

PENGEMBANGAN SISTEM *AUTOMATED ESSAY SCORING* MENERAPKAN *NATURAL LANGUAGE PROCESSING* BERBASIS WEB

Hali Putri Aisyah¹, Afrida Helen², Rudi Rosadi³

Program Studi Teknik Informatika, Universitas Padjadjaran

Jl. Raya Bandung Sumedang KM.21, Hegarmanah, Kec. Jatinangor, Kabupaten Sumedang, Jawa Barat 45363

¹hali20001@mail.unpad.ac.id ²helen@unpad.ac.id ³r.rosadi@unpad.ac.id

Penelitian bertujuan untuk meningkatkan efektivitas penilaian esai dengan mengintegrasikan word embedding word2vec, model Bi-LSTM dengan tampilan berbasis web. Kemampuan NLP dalam memahami bahasa manusia memberikan keuntungan model dalam memahami konteks kalimat. Model word-embedding metode word2vec untuk mendapatkan representasi vektor kata yang semantik dan implementasi Bi-LSTM untuk memahami konteks kalimat secara sekuensial. Hasil pelatihan model terhadap dataset sejarah menunjukkan kemampuannya dalam memprediksi data input baru dengan hasil nilai F1 model 0.8943. Penelitian ini tidak hanya meningkatkan efektivitas pengajar, tetapi juga memberikan umpan balik instan kepada siswa. Simpulan penelitian menegaskan bahwa integrasi word2vec dan Bi-LSTM dapat meningkatkan kualitas penilaian esai menyeluruh.

Kata kunci— penilaian esai otomatis, Word2Vec, Bi-LSTM, semantik, efektivitas, NLP, web, pengajar, siswa.

The research aimed to enhance the effectiveness essay scoring by integrating word embedding Word2Vec and Bi-LSTM model with web-based. The NLP's ability to understanding human language give an advantage in comprehending sentence contexts. The Word2Vec method was used to obtain semantic vector representations of word and Bi-LSTM implementation to understand contexts sequentially. The training results model on the history dataset demonstrated its ability to predict new input data with F1 score 0.8943. This research improves teaching effectiveness and provides instant feedback to students. The research conclusion affirms that the integration of Word2Vec and Bi-LSTM can enhance quality of essay scoring.

Keywords— automated essay scoring, Word2Vec, Bi-LSTM, semantic, effectiveness, NLP, website, teacher, students

I. PENDAHULUAN

Ujian esai merupakan salah satu bentuk variasi dari penilaian peserta ujian dalam menjawab pertanyaan atau dalam mengatasi masalah. Menurut Kamus Besar Bahasa Indonesia (KBBI), esai merupakan sebuah nomina yang berarti karangan prosa yang membahas suatu permasalahan secara sepintas dari sudut pandang pribadi penulisnya. Dalam proses penilaiannya, jawaban bentuk esai biasanya dilakukan secara manual. Kekurangan dari penilaian secara manual adalah waktu, human error dan ketidakonsistenan pengujian dalam mengoreksi. Ditengah perkembangan zaman saat ini, ada cara yang dapat digunakan untuk mengatasi kekurangan pengoreksian esai secara manual tanpa mengurangi esensinya.

Salah satu contoh penerapannya adalah dengan memanfaatkan NLP dalam mengoreksi esai.

Natural Language Processing, disingkat NLP, adalah sebuah upaya untuk mengekstrak representasi makna yang lebih lengkap dari teks bebas [1]. Langkah-langkah umum yang dilakukan adalah pengumpulan data, pra-pemrosesan data, pembuatan model kata, pembangunan matriks kata, pelatihan model, pengembangan model, validasi dan evaluasi, serta optimasi dan *fine-tuning*.

Ada banyak model yang dapat digunakan dalam membangun sistem ini, seperti TF-IDF dan *word embedding*. Namun, TF-IDF masih memiliki kekurangan dalam mengekstraksi informasi dari konteks kata-kata. Perlu adanya pemahaman yang jelas mengenai konteks kata dalam kalimat, *word-embedding* berguna dalam memahami konteks teks dimana setiap kata diubah menjadi vektor yang membantu dalam menemukan kemiripan kata. Sehingga, pembangunan sistem penilaian esai otomatis ini, penulis memilih model *word2vec* dalam implementasi pembuatan vector kata dan pembangunan konteks *word embedding*. Pembangunan sistem *Automated Essay Scoring* (AES) akan dibangun berbasis web yang sehingga memudahkan interaksi pengajar sebagai pengguna dengan aplikasi.

II. TINJAUAN PUSTAKA

A. Automated Essay Scoring

Automated Essay Scoring (AES) merupakan salah satu sistem penilaian berbasis komputer secara otomatis menilai tanggapan siswa dengan mempertimbangkan aspek atau fitur yang sesuai. Contoh sistemnya adalah *E-rater* and *Intellimetric* dan *Bayesian Essay Test Scoring System* (BESTY). Sejak dekade terakhir, sistem penilaian esai otomatis ini menggunakan teknik *natural language processing* (NLP) dan berbasis regresi [2]. Penilaian esai otomatis pada machine learning dapat dilihat sebagai masalah teks klasifikasi atau regresi, tergantung pada jenis datanya.

Pembangunan model yang akurat tidaklah mudah, banyak faktor yang memengaruhinya seperti label, validasi, model *training* dan parameter proses *tuning* [3]. Pada jenis ini, model penilaian esai otomatis dilatih menggunakan esai yang sudah memiliki skor. Model terlatih ini kemudian digunakan untuk memprediksi skor esai. Model yang terdapat pada jenis klasifikasi ini adalah *RNN-based model*, *RNN-based model*

dengan nilai spesifik *word-embedding*, Hirarki representasi model, model koheren, *BERT-based models*, dan lain lain [4].

