELEVAN

Posisi	Deskripsi Sekuens	Kode MK	Nama MK
Web Developer	Mengembangkan halaman <i>web</i> dasar menggunakan <i>HTML</i> dan <i>CSS</i> .	IN212	Web Dasar
	Menerapkan teknik berpikir komputasional untuk mendesain solusi inovatif bagi tantangan teknis.	IN216	Sistem Operasi Komputer
	Menerapkan dan menggunakan <i>API</i> dari <i>cloud service</i>	IN271	Cloud Computing

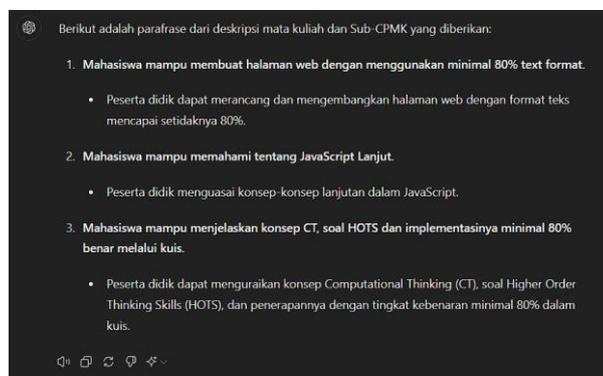
B. Teknik Pra-pemrosesan

Dataset corpus mata kuliah digunakan sebagai sumber data yang nantinya akan dicari dan direlevansikan oleh teks kompetensi yang dibutuhkan perusahaan, terutama pada bagian deskripsi dan Sub-CPMK pada mata kuliah. Pra-pemrosesan *dataset corpus* mata kuliah dimulai dengan penghapusan bagian CPMK. Bagian CPMK mengandung banyak istilah umum yang tidak mencerminkan tujuan keterampilan spesifik dari mata kuliah tersebut, sehingga data CPMK tidak akan digunakan. Fokus utama dialihkan ke Sub-CPMK mata kuliah, yang berfokus pada keterampilan teknis (*hard skill*) yang dihasilkan dari proses pembelajaran, meskipun beberapa keterampilan non-teknis (*soft skill*) juga muncul. Pada data Sub-CPMK, terdapat beberapa entri yang hanya menunjukkan pemahaman mahasiswa, seperti 'menjawab soal kuis' atau 'mengerjakan tugas besar', yang tidak relevan dengan output keterampilan teknis yang diinginkan. Selain itu, ditemukan mata kuliah yang berbeda namun memiliki Sub-CPMK yang sama persis. Mata kuliah seperti 'Seminar Tugas Akhir', 'Tugas Akhir', 'Kapita Selekta', dan 'Kerja Praktek' dihapus dari *dataset* karena berhubungan dengan urusan akademik dan mengandung istilah umum yang tidak relevan dengan keterampilan spesifik. Permasalahan ini diselesaikan dengan menghapus data-data tersebut.

Sebelum masuk ke tahap pra-pemrosesan lebih lanjut, dilakukan parafrase teks pada deskripsi mata kuliah dan semua teks Sub-CPMK nya. Teknik parafrase ini dibantu menggunakan alat *ChatGPT*. Penggunaan *ChatGPT* untuk parafrase sangat bagus karena mampu menghasilkan teks yang relevan secara kontekstual, setara secara semantik, dan beragam secara linguistik [19]. Peningkatan kualitas melalui teknik *prompt engineering* memungkinkan *prompt* yang dirancang dengan baik untuk memandu model menghasilkan variasi leksikal, sintaksis, dan struktur kalimat yang lebih luas, sambil tetap menjaga kejelasan dan keakuratan makna asli. Contoh *prompting* dan hasil *output*-nya untuk parafrase teks dapat dilihat pada Gambar 2 dan Gambar 3.

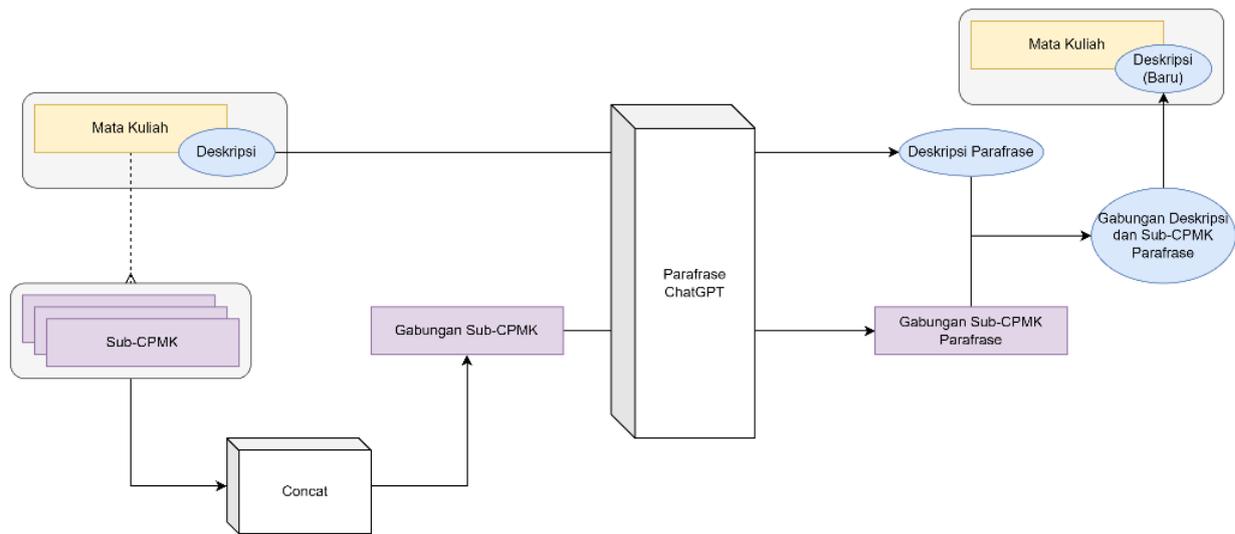


Gambar 2. Contoh *Prompting ChatGPT* untuk Parafrase Teks



Gambar 3. Contoh Hasil Output dari Prompting ChatGPT

Parafrase teks ini diperlukan untuk memperbaiki, menstrukturkan ulang, dan memperkaya teks sehingga model dapat mengenali pola kalimat. Meskipun menggunakan *ChatGPT*, hasil *output* teks parafrasenya tetap harus diperhatikan dan ditinjau kembali sebelum dimasukkan ke dalam *dataset* RPS kembali. Setelah memparafrase semua teks termasuk deskripsi dan Sub-CPMK, dilakukan penggabungan Sub-CPMK ke dalam deskripsi mata kuliah. Setiap Sub-CPMK dalam satu mata kuliah digabung menjadi satu *string* panjang, kemudian dilakukan penghapusan karakter spesial dan di-*append* ke deskripsi mata kuliah. Langkah ini bertujuan untuk memperkaya konten deskripsi mata kuliah dengan rincian capaian pembelajaran yang spesifik. Alur pra-pemrosesan data deskripsi setiap mata kuliah dapat dilihat pada Gambar 4.



Gambar 4. Alur Pra-pemrosesan Deskripsi Mata Kuliah

Pra-pemrosesan data untuk *fine-tuning* model *S-BERT* menggunakan kombinasi dari *dataset* RPS dan *dataset* konsolidasi alih kredit deskripsi pekerjaan. Data ini diolah menjadi tiga jenis tipe *dataset*: *pair*, *pair-class*, dan *pair-score*. Seiring dengan dilakukannya proses *fine-tuning* menggunakan *dataset-dataset* tersebut, akan dilakukan duplikasi *dataset* dari total data aslinya, gunanya untuk meningkatkan performa dari model yang sudah di-*fine-tuning* [20].

Dataset pair digunakan untuk melatih model dengan metode pembelajaran kontrasitif, di mana model dilatih untuk membedakan antara pasangan kalimat yang bermakna serupa dan yang tidak. Contohnya, jika ada dua kalimat yang memiliki makna serupa, model dilatih untuk memberikan vektor *embedding* yang lebih dekat satu sama lain di ruang vektor. *Dataset pair* terdiri dari dua kolom utama: *'anchor'* dan *'positive'*. Kolom-kolom ini diisi dengan kombinasi dari deskripsi mata kuliah, Sub-CPMK, dan deskripsi pekerjaan. Contoh pasangan data yang dihasilkan dapat dilihat pada Tabel 4.

TABEL 4
CONTOH ISI DATASET PAIR

<i>Anchor</i>	<i>Positive</i>
Mata kuliah ini mengajarkan mahasiswa agar mampu mendemonstrasikan pemrograman Basis Data menggunakan <i>Views, View, Stored Procedure, Function, Trigger, Cursor</i> .	Mampu mengenal <i>Microsoft SQL Server</i> .
Mata kuliah ini bertujuan agar mahasiswa mampu menerapkan pendekatan sistem cerdas melalui metode pembelajaran mesin yang sesuai, menentukan metode pengolahan data, serta menerapkan dan mengevaluasi performa algoritma cerdas yang digunakan dalam berbagai kasus dengan problem dan <i>dataset</i> yang riil.	Menguasai konsep penerapan kriteria dalam pembentukan model saat proses pembelajaran mesin.
Mata kuliah ini menyiapkan mahasiswa untuk pembuatan game dengan memberikan pengetahuan-pengetahuan dasar pemrograman yang diorientasikan ke arah pemrograman game.	Mahasiswa mengetahui aspek aspek pembuatan <i>character: Character Design, PixelArt, Animation, SpriteSheet</i> .

Dataset pair-class digunakan untuk melatih model dalam mengklasifikasikan jenis hubungan semantik antara dua kalimat. Dalam kasus ini, model dilatih untuk mengklasifikasikan jenis hubungan semantik antara premis, hipotesis, dan label klasifikasinya. *Dataset* ini terdiri dari tiga kolom utama: '*premise*', '*hypothesis*', dan '*label*'. Kolom-kolom ini diisi dengan kombinasi dari deskripsi mata kuliah, Sub-CPMK, dan deskripsi pekerjaan. Contoh pasangan data yang dihasilkan dapat dilihat pada Tabel 5.

TABEL 5
CONTOH ISI DATASET PAIR-CLASS

<i>Premise</i>	<i>Hypothesis</i>	<i>Label</i>
Mata kuliah ini memberikan landasan matematika yang merupakan fondasi utama bagi teknik informatika dan ilmu komputer. Lebih spesifik, MK ini bertujuan untuk membekali mahasiswa skill matematika sebagai fondasi utama keilmuannya. Materi yang dibahas, meliputi logika proposisi & kuantifikasi, himpunan, relasi, fungsi, induksi matematika, kombinatorik, relasi rekurensi, kompleksitas algoritma, teorema <i>Big-Oh</i> , graf dan pohon, dan teori otomata.	Menentukan jalur terpendek dari suatu simpul ke semua simpul dalam suatu graf dengan algoritma Dijkstra.	IN232 (Matematika Diskrit)
Mata kuliah ini menyiapkan mahasiswa untuk pembuatan <i>game</i> dengan memberikan pengetahuan-pengetahuan dasar pemrograman yang diorientasikan ke arah pemrograman <i>game</i> .	Mahasiswa mengetahui dan dapat merancang antarmuka pengguna untuk permainan.	IN266 (Pemrograman <i>Game</i>)
Mahasiswa memahami tujuan dan fungsi <i>Business Process and Functional Modeling: activity diagram, swimlanes</i> .	Desain perangkat lunak.	IN032 (Rekayasa Perangkat Lunak)

Dataset pair-score digunakan untuk menilai tingkat kesamaan antara teks pertama yaitu berupa deskripsi mata kuliah, Sub-CPMK, dan kompetensi pekerjaan terhadap teks yang kedua yaitu berupa deskripsi mata kuliah, Sub-CPMK, dan kompetensi pekerjaan yang relevan atau yang tidak relevan dengan teks pertama, kemudian diberikan skor masing-masing. Dalam proses pembuatan *dataset pair-score*. Salah satu pendekatan yang efektif dalam menentukan skor tersebut adalah melalui penilaian manual oleh manusia (*human annotation*). Penilaian manual melibatkan sejumlah penilai manusia yang ditugaskan untuk memberikan skor berdasarkan persepsi mereka tentang kesamaan atau relevansi antara dua kalimat.

Dataset pair-score terdiri dari tiga kolom utama: '*sentence_1*', '*sentence_2*', dan '*score*'. Kolom-kolom ini diisi dengan kombinasi dari deskripsi mata kuliah, Sub-CPMK, dan deskripsi pekerjaan. Contoh pasangan data yang dihasilkan dapat dilihat pada Tabel 6.

TABEL 6
CONTOH ISI DATASET PAIR-SCORE

<i>Sentence 1</i>	<i>Sentence 2</i>	<i>Score</i>
Mata kuliah ini mengajarkan bagaimana menggunakan konsep <i>markup language</i> sebagai basis penulisan representasi lambang dalam pemrograman <i>web</i> , memanfaatkan <i>client side scripting</i> untuk membuat halaman <i>web</i> statis, menggunakan teknologi yang cocok untuk membuat tata letak dan antarmuka pengguna halaman <i>web</i> , dan mengevaluasi fungsionalitas halaman <i>web</i> berdasarkan persyaratan pengembangan yang telah disusun.	Mahasiswa mampu menggunakan <i>frame</i> untuk menampilkan beberapa halaman <i>web</i> dalam satu jendela peramban.	1.0
Mata kuliah ini disusun agar mahasiswa memahami konsep dasar, prinsip-prinsip dan struktur basis data, juga penerapannya pada perangkat lunak yang digunakan untuk mengelola dan memanggil basis data (<i>Database Management System/DBMS</i>). Materi meliputi	Menerangkan definisi-definisi dalam Relasi dan Fungsi sehingga mampu mendemonstrasikan keterampilan <i>problem solving</i> untuk soal-soal pembuktian definisi secara sistematis.	1.0

pengenalan basis data, normalisasi data, desain *Entity Relationship Diagram (ERD)*, dan penerapan *ERD* ke tabel.

Mahasiswa mampu mengenal penelitian dan cara memulainya.	Mahasiswa mengetahui definisi dan manfaat <i>cloud computing</i> , <i>AWS Console</i> , <i>virtual servers</i> , <i>content delivery</i> , <i>virtual storage</i> , <i>cloud security</i> , <i>monitoring the cloud</i> , <i>load balancers</i> , <i>Elastic Beanstalk</i> , <i>CloudFormation</i> , <i>emerging technologies</i> , <i>billing</i> , <i>support</i> , dan optimisasi <i>cloud</i> dengan <i>AWS CDK</i> .	0.0
----------------------------------------------------------	---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------	-----

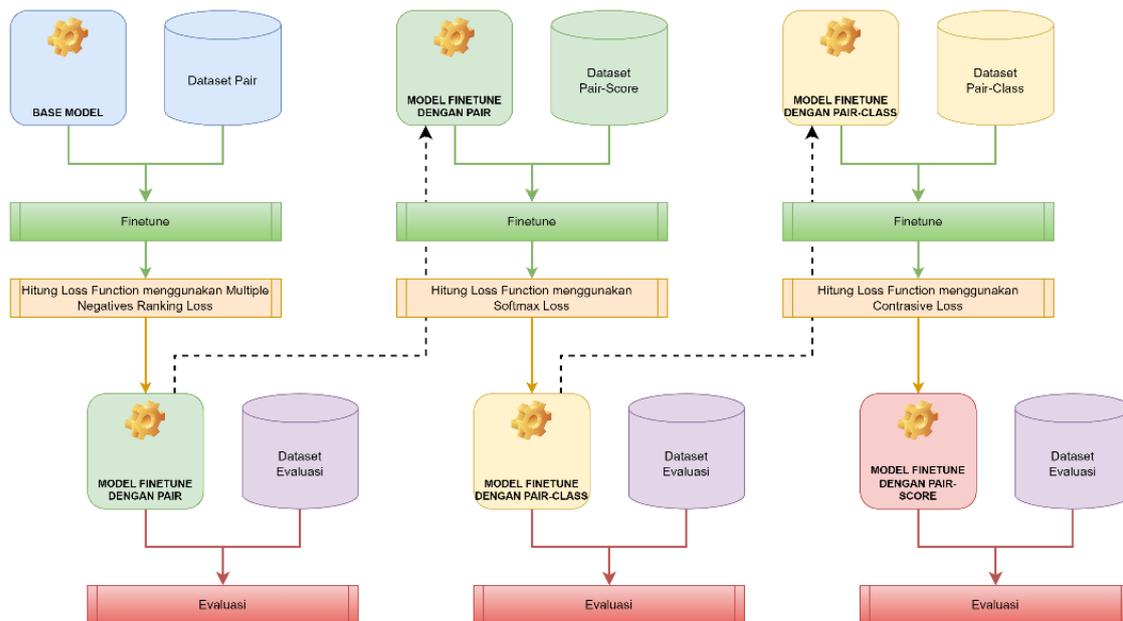
Dataset evaluasi memainkan peran krusial dalam perancangan dan pengembangan model *S-BERT*. Evaluasi dilakukan untuk memastikan bahwa model yang telah di-*fine-tune* dapat bekerja dengan baik dalam konteks yang diharapkan. *Dataset* ini terdiri dari kolom *query descriptions*, *corpus text* dan *score*. Tabel 7 merupakan contoh isi sampel dari *dataset* evaluasi.