B. Natural Language Processing

Natural Language Processing atau pemrosesan bahasa alami adalah sebuah upaya untuk mengekstrak representasi makna yang lebih lengkap dari teks bebas. Hal hal urusan kata dalam NLP biasanya menggunakan representasi pengetahuan seperti leksikon kata-kata dan maknanya serta sifat tata bahasa dan seperangkat aturan tata bahasa dan sumber daya lain seperti ontologi entitas dan tindakan, atau tesaurus sinonim atau singkatan [1].

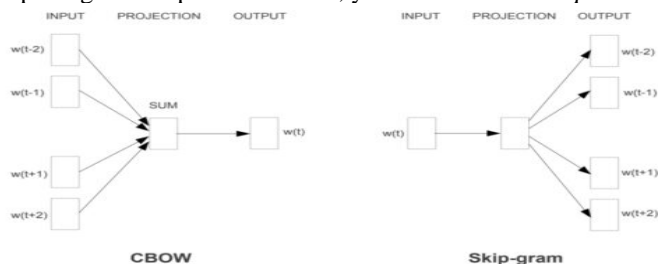
Pendekatan pada NLP terbagi dalam empat kategori, yaitu : simbolik statistik, koneksionisme, dan hybrid. Perbedaan pendekatan ini dilihat dari aspek pemrosesan dan sistem, ketahanan, fleksibilitas, dan kesesuaian untuk berbagai tugas [5]. Ringkasnya, pendekatan yang terdapat pada NLP menunjukkan karakteristik yang berbeda, sehingga beberapa masalah mungkin dapat ditangani dengan satu pendekatan sementara masalah lain dengan pendekatan lainnya. Hal ini karena dalam beberapa kasus, tugas tertentu dengan satu pendekatan mungkin terbukti memadai semetara dalam kasus lain bisa sangat rumit.

C. Word2Vec

Word2Vec adalah metode statistik untuk mempelajari *word embedding* mandiri secara efisien dari *corpus text*. *Word2vec* melibatkan analisis vektor yang dipelajari dan eksplorasi matematika vektor pada representasi kata-kata [6]. Matriks *embedding* merupakan struktur data matematis untuk merepresentasikan kata-kata atau entitas dalam vektor numerik. Matriks yang baik akan meningkatkan akurasi model, karena memungkinkan matriks model mewakili kata-kata dalam konteks. Diketahui bahwa arsitektur *word2vec* model terdiri dari CBOW dan Skip-gram. Dua model pembelajaran ini dijelaskan sebagai berikut:

- 1) *Continuous Bag-of-words*, atau CBOW Model. Model ini mempelajari *embedding* dengan memprediksi kata saat ini berdasarkan konteksnya.
- 2) *Continuous Skip-Gram Model*. Model ini belajar dengan memprediksi kata-kata disekitarnya yang diberi kata saat ini.

Manfaat utama dari pendekatan ini adalah *word embedding* yang berkualitas tinggi dapat dipelajari secara efisien, memungkinkan *embedding* yang lebih besar untuk dipelajari dari kumpulan teks yang jauh lebih besar (miliaran kata). Pada Gambar 1 merupakan dua model arsitektur yang dapat digunakan pada *word2vec*, yaitu CBOW dan *Skip-Gram*



Gambar 1 Arsitektur CBOW (kiri) dan Skip-gram (kanan)

D. Bi-LSTM

Bidirectional Long Short Term Memory atau LSTM dua arah merupakan penambahan LSTM standar guna meningkatkan performa model pada masalah klasifikasi. Model ini melatih dua LSTM pada data latih. LSTM pertama pada data latih asli dan yang lainnya merupakan replika dari data latih yang dibalik. Dengan ini, akan lebih banyak makna yang ditambahkan ke jaringan dan mencapai hasil yang lebih cepat. Konsep yang terdapat pada model ini sangat mudah, yaitu menduplikasi lapisan berulang pertama dalam jaringan, kemudian memberikan data masukan dalam bentuk asli ke lapisan pertama dan replika terbalik dari data masukan ke lapisan duplikat. Hal ini memecahkan masalah hilangnya gradien di RNN standar [7].

Pelatihan Bi-LSTM merupakan semua informasi masukan masa lalu dan masa depan yang tersedia dalam jangka waktu tertentu. Bi-LSTM memproses data masukan dalam dua arah (dari kiri ke kanan dan kanan ke kiri) menggunakan lapisan tersembunyi maju dan lapisan tersembunyi mundur. Dalam *python*, penggunaannya dapat menggunakan pustaka *Keras* dengan pembungkus lapisan dua arah untuk mengimplementasikan Bi-LSTM dengan mengambil lapisan LSTM pertama sebagai argumen [8].

E. LSTM

Long Short-Term Memory merupakan salah satu jenis *deep learning* yang digunakan untuk mendemonstrasikan hasil *world-class* dalam domain yang kompleks seperti terjemahan bahasa, *automatic image captioning* (pembuatan teks gambar otomatis), dan *text generation* (pembuatan teks). LSTM merupakan tipe dari *Recurrent Neural Network* (RNN). RNN merupakan salah satu jenis khusus neural network yang dirancang untuk masalah sekuensial (urutan). RNN dirancang untuk tambahan *looping* pada arsitektur. Contohnya, pada layer tertentu, setiap neuron dapat meneruskan sinyal ke belakang (*sampling*) dan meneruskan ke lapisan berikutnya. Jadi, *output* yang didapatkan tadi dapat dimasukkan sebagai input ke jaringan dengan vektor selanjutnya.