TABEL 7
CONTOH ISI DATASET EVALUASI

<i>Query Description</i>	<i>Corpus Text</i>	<i>Score</i>
Memiliki kemampuan bahasa (Inggris, Bahasa Indonesia) pada tingkat (dasar/fasih).	Mahasiswa dapat menguraikan struktur <i>grammar</i> untuk bentuk <i>present</i> , <i>past</i> , dan <i>future sentences</i> .	1
Membuat dan mengembangkan sistem atau aplikasi sesuai kebutuhan <i>user</i> dan perusahaan.	Mata kuliah ini disusun agar mahasiswa memahami konsep dasar, prinsip-prinsip dan struktur basis data, juga penerapannya pada perangkat lunak yang digunakan untuk mengelola dan memanggil basis data (<i>Database Management System/DBMS</i>). Materi meliputi pengenalan basis data, normalisasi data, desain <i>Entity Relationship Diagram (ERD)</i> , dan penerapan <i>ERD</i> ke tabel.	1
Menguasai <i>basic 3D game</i> optimasi seperti <i>LOD</i> dan <i>shader</i> .	Mahasiswa memahami tujuan <i>basic</i> pembuatan <i>shader</i> .	1

C. Perancangan Model

Perancangan model melibatkan proses *fine-tuning* dan evaluasi model. Hal ini mencakup beberapa langkah penting untuk mengoptimalkan kinerja model terhadap pengenalan pola untuk dapat mengenali teks kompetensi (*query*) dan deskripsi gabungan dengan Sub-CPMK pada *dataset corpus* mata kuliah. Dalam tahap ini, digunakan tiga jenis *dataset* untuk *fine-tuning* yang telah dijelaskan sebelumnya, yaitu *dataset pair*, *pair-class*, dan *pair-score*, masing-masing dengan perannya yang spesifik dalam pelatihan model. *Dataset pair* digunakan untuk mencocokkan pasangan teks, *pair-class* untuk klasifikasi pasangan teks, dan *pair-score* untuk memberikan penilaian pada pasangan teks. Tahapan lengkap dari proses *fine-tuning* model ini dapat dilihat pada Gambar 5, yang menggambarkan alur kerja secara rinci dan sistematis.



Gambar 5. Diagram Alur Proses *Fine-Tuning* dan Evaluasi Model

Proses dimulai dengan penggunaan *base model*, yang merupakan model dasar berbahasa Indonesia dengan nama model yaitu ‘*indobenchmark/indobert-base-p1*’ yang telah dilatih sebelumnya. Model ini berfungsi sebagai titik awal sebelum dilakukan *fine-tuning* lebih lanjut. *Base model* ini kemudian akan di-*fine-tune* menggunakan tiga jenis *dataset* yang berbeda untuk meningkatkan kemampuannya dalam mengenali kesamaan semantik antara kalimat. Langkah pertama dalam *fine-tuning* adalah menggunakan *dataset pair*. *Dataset pair* ini terdiri dari pasangan kalimat yang digunakan untuk melatih model dalam menangkap kesamaan dasar antara kalimat. Proses *fine-tuning* dengan *dataset pair* bertujuan untuk membuat model lebih peka terhadap similaritas semantik yang mendasar. Hasil dari proses ini adalah model yang telah dioptimalkan dengan *dataset pair*, yang kemudian akan dievaluasi menggunakan *dataset evaluasi* untuk mengukur kinerjanya. Setelah *fine-tuning* dengan *dataset pair*, model yang dihasilkan digunakan sebagai dasar untuk *fine-tuning* selanjutnya dengan *dataset pair-class*. *Dataset pair-class* digunakan untuk melatih model dalam mengklasifikasikan hubungan semantik antara pasangan kalimat dengan mata kuliah yang relevan. Proses ini bertujuan untuk memperdalam pemahaman model tentang jenis-jenis hubungan semantik terhadap mata kuliah yang relevan yang lebih kompleks. Model yang telah di-*fine-tune* dengan *dataset pair-class* kemudian dievaluasi kembali untuk mengukur peningkatan kinerjanya. Langkah berikutnya adalah *fine-tuning* dengan *dataset pair-score*. Model yang telah di-*fine-tune* dengan *dataset pair-class* di-*fine-tune* kembali menggunakan *dataset pair-score*. *Dataset pair-score* digunakan untuk memberikan penilaian numerik terhadap tingkat kesamaan antara pasangan kalimat. Proses ini membantu model untuk mengukur similaritas dengan lebih presisi melalui skala numerik. Model yang dihasilkan dari *fine-tuning* dengan *dataset pair-score* kemudian dievaluasi menggunakan *dataset evaluasi* yang sama.

Setiap model yang telah di-*fine-tune* melalui tahapan tersebut dievaluasi menggunakan *dataset evaluasi* yang konsisten. Evaluasi ini bertujuan untuk mengukur kinerja model dalam mengenali dan mencocokkan *query* kompetensi dengan deskripsi mata kuliah. Proses evaluasi ini penting untuk memastikan bahwa model yang dihasilkan memiliki kemampuan yang baik dalam menangkap relevansi semantik antara kompetensi pekerjaan dan deskripsi mata kuliah.

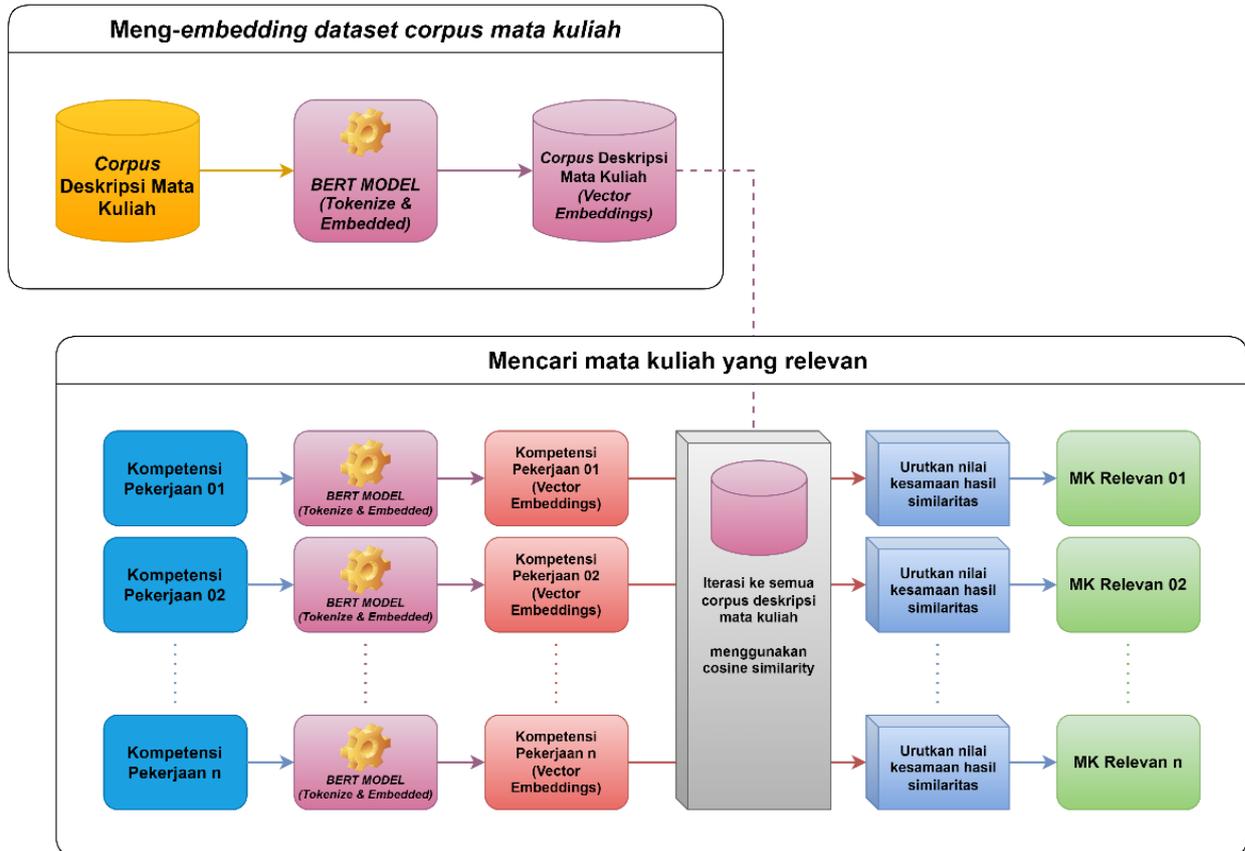
Secara keseluruhan, proses *fine-tuning* dan evaluasi model ini dirancang untuk mengoptimalkan kemampuan model *S-BERT* dalam mengenali pola teks yang kompleks dan memastikan relevansi antara kompetensi pekerjaan dan deskripsi mata kuliah. Dengan menggunakan berbagai jenis *dataset* secara berurutan, model diharapkan dapat mencapai kinerja yang lebih baik dan menghasilkan hasil yang lebih akurat dalam mengidentifikasi similaritas semantik antara kompetensi pekerjaan dengan mata kuliah yang relevan.