Unit komputasi LSTM disebut dengan sel memori, sel LSTM terdiri dari *weight* and *gate*. Memori sel memiliki *parameter weight* untuk *input*, *output* dan *internal state* yang dibangun melalui *time step*. *Input weight* digunakan untuk *current time step*. *Output weight* digunakan untuk last time step. *Internal State* digunakan dalam kalkulasi *output* langkah saat ini. Kunci dari memori sel adalah *gate*, yang merupakan fungsi pembobotan untuk mengatur aliran informasi dalam sel. LSTM terbatas dalam hal memori atau lebih tepatnya bagaimana ingatan dapat disalahgunakan, seperti dipaksa untuk mengingat satu pengamatan dalam jumlah langkah waktu masukan yang sangat panjang sehingga akan menyebabkan model LSTM mengalami kegagalan [6].

F. F1-Score

Skor F1 adalah metode yang mengintegrasikan recall dan precision model serta merupakan rata-rata harmonis dari keduanya [9]. *F1-score* digunakan untuk mengukur model klasifikasi (prediksi kelas atau label). Cara kerja dari perhitungan ini yaitu, presisi mengukur berapa banyak

prediksi positif model benar. Sedangkan recall mengukur sejauh mana model dapat menemukan semua kasus positif. *F1-Score* memberikan keseimbangan antara kedua metrik. Perhitungan F1 skor dapat dilihat seperti berikut.

$$F1_{Score} = \frac{2 * (Presisi * Recall)}{Presisi + Recall}$$

Keuntungan *F1-Score* yaitu cocok digunakan pada situasi ketika kelas tidak seimbang yaitu seperti kasus dimana satu kelas memiliki jumlah sample jauh lebih banyak daripada yang lain. *F1-Score* cocok untuk kasus klasifikasi karena memperhitungkan kedua jenis kesalahan (*False positives* dan *False Negative*) dan memberikan gambaran yang lebih baik tentang kinerja model pada kelas minoritas.

G. Flask

Flask (source code) merupakan *framework* web yang relatif kecil dan memungkinkan untuk dibaca membaca dan memahami *source code*. *Flask* dirancang sebagai *framework* yang diperluas sejak awal dan menyediakan inti yang kuat dengan layanan dasar dan dapat memilih paket ekstensi yang diinginkan. Dibandingkan dengan *Django framework*, flask dinilai lebih pythonic dan dalam situasi umum lebih eksplisit. Flask mudah digunakan bagi pemula karena kode boilerplate untuk mengaktifkan dan menjalankan aplikasi sederhana hanya sedikit [10].

Berikut beberapa konsep dan fitur utama yang terdapat dalam *Flask*, yaitu *Routing*, *View Function*, *Template Engine*, *Request* dan *Response Objects*, dan *Blueprints*.

III. METODOLOGI

A. Pengumpulan Data

Pengumpulan data dilakukan dengan mengumpulkan data secara mandiri untuk mata pelajaran sejarah SMA di salah satu sekolah. Dataset berbahasa Indonesia dan akan digunakan untuk membuat vektor kata. Dataset dikumpulkan dalam format .csv dengan kolom terdiri dari *RES_ID*, *RESPONSE* dan *LABEL*.

B. Pra-pemrosesan Data

Dataset yang dikumpulkan masih berbentuk kalimat dan mentah dan perlu diproses terlebih dahulu untuk menghasilkan kumpulan kata yang dapat di proses oleh model. Tahapan yang dilakukan dapat dilihat pada Gambar 2.



Gambar 2 Tahapan Pra-pemrosesan Data

- 1) *Trimming* atau *stripping*, fungsi yang digunakan adalah `'text.strip()'` untuk menghapus karakter spasi yang mungkin ada di awal dan di akhir teks.
- 2) *Lowercasing*, fungsi yang digunakan adalah `'text.lower()'` untuk pengonversian teks ke huruf kecil.
- 3) *Punctuation removal* atau *symbol removal*, fungsi yang digunakan adalah `'re.sub('[^0-9a-zA-Z]+', '', text)'` yang akan menghapus karakter selain huruf dan angka.

- 4) *Whitespace normalization* atau *whitespace compression*, dengan menggunakan fungsi `'text = re.sub(' +', ' ', text).strip()'` spasi yang berlebihan atau berturut akan digantikan dengan satu spasi tunggal.

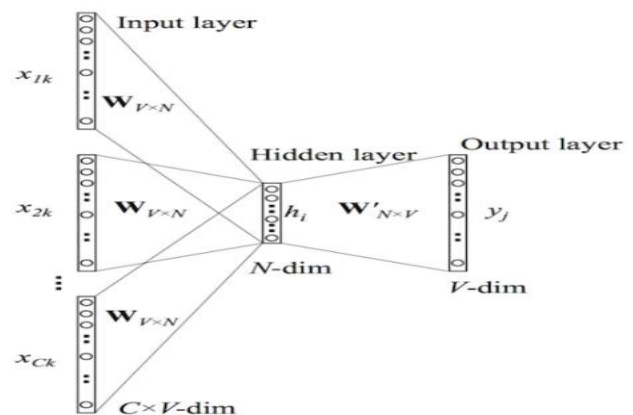
C. Pembuatan Model Word2Vec

Model *word2vec* merupakan model yang dirancang untuk mengubah kata-kata dalam teks menjadi vektor numerik sehingga model dapat memahami dan membandingkan makna kata-kata dalam sebuah teks. Proses dari pembuatan *word2vec* menggunakan *library gensim* pada *python*.