Proses pencarian mata kuliah relevan dimulai dengan meng-*embed dataset corpus* mata kuliah menggunakan model *S-BERT*. *Dataset corpus* mata kuliah yang telah dibersihkan sebelumnya, diolah dengan model *S-BERT* untuk menghasilkan representasi vektor (*embedding*) dari setiap elemen dalam *dataset corpus* mata kuliah tersebut. *Embedding* ini untuk memproses dan membandingkan teks secara lebih efektif berdasarkan konteks semantiknya.

Langkah pertama adalah mengonversi *dataset corpus* mata kuliah yang telah dibersihkan menjadi *embedding* menggunakan model *S-BERT*. Model *S-BERT* ini bertugas mengubah teks deskripsi dari *dataset corpus* mata kuliah menjadi

representasi vektor yang kaya akan informasi semantik. Hasilnya adalah sekumpulan data deskripsi mata kuliah yang telah di-embed kedalam representasi nilai vektor, yang siap untuk digunakan dalam proses pencarian mata kuliah yang relevan.

Selanjutnya, proses pencarian mata kuliah dimulai dengan menerima *query*. *Query* yang dimaksud adalah teks dari kompetensi yang dibutuhkan perusahaan dari *dataset* hasil konsolidasi alih kredit terhadap mata kuliah. Selain mengambil teks kompetensi dari *dataset* hasil konsolidasi alih kredit terhadap mata kuliah, diambil juga beberapa contoh dari situs pencarian kerja seperti *LinkedIn*, *JobStreet*, dan situs web resmi MBKM untuk skema magang. Setiap *query* atau teks kompetensi ini juga diolah dengan model *S-BERT* yang sama untuk menghasilkan *embedding* yang sesuai. Model *S-BERT* mengubah teks kompetensi menjadi vektor *embedding*, sehingga memungkinkan perbandingan yang lebih mudah dan akurat dengan *embedding* dari *dataset corpus* mata kuliah. Alur proses pencarian mata kuliah relevan dapat dilihat pada Gambar 6.



Gambar 6. Alur Pencarian Mata Kuliah Relevan

Dengan *embedding* dari kompetensi (*query*) dan *dataset corpus* mata kuliah yang telah dihasilkan, langkah berikutnya adalah melakukan pencarian mata kuliah relevan. Dilakukan iterasi melalui seluruh mata kuliah yang ada dalam *dataset corpus* mata kuliah yang telah di-embed, dan menghitung kesamaan menggunakan fungsi *cosine similarity* antara *embedding* dari kompetensi (*query*) dan *embedding* dari setiap mata kuliah pada *dataset corpus* mata kuliah.

Hasil dari pencarian ini adalah daftar mata kuliah yang diurutkan berdasarkan tingkat kesamaannya dengan teks kompetensi (*query*). Mata kuliah yang memiliki kesamaan tertinggi ditempatkan di urutan paling atas, dan yang terendah di urutan paling bawah. Setiap teks kompetensi (*query*) akan menghasilkan satu mata kuliah yang relevan.

III. HASIL DAN PEMBAHASAN

A. Persiapan

Untuk mendapatkan hasil yang relevan dari mata kuliah, disiapkan daftar kompetensi pekerjaan yang dijadikan *query-query* untuk diuji coba. Tabel 8 memaparkan teks-teks *query* tersebut.

TABEL 8
DAFTAR KOMPETENSI (QUERY)

No.	Query
1	Dapat memahami dan mengimplementasikan, dan mengamankan jaringan <i>network</i>
2	Dapat memprioritaskan pekerjaan yang dilakukan guna menghasilkan aplikasi yang dapat digunakan sesuai target masing-masing <i>sprint</i>
3	Mengidentifikasi kebutuhan bisnis dan teknis untuk proyek <i>web</i>
4	Membuat rencana proyek, termasuk <i>timeline</i> dan <i>milestone</i>
5	Menguasai <i>SQL</i> dan mampu melakukan <i>backup database</i>
6	Menguasai bahasa pemrograman <i>Kotlin</i> dan mengimplementasikan kedalam proyek berbasis <i>mobile</i>
7	Menguasai <i>Database SQL, MySQL, PostgreSQL, Oracle</i>
8	Memiliki kemampuan analisa program yang baik
9	Menguasai bahasa <i>PHP Framework, Laravel, CodeIgniter, MySQL, PostgreSQL, jQuery, Javascript, CSS, WordPress, Bootstrap, MySQL, Typescript, ECMAScript, dan AWS</i>
10	Memiliki kemampuan bahasa (Inggris, Bahasa Indonesia) pada tingkat (dasar/fasih)

B. Fine-tuning dan Evaluasi Model

Pada bagian ini akan dibahas beberapa hasil dari *fine-tuning model* secara sekuensial kemudian memvalidasi hasilnya. Pada awalnya, akan dilakukan *fine-tuning model* dengan menggunakan *dataset pair*, yang berisi pasangan *input-output* yang relevan. Kemudian, model akan di-*fine-tune* menggunakan *dataset pair-class*, yang tidak hanya berisi pasangan data tetapi juga dilengkapi dengan label kelas untuk mengklasifikasikan pasangan tersebut. Terakhir, model akan di-*fine-tune* menggunakan *dataset pair-score*, di mana setiap pasangan data diberi skor tertentu untuk menunjukkan relevansi atau tingkat kesamaan.

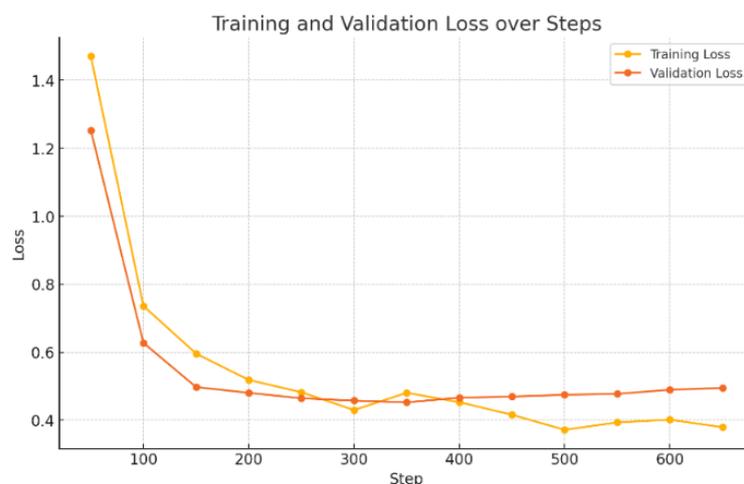
Untuk melakukan *fine-tuning model* secara sekuensial, diperlukan spesifikasi sebagai berikut: sistem operasi berbasis Linux yang dijalankan melalui *platform Windows Subsystem for Linux (WSL)*, *GPU NVIDIA GeForce 3070* dengan *CUDA* yang sudah terpasang dan memiliki *VRAM* sebesar 8GB, *RAM* sebesar 16GB, serta prosesor *AMD Ryzen 7 5800H*.

Selama seluruh proses *fine-tuning* ini, beberapa *hyperparameter* kunci akan digunakan secara konsisten di setiap tahap untuk menjaga keseragaman dan memastikan evaluasi yang adil. *Hyperparameter* tersebut meliputi:

1. *Epochs*: Setiap tahap *fine-tuning* akan berlangsung selama 10 *epochs*.
2. *Train Batch Size*: Ukuran *batch* untuk pelatihan ditetapkan pada 32.
3. *Evaluation Batch Size*: Ukuran *batch* untuk evaluasi juga ditetapkan pada 32.
4. *Learning Rate*: *Learning rate* ditetapkan pada 0.00005 (5e-5).

Evaluasi yang dilakukan menggunakan metrik akurasi dari *cosine similarity* dan *dot product*. Penggunaan metrik *cosine similarity* dan *dot product* bersama-sama memberikan pemahaman yang lebih menyeluruh tentang performa model [21].

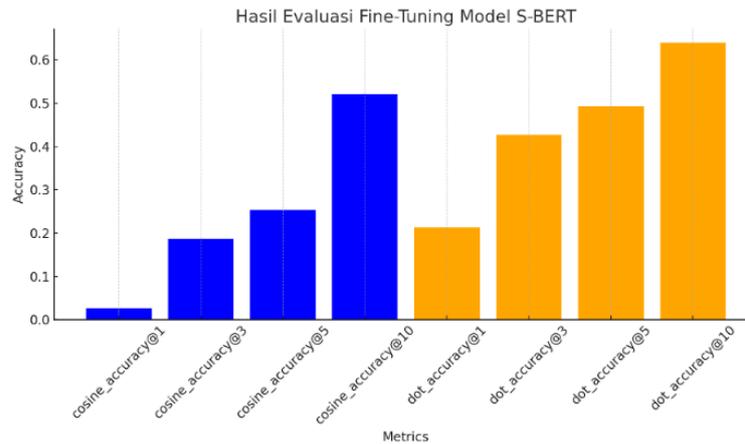
Hasil dari *fine-tuning base model* menggunakan model dari '*indobenchmark/indobert-base-p1*' ini memberikan gambaran tentang bagaimana model *S-BERT* telah dilatih dan divalidasi menggunakan *dataset pair*. Pada awal pelatihan, baik *training loss* maupun *validation loss* cukup tinggi, yang menunjukkan bahwa model memiliki performa awal yang buruk dalam memahami dan memprediksi hubungan antara *anchor* dan *positive*. Grafik gambar bisa dilihat pada Gambar 7.



Gambar 7. Grafik Performa *Fine-Tuning Model* Menggunakan *Dataset Pair*

Seiring bertambahnya langkah pelatihan (*steps*), kedua *loss* tersebut menurun drastis. Penurunan yang signifikan pada langkah-langkah awal menunjukkan bahwa model dengan cepat belajar dari data yang disediakan. Setelah sekitar 100 langkah, laju penurunan menjadi lebih lambat, dan nilai *loss* mulai stabil di sekitar langkah ke-300. Pada titik ini, *training loss* lebih rendah dibandingkan *validation loss*, namun keduanya berada pada kisaran yang stabil, menunjukkan bahwa model telah mencapai tingkat generalisasi yang baik. Stabilitas nilai *loss* ini mengindikasikan bahwa model telah menemukan pola-pola penting dalam data pelatihan tanpa mengalami *overfitting*. Dengan kata lain, model tidak hanya menghafal data pelatihan tetapi juga mampu menerapkan pengetahuannya pada data baru yang serupa (data validasi).

Evaluasi model hasil *fine-tuning* dilakukan setelah melatih *base model* menjadi model hasil *fine-tuning* dengan *dataset pair*. Grafik hasil evaluasi dapat dilihat pada Gambar 8.



Gambar 8. Hasil Evaluasi Model *Fine-tune* Menggunakan *Dataset Pair*

1. '*cosine_accuracy@1*' menunjukkan akurasi yang paling rendah dengan nilai 0.0267. Hal ini menunjukkan bahwa model memiliki kesulitan dalam mengidentifikasi pasangan yang benar dengan hanya satu pilihan.
2. '*cosine_accuracy@3*' dan '*cosine_accuracy@5*' mengalami peningkatan signifikan masing-masing sebesar 0.1867 dan 0.2533. Ini menunjukkan bahwa dengan menambah jumlah pilihan, model menjadi lebih baik dalam mengidentifikasi pasangan yang benar.
3. '*cosine_accuracy@10*' mencapai nilai tertinggi 0.52, yang berarti lebih dari separuh prediksi dalam 10 pilihan teratas adalah benar.
4. '*dot_accuracy@1*' memiliki nilai yang jauh lebih tinggi dibandingkan '*cosine_accuracy@1*', yaitu 0.2133. Hal ini menunjukkan bahwa metrik *dot product* lebih efektif dalam situasi dengan satu pilihan.
5. '*dot_accuracy@3*' dan '*dot_accuracy@5*' menunjukkan peningkatan yang signifikan, masing-masing sebesar 0.4267 dan 0.4933. Ini memperlihatkan bahwa metrik *dot product* konsisten lebih unggul dibandingkan *cosine* dalam semua skenario.
6. '*dot_accuracy@10*' mencapai nilai tertinggi 0.64, menunjukkan bahwa hampir dua pertiga dari prediksi dalam 10 pilihan teratas adalah benar.

Dari hasil evaluasi ini, dapat disimpulkan bahwa metrik *dot product* umumnya memberikan performa yang lebih baik dibandingkan *cosine similarity* pada model yang telah di *fine-tuning* dengan *dataset pair*. Model menunjukkan peningkatan performa yang signifikan ketika jumlah pilihan meningkat, terutama pada metrik *dot product*.

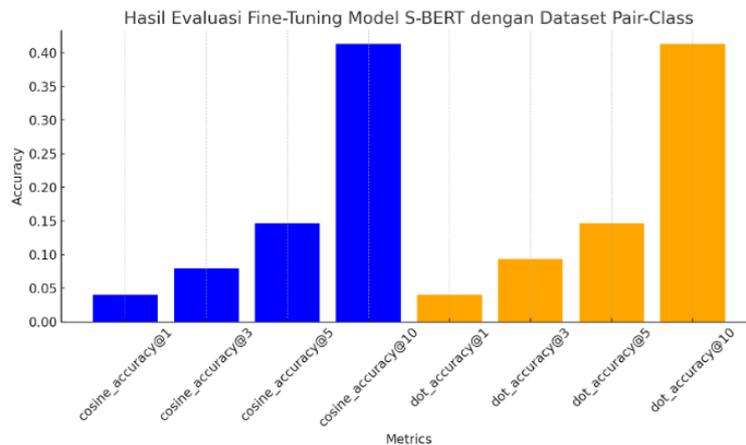
Kemudian dilakukan *fine-tuning* model menggunakan *dataset pair-class*. Hasil *fine-tuning* dengan *dataset pair-class* ini memberikan gambaran tentang bagaimana model *S-BERT* yang sudah dilatih sebelumnya pada *dataset pair*, sekarang dilatih ulang (*fine-tuning*) menggunakan *dataset pair-class*. *Dataset* ini mengandung elemen '*premise*', '*hypothesis*', dan '*label*' yang digunakan untuk tugas klasifikasi teks mata kuliah. Pada awal pelatihan, nilai *loss* baik untuk data pelatihan maupun validasi sangat tinggi, menunjukkan bahwa model pada awalnya mengalami kesulitan dalam memprediksi dengan akurasi yang baik. Gambar 9 merupakan grafik hasil *fine-tuning model* menggunakan *dataset pair-score*.



Gambar 9. Grafik Performa *Fine-Tuning* Model Menggunakan *Dataset Pair-Class*

Hanya dalam beberapa langkah pertama, baik *training loss* maupun *validation loss* turun drastis dan mencapai nilai yang sangat rendah. Setelah langkah awal tersebut, nilai *loss* tetap stabil di kisaran yang sangat rendah. Hal ini mengindikasikan bahwa model dengan cepat belajar dari *dataset pair-class* dan segera mencapai titik stabil dimana model dapat memprediksi dengan sangat baik. Dari grafik tersebut, dapat disimpulkan bahwa *fine-tuning* model *S-BERT* yang sudah dilatih menggunakan *dataset pair* menjadi sangat efektif ketika diterapkan pada *dataset pair-class*. Model dengan cepat beradaptasi dan mencapai performa yang baik dalam beberapa langkah awal. *Hyperparameter* yang digunakan memungkinkan pelatihan yang stabil dan efisien, menjaga *balance* antara kecepatan pelatihan dan kemampuan generalisasi model.

Evaluasi model hasil *fine-tuning* ini dilakukan setelah melatih model hasil *fine-tuning* dari *dataset pair* menjadi model hasil *fine-tuning* dengan *dataset pair-class*. Grafik hasil evaluasi dapat dilihat pada Gambar 10.



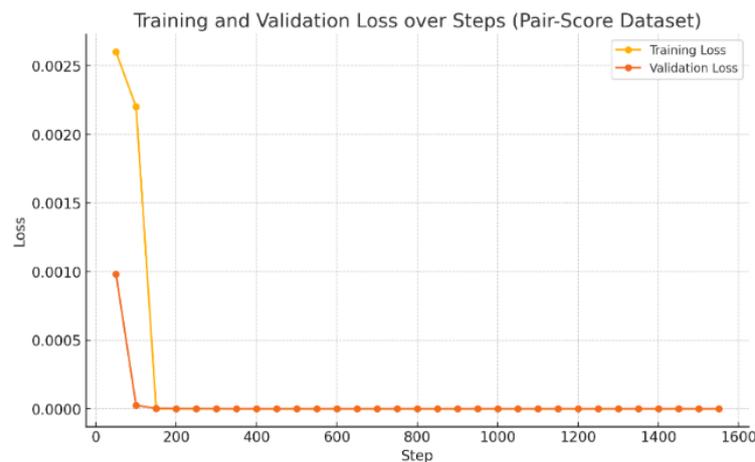
Gambar 10. Hasil Evaluasi Model *Fine-tune* Menggunakan *Dataset Pair-Class*

1. '*cosine_accuracy@1*' menunjukkan akurasi sebesar 0.04, mengindikasikan bahwa model memiliki tingkat kesulitan dalam mengidentifikasi pasangan yang benar dengan hanya satu pilihan.
2. '*cosine_accuracy@3*' mengalami peningkatan menjadi 0.08, meskipun masih cukup rendah.
3. '*cosine_accuracy@5*' meningkat lebih lanjut ke 0.1467, menunjukkan adanya peningkatan performa dengan penambahan jumlah pilihan.
4. '*cosine_accuracy@10*' mencapai nilai tertinggi 0.4133, memperlihatkan bahwa model mampu melakukan prediksi yang benar lebih dari 40% dalam 10 pilihan teratas.
5. '*dot_accuracy@1*' memiliki nilai yang sama dengan '*cosine_accuracy@1*', yaitu 0.04, menunjukkan performa awal yang serupa.