D. Pembuatan Matriks Embedding

Diketahui bahwa arsitektur *word2vec* model terdiri dari CBOW dan *Skip-gram*, pada pembangunan model yang akan digunakan dalam program penilaian esai otomatis ini, model yang bangun menggunakan arsitektur CBOW. CBOW (*continuous bag of words*) merupakan model arsitektur untuk kata target "*current*" (berada ditengah) berdasarkan pada konteks kata sumber (kata disekitarnya). Tahap pertama, ubah kata menjadi vektor, tentukan dimensi dari vektor (*embedding dimension*). Contohnya, jika kita menentukan dimensi sebanyak 2, maka kata "halo" akan merepresentasikan vektor [0.213, 0.4543].

Untuk melatih representasi vektor, CBOW menggunakan target dan *windows*. Tahapan pembuatan matriks *word embedding* dapat dilihat pada Gambar 4. CBOW memiliki sebuah *input layer*, *hidden layer*, dan *output layer*. Input merupakan *one-hot-encode* bobot dan bias dari *hidden layer word-embedding*. Kemudian mengambil dimensi dan membuat prediksi terhadap kata di sekitarnya.



Gambar 3 Tahapan Pembuatan *Matrix Word Embedding*

E. Pelatihan Model

Pada sistem penilaian esai otomatis ini, digunakan model *neural network* untuk melatih model dalam menilai atau memprediksi teks/jawaban esai. Model *neural network* yang digunakan ini adalah Bi-LSTM (*Bidirectional Long Short Term Memory*), pustaka *deep learning* representasi vektor data.

Pada pelatihan model ini, digunakan *sequence_padding*, yaitu sebuah fungsi yang ada pada pustaka *keras deep learning*. Jenis dari *padding* ini dapat disesuaikan dengan kebutuhan, hal ini karena *padding* memengaruhi representasi

teks dan kinerja dari model yang dibangun. Pada aplikasi penilaian esai otomatis ini, digunakan *pre-sequence*. Jenis *padding* ini berguna dalam meningkatkan kemudahan pemahaman karena membuat representasi teks lebih intuitif untuk sebagian besar.

F. Pembangunan Model

Pada tahapan akhir dalam pembangunan penilaian esai otomatis ini, setelah model *Bi-LSTM* ini didapatkan, model kemudian di simpan untuk digunakan kembali dalam memprediksi esai yang sesuai dengan topik dataset dalam model. Prediksi model dapat dilakukan dengan metode *model.predict(X_test)* yang ada dalam model *Keras*. Model prediksi ini dapat disimpan ke dalam database atau diintegrasikan dalam *website*. Model mampu memprediksi teks dengan panjang kata maksimal sesuai pada saat model dilatih.

G. Validasi dan Evaluasi

Tahapan ini digunakan untuk menguji seberapa baik model yang telah dibangun berfungsi pada data baru yang belum pernah dilihat sebelumnya. Tujuannya untuk mengukur keakuratan model dalam memprediksi skor esai. Perhitungan metrik dapat dilakukan untuk mengukur seberapa baik prediksi model yang cocok dengan skor manual menggunakan *F1-Score*. Penilaian F1 ini akan menghitung keseimbangan metrik presisi dan *recall*.

H. Optimasi dan *Fine-tuning*

Pada tahap ini, kinerja model akan dioptimalkan dengan menyetel parameter dan melakukan *fine-tuning* untuk mencapai hasil yang lebih baik. Hal ini dilakukan agar model menjadi lebih akurat dan efektif dalam memprediksi skor esai. Beberapa langkah yang dapat digunakan adalah melakukan penyetelan parameter, *optimizer*, penggunaan data validasi, melakukan penyesuaian model, learning rate, jumlah epoch, ukuran batch, dan lainnya. Lakukanlah validasi dan evaluasi setelah penyesuaian model.

I. Pembangunan Web

Pembangunan web memungkinkan untuk memudahkan sistem penilaian otomatis digunakan oleh pengajar dan siswa. Pembangunan web akan menggunakan *Flask* sebagai *framework backend python*. Kemudian model yang telah di kembangkan akan di integrasikan dengan *python* sebagai *backend*. Rancangan untuk website terdiri dari fitur utama tambah soal, daftar soal, kerjakan soal, daftar hasil kerja siswa dan hasil pengerjaan siswa.

IV. HASIL DAN ANALISA

A. Pembangunan Kosa Kata *Word2vec*

Tahapan pertama dalam pembangunan kosa kata *word2vec* adalah pra-pemrosesan untuk kolom *RESPONSE* pada dataset. *Import* sebuah *library* bernama *gensim* yang kemudian akan digunakan untuk membangun model *word2vec* (*word embedding*).

B. Pembangunan Model Bi-LSTM

Model *Bi-LSTM* yang akan dibangun ini berisi dataset mengenai jawaban siswa dengan topik mata pelajaran sejarah. Kata yang sudah diproses akan dikumpulkan dalam satu array bernama *data_stimulus*. Selanjutnya, lakukan pencarian jumlah kata unik sebagai *max_features* yang terdapat dalam *data_stimulus*. Kemudian tahapan selanjutnya adalah melakukan tokenisasi untuk data *train* dan data *test* dengan memanggil fungsi *Tokenizer* yang tersedia pada *library keras.preprocessing.text*. Tahap tokenisasi ini berguna untuk memecah teks menjadi potongan yang lebih kecil, seperti kata atau sub kata. Potongan ini akan disebut dengan “token”.