6. *'dot_accuracy@3'* sedikit lebih tinggi daripada *'cosine_accuracy@3'* dengan nilai 0.0933, memperlihatkan bahwa metrik *dot product* sedikit lebih efektif dalam konteks tiga pilihan.
7. *'dot_accuracy@5'* memiliki nilai yang sama dengan *'cosine_accuracy@5'*, yaitu 0.1467.
8. *'dot_accuracy@10'* juga mencapai nilai yang sama dengan *'cosine_accuracy@10'*, yaitu 0.4133.

Hasil evaluasi menunjukkan bahwa baik *cosine accuracy* maupun *dot accuracy* memiliki pola peningkatan yang serupa seiring dengan bertambahnya jumlah pilihan (*k*). Namun, *dot accuracy* sedikit lebih unggul pada nilai *k=3*. Pada nilai *k=5* dan *k=10*, kedua metrik menunjukkan performa yang sama. Dari hasil ini, dapat disimpulkan bahwa model yang telah di *fine-tuning* menggunakan *dataset pair-class*, baik *cosine similarity* maupun *dot product* memberikan hasil yang cukup sebanding.

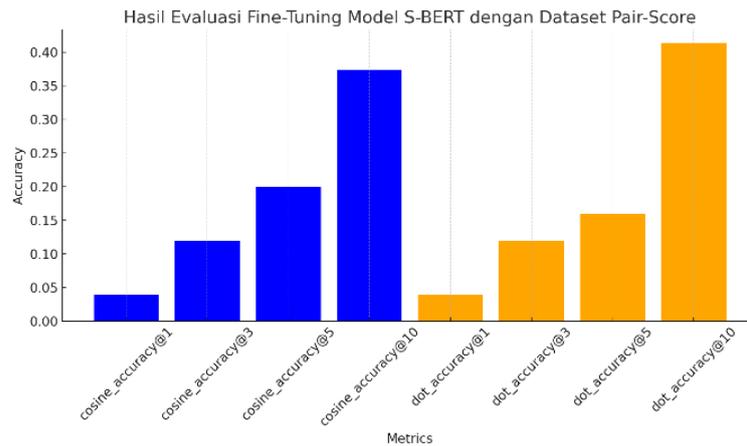
Terakhir dilakukan *fine-tuning* model menggunakan *dataset pair-score*. Hasil *fine-tuning* menggunakan *dataset pair-score* ini memberikan gambaran tentang bagaimana model *S-BERT* yang sudah dilatih sebelumnya pada *dataset pair-class*, sekarang dilatih ulang (*fine-tuning*) menggunakan *dataset pair-score*. *Dataset* ini mengandung elemen *'sentence_1'*, *'sentence_2'*, dan *'score'* yang digunakan untuk tugas regresi atau pemeringkatan teks. Pada awal pelatihan, nilai *loss* baik untuk data pelatihan maupun validasi sangat tinggi, menunjukkan bahwa model pada awalnya mengalami kesulitan dalam memprediksi dengan akurasi yang baik. Grafik gambar bisa dilihat pada Gambar 11.



Gambar 11. Grafik Performa *Fine-Tuning* Model Menggunakan *Dataset Pair-Score*

Hanya dalam beberapa langkah pertama, baik *training loss* maupun *validation loss* turun drastis dan mencapai nilai yang sangat rendah. Setelah langkah awal tersebut, nilai *loss* tetap stabil di kisaran yang sangat rendah. Hal ini mengindikasikan bahwa model dengan cepat belajar dari *dataset pair-score* dan segera mencapai titik stabil dimana model dapat memprediksi dengan sangat baik. Dari grafik ini, kita bisa menyimpulkan bahwa *fine-tuning* model *S-BERT* yang sudah dilatih menggunakan *dataset pair-class* menjadi sangat efektif ketika diterapkan pada *dataset pair-score*. Model dengan cepat beradaptasi dan mencapai performa yang baik dalam beberapa langkah awal. *Hyperparameter* yang digunakan memungkinkan pelatihan yang stabil dan efisien, menjaga *balance* antara kecepatan pelatihan dan kemampuan generalisasi model.

Evaluasi model hasil *fine-tuning* menggunakan *dataset pair-score* ini dilakukan setelah melatih model hasil *fine-tuning* dari *dataset pair-class* menjadi model hasil *fine-tuning* dengan *dataset pair-score*. Grafik hasil evaluasi dapat dilihat pada Gambar 12.



Gambar 12. Hasil Evaluasi Model *Fine-tune* Menggunakan *Dataset Pair-Score*

1. '*cosine_accuracy@1*' menunjukkan akurasi sebesar 0.04, mengindikasikan bahwa model memiliki tingkat kesulitan dalam mengidentifikasi pasangan yang benar dengan hanya satu pilihan.
2. '*cosine_accuracy@3*' mengalami peningkatan menjadi 0.12, menunjukkan adanya perbaikan performa dengan penambahan jumlah pilihan.
3. '*cosine_accuracy@5*' meningkat lebih lanjut ke 0.2, menunjukkan peningkatan performa yang lebih signifikan.
4. '*cosine_accuracy@10*' mencapai nilai tertinggi 0.3733, memperlihatkan bahwa model mampu melakukan prediksi yang benar hampir 40% dalam 10 pilihan teratas.
5. '*dot_accuracy@1*' memiliki nilai yang sama dengan '*cosine_accuracy@1*', yaitu 0.04, menunjukkan performa awal yang serupa.
6. '*dot_accuracy@3*' juga memiliki nilai yang sama dengan '*cosine_accuracy@3*', yaitu 0.12, memperlihatkan bahwa metrik *dot product* juga menunjukkan perbaikan performa dengan penambahan jumlah pilihan.
7. '*dot_accuracy@5*' sedikit lebih rendah daripada '*cosine_accuracy@5*' dengan nilai 0.16, menunjukkan bahwa dalam konteks ini, *cosine similarity* lebih unggul.
8. '*dot_accuracy@10*' mencapai nilai tertinggi 0.4133, yang sedikit lebih baik daripada *cosine_accuracy@10*, menunjukkan efektivitas metrik *dot product* pada 10 pilihan teratas.

Hasil evaluasi menunjukkan bahwa baik *cosine accuracy* maupun *dot accuracy* memiliki pola peningkatan yang serupa seiring dengan bertambahnya jumlah pilihan (k). Pada nilai $k=5$, *cosine similarity* menunjukkan performa yang lebih baik dibandingkan *dot product*, namun pada nilai $k=10$, *dot product* sedikit lebih unggul. Dari hasil ini, dapat disimpulkan bahwa model yang telah di *fine-tuning* menggunakan *dataset pair-score*, baik *cosine similarity* maupun *dot product* memberikan hasil yang kompetitif.

Berdasarkan hasil evaluasi di atas, dapat disimpulkan bahwa model yang di *fine-tuning* menggunakan *dataset pair* umumnya menunjukkan performa yang lebih baik dibandingkan dengan *dataset pair-class* dan *dataset pair-score*. Hal ini terutama terlihat pada metrik *dot accuracy*, di mana *dataset pair* secara konsisten memberikan hasil terbaik di semua nilai k . Namun, pada metrik *cosine accuracy*, performa *dataset pair* tidak selalu unggul, terutama pada nilai $k=1$, di mana *dataset pair-class* dan *pair-score* memberikan hasil yang lebih baik. Secara keseluruhan, *dataset pair* menunjukkan keunggulan dalam mengukur similaritas antara kompetensi pekerjaan dan mata kuliah, terutama ketika menggunakan metrik *dot product*.

C. Hasil Pencarian Menggunakan *Fine-tuned Model*

Pengujian *query-query* kompetensi menggunakan model yang telah di *fine-tune* menunjukkan hasil skor yang sangat tinggi. Namun, meskipun skor menunjukkan kesesuaian yang tinggi, beberapa hasil relevansi ke mata kuliah ternyata tidak sesuai. Berikut adalah hasil pengujian yang dapat dilihat pada Gambar 13.

	Query Aktivitas	Kode MK	Nama MK	SKS	Semester	Deskripsi	Skor
8	Menguasai bahasa PHP Framework, Laravel, Codel...	IN240	Pemrograman Web Lanjut	4	4	Mata kuliah ini mengajarkan implementasi serve...	99.98
4	Menguasai SQL dan mampu melakukan backup database	IN251	Basis Data 2	2	5	Mata kuliah ini mengajarkan pemrograman Basis ...	99.94
6	Menguasai Database SQL, MySQL, PostgreSQL, Oracle	IN240	Pemrograman Web Lanjut	4	4	Mata kuliah ini mengajarkan implementasi serve...	99.94
9	Memiliki kemampuan bahasa (Inggris, Bahasa Ind...	IN213	Bahasa Inggris	2	1	Mata kuliah ini mengajarkan kemampuan berbahas...	99.94
5	Menguasai bahasa pemrograman Kotlin dan mengim...	IN262	Pemrograman Mobile	3	6	Mata kuliah ini memberikan pengetahuan dan ket...	99.93
3	Membuat rencana proyek, termasuk timeline dan ...	IN055	Manajemen Proyek	3	5	Mata kuliah ini menawarkan teknik pengelolaan ...	99.92
1	Dapat memprioritaskan pekerjaan yang dilakukan...	IN254	Proyek Perangkat Lunak	3	5	Mata kuliah ini mengajarkan aspek-aspek peranc...	99.91
2	Mengidentifikasi kebutuhan bisnis dan teknis u...	IN055	Manajemen Proyek	3	5	Mata kuliah ini menawarkan teknik pengelolaan ...	99.89
7	Memiliki kemampuan analisa program yang baik	IN222	Arsitektur Komputer Modern	2	2	Mata kuliah ini bertujuan memberikan kemampuan...	99.89
0	Dapat memahami dan mengimplementasikan, dan me...	IN221	Arsitektur & Keamanan Jaringan	3	2	Mata kuliah ini bertujuan memberikan kemampuan...	99.88

Gambar 13. Hasil Pengujian Query

Poin-poin berikut merupakan penjelasan hasil dari pencarian kompetensi (*query*) menggunakan model yang sudah di-*fine-tune*:

1. Arsitektur & Keamanan Jaringan (IN221) mencatat skor sangat tinggi 99.88 pada *query* terkait pemahaman dan implementasi jaringan serta keamanan *network*. Ini menunjukkan relevansi yang sangat baik antara deskripsi mata kuliah dan kompetensi yang dibutuhkan.
2. Proyek Perangkat Lunak (IN254) mencatat skor hampir sempurna 99.91 pada *query* terkait memprioritaskan pekerjaan untuk aplikasi *sprint*, menunjukkan kesesuaian yang luar biasa.
3. Manajemen Proyek (IN055) mencatat skor 99.89 dan 99.92 pada *query* terkait mengidentifikasi kebutuhan bisnis dan teknis untuk proyek *web* serta membuat rencana proyek, termasuk *timeline* dan *milestone*. Ini menunjukkan relevansi yang sangat tinggi dan kesesuaian yang sangat baik dengan kompetensi yang dibutuhkan.
4. Basis Data 2 (IN251) mencatat skor hampir sempurna 99.94 pada *query* terkait penguasaan *SQL* dan kemampuan melakukan *backup database*. Ini menunjukkan relevansi yang sangat baik.
5. Pemrograman *Mobile* (IN262) mencatat skor 99.93 pada *query* terkait penguasaan bahasa pemrograman *Kotlin*, menunjukkan kesesuaian yang sangat kuat dengan kompetensi yang dibutuhkan.
6. Pemrograman *Web Lanjut* (IN240) mencatat skor tinggi 99.94 pada *query* terkait penguasaan *Database SQL* dan teknologi lainnya. Namun, ini adalah contoh dari hasil yang kurang relevan. Mata kuliah ini seharusnya lebih berfokus pada pemrograman *web* lanjutan, bukan *database*.
7. Arsitektur Komputer Modern (IN222) mencatat skor 99.89 pada *query* terkait kemampuan analisa program yang baik, menunjukkan kesesuaian yang sangat baik dengan kompetensi yang dibutuhkan.
8. Pemrograman *Web Lanjut* (IN240) mencatat skor 99.98 pada *query* terkait penguasaan berbagai bahasa pemrograman dan teknologi. Ini menunjukkan kesesuaian yang tinggi, namun tidak sepenuhnya relevan dengan fokus mata kuliah tersebut.
9. Bahasa Inggris (IN213) mencatat skor hampir sempurna 99.94 pada *query* terkait kemampuan bahasa, menunjukkan relevansi yang sangat tinggi.

IV. SIMPULAN

Penelitian ini telah menganalisis penggunaan model *S-BERT* untuk melakukan pencarian dan mengukur similaritas semantik antara kompetensi pekerjaan dengan deskripsi mata kuliah. Berdasarkan hasil *fine-tuning* model '*indobenchmark/indobert-base-pl1*', beberapa model hasil *fine-tuning*, dan evaluasi model, ditemukan beberapa temuan penting terkait kinerja dan relevansi hasil pencarian model. Model yang telah di-*fine-tuning* mampu memberikan hasil yang lebih akurat dibandingkan dengan *base model*. Model ini mampu memberikan peningkatan kinerja yang signifikan, namun masih terdapat beberapa *query* yang hasil pencariannya tidak sepenuhnya relevan. Hal ini menunjukkan bahwa meskipun *fine-tuning* membantu meningkatkan performa, masih ada ruang untuk perbaikan dalam menangkap kesamaan semantik yang lebih mendalam. Contoh hasil yang relevan dari model yang di-*fine-tune* seperti "Menguasai *SQL* dan mampu melakukan *backup database*" menghasilkan mata kuliah "Basis Data 2". Ini menunjukkan bahwa model mampu menangkap kesamaan semantik dengan baik ketika deskripsi mata kuliah jelas dan komprehensif. Akan tetapi pada contoh *query* seperti "Memiliki kemampuan analisa program yang baik" menghasilkan mata kuliah "Dasar Pemrograman" yang bersifat tidak tepat karena pada nyatanya, mata kuliah Dasar Pemrograman tidak mengajarkan mengenai analisa program. Berdasarkan hasil pengujian pencarian menggunakan *query*, dapat disimpulkan bahwa model *S-BERT* yang di-*fine-tuning* menggunakan *dataset pair-score* memberikan hasil pencarian yang paling relevan dan akurat, dengan skor yang sangat tinggi pada hampir semua *query*. Model *pair-class* juga menunjukkan peningkatan performa yang signifikan dibandingkan dengan model *pair*, namun tidak sebaik model *pair-score*. Dengan demikian, penggunaan *dataset pair-score* dalam *finetuning* model *S-BERT* terbukti lebih efektif dalam mengidentifikasi mata kuliah yang relevan dengan kompetensi pekerjaan. Pada pengujian hasil evaluasi akhir model *fine-tuning* masing-masing jenis *dataset* yang dimulai dari *dataset pair*, *pair-class* dan *pair-score*, ditemukan hasil

terbaik yaitu terlihat dari perhitungan metrik akurasi *dot product* lebih unggul daripada akurasi *cosine similarity* dengan nilai untuk *dot product* yaitu 0.4133 pada $k=10$ sedikit lebih tinggi daripada nilai akurasi *cosine similarity* yaitu 0.3733 pada nilai k yang sama. Ini menyatakan model hasil *fine-tuning* bertahap ini lebih unggul untuk similaritas teks kompetensi pekerjaan dan mata kuliah menggunakan *dot product* daripada menggunakan *cosine similarity*.

Untuk meningkatkan kinerja model *S-BERT* dalam melakukan pencarian dan mengukur similaritas semantik antara kompetensi pekerjaan dan deskripsi mata kuliah, beberapa langkah perbaikan dapat diambil di masa depan. Langkah pertama yang sangat penting adalah memperbaiki *dataset RPS*. Deskripsi dalam RPS perlu dirinci lebih spesifik dan kontekstual. Dengan deskripsi yang lebih terperinci, model akan lebih mudah menangkap konteks spesifik dari setiap *query*, sehingga mampu memberikan hasil pencarian yang lebih akurat dan relevan. Selanjutnya, penting untuk meningkatkan konsistensi dalam data konsolidasi alih kredit. Hal ini dapat dicapai dengan standarisasi terminologi dan struktur data. Standarisasi ini akan membantu dalam menjaga konsistensi penulisan dan format data, yang pada gilirannya akan meningkatkan kemampuan model dalam memahami dan memproses data dengan lebih efektif. Selain itu, penambahan data berkualitas merupakan langkah krusial lainnya. Mengumpulkan lebih banyak data konsolidasi alih kredit yang berkualitas dari berbagai institusi akan sangat membantu. Data tersebut harus representatif dan terperinci, memastikan bahwa model memiliki cukup informasi untuk mengidentifikasi kesamaan semantik dengan lebih baik. Terakhir, standarisasi format penulisan data sangat diperlukan. Mengikuti format penulisan data yang seragam akan memudahkan model dalam memproses dan memahami data secara konsisten.