Import library Gensim untuk memproses dan memodelkan teks untuk *word embedding* (pemodelan vektor kata). *Import library Sastrawi* untuk memproses Bahasa Indonesia untuk *stemming*, yaitu proses mengembalikan kata ke bentuk dasarnya. Kata yang sudah di *stemming* akan diidentifikasi apakah kata tersebut terdapat dalam kamus *word2vec* atau tidak yang kemudian kata-kata tersebut akan dihitung frekuensi kemunculannya.

Menggunakan *library Keras* dilakukan pembuatan dan pelatihan model jaringan saraf untuk *Bi-LSTM*. Dibuatlah fungsi *F1-Metric* yang berguna untuk mengevaluasi selama pelatihan model. Fungsi ini akan dihitung berdasarkan pada *prediction* dan *true* (*y_true*, *y_pred*). Selanjutnya, melakukan perancangan terhadap arsitektur model jaringan saraf. Model-model ini memiliki lapisan seperti *embedding*, *Bi-LSTM* dan lapisan kepadatan (*dense*). Mengimplementasikan optimizer bernama ‘adam’ dan metrik evaluasi F1 yang telah didefinisikan sebelumnya.

Setelah model dan F1 metrik nilai dirancang, selanjutnya melakukan proses pelatihan model menggunakan teknik *cross-validation*. Proses ini menggunakan beberapa *callbacks* untuk mengelola pelatihan dan penggunaan model dalam membuat prediksi pada data validasi dan uji. Model yang telah dilatih dapat di save agar dapat digunakan kembali untuk memprediksi nilai esai. Penggunaan model yang telah di save dapat dilakukan dengan *load model* melalui *library keras.model* dan memasukan *f1_metric*.

C. Validasi dan Evaluasi Model

Pada model dataset yang sudah dilatih, akan dilakukan proses validasi dan evaluasi model. Proses evaluasi dilakukan dengan menghitung metrik F1. Pada Tabel 1 dibawah ini, terdapat berbagai jenis eksperimen untuk beberapa variable yang digunakan pada saat pelatihan model untuk dataset sejarah.

Tabel 1 Evaluasi Model untuk Dataset dan Variable Berbeda

Maxlen	Presentasi Kelas 1 dan 0	Jumlah Data Latih	Jumlah Epoch	Presisi	Recall	F1-Score
100	63:37	478	50	0.8722	0.9009	0.8863
189	70:30	377	50	0.8591	0.9363	0.8960
189	63:37	478	35	0.875	0.9009	0.8878
189	63:37	478	50	0.8753	0.9042	0.8896

Beberapa hal yang memengaruhi nilai F1 yaitu, panjang kata yang digunakan pada saat melatih model. Semakin banyak token yang dimiliki oleh data, semakin bagus juga pembelajaran model, hal ini selaras dengan panjang maksimum input kata yang disediakan model. Jumlah data dan presentasi kelas 0 dan 1 akan memengaruhi nilai F1. Jumlah *epoch* pada saat melatih model berpengaruh untuk nilai F1, namun jumlah *epoch* untuk dua dataset mengalami kenaikan nilai F1 pada saat jumlah epoch pelatihan dinaikan.

D. Optimasi dan *Fine-tuning* Model

Optimasi dapat meningkatkan kinerja model dengan mengatur parameter atau arsitektur model untuk mencapai hasil terbaik. Serangkaian pengaturan parameter yang sudah dilakukan, menghasilkan nilai F1 seperti berikut ini.

Tabel 2 Hasil Nilai F1 Pengaturan Parameter

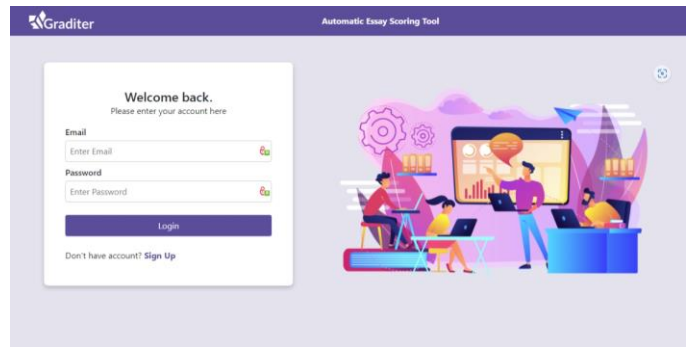
Jumlah Unit Neuron LSTM	Jumlah Unit Neuron Dense	Dropout Layer LSTM	Presisi	Recall	F1-Score
24	24	0.5	0.8798	0.8943	0.8870
24	24	0.1	0.8823	0.8910	0.8866
50	50	0.1	0.8753	0.9042	0.8896
50	100	0.1	0.8814	0.9075	0.8943

Berdasarkan pada hasil eksperimen diatas, pengaturan jumlah unit neuron menghasilkan nilai F1 yang berbeda-beda. Selain itu, mengubah nilai pada parameter dropout akan mempengaruhi nilai F1. Seperti contoh diatas, pada saat lapisan LSTM dan Dense 24 dan dropout 0.5 nilai F1 adalah 0.8870, sedangkan ketika nilai dropout diturunkan, nilai F1 menurun menjadi 0.8866. Selain itu, menaikkan jumlah unit layer pada lapisan akan mempengaruhi nilai F1, semakin tinggi nilai unit neuron, semakin tinggi juga nilai F1. Pada eksperimen diatas, nilai F1 tertinggi ketika nilai unit *neuron* LSTM berjumlah 50 dan unit *neuron* Dense berjumlah 100, yaitu 0.8943.

E. Impelementasi Program Pembangunan Web

Tahap pembangunan web merupakan tahap pengintegrasian model penilaian esai otomatis dengan antarmuka yang akan disajikan dalam bentuk *website*. Pembangunan web bertujuan untuk memudahkan pengguna berinteraksi pada sistem. Pembangunan *website* ini dilakukan dengan menggunakan HTML, CSS dan *Flask (Framework Backend Python)*. Dalam mendukung penyimpanan data, digunakan *platform database MongoDB*. *Website* ini bernama *Graditer* yang terdiri atas beberapa halaman, halaman utama yaitu login yang akan digunakan untuk admin maupun siswa. Untuk halaman admin terdiri atas dashboard admin, daftar esai, daftar pengguna dan edit *profile*. Untuk halaman siswa terdiri atas daftar esai, kerjakan esai dan edit profile.

Gambar merupakan 4 halaman utama sekaligus halaman login untuk pengguna aplikasi. Pengguna baru bisa mendaftarkan diri dengan *sign up*. *Role* pada aplikasi terdiri dari dua, yaitu admin dan *user* sebagai seorang siswa.



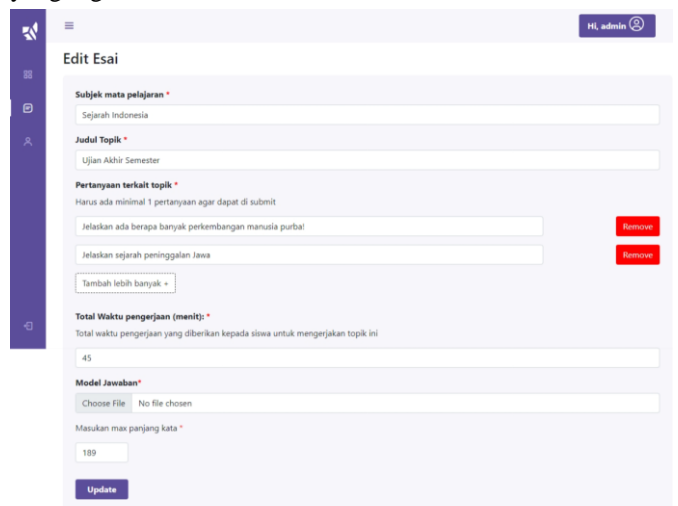
Gambar 4 Halaman Login

Gambar 5 merupakan Dashboard Admin, dimana admin bisa melakukan tambah esai dan tambah pengguna di aplikasi.



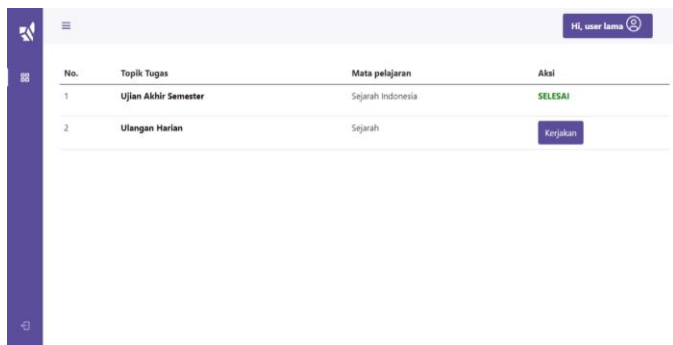
Gambar 5 Dashboard Admin

Gambar 6 merupakan halaman untuk menambahkan esai. Pada saat menambah pertanyaan, hanya pertanyaan yang telah dilatih saja yang tersedia di dropdown. Admin akan diminta untuk upload model dan memasukkan panjang kata maksimum yang digunakan saat melatih model.



Gambar 6 Halaman Tambah Esai

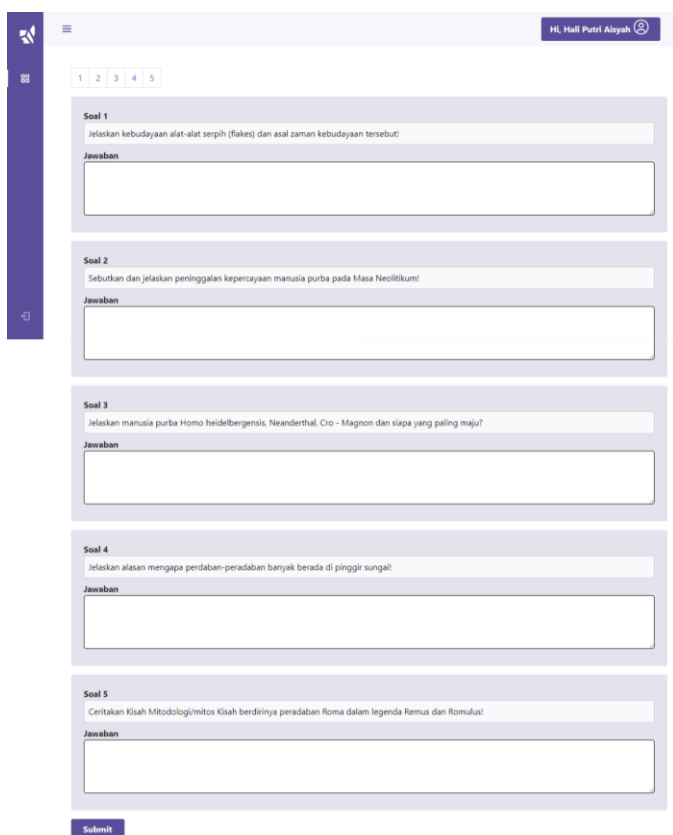
Pada Gambar 7 merupakan Halaman Dashboard Siswa. Pada halaman ini, siswa akan mengerjakan esai yang tersedia dalam halaman. Siswa yang diizinkan untuk mengerjakan esai dipilih melalui sisi admin. Siswa akan mendapatkan umpan balik instan berupa nilai setelah esai di kerjakan.