Dengan format yang terstandarisasi, model akan dapat bekerja lebih efisien dan memberikan hasil yang lebih akurat. Untuk memaksimalkan manfaat dari temuan ini, disarankan untuk mengembangkan hasil penelitian ini menjadi sebuah aplikasi praktis yang dapat digunakan dalam kegiatan MBKM dan skenario lainnya. Aplikasi yang dikembangkan bertujuan untuk membantu mahasiswa, dosen, dan perusahaan dalam:

1. Menilai kecocokan antara kompetensi yang dibutuhkan oleh perusahaan dengan mata kuliah yang telah diambil oleh mahasiswa.
2. Membantu mahasiswa merencanakan mata kuliah yang akan diambil untuk memenuhi kompetensi yang diinginkan dalam industri tertentu.
3. Menyediakan rekomendasi mata kuliah bagi mahasiswa berdasarkan kompetensi yang ingin mereka capai.

Pengembangan aplikasi berbasis model *S-BERT* yang telah di *fine-tuning* ini memiliki potensi besar untuk meningkatkan efektivitas pencarian mata kuliah yang relevan dengan kompetensi pekerjaan. Dengan fitur-fitur yang diusulkan, aplikasi ini tidak hanya akan membantu mahasiswa dalam merencanakan karir mereka tetapi juga akan memperkuat kolaborasi antara dunia akademis dan industri. Langkah selanjutnya adalah mengembangkan prototipe aplikasi dan melakukan uji coba untuk memastikan keefektifannya sebelum diluncurkan secara luas.

Selain pengembangan model dan aplikasi untuk konteks konsolidasi alih kredit MBKM, dengan cara dan metode yang sama, konsep ini dapat diimplementasikan kedalam bentuk alat atau aplikasi yang memerlukan similaritas antar teks misalnya seperti pengembangan model untuk pencarian pekerjaan dengan *query* yang diberikan, similaritas sebuah *Curriculum Vitae* (CV) dengan pekerjaan dan posisi yang diambil oleh kandidat, dan lain-lain.

UCAPAN TERIMA KASIH

Penulis ingin menyampaikan penghargaan dan terima kasih yang sebesar-besarnya kepada Universitas Kristen Maranatha, khususnya Program Studi S2 Ilmu Komputer, atas segala dukungan dan fasilitas yang telah diberikan selama proses penelitian ini. Dukungan dari institusi ini telah memberikan kontribusi yang sangat berarti dalam penyelesaian karya ilmiah ini. Terima kasih atas kesempatan dan kepercayaan yang telah diberikan kepada penulis untuk mengembangkan penelitian ini.

DAFTAR PUSTAKA

- [1] T. M. Fuadi and I. Irdalisa, "Merdeka Belajar Kampus Merdeka: Application in Education Faculty," *AL-ISHLAH J. Pendidik.*, vol. 13, no. 3, pp. 2747–2756, 2022.
- [2] A. A. Yunanto, S. Rochimah, S. Arifiani, and M. S. Dhuha, "Design and Implementation of Rencana Pembelajaran Semester (RPS) Document Generation System," in *2021 International Conference on Advanced Mechatronics, Intelligent Manufacture and Industrial Automation (ICAMIMIA)*, IEEE, 2021, pp. 125–130.
- [3] Sukirman, M. Zaenuri, and L. Q. Hasanah, "Pengembangan Rencana Pembelajaran Semester (RPS) Mata Kuliah Mahārat al-Istimā' Berbasis Kecakapan Abad 21," *Al-Ma'rifah*, vol. 20, no. 1, 2023.
- [4] L. Diner, "Evaluasi Rencana Pembelajaran Semester Mata Kuliah Nihonjijo Dan Seminar Desain Skripsi," *Anterior J.*, vol. 19, no. 2, pp. 1–6, 2020.
- [5] S. Saudah, B. E. Putranti, E. K. Nurnawati, and A. Pranoto, "PENERAPAN MERDEKA BELAJAR KAMPUS MERDEKA (MBKM) DALAM MEMBANGUN KARAKTER MAHASISWA Studi Kasus di Program Studi Teknik Lingkungan Institut Sains & Teknologi AKPRIND Yogyakarta," *JMPA (Jurnal Manaj. Pendidik. Al-Multazam)*, vol. 4, no. 1, p. 7, 2022.
- [6] J. Vig, "A Multiscale Visualization of Attention in the Transformer Model," in *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics: System Demonstrations*, Association for Computational Linguistics, 2019, pp. 37–42.
- [7] A. Vaswani *et al.*, "Attention Is All You Need," *31st Conf. Neural Inf. Process. Syst. (NIPS 2017)*, no. Nips, 2017, [Online]. Available: <https://arxiv.org/abs/1706.03762>

- [8] C. Yeh, Y. Chen, A. Wu, C. Chen, F. Viégas, and M. Wattenberg, "AttentionViz: A Global View of Transformer Attention," *IEEE Trans. Vis. Comput. Graph.*, pp. 1–11, 2023.
- [9] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova, "BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding," in *Proceedings of the 2019 Conference of the North*, Association for Computational Linguistics, 2019, pp. 4171–4186.
- [10] S. Aftan and H. Shah, "A Survey on BERT and Its Applications," in *2023 20th Learning and Technology Conference (L&T)*, IEEE, 2023, pp. 161–166.
- [11] F. Koto, A. Rahimi, J. H. Lau, and T. Baldwin, "IndoLEM and IndoBERT: A Benchmark Dataset and Pre-trained Language Model for Indonesian NLP." [Online]. Available: <https://arxiv.org/abs/2011.00677>
- [12] Y. Tawil and S. Alqaraleh, "BERT Based Topic-Specific Crawler," in *2021 Innovations in Intelligent Systems and Applications Conference (ASYU)*, IEEE, 2021, pp. 1–5.
- [13] D. Sebastian, H. D. Purnomo, and I. Sembiring, "BERT for Natural Language Processing in Bahasa Indonesia," in *2022 2nd International Conference on Intelligent Cybernetics Technology & Applications (ICICyTA)*, IEEE, 2022, pp. 204–209.
- [14] H. S. Walsh and S. R. Andrade, "Semantic Search With Sentence-BERT for Design Information Retrieval," in *Volume 2: 42nd Computers and Information in Engineering Conference (CIE)*, American Society of Mechanical Engineers, 2022.
- [15] W. Yang, H. Zhang, and J. Lin, "Simple Applications of BERT for Ad Hoc Document Retrieval," 2019, [Online]. Available: <http://arxiv.org/abs/1903.10972>
- [16] Z. A. Yilmaz, S. Wang, W. Yang, H. Zhang, and J. Lin, "Applying BERT to Document Retrieval with Birch," in *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP): System Demonstrations*, Association for Computational Linguistics, 2019, pp. 19–24.
- [17] D. Lavi, V. Medentsiy, and D. Graus, "conSultantBERT: Fine-tuned Siamese Sentence-BERT for Matching Jobs and Job Seekers," 2021, [Online]. Available: <https://arxiv.org/abs/2109.06501>
- [18] E. Abdollahnejad, M. Kalman, and B. H. Far, "A Deep Learning BERT-Based Approach to Person-Job Fit in Talent Recruitment," in *2021 International Conference on Computational Science and Computational Intelligence (CSCI)*, IEEE, 2021, pp. 98–104.
- [19] M. K. Pehlivanoglu, M. A. Syakura, and N. Duru, "Enhancing Paraphrasing in Chatbots Through Prompt Engineering: A Comparative Study on ChatGPT, Bing, and Bard," in *2023 8th International Conference on Computer Science and Engineering (UBMK)*, IEEE, 2023, pp. 432–437.
- [20] D. Alexandrov, A. Zakharova, and N. Butakov, "Does Noise Really Matter? Investigation into the Influence of Noisy Labels on Bert-Based Question Answering System," in *2023 IEEE 17th International Conference on Semantic Computing (ICSC)*, IEEE, 2023, pp. 33–40.
- [21] K. V. S, N. L, A. Raj, R. S, and A. D. S, "Abstractive Text Summarizer: A Comparative Study on Dot Product Attention and Cosine Similarity," in *2021 Fourth International Conference on Electrical, Computer and Communication Technologies (ICECCT)*, IEEE, 2021, pp. 1–8.