No.	Topik Tugas	Mata pelajaran	Aksi
1	Ujian Akhir Semester	Sejarah Indonesia	SELESAI
2	Ulangan Harian	Sejarah	Kerjakan

Gambar 7 Dashboard Siswa

Pada Gambar 8 merupakan Halaman Pengerjaan Esai. Siswa akan diberikan soal yang ada pada esai dan mengisinya.



1 2 3 4 5

Soal 1
Jelaskan kebudayaan alat-alat serpih (flakes) dan asal zaman kebudayaan tersebut!

Jawaban

Soal 2
Sebutkan dan jelaskan peninggalan kepercayaan manusia purba pada Masa Neolitikum!

Jawaban

Soal 3
Jelaskan manusia purba Homo heidelbergensis, Neanderthal, Cro - Magnon dan siapa yang paling maju?

Jawaban

Soal 4
Jelaskan alasan mengapa peradaban-peradaban banyak berada di pinggir sungai!

Jawaban

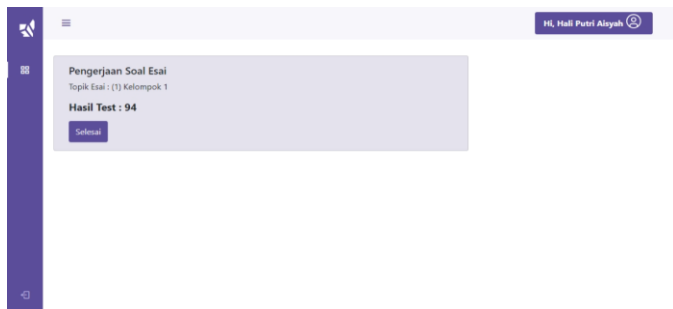
Soal 5
Ceritakan Kisah Mitologi/mitos Kisah berdirinya peradaban Roma dalam legenda Remus dan Romulus!

Jawaban

Submit

Gambar 8 Halaman Pengerjaan Esai

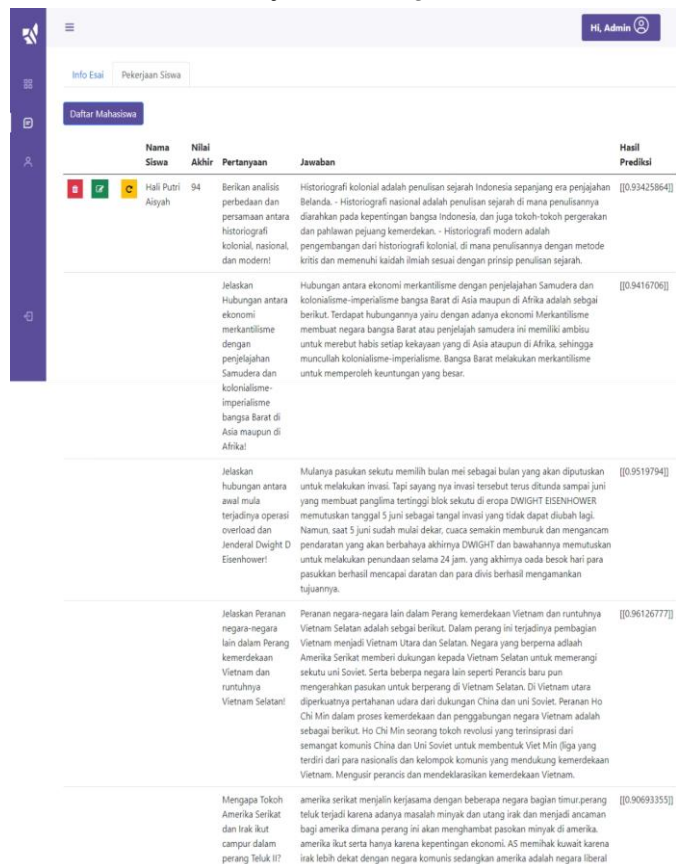
Pada Gambar 9 merupakan Halaman Hasil yang berisi umpan balik nilai siswa.



Pengerjaan Soal Esai
Topik Esai : (1) Kelompok 1
Hasil Test : 94
Selesai

Gambar 9 Halaman Hasil

Pada Gambar 10 merupakan Halaman Hasil Pekerjaan Siswa yang dapat dilihat dari sisi admin. Pada halaman ini admin dapat melihat prediksi nilai siswa dan hasil akhir yang telah dikonversikan menjadi nilai *integer*.



Nama Siswa	Nilai Akhir	Pertanyaan	Jawaban	Hasil Prediksi
Hali Putri Aisyah	94	Berikan analisis perbedaan dan persamaan antara historiografi kolonial, nasional, dan modern!	Historiografi kolonial adalah penulisan sejarah Indonesia sepanjang era penjajahan Belanda. - Historiografi nasional adalah penulisan sejarah di mana penulisannya diarahkan pada kepentingan bangsa Indonesia, dan juga tokoh-tokoh pergerakan dan pahlawan pejuang kemerdekaan. - Historiografi modern adalah pengembangan dari historiografi kolonial, di mana penulisannya dengan metode kritis dan memenuhi kaidah ilmiah sesuai dengan prinsip penulisan sejarah.	[[0.93425864]]
		Jelaskan Hubungan antara ekonomi merkantilisme dengan penjajahan Samudera dan kolonialisme-imperialisme bangsa Barat di Asia maupun di Afrika!	Hubungan antara ekonomi merkantilisme dengan penjajahan Samudera dan kolonialisme-imperialisme bangsa Barat di Asia maupun di Afrika adalah sebagai berikut. Terdapat hubungannya yaitu dengan adanya ekonomi merkantilisme membuat negara bangsa Barat atau penjajahan samudera ini memiliki ambisi untuk merebut habis setiap kekayaan yang di Asia ataupun di Afrika, sehingga munculah kolonialisme-imperialisme. Bangsa Barat melakukan merkantilisme untuk memperoleh keuntungan yang besar.	[[0.9416706]]
		Jelaskan hubungan antara awal mula terjadinya operasi overload dan Jenderal Dwight D Eisenhower!	Mulanya pasukan sekutu memilih bulan mei sebagai bulan yang akan diputuskan untuk melakukan invasi. Tapi sayang nya invasi tersebut terus ditunda sampai juni yang membuat panglima tertinggi blok sekutu di eropa DWIGHT EISENHOWER memutuskan tanggal 5 juni sebagai tanggal invasi yang tidak dapat diubah lagi. Namun, saat 5 juni sudah mulai dekat, cuaca semakin memburuk dan mengancam pendaratan yang akan berbahaya akhirnya DWIGHT dan bawahannya memutuskan untuk melakukan penundaan selama 24 jam, yang akhirnya oada besok hari para pasukan berhasil mencapai daratan dan para divisi berhasil mengamankan tujuannya.	[[0.9519794]]
		Jelaskan Peranan negara-negara lain dalam Perang kemerdekaan Vietnam dan runtuhnya Vietnam Selatan!	Peranan negara-negara lain dalam Perang kemerdekaan Vietnam dan runtuhnya Vietnam Selatan adalah sebagai berikut. Dalam perang ini terjadinya pembagian Vietnam menjadi Vietnam Utara dan Selatan, Negara yang bernama adiah Amerika Serikat memberi dukungan kepada Vietnam Selatan untuk memerangi sekutu uni Soviet. Serta beberapa negara lain seperti Perancis baru pun mengerahkan pasukan untuk berperang di Vietnam Selatan. Di Vietnam utara diperkuatnya pertahanan udara dari dukungan China dan uni Soviet. Peranan Ho Chi Min dalam proses kemerdekaan dan penggabungan negara Vietnam adalah sebagai berikut. Ho Chi Min seorang tokoh revolusi yang terinspirasi dari semangat komunis China dan Uni Soviet untuk membentuk Viet Min (jga yang terdiri dari para nasionalis dan kelompok komunis yang mendukung kemerdekaan Vietnam. Mengusir perancis dan mendeklarasikan kemerdekaan Vietnam.	[[0.9612677]]
		Mengapa Tokoh Amerika Serikat dan Irak ikut campur dalam perang Teluk II?	amerika serikat menjalin kerjasama dengan beberapa negara bagian timur.perang teluk terjadi karena adanya masalah minyak dan utang irak dan menjadi ancaman bagi amerika dimana perang ini akan menghambat pasokan minyak di amerika. amerika ikut serta hanya karena kepentingan ekonomi. AS melihat kuwait karena irak lebih dekat dengan negara komunis sedangkan amerika adalah negara liberal	[[0.9069335]]

Gambar 10 Halaman Hasil Pekerjaan Siswa

V. SIMPULAN DAN SARAN

Pada penelitian yang dilakukan didapat simpulan sebagai berikut :

1. Aplikasi penilaian esai otomatis ini menerapkan NLP dalam mengevaluasi jawaban siswa secara semantik dalam waktu yang lebih singkat jika dibandingkan dengan penilaian manual.
2. *Word2vec* memiliki kemampuan untuk menghasilkan representasi vektor kata, yang menangkap makna dan hubungan semantik antar kata.
3. Penilaian esai otomatis dapat mengurangi waktu penilaian secara signifikan, konsisten, dan objektif.

Ada juga saran dari penelitian ini karena masih memiliki banyak kekurangan dan keterbatasan sehingga saran dari peneliti untuk penelitian selanjutnya adalah sebagai berikut :

1. Dapat dilakukan penelitian lebih lanjut untuk mengembangkan algoritma penilaian esai secara otomatis dalam upaya pemahaman konten dan kemampuan sistem mengatasi bahasa yang kompleks.

2. Dapat dilakukan uji validasi yang lebih mendalam dan membandingkan hasil penilaian otomatis dengan penilaian manusia sehingga dapat memastikan akurasi dan keandalan sistem.

REFERENSI

- [1] A. Kao dan S. R. Poteet, *Natural Language and Text Mining*, London: Springer-Verlag, 2007.
- [2] D. Ramesh dan K. S. Suresh, "An automated essay scoring systems: a systematic literature review," *Artificial Intelligence Review*, 2022.
- [3] R. A. Rajagede, "Improving automatic essay scoring for Indonesian language using simpler model and richer feature," 2021.
- [4] M. Uto, *A review of deep-neural automated essay scoring models*, 2021.
- [5] J. Brownlee, *Long Short-Term Memory Networks With Python, Machine Learning Mastery*, 2017.
- [6] E. D. Liddy, *Natural Language Processing*, In *Encyclopedia of Library and Information Science*, 2001.
- [7] Y. Imrana, Y. Xiang, L. Ali dan Z. Abdul-Rauf, "A bidirectional LSTM deep learning approach for intrusion detection," *Expert Systems with Applications*, 2021.
- [8] A. Graves, A.-R. Mohamed dan G. Hinton, "Speech recognition with deep recurrent neural networks," 2013.
- [9] V. Sheth, U. Tripathi dan A. Sharma, "A Comparative Analysis of Machine Learning Algorithms for," *4th International Conference on Innovative Data Communication Technology and* , 2022.
- [10] M. Grinberg, *Flask Web Development (Developing Web Application with Python)*, O'Reilly Media, 2018.